CLUSTERING TECHNIQUES FOR DNA COMPUTING READOUT METHOD BASED ON REAL-TIME POLYMERASE CHAIN REACTION

MUHAMMMAD FAIZ MOHAMED SAAID

UNIVERSITI TEKNOLOGI MALAYSIA

CLUSTERING TECHNIQUES FOR DNA COMPUTING READOUT METHOD BASED ON REAL-TIME POLYMERASE CHAIN REACTION

MUHAMMAD FAIZ MOHAMED SAAID

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Master of Engineering (Electrical)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JULY 2009

Especially for:

Haji Mohamed Saaid bin Abdul Manap

Dr. Hajah Zurinah binti Hassan

Ainul Fadzilah

Siti Raihani binti Mohamed Saaid

Ainul Huda binti Mohamed Saaid

Muhammad Taufiq bin Mohamed Saaid

Aina Mastura binti Mohamed Saaid

Ilham Rania

ACKNOWLEDGEMENT

I would like to take this opportunity to thank especially my supervisor DR ZUWAIRIE IBRAHIM that guides me to fully understand the project title and some basic concept that very new to me. I would also like to express my gratitude to my co-supervisor DR AZLI YAHYA for guidance and supports.

Secondly, I would like to thank Associate Professor Nor Haniza Sarmin, Professor Marzuki Khalid for their discussions to complete the whole thesis. Also, special thanks go to Associate Professor John Anthony Rose and Professor Junzo Watada for their guidance's during the writing of this thesis.

Special indebtedness goes to all my friends in Universiti Teknologi Malaysia (UTM) especially in the robotics lab, and robotics lab technicians for their assistance and understanding. Not forgetting also their moral support that knowing this project is very difficult for me to understand.

I wish to thank Ministry of Science, Technology and Enviroment (MOSTI), Ministry of Higher Education (MOHE), Research Management Centre (RMC) and UTM for financial supports and facilities.

Finally, my very special thank goes to DR. ZURINAH HASSAN, MR. MOHAMED SAAID, and AINUL FADZILAH, that give me their endlessly support so that I can survive in academic years in UTM. It was so hard to study in UTM without their help and care. Thank you.

ABSTRACT

In the first experiment of Deoxyribonucleic Acid (DNA) computation, Adleman has solved a seven nodes Hamiltonian Path Problem (HPP) by applying some biotechnology techniques such as hybridization and polymerase chain reaction (PCR). In that experiment, graduated PCR has been used to visualize the Hamiltonian path. In other research work, a novel readout method tailored specifically to the HPP in DNA computing was proposed, which employs a hybrid *in vitro-in silico* approach. In the *in vitro* phase, TaqMan-based real-time PCR reactions are performed in parallel, to investigate the ordering of pairs of nodes in the Hamiltonian path, in terms of relative distance from the DNA sequence encoding the known start node. The resulting relative orderings are then processed *in silico*, which efficiently returns the complete Hamiltonian path. However, this method used manual classification to distinguish the two different reactions of real-time PCR. In this thesis, clustering techniques are implemented during the *in silico* phase. Clustering is crucial to identify automatically two different reactions produced by real-time PCR. K-means, Fuzzy C-means (FCM), and Alternative Fuzzy C-means (AFCM) clustering algorithms are implemented to differentiate the output of real-time PCR. Results show that K-means and FCM clustering algorithms are capable to classify the two different reactions of real-time PCR. In addition, it has been shown that AFCM clustering algorithm is better than FCM and K-means in term of handling outliers in the real-time PCR output data. Application of clustering techniques have improved the *in silico* information processing of the readout method.

ABSTRAK

Dalam eksperimen pertama pengkomputeran Asid Deoksiribonukleik (*DNA)*, Adleman telah menyelesaikan Masalah Laluan Hamiltonian (HPP) tujuh nod dengan mengaplikasikan beberapa teknik bioteknologi seperti penghibridan dan tindak balas rantai polimerase (PCR). Dalam eksperimen tersebut, kaedah PCR berperingkat telah digunakan untuk mengimbas laluan Hamiltonian. Dalam penyelidikan lain, kaedah terbaru baca-keluar yang disesuaikan secara spesifik untuk HPP dalam *DNA computing* dibincangkan, yang menggunakan pendekatan hybrid *in vitro-in silico*. Dalam fasa *in vitro,* tindak balas PCR masa nyata berdasarkan TaqMan dijalankan secara serentak, untuk mencari turutan pasangan nodan dalam HPP, dengan mengambilkira jarak, secara relatif, dari jujukan DNA yang mengekod nodan permulaan yang telah diketahui. Hasil dari turutan secara relatif diproses secara *in silico*, yang mana menghasilkan HPP yang lengkap dengan cekap. Bagaimanapun, kaedah baca-keluar tersebut menggunakan klasifikasi manual untuk membezakan dua tindak balas berbeza PCR masa nyata. Dalam thesis ini, teknik pengerumunan dijalankan semasa fasa *in silico*. Pengerumunan sangat penting dalam mengenal pasti secara automatik dua tindak balas berbeza yang dihasilkan oleh PCR masa nyata. K-min, C-min Kabur (FCM), dan C-min Kabur Alternatif (AFCM) dijalankan untuk membezakan keluaran PCR masa nyata. Hasil menunjukkan algoritma pengerumunan K-min dan FCM mampu mengklasifikan dua tindak balas berbeza PCR masa nyata. Hasil lain pula menunjukkan algoritma pengerumunan AFCM adalah lebih baik berbanding FCM dan K-min dari segi pengendalian nilai tersisih yang wujud dalam data PCR masa nyata. Aplikasi teknik pengerumunan telah memperbaiki pemprosesan maklumat *in silico* bagi kaedah baca-keluar.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

°C	-	degree celcius
T_s	-	DNA strand
S	-	DNA strand
S*	-	DNA complement of S
F	-	DNA strand
G	-	directed graph
V	-	set of vertices
e_{ij}	-	edges
V_{in}	-	start node
V_{out}	-	end node
nm	-	nanometer
kg	-	kilogram
v_{i}	-	double stranded DNA
V_i	-	node
$\lvert V \rvert$	-	number of nodes
L	-	array of location of nodes
A	-	array of aggregation values
N	-	array of Hamiltonian path node
μl	-	microliter
$\bar{v_i}$	-	reverse primer
μM	-	micro Molar
rpm	-	revolution per minute
s	-	second
J	-	cost function
U	-	partition matrix

Y	-	set of cluster centers
X	-	set of data
C	-	number of clusters
N	-	number of data
m	-	fuzziness value index
x	-	data point
y	-	cluster center
μ	-	membership value
$d\ (x,y)$	-	distance
ε	-	error
t	-	iteration step
GHz	-	Giga Herzt
GB	-	Giga Byte
η	-	scale parameter
β	-	positive constant

LIST OF ABBREVIATIONS

DNA	-	Deoxyribonucleic acid
PCR	-	Polymerase Chain Reaction
HPP	-	Hamiltonian Path Problem
A	-	Adenine
C	-	Cytosine
G	-	Guanine
T	-	Thymine
ssDNA	-	single-stranded DNA
dsDNA	-	double stranded DNA
ATP	-	Adenosine-5'-triphosphate
NAD^+	-	Nicotinamide adenine dinucleotide
PO_4^-	-	phosphate
dNTP	-	deoxynucleotide triphosphate
NP	-	Nondeterministic polynomial
RNA	-	Ribonucleic acid
PAGE	-	Polyacrylamide Gel Electrophoresis
UV	-	ultra violet
SAT	-	satisfiability problem
SA	-	simulated annealing
EA	-	Evolutionary Algorithm
ACO	-	Ant Colony Optimization
PSO	-	Particle Swarm Optimization
AFM	-	Atomic Force Microscope
DHP	-	Directed Hamiltonian Path
FCM	-	Fuzzy C-Means

AFCM	-	Alternative Fuzzy C-Means
EtBr	-	ethidium bromide.
FAM	-	6-carboxyfluorescein
TAMRA	-	tetramethylrhodamine
FRET	-	fluorescence resonance energy transfer
R	-	reporter dye and
Q	-	quencher dye
Taq	-	Thermus aquaticus
bp	-	base pairs
POA	-	Parallel Overlap Assembly
ddH2O	-	double distilled water
$MgCl_2$	-	magnesium chloride
dUTP-2'	-	deoxyuridine 5'-triphosphate
dTTP	-	deoxythymidine triphosphate
EM	-	Expectation Maximization
PCA	-	Principal Component Analysis
PCM	-	Possibilistic C-Means
TSP	-	Travelling Salesman Problem
SPP	-	Shortest Path Problem

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Deoxyribonucleic Acid (DNA)

DNA is a polymer, which is linked together from a series of monomers. Monomers, which form the structure of nucleic acids, are called nucleotides. Each nucleotide contains a sugar (deoxyribose), a phosphate group, and one of four bases: Adenine (A), Thymine (T), Guanine (G), or Cytosine (C), as shown in Figure 1.1 [1].

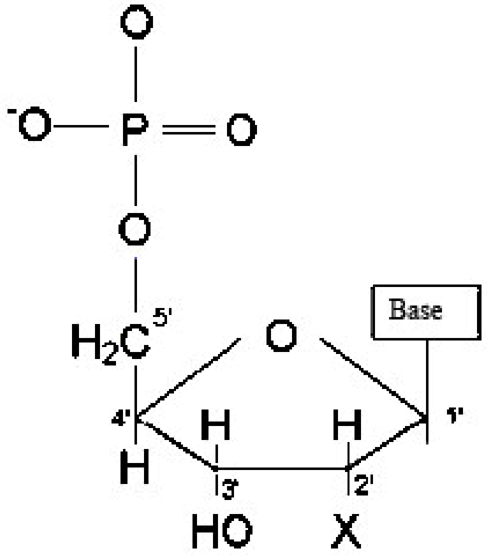

Figure 1.1 A nucleotide

Single-stranded DNA (ssDNA) is a sequence of nucleotides. This sequence, which forms a negatively charged backbone, is linked by 5’-phosphate with 3’-hydroxyl to form a phosphodiester bond, which is a strong covalent bond. Hence, each end of a single strand is easily identified by a 5’ and 3’. Figure 1.2 shows three different nucleotides that are linked to form a single-stranded DNA [1].

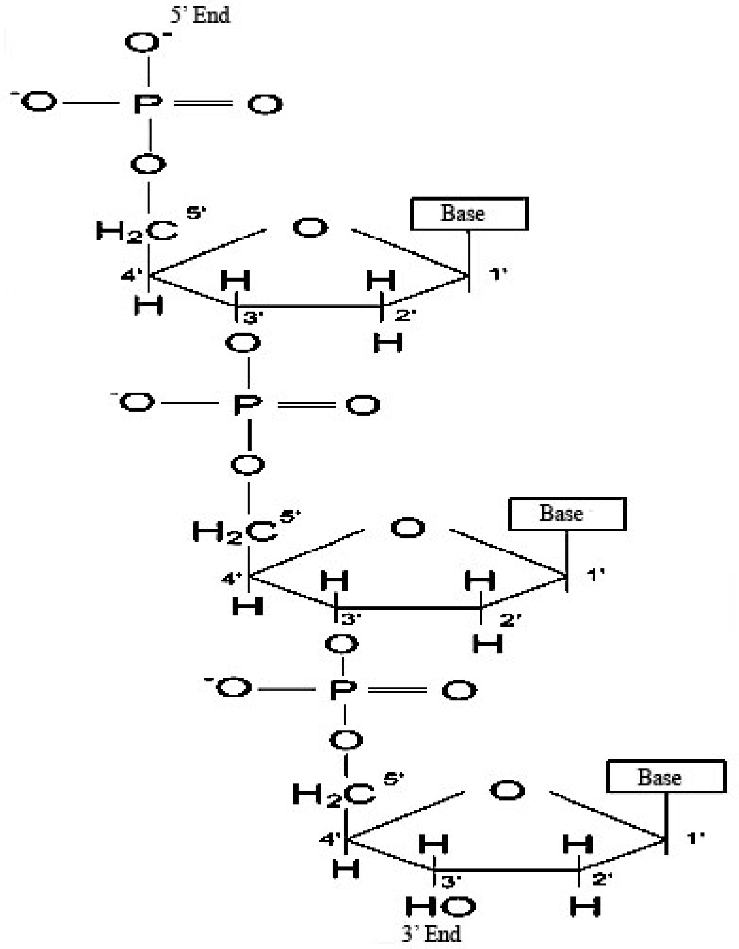

Figure 1.2 A single-stranded DNA

Figure 1.3 shows the two single-stranded DNAs, which are held together by hydrogen bonds between pairs of bases. In this figure, Adenine (A) is paired with Thymine (T) (2 hydrogen bonds) and Cytosine (C) with Guanine (G) (3 hydrogen bonds) [2]. Hybridization or annealing occurs when a sequence of nucleotides bonds to the nucleotides of another sequence, starting from the 5’-end (the ribose end) of one sequence and the 3’-end (the phosphate end) of the other sequence. These sequences are tied together in a helical structure notably known as the double helix structure [2]. The nucleotides only form stable bonds in certain combinations: A hydrogen-bonds to T, and G hydrogen-bonds to C. Thus, A is the Watson-Crick complement of T, and G is the Watson-Crick complement of C. A single-stranded of DNA sequence that contains *n* bases has length of *n*-mer.

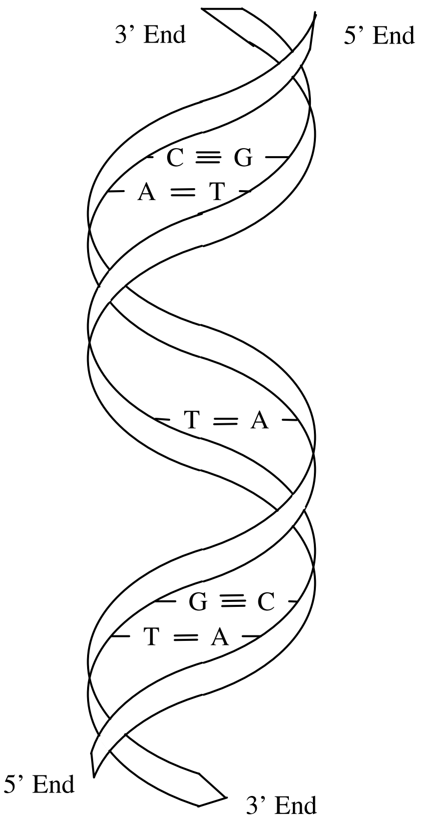

Figure 1.3 Double helix structure of DNA

1.2 Basic Biotechnology

1.2.1 Synthesizing DNA

A short single-stranded DNA is called oligonucleotide or oligo in simple term. Usually, 70-80 sequences can be chemically synthesized based on current technology, which produce less error. Lately, it is possible to get a test tube containing approximately 10^{18} DNA molecules with a desired sequence.

1.2.2 Hybridization and Denaturation

Hybridization is defined as a sequence-specific annealing of two or more single stranded DNAs, forming a double-stranded DNA (dsDNA) product. From DNA computing point of view, hybridization performs computation. Thus, the specific recognition property is very useful for the computation at molecular level. Hybridization can be done by cooling down the test tube reaction solution [3].

Three types of hybridization could occur: bi-molecular hybridization, multi-molecular hybridization, and uni-molecular hybridization. Bi-molecular hybridization involves two kinds of ssDNAs to form a double helix structure of DNA as shown in Figure 1.4 [4]. Meanwhile, three or more strands are involved in the multi-molecular hybridization. Uni-molecular hybridization or self-hybridization could lead to hairpin formation as shown in Figure 1.5. This would happen if a complementary subsequence exists in the same ssDNAs.

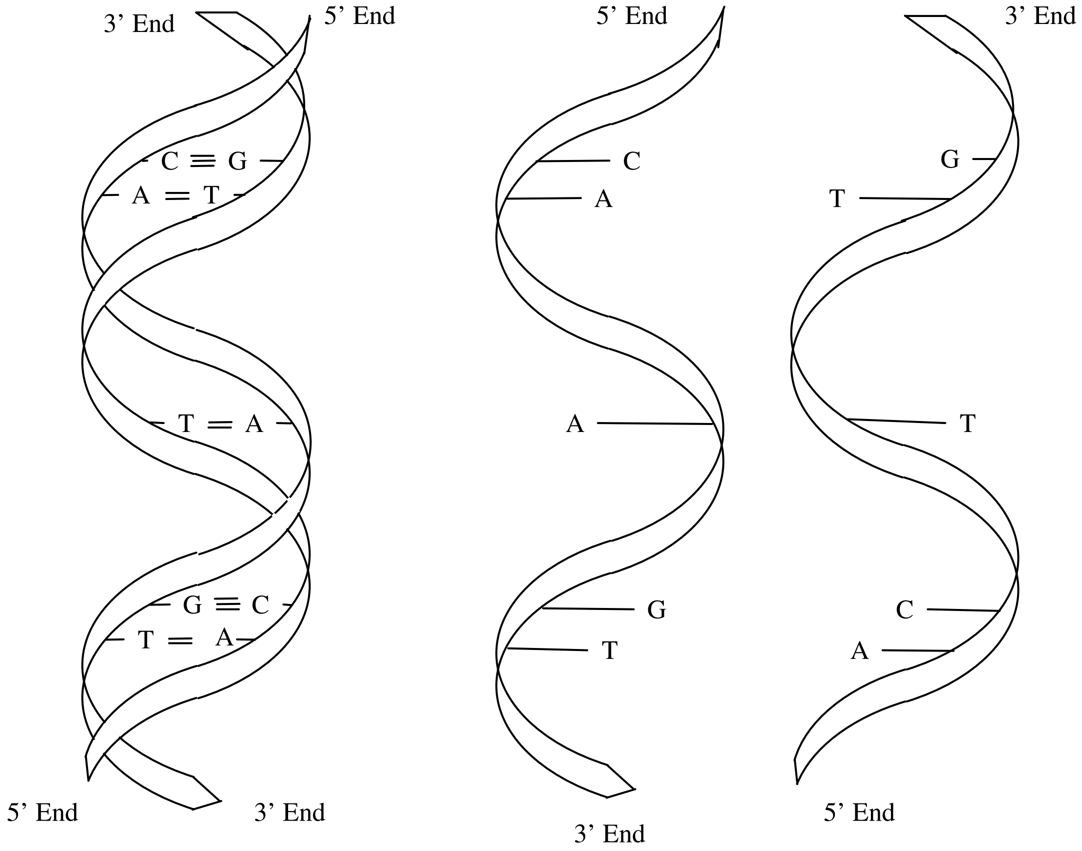

Figure 1.4 Bi-molecular hybridization and denaturation of DNA

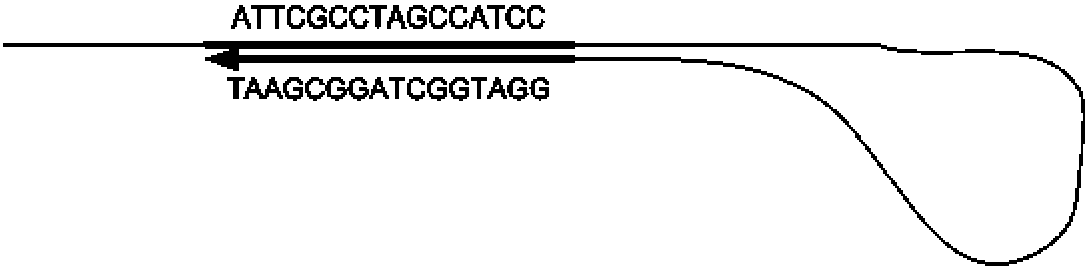

Figure 1.5 An example of hairpin formation of DNA

In denaturation, dsDNAs can be separated by heating up the solution to about 85-95°C. As shown in Figure 1.4, two strands can be separated without breaking the single strands dsDNAs as the hydrogen bonds between complementary nucleotides are much weaker than the covalent bonds between nucleotides adjacent in the two strands [5].

1.2.3 Ligation

Ligation is a process of connecting two single-strand fragments in series. A enzyme called where ligase, such as T4 DNA ligase, is used as ‘glue’ to stick the covalent bonds between the adjacent fragments [6]. The basic concept of ligation is shown in Figure 1.6. During the ligation process, strand A and strand B are placed adjacently with each other without gap and hybridized partially with strand C. The final product of ligation is a ‘new’ strand AB. In addition, strand A must have a 5’ PO_4. Usually, either Adenosine-5'-triphosphate (ATP) or Nicotinamide adenine dinucleotide (NAD^+) can be used to supply the energy in ligation.

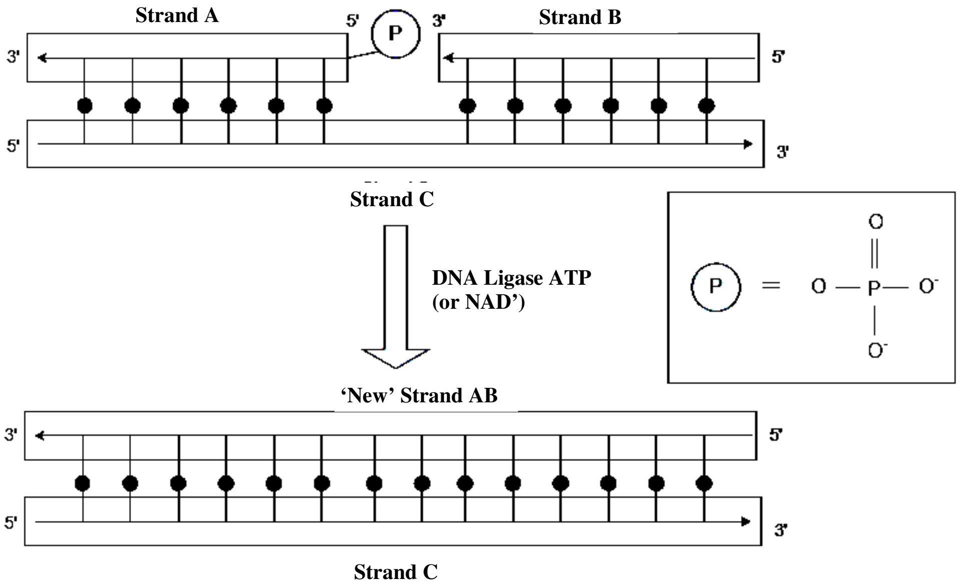

Figure 1.6 Ligation

1.2.4 Polymerization

Polymerization involves a template strand to be copied, a primer strand to be 3’-extended, and incoming deoxynucleotide triphosphate (dNTP) monomers, which act as both base and energy sources, and DNA polymerase. The polymerization process is depicted in Figure 1.7. Firstly, a primer hybridizes at a specific location on the template and initiate DNA polymerase at the particular location. After that, DNA polymerase copies the nucleotides one by one, by moving along the template DNA strand. DNA polymerase can only synthesize in the 5’ to 3’ direction. Note that there is no 3’ to 5’ copying operation ever observed [7].

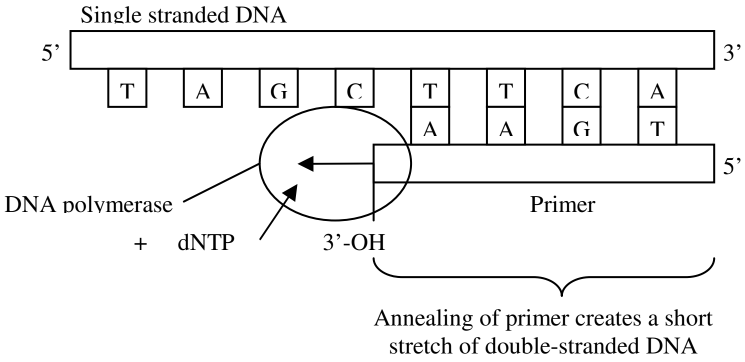

Figure 1.7 DNA polymerization

1.2.5 Polymerase Chain Reaction (PCR)

PCR is a sensitive copying machine for DNA. It also can be applied for DNA detection. A million or even billion of similar molecules can be produced by PCR process. It can produce 2^n copies of the same molecules in n steps. ‘Primers’, which are usually about 20 bases long are attached on the specific start and end site of the template for replication. PCR usually runs for 30-40 cycles of 3 phases: denaturation of DNA at about 95°C, annealing at 55°C, and extension at 74°C [8]. It takes about two to three hours normally in order to complete the cycles. Figure 1.8 shows the process of PCR up to third cycles.

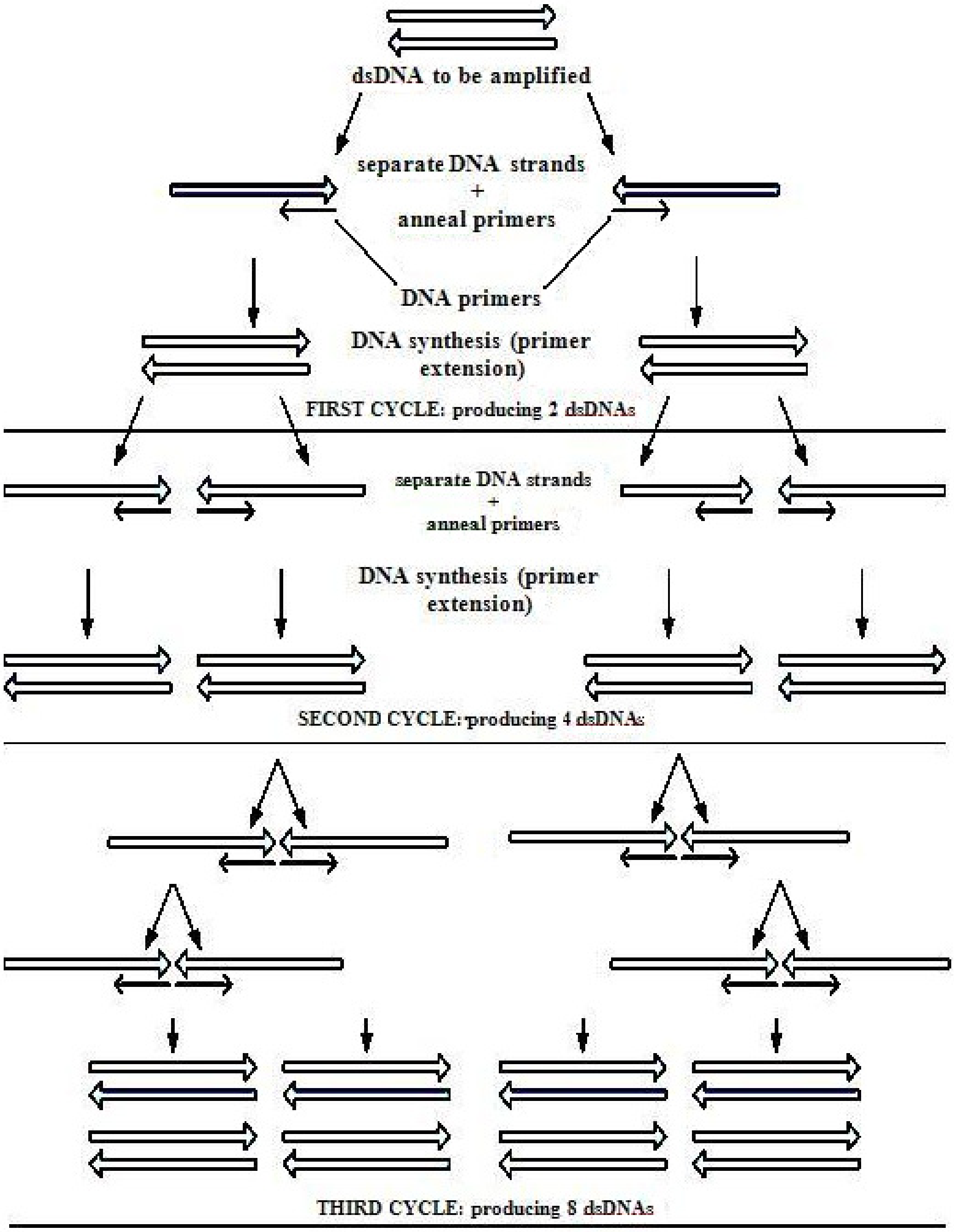
dsDNA to be amplified
separate DNA strands
+
anneal primers
DNA primers
DNA synthesis (primer extension)
FIRST CYCLE: producing 2 dsDNAs
separate DNA strands
+
anneal primers
DNA synthesis (primer extension)
SECOND CYCLE: producing 4 dsDNAs
THIRD CYCLE: producing 8 dsDNAs

Figure 1.8 Polymerase chain reaction

1.2.6 Gel Electrophoresis

DNA strands can be separated in terms of its length by means of gel electrophoresis. In fact, the molecules are separated according to their weight, which is almost proportional to their length [5]. This technique is based on the characteristic of DNA molecules, which are negatively charged [9]. DNA molecules move towards the positive electrode at different speed in the electric field. In this case, longer molecules will remain behind the shorter ones, as shown in Figure 1.9 [10]. The speed of DNA mixture in a gel depends heavily on the gel porosity and the magnitude of the electrical field. Polyacrylamide gel is used for separation of shorter dsDNAs, which range from 10 bps until 500 bps. Meanwhile, agarose gel is frequently used for longer dsDNAs, which is more than 500-bps. An example of the output of gel electrophoresis is depicted in Figure 1.10 [11]. In DNA computing, this technique is used to visualize the results of computation. Normally, at the end of this process, the gel is photographed for convenience.

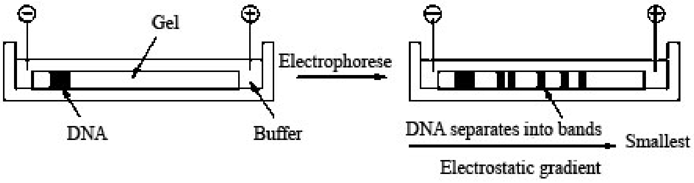

Figure 1.9 Gel electrophoresis

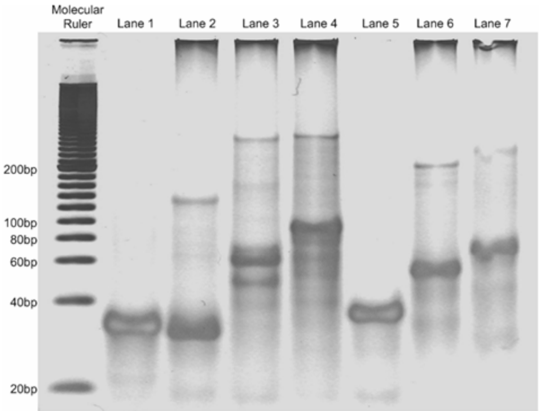

Figure 1.10 Example of a gel image

1.2.7 DNA Extraction

A ssDNA can also be isolated by sequence based on specificity of hybridization. Figure 1.11 shows an example of DNA extraction [11]. In a DNA mixture *T*, the objective of this operation is to remove the subset T_S of strands in *T* containing the subsequence *S* = AGCATA. Before the extraction, biotinylized strand, *F* with *S**, where * denotes Watson-Crick complementation, is attached to streptavidin-coated magnetic beads. Then, strand *F* is mixed with the mixture *T,* allowing strands *F* to hybridize to strands in *T* containing *S*. After the hybridization, the strands *F* can be separated magnetically, from the DNA mixture *T*. At the same time, the subset of *T*, which is hybridized with *S**, will also be removed from the DNA mixture *T*. Finally, the strand T_S can be recovered by melting or washing the strand *F* [11].

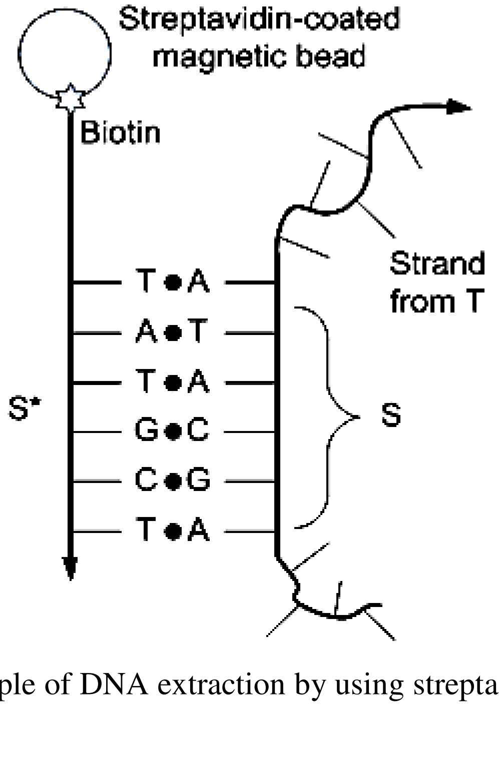

Figure 1.11 An example of DNA extraction by using streptavidin-coated magnetic bead.

1.3 DNA Computing Paradigm

1.3.1 Hamiltonian Path Problem (HPP)

Hamiltonian Path Problem (HPP) is a famous NP-complete problem, in computer science. HPP is an NP-complete problem; where there is no such efficient algorithm exist in order to solve this problem. It is a problem of directed graphs, $G = (V, E)$, which has a set of vertices, $V = \{V_\mathrm{i}\}$ and a set of 1-way directed edges, e_ij, connecting two vertices, from V_i to V_j, denoted as $(V_\mathrm{i},V_\mathrm{j}) \in V$. Furthermore, two vertices, which are start vertex, V_in, and finished vertex, V_out, are distinguished. Figure 1.12 (a) shows a simple example of directed graph for HPP, which has been selected by Adleman. This graph consists of 7 vertices, 12 edges, $V_\mathrm{in} = 0$, and $V_\mathrm{out} = 6$. The problem is to find a path between V_in and V_out through G, which passes through each vertex in V exactly once. Figure 1.12(b) shows the satisfying path, which is 0→1→2→3→4→5→6.

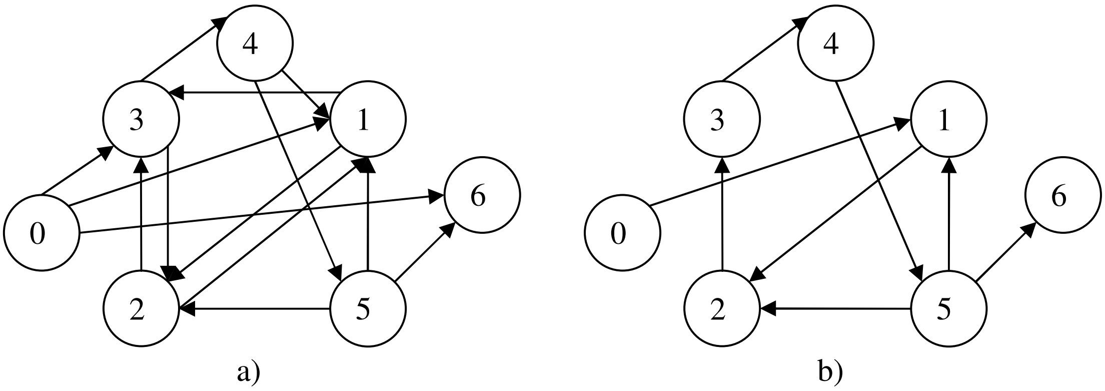

Figure 1.12 a) A directed graph for Hamiltonian path problem, b) The answer of Hamiltonian path problem.

1.3.2 From Turing Machine to DNA Computing

In 1936, Alan Turing designed the Turing Machine [12], a rule-based device that moves over a limitless tape with symbols written on it and can read, write, and rewrite these symbols. The Turing machine marks the beginning of modern computer science and represents as a universal model of computation. A decade later, John von Neumann described the architecture of the first practical programmable computer [13]. It made use of electrical implementation of Boolean logic circuits by using "0" and "1" as the absence and presence of electrical signals. Transistor stands as a basic component in modern integrated circuit, which integrated circuit is widely used in many practical programmable computers. However, in 1965, Moore [14] observed an exponential growth in the number of transistors per integrated circuit against time. This is the definition of Moore's Law, meaning that more and more transistors can be crammed into a single chip until the silicon itself reaches its limitation. From the observation, researchers have been searching for alternative medium for computation.

The notion that single molecules or atoms could be used to construct computer components was first conceived by Richard Feynman in his talk in 1959 [15]. Later scientists began to realize that natural biomolecular process within living cells, such as DNA duplication, transcription, and translation, could realize Turing machine-like information processing operations using DNA, RNA, and enzymes [16]. The concept that DNA molecules and enzymatic DNA processing could be used to store information and perform computation was then theoretically discussed by T. Head in 1987 [17] and 1992 [18]. The possibility that DNA computation could be applied to solve complex mathematical problems was demonstrated by Adleman in 1994 [19]. In that paper, he launched a novel *in vitro* approach to solve the HPP with seven vertices by DNA molecules. He encoded the information of the vertices by generating randomized DNA sequences. The computation is performed by a series of primitive bio-molecular reactions involving hybridization, denaturation, ligation, magnetic bead separation, and PCR. The output of computation, also in the form of DNA molecules can be read and "printed" by electrophoretical fluorescence method such as agarose gel electrophoresis or polyacrlamide gel electrophoresis

(PAGE).

In the first experimental of DNA computing, Adleman implement the non-deterministic algorithm for solving directed HPP shown in Figure 1.13. The algorithm consists of five steps as follows:

Step 1: Generate all paths randomly in large quantity.
Step 2: Eliminate all paths that do not begin with v_{in} and end in v_{out}.
Step 3: Eliminate all paths that do not involve exactly n vertices.
Step 4: For each of the n vertices v, eliminate all paths that do not involve v.
Step 5: The answer is ‘YES’ if any path remains, otherwise ‘NO’.

Adleman proved that this algorithm can be implemented in molecular level. Adleman used a set of 20-mer oligonucleotides, or oligos, to encode each vertex and edge, which is randomly designed in advance. To implement the Step 1 in molecular level, all the oligos representing the edges and vertices are poured in a single test tube. Then, hybridization and ligation reaction are applied to the mixture, resulting formation of DNA molecules encoding a lot of random paths of the graph. Step 2 is implemented whereby the product of Step 1 is amplified by using PCR using the oligos that encode start node and end node, respectively. As a result, all formations that begin with from V_0 and end with V_6 will be exponentially amplified. Then, gel electrophoresis is implemented to separate the amplified products in term of length. The double-stranded DNAs (dsDNAs) of 140 base-pair (bp) representing the formation of path, which starts with V_0 and ends with V_6, are excised and extracted from the gel. Next, Step 4 can be implemented by affinity-purify of the product of Step 3 with a biotin-avidin magnetic beads system for *7* times. At each time, the DNA molecules that contain subsequence node are selected and separated from the solution. Lastly, the last step can be made with the use of 260 nm ultra-violet (UV) source in order to check whether there are DNA molecules survived in the test tube after Step 1 to Step 4 are accomplished. The answer of the HPP is ‘YES’ if any DNA molecules remain, otherwise, ‘NO’. The final result of the computation was displayed on gel elctrophoresis using a technique called graduated PCR. The whole procedures of Adleman HPP base-DNA computing are depicted in Figure 1.13.

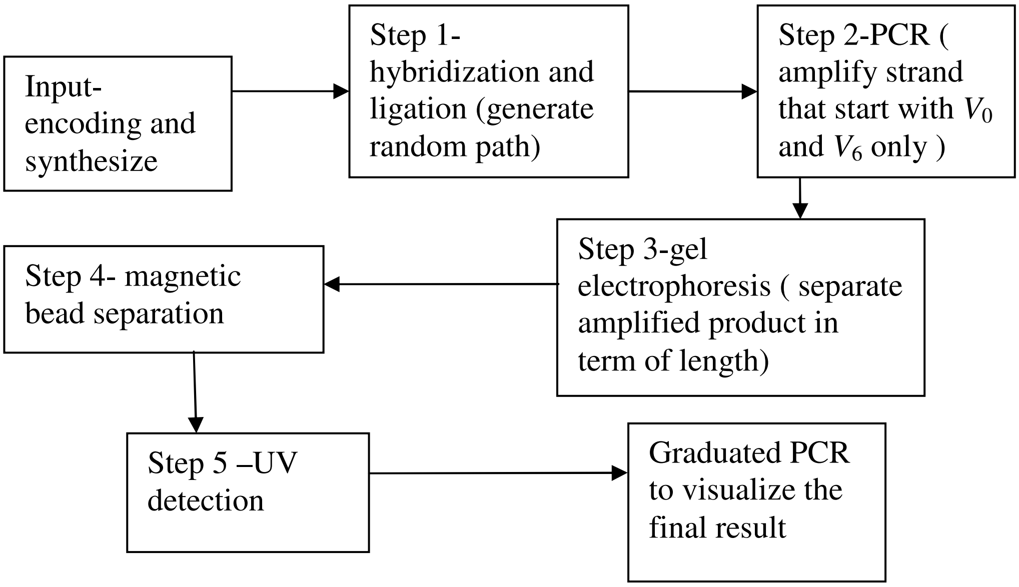
Input- encoding and synthesize
Step 1- hybridization and ligation (generate random path)
Step 2-PCR (amplify strand that start with V_0 and V_6 only)
Step 3-gel electrophoresis (separate amplified product in term of length)
Step 4- magnetic bead separation
Step 5 –UV detection
Graduated PCR to visualize the final result

Figure 1.13 The overall procedure of Adleman HPP base DNA computing.

1.4 Emergence of DNA Computing

DNA computing emerged as an attractive research, which contains the element of computer science, molecular biology, nanotechnology, and chemical engineering. The main benefit of using DNA computing to solve complex problems is the use of massive parallelism, where DNA computing is capable to solve such problems through a single parallel process. Meanwhile, silicon machines compute a problem by executing single task at once [20].

The extreme compactness of DNA as a data storage medium can be an alternative for today's memory. A mole contains 6.02 x 10^{23} DNA base monomers, and the mean molecular weight of a monomer is approximately 350 grams/mole. Hence, 1 gram of DNA comprises 2.1 x 10^{21} DNA based. In addition, 4 DNA bases can encode 2 bits, which give approximately 4.2 x 10^{21} bits in 1 gram DNA compare to conventional memory technologies capacity, which roughly 10^9 bits per gram.

Indeed, DNA has the capability of data storage which in 10^{12} times more compact than current storage technologies [21].

From the energy consumption point of view, DNA computation is expected to use very little energy [22], as DNA molecules release energy when they anneal together. Adleman noted that enzyme-based DNA computing use very low energy; where one ATP pyrophosphate cleavage per ligation provides an efficiency of roughly 2 x 10^{19} operations per joule. However, supercomputers of that time performed approximately 10^{9} operations per joule [19, 23].

Subsequent to Adleman's experiment, various models of computation have also been carried out via bio-molecular experiments. Lipton extended the Adleman DNA algorithm and proposed a mix-and-split model of DNA computing for solving satisfiability problem (SAT) for propositional formulas [24]. Later, Liu *et al.* designed and implemented a surface-based DNA computation also for SAT [25]. In addition, DNA memory as reported by Baum [26], exploits the capability of DNA effectively to perform associative search.

From the biotechnology aspect, the first practical DNA computer for gene expression has been developed by Akira Suyama [27]. Furthermore, biochemical sensing, genetic engineering, and medical diagnosis and treatment are claimed to be the future of DNA computing, based on the works carried out by Benenson *et al* on the DNA-based automata [28].

Apart from wet-lab experiments, where real DNAs is used to perform the computation, simulation of DNA computing is useful to support DNA computing algorithm design and to decrease the costs and efforts of laboratory experiments. Reliability, performance benchmarks, user interfaces, and accessibility are to be the most important criteria for the development of DNA computing simulator [29].

Peptide computing is a form of computing which uses peptides and molecular biology, based on the affinity of antibodies towards peptide sequences [30]. Similar to DNA computing, the parallel interactions of peptide sequences and antibodies have been used by this model to solve computational problems. Another important works initiated from Adleman DNA computer is a membrane computing by Gheorge Paun. Membrane computing is developed extensively from mathematical point of view, to establish a model called P systems, which is inspired from the cell biochemistry [31].

DNA computing also requires good sequences for input molecules, as errors usually occur in hybridization and annealing. Various kinds of strategies for DNA sequence design has been proposed to date. Hartemink *et al.* [32] designed sequences for the programmed mutagenesis, using exhaustive search method "SCAN". Furthermore, Tanaka *et al.* [33] generated the DNA sequence using simulated annealing (SA) based on some fitness criteria. Marathe *et al.* implemented a dynamic programming approach to design a set of DNA sequences based on Hamming distance [34]. Feldkemp *et al.* [35] used a directed graph to design DNA sequences. Evolutionary algorithm (EA) also has been implemented for optimizing DNA sequences [36-38]. Recently, swarm intelligence approaches such as ant colony optimization (ACO) and particle swarm optimization (PSO) were employed to optimize a set of DNA sequences [39-40].

Existing models of DNA computation are based on various combinations of bio-operations, which are *synthesizing*, *mixing*, *annealing* (*hybridization*), *melting* (*denaturation*), *amplifying* (*copying*), *separating*, *extracting*, *cutting*, *ligating*, *substituting*, *detecting*, and *reading* [41]. Based on this model, the DNA computation implementation can be classified by three important aspects: nucleic acid design, DNA algorithms, and readout method. The first step for wet-lab experiment of DNA computation is to find a good set of DNA sequences. After that, the desired sequences are synthesized based on the specific problem. Then, the computational part of the DNA algorithms is performed, where *mixing*, *annealing* (*hybridization*), *melting* (*denaturation*), *amplifying* (*copying*), *separating*, *extracting*, *cutting*, *ligating*, *substituting,* and *detecting* are fully applied to implement the algorithm for the computation. The final part of the implementation is visualization of the output result, where the *readout* operation can be implemented by utilizing the biotechnology, such as DNA sequencing. The readout method implementation issue is stated in [42] as an important drawback of current DNA computation, which requires the developments of high-throughput screening technologies to overcome the limitation imposed by existing readout methods. However, readout problem receive less attention from researchers, instead of computational part of DNA computing.

There are several papers dealing with readout method for DNA computation. Wang *et al* [43] described the DESTROY and READOUT operation in surface based DNA computing. In the READOUT operation, two methods were proposed for visualization of surface based DNA computing. The first method is to implement the conventional electrophoresis-based DNA sequencing. Another method proposed by Wang is the hybridization to word-specific addressed arrays. In [44], Wang *et al,* proposed a structure-specific cleavage-based readout strategy for surface-based DNA computing. The proposed method was implemented to display the solution of a 4-variable/3-satisfiability (SAT) problem. Recently, Lee *et al.* [45] implemented a gold nanoparticle aggregation for logic-based biomolecular detection and DNA computing, where the results of DNA computing process were displayed based on a color changing process induced by gold nanoparticle aggregation. For specific problem of HPP based DNA computation, Woods *et al.* [46] proposed a universal biochip for readout of multiple solutions of HPP. Meanwhile, Ibrahim *et al.* [47] implemented a TaqMan-based real-time PCR for visualizing the Hamiltonian path which encoded in double-stranded DNA sequences.

1.5 Reviews of Output Visualization Technologies in DNA Computing

1.5.1 Polymerase Chain Reaction

Since the pioneering work by Adleman in 1994, polymerase chain reaction (PCR) and gel electrophoresis are extensively used in detection and readout method for experimental DNA computing. PCR and gel electrophoresis has been utilized as a readout methodology for satisfiability problem (SAT problem) based DNA computing [48-50]. Moreover, PCR and gel electrophoresis have been used for screening the output of RNA solution of chess problem [51]. In the DNA computing playing poker by Woods [52], different lengths that indicate payoffs of each player is separated via denaturing gel electrophoresis, where the readout can be done by quantifying the amount of DNA in each band of the gel.

Adleman performed the technique so called graduated PCR, where different PCR reactions are performed that encode the ordering of the HPP [19]. Since that, graduated PCR technique for readout method in DNA computing is reported in literatures. Yoshida *et al.* [48] reported that graduated PCR was used to perform the readout operation for 3-SAT problem. Meanwhile, Braich *et al.* [49] performed several PCR amplification methods (similar to graduated PCR) to extract the strands representing the answer to the 20-variable 3-SAT problem. Graduated PCR also has been utilized in automated DNA computer for solving *n*-variable 3-SAT problem [50]. Ibrahim *et al.* used graduated PCR for visualizing output of DNA computation for the shortest path problems [53-55]. Morimoto *et al.* used graduated PCR to readout answer for solid phase method DNA computation, where the Hamiltonian paths was determined by comparing the elution time of each of the PCR reaction. The fluorescence level was then visualized on the electropherograms [56].

1.5.2 DNA Sequencing

DNA sequencing is the most straight forward method for readout computation of molecular computing. The basic of *sequencing* is to use PCR and gel electrophoresis, to return the sequence of a particular strand. As a result, the location of each base in the DNA strand can be directly read. Considering the advantage of DNA sequencing, it has been widely used in many implementation of DNA computing. For instance, Lee *et al.* used DNA sequencing method for readout operation for temperature gradient-based DNA computing, where *cloning* and *sequencing* operations are utilized to extract the shortest path of the TSP [57,58].

DNA sequencing also has been used to readout the answer for maximal clique problem [59]. Furthermore, the issue on using DNA sequencing for readout method for DNA computing application has been addressed by Mir in 1996 [60]. In another implementation of HPP based on DNA computing perform on microfluidic device, Ledesma *et al.* utilized a DNA sequencing microchip to readout the final solution obtained from the computation [61].

The basic idea of the most widely used sequencing method is to use PCR and gel electrophoresis. Assume there is a homogeneous solution, that is, a solution containing mainly copies of the strand to be sequenced with very few contaminants (other strands). To detect the positions of A's in the target strand, a blocking agent is used to prevent the templates from being extended beyond A's during PCR. As a result of this modified PCR, a population of subsequences is obtained, each corresponding to a different occurrence of A in the original strand. By separating the resultant solution using gel electrophoresis, the positions where the bases A occurs in the strand will be known. The process can then be repeated for each of C, G, and T, to yield the sequence of the strand [62].

1.5.3 Biochip

From the literature review, biochip technology has been proposed for readout method in DNA computing. For example, HPP readout by biochip hybridization has been suggested in [63], [64], and [46]. Wood [64] utilized DNA chip to visualize the output of HPP, where the Held-Karp DNA based algorithm has been used to find the Hamiltonian path. Furthermore, Wood *et al.* suggested a universal biochip for readout method, which emphasizing on reading out multiple solutions of HPP [46]. In another implementation of DNA computing, biochip readout technique has been proposed to observe the decision nodes of 3-person poker based on DNA computing [65].

1.5.4 Fluorescence Detection

Fluorescence detection is widely used in biotechnology application. Fluorescent dyes have been used together with PCR to visualize the amplification process. In addition, fluorescence is extensively used to detect the hybridization of DNA. In advanced application of DNA sequencing, four different fluorescent dyes are used, one for each base, which allows all four bases to be processed simultaneously. As the fluorescent molecules pass a detector near the bottom of the gel, signals from the detector can be sent directly to an electronic computer [62]. Moreover, fluorescent detection can also be integrated with biochip technology for better output visualization.

Fluorescence technologies have also been used in many applications of DNA computing readout method. For example, Stojanovic *et al.* exploited two different colors of fluorescence dyes that represent the output of half-adder made from DNA based logic gates [66]. In [67], the readout process for 3 bit 4 clause SAT problem based DNA computing performed on microfludic processor was done by comparing the relative flourescence of the two chambers of the microfludic processor. Ibrahim *et al.* proposed a readout method of Hamiltonian Path Problem based on real-time PCR. In this method, TaqMan fluorescence probe have been utilized for visualizing the amplification of PCR [47, 68].

1.5.5 Atomic Force Microscope

Atomic Force Microscope (AFM) [69] is one of the foremost tools for imaging, measuring, and manipulating matter at nanoscale. The advantage is that the the advantage of imaging almost any type of surface, including polymers, ceramics, composites, glass, and biological samples. In DNA computing applications, AFM has been implemented to visualize the DNA double-crossover crystals structure in

DNA computing by self-assembly [70]. In another work by Rothemund *et al.* [71], a DNA Sierpinski Triangle, which performs the XOR computation, was visualized by AFM.

1.6 Problem Statement

In general, given a double-stranded DNA sequence which contains a several subsequences, with the start and end sequences are already known, the problem is to determine the ordering of the intermediate sequences.

In the first DNA computing experiment by Adleman [19], graduated PCR have been employed to readout the answer of final DNA computation. In this case, one only knows that a Hamiltonian path begins from node 0 and ends at node 6. However, the information of the nodes that passed through is unknown.Hence, graduated PCR is used to allow one to "print" the result of the computation. Graduated PCR was performed by running six different PCR operations to the solution of seven nodes HPP. However, this method is very time consuming. As such, Ibrahim *et al.* [47,68] claimed that graduated PCR was very time consuming method.

In [46], a technique for reading out arbitrary graphs with up to *n* nodes using an *n* x *n* biochip incorporating standardized DNA sequences was proposed, which made the biochip universal for all graphs of the size. Such graph can be Directed Hamiltonian Path (DHP) in the large, with all graphs can be superimposed each other. The superposition of graphs can be diluted by detecting n^2 different quantum dot barcode labels within the spots on the universal biochip. Then, the partial readout of special class of permutation graphs is subjected to computer-based heuristics for isolating individual graphs from a collection of graphs. However, this method is not experimentally verified in the laboratory.

Ibrahim *et al* [47,68] implemented a TaqMan based real-time PCR for reading out DNA solution that encodes the Hamiltonian path. The readout method consists of *in vitro* computation and *in silico* information processing. Several TaqMan reactions were performed to investigate the order of the Hamiltonian path in the *in vitro* computation part. The output of the real-time PCR can be distinguished as either "YES" or "NO" reaction. After that, the output from the *in vitro* computation was subjected into *in silico* algorithm to produce the Hamiltonian path. However, the TaqMan "YES" and "NO" reactions are identified manually. Based on this problem, an automatic classification procedure could be employed to improve the *in silico* part of the readout procedure. In addition, the *in silico* algorithm in [47] and [68] can be further improved. The final result of the previous algorithm shows only the location of each node for Hamiltonian path, where additional steps are required to show the actual Hamiltonian path.

1.7 Objective

The objective of this research is to improve the *in silico* information processing of the readout method of DNA computer based on real-time PCR. In this research, clustering algorithms are implemented to automatically classify the "YES" and "NO" reactions.

The motivation behind this project is the output visualization of HPP, computed on a DNA computer, using real-time PCR. The real-time PCR is able to show the PCR amplification output at each cycle. Previously, graduated PCR, which was originally demonstrated by Adleman [19], was employed to perform the computation. The major problem of using graduated PCR is that the amplification process for the *in vitro* computation cannot be viewed online. DNA biochip based methodology, which makes use of biochip hybridization for the same purpose has been proposed [46]. However, this method is more costly, and has yet to be experimentally implemented.

1.8 Scope of Work

Figure 1.14 provides an overview of scope of work and contribution in this thesis. In this figure, DNA computing can be viewed as the main field in this research, however, the readout method based on real-time PCR is only applied on HPP. Particularly, the real-time PCR readout method are performed on LightCycler System and DNA Engine Opticon 2 System. Implementation based on the LightCycler System includes two different six nodes of Hamiltonian path. Meanwhile, experiments of three different seven nodes of Hamiltonian path are conducted on DNA Engine Opticon 2 System. Only five different paths of HPP are carried out in this thesis, as those paths are taken from the previous research conducted in [47] and [68]. Clustering algorithms are then implemented to both output of real-time PCR for automatic classification of TaqMan reactions. For the LightCycler System-based implementation, K-means [72] and Fuzzy C-Means (FCM) [73] clustering algorithm are employed to group the TaqMan reactions into "YES" and "NO" groups. Subsequently, FCM and Alternative FCM (AFCM) [74] are applied to the output of DNA Engine Opticon 2 System.

Figure 1.15 shows the overall process of DNA computing readout method based on real-time PCR. The first stage is the preparation of input molecules for real-time PCR experiment. Then, the *in vitro* part of the readout method based on real-time PCR are performed on LightCycley System and DNA Engine Opticon 2 System. In the *in silico* phase, clustering algorithms are applied to automatically classify the TaqMan reaction. Subsequently, the information produced from the clustering algorithm is subjected to the *in silico* algorithm for extracting the desired Hamiltonian path.

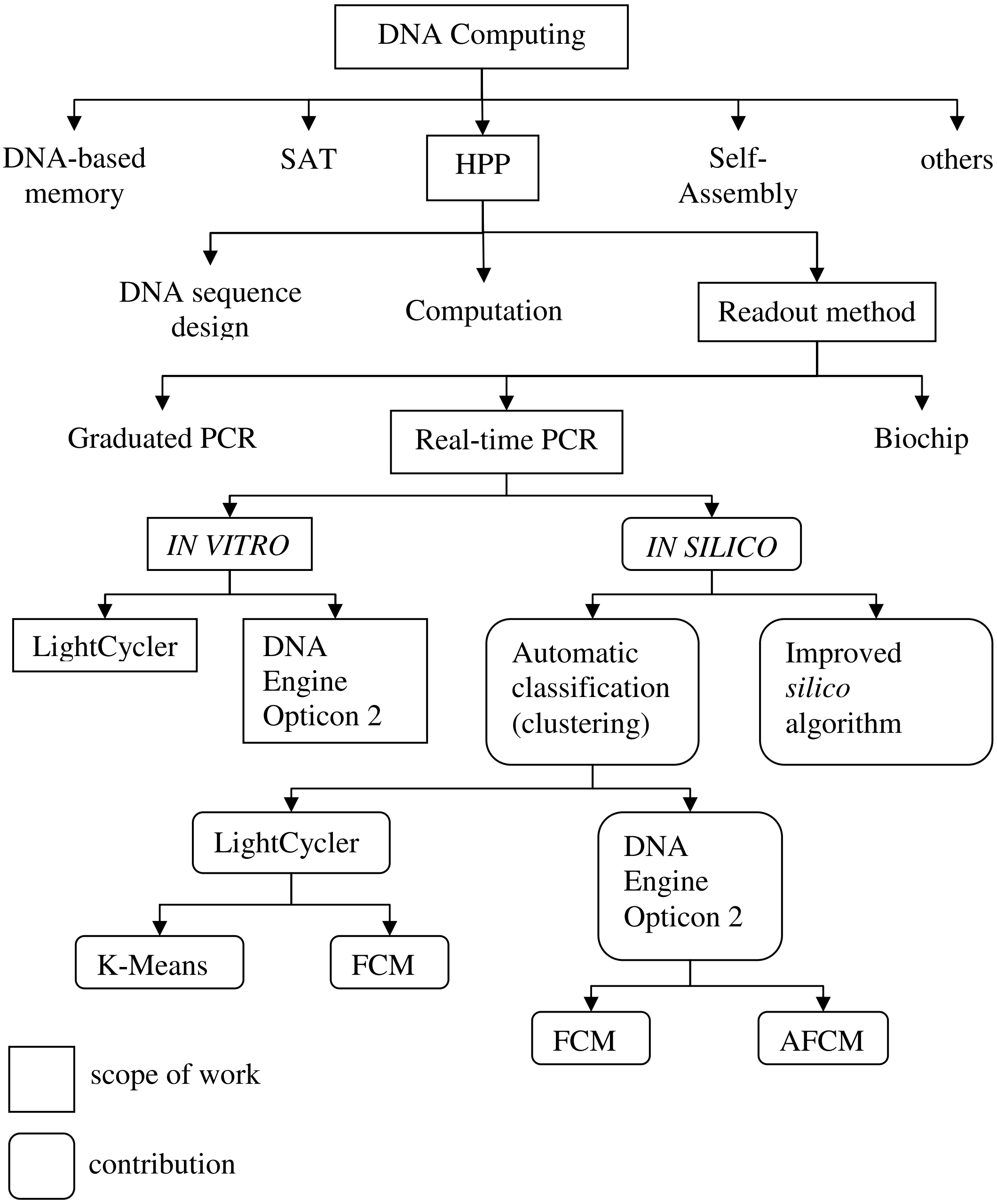

Figure 1.14 Scope of work and contributions

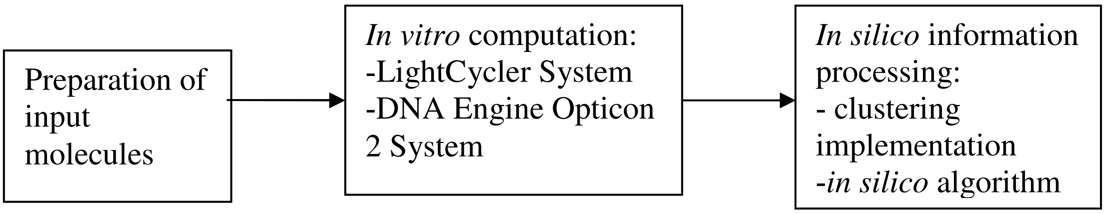
Preparation of input molecules
In vitro computation:
-LightCycler System
-DNA Engine Opticon 2 System
In silico information processing:
- clustering implementation
-in silico algorithm

Figure 1.15 The whole process of readout method based on real-time PCR

1.9 Contribution

From Figure 1.14, contributions in this thesis are highlighted in rounded box. In this research, the improvement of *in silico* information processing of the readout method is the major contribution, which can be divided into three parts. The first contribution of this thesis is the clustering implementation on real-time PCR output generated by LightCycler System. K-means and FCM are employed to classify the TaqMan reactions. The performance of two different methods are analyzed in term of consistency. Based on the consistency criteria, FCM shows better performance than the K-means algorithm.

The second contribution of this thesis is the FCM clustering, implemented to the output of the DNA Engine Opticon 2 System. However, misclassification could occurr, due to the nature of the data produced from the DNA Engine Opticon 2 System. Noise or an outlier is figured out as the main problem of the clustering process. AFCM, which the improve version of FCM, is implemented to the same data to overcome the noise or outlier problems.

A minor contribution or the last part of the contribution is the modified *in silico* algorithm, which directly display the desired Hamiltonian path. As discussed in the earlier section, the previous algorithm only shows the location of each nodes of the Hamiltonian path. Practically, the *in silico* algorithm can be programmed in the computerized application, where the binary input consist of "YES" and "NO" are processed to computed the actual order of Hamiltonian path. Base on the modified algorithm, the Hamiltonian path can be directly viewed for convenience.

1.10 Publication List

This thesis contains materials from several conference publications and a journal article. Some of the text and figures in this thesis come directly from those articles, although most of it has undergone revision, and occasionally correction, for incorporation into this thesis.

Chapter 1 is based on

Saaid, M. F. M.**,** Ibrahim, Z., Khalid, M. and Sarmin, N. H. DNA Computing Readout Approaches: A Review. *The Second International Conference on Control, Instrumentation and Mechatronic Engineering (CIM09).* June 2-3, 2009. Malacca, Malaysia: 2009. (accepted)

Chapter 2 is based on:

Ibrahim Z., Rose, J. A., Tsuboi, Y., Ono, O. and Khalid, M. A New Readout Approach in DNA Computing Based on Real-Time PCR with TaqMan Probes. In: Mao, C. and Yokomori, T. ed. *Lecture Notes in Computer Science (LNCS).* Springer-Verlag. 4287: 350-359; 2006.

Ibrahim, Z., Rose, J. A., Suyama, A. and Khalid, M. Experimental Implementation and Analysis of a DNA Computing Readout Method Based on Real-Time PCR with TaqMan Probes. *Natural Computing Journal*, *Springer*, 2008. 7(2): 277-286.

Saaid, M. F. M., Ibrahim, Z. and Sarmin, N. H. An Improved *In Silico* Algorithm for Output Visualization of DNA Computing based on Real-Time PCR. *Asia Modelling Symposium*, AMS 2008, *Second Asia International Conference on Modelling & Simulation*. May 13-15, 2008. Kuala Lumpur, Malaysia: IEEE. 2008. 879-884.

In this chapter, the explanations of readout method based on real-time PCR are takes solely from those materials above.

Chapter 3 is based on:

Saaid, M. F. M., Ibrahim, Z., Khalid, M., Sarmin, N. H. and Rose, J. A. K-Means Clustering for DNA Computing Readout Method Implemented on LightCycler System**.** *3rd Southeast Asia Technical University Consortium (SEATUC).* February 25-26*,* 2009. Universiti Teknologi Malaysia, Malaysia. 2009. (accepted)

Saaid, M. F. M., Ibrahim, Z., Khalid, M., Sarmin, N. H. and Rose., J. A. Fuzzy C-Means Clustering for DNA Computing Readout Method Implemented on LightCycler System. *International Conference on Instrumentation, Control and Technology*, *SICE 2008*. August 20-22, 2008. University of Electro-Communications, Chofu City, Tokyo, Japan: IEEE. 2008. 676-681.

Chapter 4 is based on:

Saaid, M. F. M., Ibrahim, Z., Khalid, M. and Yahya, A. Alternative Fuzzy C-Means Clustering for DNA Computing Readout Method Implemented on DNA Engine Opticon 2 System. *The Fourth International Conference on Signal-Image Technology & Internet–based Systems (SITIS 2008).* November 30-4, 2008. Bali, Indonesia: IEEE. 2008. 498-503.

1.11 Thesis Organization

This thesis is organized as follows. Chapter 2 provides detailed explanations of DNA computing readout method for HPP based on real-time PCR. After that, Chapter 3 and Chapter 4 discuss the clustering implementation on LightCycler System output and DNA Engine Opticon 2 System output, respectively. Finally, Chapter 5 ends this thesis with conclusions as well as some research directions based on this research. Finally, the references are placed at the back of this thesis, with additional appendices.

REFERENCES

1. Hames, L. and Hooper, N. M. *Biochemistry.* 3rd. ed. Taylor and Francis. 2005
2. Reece, R. J. *Analysis of Genes and Genomes*. Wiley. 2004
3. Ausubel, F. andStruhl, K. *Short Protocol in Molecular Biology: A Compendium of Methods from Current Protocols in Molecular Biology*. 3rd. ed. Wiley & Sons. 1995
4. Passarge, E. *Color Atlas of Genetics.* 3rd. ed. Thieme. 2007
5. Calude, C. S. and Paun, G. *Computing with cells and atoms - An introduction to quantum, DNA, and membrane computing*. New York: Taylor & Francis Inc. 2001
6. Zucca, M. *DNA based Computational Models*, Ph.D. Thesis. Politecnico Di Torino, Italy; 2000
7. Velden, F. H. P. V. *Biomolecular Computing and Their Simulations*. Master Thesis. University of Amsterdam, The Netherlands; 2005
8. Fitch, J. P. *An Engineering Introduction to Biotechnology*. SPIE. 2002
9. Paun, G., Rozenberg, G. and Salooma, A. *DNA Computing: New Computing Paradigms.* New York ;Springer. 1998
10. Amos, M. *DNA computation*. Ph.D. Thesis. The University of Warwick, UK; 1997
11. Ibrahim, Z. *Concentration-Controlled Length-based DNA computing for Weighted Graph Problems with Novel Readout Approach using Real-Time PCR*. Ph.D. Thesis. Meiji University; 2006
12. Turing, L. M. On Computable Numbers, With An Application to the Entcheidungs Problem. *Proc. Lond. Math. Soc*, 1936. 42: 230-265.
13. Neumann, J. V. First draft of a report on EDVAC. 1945.

14. Moore, G. E. Craming More Components onto Integrated Circuits. *Electronics*, 1965. 38(8).

15. R. P. Feynman, R. P. There's Plenty of Room at the Bottom. In: Gilbert, D. H. ed. *Minaturization*. New York: Reinhold Publishing Corporation. 282-296; 1961

16. C. H. Bennet, C. H. The Thermodynamics of Computation- Review. *Int. J. Theoret. Phys.* 1982. 21: 905-940.

17. Head, T. Formal Language Theory and DNA: An Analysis of the Generative Capacity of Special Recombinant Behaviors. *Bull. Math. Biol.* 1987. 49; 737-759.

18. Head, T. Splicing Systems and DNA. In: *Handbook of Formal Language.* Berlin: Springer-Verlag. 371-383; 1992.

19. Adleman, L. Molecular Computation of Solutions to Combinatorial Problems. *Science.* 1994. 266: 1021-1024.

20. Ito, Y and Fukusaki, E. DNA as a Nanomaterial. *J. Mol. Catal. B: Enzymatic*. 2004. 28: 155-166.

21. Reif, J. H., LaBean, T. H., Pirrung, M., Rana, V. S., Guo, B., Kingsford, C. and Wickham, G. S. Experimental Construction of Very Large Scale DNA Databases with Associative Search Capability. *Proc. a DIMACS Workshop: DNA Based Computers*. 2001. 231-247.

22. Maley, C. C. DNA Computation: Theory, Practice and Prospects. *Eval. Comput.* 1998. 6(3): 201-230.

23. Fu, P. Biomolecular Computing; Is it ready to take off?. *Biotechnology Journal.* 2007. 2(1): 91-101.

24. Lipton, R. J. DNA Solution of Hard Computational Problems. *Science*. 1995. 268: 1021-1023.

25. Liu, Q., Frutos, A. G., Wang, L., Condon, A. E., Corn, R. M. and Smith, L. M. DNA Computing on Surfaces. *Nature*. 2000. 403: 175-179.

26. Baum, E. B. Building an Associative Memory Vastly Larger Than the Brain. *Science*. 1995. 268: 583-585.

27. Normile, D. Molecular Computing: DNA-based Computer Takes Aim at Genes. *Science*. 2002. 295: 951.

28. Benenson, Y., Gil, Y. B., Ben-Dor, U., Adar, R. and Shapiro, E. An Autonomous Molecular Computer for Logical Control of Gene Expression. *Nature*. 2004. 429: 423-429.

29. Blain, D., Garzon, M., Shin, S. Y., Zhang, B. T., Kashiwamura, S., Yamamoto, M., Kameda, A. and Ohuchi, A. Development, Evaluation and Benchmarking of Simulation Software for Biomolecule-based Computing. *Natural Computing, Kluware Academic Publishers*. 2004. 3: 427-442.

30. Balan, M. S. and Jurgensen, H. On the Universality of Peptide Computing. *Natural Computing, Springer Netherlands*. 2008. 7(1): 71-94.

31. Paun, G. From Cells to Computers: Membrane Computing- A Quick Overview. *Lecture Notes in Computer Science*, *Springer Berlin/ Heidelberg*. 2005. 3384: 268-280.

32. Hartemink, A. J., Gifford, D. K. and Khodor, J. Automated Constraint Based Nucleotide Sequence Selection for DNA Computation. *Proc. 4th DIMACS Workshop DNA Based Comput*. 1998. 227-235.

33. Tanaka, F., Nakatsugawa, M., Yamamoto, M., Shiba, T. and Ohuchi, A. Developing Support System for Sequence Design in DNA Computing. *Proc. 7th Int. Workshop DNA Based Comput.* 2001. 340-349.

34. Marathe, A., Condon, A. E. and Corn, R. M. On Combinatorial DNA Word Design. *Proceedigs of the 5th International Meeteing on DNA Based Computers.* 1999.

35. Feldkamp, U., Saghafi, S., Banzhaf, W. and Rauhe, H. DNA Sequence Generator-A Program for the Construction of DNA Sequences. *Proc. 7th Int. workshop DNA Based Comput.*. 2001. 179-188.

36. Deaton, R., Murphy, R. C., Rose, J. A., Garzon, M., Franceschetti, D. T. and Stevens Jr., S. E. Genetic Search for Reliable Encodings for DNA-based Computation. *First Conference on Genetic Programming.* 1996.

37. Arita, M., Nishikawa, A., Hagiya, M., Komiya, K., Gouzu, H. and Sakamoto, K. Improving Sequence Design for DNA Computing. *Proc. Genetic Evol. Comput. Conf. (GECCO).* 2000. 875-882.

38. Shin, S. Y., Lee, I. H., Kim, D. and Zhang, B. T. Multiobjective Evolutionary Optimization of DNA Sequences for Reliable DNA Computing. *IEEE Transaction on Evolutionary Computation*. 2005. 9(2): 143-158.

39. Kurniawan, T. B., Khalid, N. K., Ibrahim, Z., Khalid, M. and Middendorf, M. Evaluation of Ordering Methods for DNA Sequence Design Based on Ant Colony System. *Second International Second Asia International Conference on Modelling & Simulation, AMS 2008*. May 13-15, 2008. Kuala Lumpur, Malaysia: 2008. 905-910.

40. Khalid, N. K., Kurniawan, T. B., Ibrahim, Z., Yusof, Z. M., Khalid, M. and Engelbrecht, A. P. A Model to Optimize DNA Sequences Based on Particle Swarm Optimization. *Second International Second Asia International Conference on Modelling & Simulation, AMS 2008*. May 13-15, 2008. Kuala Lumpur, Malaysia. 2008. 534-539.

41. Kari, L. DNA computing in vitro and in vivo. *Future Generation Computer System.* 2001. 17: 823-834.

42. Henkel, C.V. *Experimental DNA computing*. Ph.D. Thesis, Leiden University; 2005

43. Wang, L., Liu, Q., Frutos, A. G., Gillmor, S. D., Theil, A. J., Strother, T. C., Condon, A. E., Corn, R. M., Lagally, M. G. and Smith, L. M. Surface-based DNA Computing Operations: DESTROY and READOUT. *BioSystems*. 1999. 52: 181-191.

44. Wang, L. M., Hall, J. G., Lu, M. C., Liu, Q. H. and Smith, L. M. A DNA Computing Readout Operation based on Structure-specific Cleavage. *Nat. Biotechnol*. 2001. 19: 1053-1059.

45. Lee, I. H., Yang, K. A., Lee, J. H., Park, J. Y., Lee, J. H. and Zhang, B. T. The use of Gold Nanoparticle Aggregation for DNA Computing and Logic-based Biomolecular Detection. *Nanotechnology*. 2008. 19.

46. Wood, D. H., Clelland, C. L. T. and Bancroft, C. Universal Biochip Readout of Directed Hamiltonian Path Problems. *Lecture Notes in Computer Science*. 1999. 2568: 168-181.

47. Ibrahim, Z., Rose, J. A., Suyama, A. and Khalid, M. Experimental Implementation and Analysis of a DNA Computing Readout Method Based on Real-Time PCR with TaqMan Probes. *Natural Computing Journal*, *Springer*, 2008. 7(2): 277-286.

48. Yoshida, H. and Suyama, A. Solution to 3-SAT by Breadth First Search. In. Winfree, E. and Gifforrd, D. K. ed. *DNA Based Computers*. V. American Mathematical Society, Providence, RI. 9-22; 2000

49. Braich, R. S., Chelyapov, Johnson, N. C., Rothermund, P. W. K. and Adleman, L. Solution to a 20-Variable 3-SAT problem on a DNA Computer. *Science.* 2002. 296: 499-502.

50. Johnson, C. R. Automating the DNA Computer: solving n-Variable 3-SAT Problems. *Natural Computing, Springer Netherlands*. 2002. 239-253.

51. Faulhammer, D., Cukras, A. R., Lipton, R. J. and Landweber, L. F. Molecular Computation: RNA Solutions to Chess Problems. *Proc. Natl. Acad. Sci. USA*. 2000. 91: 1385–1389.

52. Wood, D. H., Bi, H., Kimbrough, S. O., Wu, D. and Chen, J. DNA Starts to Learn Poker. In: Jonoska and Seeman. Springer. 92–103; 2002

53. Ibrahim, Z., Tsuboi, Y., Ono, O. and Khalid, M. Molecular Computation Approach to Compete Dijkstra's Algorithm. *The 5th Asian Control Conference (ASCC2004)*. July 20-23, 2004. Melbourne, Australia: 2004. 634-641.

54. Ibrahim, Z. Tsuboi, Y., Ono, O. and Khalid, M. Direct-Proportional Length-Based DNA Computing for Shortest Path Problem. *International Journal of Computer Science and applications (IJCSA). Technomathematics Research Foundation.* 2004. 1(1): 46-40.

55. Ibrahim, Z., Tsuboi, Y., Ono, O. and Khalid, M. *In vitro* Implemntation of *k*-Shortest Paths Computation with Graduated PCR. *International Journal of Computational Intelligence Research.* 2005. 1(2): 127-137.

56. Morimoto, N., Arita, M. and Suyama, A. Solid Phase DNA Solution to the Hamiltonian Path Problem. *Proceedings of the 3rd DIMACS Workshop on DNA Based Computers*. June 1997. University of Pennsylvania: 83–92.

57. Lee, J. Y., Shin, S. Y., Augh, S. J., Park, T. H. and Zhang, B. T. Temperature Gradient based DNA Computing for Graph Problems with Weighted Edges, *Lecture Notes in Computer Science*. 2003. 2568: 73-84.

58. Lee, J. Y. Shin, S. Y., Park, T. H.and Zhang, B. T. Solving Traveling Salesman Problems with DNA Molecules Encoding Numerical Values. *BioSystems*. 2004. 78(1-3): pp. 39-47.

59. Ouyang, Q., Kaplan, P. D., Liu, S.M. and Lichaber, A. DNA Solution of the Maximal Clique Problem. *Science.* 1997. 278: 446-449.

60. Mir, K. U., A Restricted Genetic Alphabet for DNA Computing. *2nd DIMACS workshop on DNA based computers.* Princeton University: 1996. 128-130.

61. Ledesma, L., Pazos, J. and Rodrıguez-Paton, A. A DNA Algorithm for the Hamiltonian Path Problem Using Microfluidic Systems. In: Jonoska, N. Paun, G. and Rozenberg, G. ed. *Aspects of Molecular Computing - Essays dedicated to Tom Head on the occasion of his 70th birthday.* LNCS Springer-Verlag. 2950. 289–296; 2004

62. Kari, L. DNA Computing: Arrival of Biological Mathematics. In: *The Mathematical Intelligencer*. Berlin, Springer. 19(2). 9–22; 1997

63. Rose, J. A., Deaton, R., Garzon, M., Murphy, R. C., Franceschetti, D. R. and Stevens, Jr. S.E. The Effect of Uniform Melting Temperatures on The Efficiency of DNA Computing. *DNA Based Computers II:DIMACS Workshop*, June 23-25, 1997. 35-42.

64. Wood, D. H. A DNA Computing Algorithm for Directed Hamiltonian Paths. *Proceedings of the Third Annual Conference on Genetic Programming*. 1998. 731-734.

65. Wood, D. H. DNA Computing Capabilities for Game Theory. *Natural Computing. Springer Netherlands.* 2003. 2(1): 85-108.

66. Stojanovic, M. N. and Stefanovic, D. Deoxyribozyme-based Half-Adder. *J. Am. Chem. Soc.* 2003. 125(22). 6673–6676.

67. Grover, W. H. and Mathies, R. A. An Integrated Microfluidic Processor for Single Nucleotide Polymorphism-based DNA Computing. *Lab on a Chip*. 2005. 5.

68. Ibrahim Z., Rose, J. A., Tsuboi, Y., Ono, O. and Khalid, M. A New Readout Approach in DNA Computing Based on Real-Time PCR with TaqMan Probes. In: Mao, C. and Yokomori, T. ed. *Lecture Notes in Computer Science (LNCS).* Springer-Verlag. 4287: 350-359; 2006.

69. Binnig, G., Quate, C. F. and Gerber, C. Atomic Force Microscope. *Physical Review Letters.* 1986. 56(9): 930–933.

70. Winfree, E., Liu, F., Wenzler, L. A. and Seeman, N. C. Design and Self-Assembly of Two-Diemsional DNA Crystals. *Nature*. 1998. 394. 539-544.

71. Rothemund, P. W. K., Papadakis, N. and Winfree, E. Algorithmic Self-Assembly of DNA Sierpinski Triangles. *PLoS Biol.* 2004. 2(12) e424: 2041-2053.

72. MacQueen, J. B. Some Methods for classification and Analysis of Multivariate Observations. *Proceedings of 5-th Berkeley Symposium on Mathematical*

Statistics and Probability. Berkeley, University of California: Press, 1. 1967. 281-297.

73. Bezdek, J. *Pattern Recognition with Fuzzy Objective Function Algorithms.* New York: Plenum Press. 1981.
74. Wu, K. L. and Yang, M. S. Alternative C-Means Clustering Algorithm. *Pattern Recognition,* 2000. 35: 2267-2278.
75. Mullis, K., Faloona, F., Scharf, S., Saiki, R., Horn, G. and Erlich, H. Specific Enzymatic Amplification of DNA *In Vitro*: The Polymerase Chain Reaction. *Cold Spring Harbor Symposium on Quantitative Biology*. 1986. 51: 263-273.
76. Higuchi R, Fockler C, Dollinger G, and Watson R. Kinetic PCR Analysis: Real-Time Monitoring of DNA Amplification Reactions. *Biotechnology.* 1993. 11: 1026–1030.
77. Monis, P. T., Giglio, S. and Saint, C. P. Comparison of SYTO9 and SYBR Green I for Real-Time Polymerase Chain Reaction and Investigation of The Effect of Dye Concentration on Amplification and DNA Melting Curve Analysis. *Analytical Biochemistry*. 2004. 340: 24-34.
78. Espy, M. J., Uhl, J. R., Sloan, L. M.; Buckwalter, S. P., Jones, M. F., Vetter, E. A., Yao, J. D. C., Wengenack, N. L., Rosenblatt, J. E., Cockerill, F. R., and Smith, T. F.. Real-Time PCR in Clinical Microbiology: Applications for Routine Laboratory Testing. *Clinical Microbiology Reviews.* 2006. 19(1): 165-256.
79. Walker, N. J. A Technique Whose Time Has Come. *Science.* 2002. 296: 557-559.
80. Wittwer, C. T., Herrman, M. G., Moss, A. A. and Rasmussen, R. P. Continuous Fluorescence Monitoring of Rapid Cycle DNA Amplification. *BioTechniques*. 1997. 22(1): 130-139.
81. Eckert, C., Landt, O., Taube, T., Seeger, K., Beyermann, B., Proba, J. and Henze, G. Potential of LightCycler Technology for Quantification of Minimal Residual Disease in Childhood Acute Lymphoblastic Leukemia. *Leukemia*. 2000. 14: 316-323.
82. Tyagi, S. and Kramer, E. R. Molecular Beacons: Probes that Fluoresce Upon Hybridization. *Nat. Biotechnol.*. 1996. 14: 303-308.
83. Tyagi, S., Bratu, D. and Kramer, E. R. Multicolor Molecular Beacons for Allele Discrimination. *Nat. Biotechnol.,* 1998. 16: 49-53.

84. Lakowicz, J. R. *Principles of fluorescence spectroscopy*, 2nd. ed. Kluwer Academic/Plenum Publishers: New York. 1999.

85. Heid, C. A. Real-Time Quantitative PCR. *Genome Research.* 1996. 6: 986-994.

86. Holland, P. M. Detection of Specific Polymerase Chain Reaction Product by Utilizing the 5'→3' Exonuclease Activity of Termus Aquaticus DNA Polymerase. *Proceedings of the National Academy of Sciences of the United States of America.* 1991. 88: 7276-7280.

87. Overbergh, L., Giulietti, A., Valckx, D., Decallonne, B., Bouillon, R. and Mathieu, C. The Use of Real-Time Reverse Transcriptase PCR for the Quantification of Cytokine Gene Expression. *Journal of Biomolecular Techniques*. 2003. 14: 33-43.

88. Li, A., Forestier, E., Rosenquist, R. and Roos, G. Minimal Residual Disease Quantification in Childhood Acute Lymphoblastic Leukemia by Real-Time Polymerase Chain Reaction using the SYBR Green Dye. *Experimental Hematolology*. 2002, 30: 1170-1177.

89. Bernard, P. S. and Wittwer, C. T. Real-Time PCR technology for cancer diagnostics, Clinical Chemistry, Vol. 48, 2002, pp. 1178-1185

90. Norton, D. M. Polymerase Chain Reaction-based Methods for Detection of Listeria Monocytogenes: Toward Real-Time Screening for Food and Environmental Samples. *Journal of AOAC International*, 2002. 85: 505-515.

91. Niesters, H. G. Quantitation of Viral Load Using Real-Time Amplification Techniques. *Methods.* 2001. 25: 419-429.

92. Ahmed, F. E. Detection of Genetically Modified Organisms in Foods. *Trends in Biotechnology*. 2002. 20: 215-223.

93. Sevall, J. S. Rapid Allelic Discrimination from Real-Time DNA Amplification. *Methods*. 2001. 25: 452-455.

94. Jain, K., Murty, M. N. and Flynn, P. J. Data Clustering: A review. *ACM Computing Surveys*, 1999. 31(3): 265-323.

95. Everitt, B. S., Landau, S. and Leese, M. *Cluster Analysis.* London: Arnold. 2001.

96. Aldridge, M. Clustering: An Overview. In: Berry, M. W. and Browne, M. *Lecture Notes in Data Mining.* Singapore: World Scientific. 99-107; 2006.

97. Jiang, D. Tang, C. and Zhang, A. Cluster Analysis for Gene Expression Data: A Survey. *IEEE Transactions on Knowledge and Data Engineering,* 2004. 16(11): 1370-1386.

98. Han, Jiawei., Kamber, and Micheline. *Data Mining: Concepts and Techniques*. Morgan Kaufmann. 2000

99. King, B. Step-wise Clustering Procedures. *Journal of the American Statistical Association*, 1967. 69: 89-101.

100. Dempster, A. P., Laird, N. M. and Rubin, D. B. Maximum Likelihood from Incomplete Data via the EM Algorithm. *J. Royal Stat. Soc. B.*, 1977. 39(1): 1–38.

101. Lu, S. Y. and Fu, K. S. A Sentence-to-Sentence Clustering Procedure for Pattern Analysis. *IEEE Trans. Systems, Man, and Cybernetics.* 1978. 8: 381-389.

102. Kohonen, T. *Self-Organization and Associative Memory*. 3rd ed. Springer information sciences series. NY: Springer-Verlag: New York. 1989

103. Raghavan, V. V. and Birchand, K. A Clustering Strategy based on a Formalism of the Reproductive Process in a Natural System. *Proceedings of the Second International Conference on Information Storage and Retrieval.* 1979. 10–22.

104. Klein, R. W. and Dubes, R. C. Experiments in Projection and Clustering by Simulated Annealing. *Pattern Recogn.,* 1989. 22: 213–220.

105. Lumer, E. and Faieta, B. Diversity and Adaptation in Populations of Clustering Ants. *Proceedings Third International Conference on Simulation of Adaptive Behavior: from animals to animates 3.* Cambridge, Massachusetts: MIT press. 1994. 499-508.

106. Omran, M., Salman, A. and Engelbrecht, A. P. Image Classification using Particle Swarm Optimization. *Conference on Simulated Evolution and Learning*. 2002, 1: 370-374.

107. Dunn, J. C. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics.* 1973. 3: 32-57.

108. Xu, R. and Wunsch, D. I. I. Survey of clustering algorithms. *IEEE Transactions on Neural Networks*. 2005. 16(3): 645-678.

109. Jackson, J. E. *A User's Guide to Principal Components*, Wiley Series on Probability and Statistics, New York: John Wiley and Sons. 1991

110. Hathaway, R., Bezdek, J. and Hu, Y. Generalized Fuzzy C-Means Clustering Strategies using L_p norm Distances, *IEEE Trans. Fuzzy Syst.*, 2000. 8(5): 576–582.

111. Krishnapuram, R. and Keller, J. M. A Possibilistic Approach to Clustering, *IEEE Trans. Fuzzy Syst.,* 1993. 1(2): 98-110.

112. Dave, R. N. Characterization and Detection of Noise in Clustering, *Patt. Rec. Letter,* 1991. 12: 657-664.

113. Banerjee, A. and Dave, R. N. The Fuzzy Mega-cluster: Robustifying FCM by Scaling Down Memberships. *Lecture Notes in Computer Science*, 2005. 3613: 444-453.