A MODIFIED COMPUTATIONAL MODEL OF ANT COLONY SYSTEM IN DNA SEQUENCE DESIGN

SERI MASTURA MUSTAZA

UNIVERSITI TEKNOLOGI MALAYSIA

A MODIFIED COMPUTATIONAL MODEL OF ANT COLONY SYSTEM IN DNA SEQUENCE DESIGN

SERI MASTURA MUSTAZA

A project report submitted in partial fulfillment of the
requirements for the award of the degree of
Master of Engineering (Mechatronics & Automatic Control)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JANUARY 2012

ACKNOWLEDGEMENT

First and foremost, the deepest gratitude of all shall be bestowed to Allah the Almighty and The Merciful for all the insight which He gave to us that lead to the completion of this project.

I would like to express my sincere gratitude to my supervisor Dr. Zuwairie Ibrahim for his supervision, patience, motivation, and immense knowledge which have guided me throughout the process of completing this research. His enthusiasm in the research area is very inspiring and has definitely enriched my growth as a student and researcher. Without his knowledge and encouragement, this study would not have been successful.

I convey a special thank you to my fellow friends especially Amar Faiz Zainal Abidin, Mohd Hafiz Othman and Lai YeeYang for the knowledge that was shared, the time the was spent and support that was provided to complete all the work and meet all the deadline throughout my years in UTM. It would not have been the same without you guys.

I would also like to thank my mum for always stood by me, giving me strength and praying for my success, as well as my dad, my brother and my sisters who always encourage me with their best wishes and prayers.

Last but not least, I would like to thank my husband whose love, support and persistent confidence in me, has tremendously help me to finish my study.

ABSTRACT

Major principle behind the development of computational intelligence is to address complex problem of real world application. Over the years, numerous computational intelligence algorithms have been developed in finding a solution to combinatorial optimization problem. Ant colony system (ACS) algorithm is one of the biologically inspired algorithms that have been applied to effectively solve various combinatorial optimization problems. In this study, ACS is going to be employed in solving DNA sequence design which is a study under the topics of DNA computing. The dependability of DNA computation is highly influenced by the information represents on the DNA strand and the strand reaction. We desire a set of stable double stranded DNA to retrieve the information encoded on the DNA sequence and to operate the computation without output error. To accomplish this, the DNA sequence design problem requires a set of objectives to be optimized and some constraints to be fulfilled. Therefore, DNA sequence design can be regarded as a constrained multi-objectives design problem. The multi-objective design problem is simplified into single-objective using the weighted sum method and objective functions used to obtain a good DNA sequence are $H_{measure}$, *similarity*, *hairpin*, and *continuity*. The sequence is subjected to two constraints which are T_m and $GC_{content.}$ The problem is modeled using finite state machine where each node represents the DNA bases {*A, C, T, G*}. In this study, 9 sets of studies have been conducted using 5, 7, 10, 15, 20, 25, 30, 35 and 40 agents/ants each with 100 independent runs. The number of iterations is set to be 300 for each set. Observation and analysis of the model with increasing number of ants was made and the performance of the model is measured by comparing the result with existing algorithm such as Genetic Algorithm (GA), Multi-Objective Evolutionary Algorithm (MOEA), Particle Swarm Optimization (PSO) etc. Based on the result, the suitable number of ants used for DNA sequence design was also proposed.

ABSTRAK

Prinsip utama di sebalik pembangunan pengkomputeran pintar adalah untuk menangani masalah kompleks yang melibatkan aplikasi dunia sebenar. Sejak kebelakangan ini, pelbagai algoritma serta perisian penkomputeran pintar telah dibangunkan dalam mencari penyelesaian kepada masalah pengoptimuman kombinatorik. Algoritma *Ant Colony System* (ACS) adalah salah satu algoritma yang telah digunakan dengan berkesan dalam menyelesaikan pelbagai masalah pengoptimuman kombinatorik. Dalam kajian ini, algoritma ACS telah digunakan dalam menyelesaikan masalah rekabentuk turutan DNA. Kebolehpercayaan pengkomputeran DNA sangat dipengaruhi oleh maklumat yang terdapat pada lembar DNA serta tindak balas antara DNA. Set DNA yang stabil adalah sangat diperlukan bagi mendapatkan maklumat yang tepat dan memastikan pengendalian pengiraan tanpa ralat. Untuk mencapai tujuan ini, masalah reka bentuk jujukan DNA memerlukan satu set objektif yang perlu dioptimumkan dan beberapa kekangan yang perlu dipenuhi. Oleh itu, masalah turutan DNA boleh dianggap sebagai masalah rekabentuk multi-objektif dan telah dipermudahkan menjadi masalah satu-objektif menggunakan kaedah jumlah wajaran. Fungsi objektif yang digunakan bagi mendapatkan turutan DNA yang baik adalah $H_{measure}$, *similarity*, *hairpin*, and *continuity* dan tertakluk kepada dua kekangan iaitu T_m and $GC_{content}$. Masalah ini dimodel menggunakan mesin keaadan terhingga dimana setiap nodus mewakili asas DNA {*A, C, T, G*}. Dalam kajian ini, 9 set kajian telah dijalankan menggunakan 5, 7, 10, 15, 20, 25, 30, 35 dan 40 bilangan agen/semut. Pemerhatian dan analisis dengan peningkatan bilangan agen telah dibuat serta prestasi model diukur melalui perbandingan dengan algoritma yang sedia ada seperti as *Genetic Algorithm* (GA), *Multi-Objective Evolutionary Algorithm* (MOEA), *Particle Swarm Optimization* (PSO) dan lain-lain. Hasil kajian ini juga digunakan bagi mencadangkan bilangan agen/semut yang sesuai bagi aplikasi masalah rekabentuk turutan DNA.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

-PO4	-	Phosphate
-OH	-	Hydroxyl
A	-	Adenine
ACO	-	Ant Colony Optimization
ACS	-	Ant Colony System
AFSA	-	Artificial Fish Swarm Algorithms
AS	-	Ant System
C	-	Cytosine
CTP	-	Course Timetabling Problem
DNA	-	Deoxyribonucleic Acid
G	-	Guanine
GA	-	Genetic Algorithms
MMAS-CTP	-	Min-Max AS version for Course Timetabling Problem
MOEA	-	Multi-Objective Evolutionary Algorithms
MOPSO	-	Multi-Objective Particle Swarm Optimization
MTTP	-	Minimum Tardy Task Problem
P-ACO	-	Population-Based Ant Colony Optimization
PSO	-	Particle Swarm Optimization
PSP	-	Project Scheduling Problem
QAP	-	Quadratic Assignment Problem
RCPSP	-	Resource Constrained Project Scheduling Problem

RLP	-	Resource Levelling Problem
RNA	-	Ribonucleic Acid
SA	-	Simulated Annealing
SI	-	Swarm Intelligence
T	-	Thymine
T_m	-	Melting Temperature
TSP	-	Travelling Salesman Problem
TWTS	-	Total Weighted Tardiness Scheduling
VRP	-	Vehicle Routing Problem
VRPPD		Vehicle Routing Problem with Pickup and Delivery
VRPTW		Vehicle Routing Problem with Time Windows

LIST OF SYMBOLS

f_{DNA}	-	the total objective functions
Λ	-	bases of DNA {A, C, G, T}
x, y	-	a DNA sequence
$\|x\|$	-	length of x (DNA sequences)
x_i $(1 \leq i \leq \|x\|)$	-	i-th nucleotide from 5'-end of sequence x
Σ	-	a set of n sequences with the same length l
Σ_i	-	i-th member of Σ
$\bar{a}$	-	complementary base of a
l	-	length of sequence
n	-	number of sequences
C_T	-	the total oligonucleotide strand concentration
R	-	the universal gas constant (Boltzmann's constant)
ΔH	-	enthalpy changes of the annealing reaction
ΔS	-	entropy changes of the annealing reaction

CHAPTER 1

INTRODUCTION

1.1 Background

Major principle behind the development of computational intelligence is to address complex problem of real world application. Over the years, numerous computational intelligence algorithms have been developed in finding a solution to combinatorial optimization problem. Most of these algorithms are nature-inspired or biologically inspired as they have been developed based on the behaviour and performance of the natural systems. Enormous success has been achieved through modelling of biological and natural intelligence which is reflected in numerous algorithms such as genetic algorithm (GA), particle swarm optimization (PSO), bee algorithm, ant colony algorithm and many more. The establishment of these algorithms and its application in solving various optimization problems have greatly improved many area of our social-economic life and have been successfully applied in complex real-world application such as pattern recognition (image, speech or handwriting), robotics, forecast and different kind of decision-making in uncertainty conditions.

Among the first algorithm that was developed is genetic algorithm (GA). GA is a search method based on the abstraction of Darwin's evolution and natural selection of biological systems and representing them in the mathematical operators: crossover or recombination, mutation, fitness, and selection of the fittest (She Yang, 2010). Ever since, genetic algorithms become so successful in solving a wide range

of optimization problems, many researchers have been motivated in producing nature inspired algorithm.

Ant colony optimization (ACO) is one of the biologically inspired algorithms that have been applied to effectively solve various combinatorial optimization problems. It was first introduced in 1992 by Marco Dorigo. It is derived from observation of real ants behaviour. The main idea behind the algorithm is the self-organizing and highly coordinated behaviour of the ants which can be exploited to solve complex computational problems (Dorigo and Struzel, 2004). Ant colony system (ACS) is an extension of ACO which was develop by Gambardella and Dorigo in 1997. In this study, ACS is going to be employed in solving DNA sequence design which is a study under the topics of DNA computing.

The next subchapter will introduced the basic of DNA and its structure followed by a review of DNA computing and the importance of DNA sequence design.

1.2 Theory of DNA

The basic building block of DNA is known to be nucleotide consisting of the five-carbon sugar deoxyribose to which one phosphate is esterified at the 5’ position of sugar ring and one nitrogenous base is attached at the 1’ site as illustrated in Figure 1.1. There are two types of nitrogenous bases present in nucleic

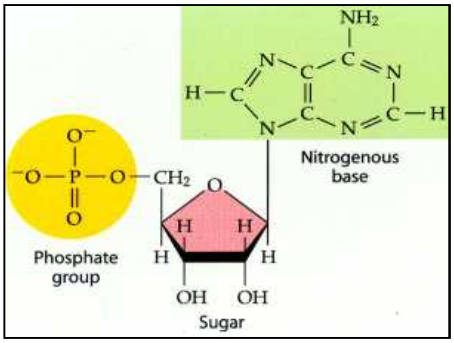

Figure 1.1: Chemical structure of DNA molecule

acid. One is pyrimidines, which contain a single ring and another one is purines, which contain two rings. There are two type of DNA pyrimidines which are thymine (T) and cytosine (C) and the two types of purines which are guanine (G) and adenine (A). The nucleotides are known to be covalently linked to one another to form a linear polymer, or strand with a backbone compose of alternating sugar and phosphate and the bases are attached to each sugar as shown in Figure 1.2. A key feature of DNA is that it has two distinctive ends: A 5' (5-prime) end and a 3' (3-prime) end. The 5' end is the phosphate group (-PO4) attached to the 5th C atom of the sugar ring and the 3' end is the hydroxyl group (-OH) at the 3rd C atom of the sugar ring. Using enzymes, the 3' and 5' ends can be linked. Through this process single stranded DNA is built.

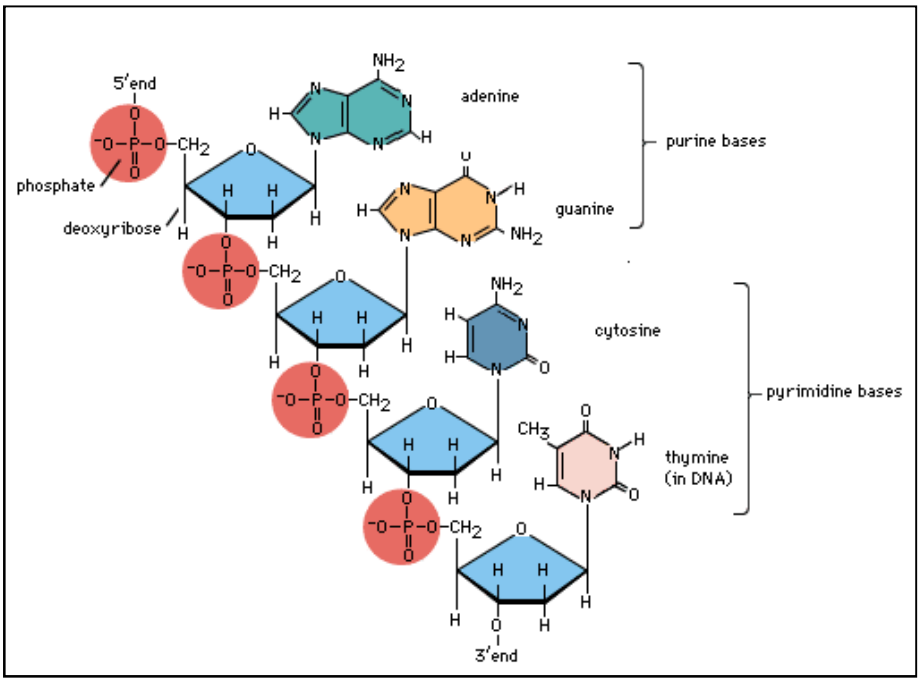

Figure 1.2: Backbone of the DNA structure

The complementary base of *A* is *T*. When coming close to one another they are kept together through 2 hydrogen bonds. The complementary base of *C* is *G*. *C* and *G* build 3 hydrogen bonds. When two strands with complementary base sequences are mixed together, they anneal and form double stranded DNA. A strand 5'-ATGC-3' and its complementary strand also called *Watson-Crick complement* 3'-TACG-5' would build double stranded DNA. This structure is also called "double helix" (Schaefer, 2002).

DNA basically has three primary functions. The first is to store genetic information. DNA contains a stored record of instruction that determine all the heritable characteristics that an organism exhibits. Second function is self-duplication and inheritance. DNA contain information for its own replication (duplication). DNA replication allows genetic instruction to be transmitted from one call to its daughter cells and from one individual to its offspring. And the third primary function is expression of the genetic message. DNA is more than a storage center, it is also a director of a cellular activity. Consequently, the information encoded in DNA has to be expressed in some form that can take part in events that are taking place within the cell.

For application in molecular computation, some basic chemical operations need to be implemented on the DNA. The operations are annealing, polymerase chain reaction (PCR) and electrophoresis. Annealing refers to hybridization of DNA to form a double stranded nucleic acid. It is often used to describe binding of primer of DNA in PCR. PCR is a technique to replicate and amplify the DNA molecule in creating a complementary copy of the template DNA strand. Agarose gel electrophoresis is a technique to separate DNA macromolecules depending on their size, and electric charge by applying electric field in moving the negatively charged molecules through an agarose matrix. Shorter molecules move faster and migrate farther than longer ones because shorter molecules migrate more easily through the pores of the gel. The application of these operations in DNA computing will be further discussed in the next subsection.

1.3 DNA Computing

DNA computing is one interdisciplinary research area that is growing fast. Lars Schaefer (2002) describes DNA as the construction plan of all life on earth. This capability suggests that DNA can be used as a storage of data. One of the main objectives of this research area is to produce, in near future, a biologically inspired computer based on DNA molecules to replace or at least beneficially complement with a silicon based computer (Watada and Bakar, 2008). DNA computing, a new computational paradigm that uses DNA molecule to solve computational problem,

promises massive parallelism and can potentially increase the speed in solving large parallel combinatorial search problems.

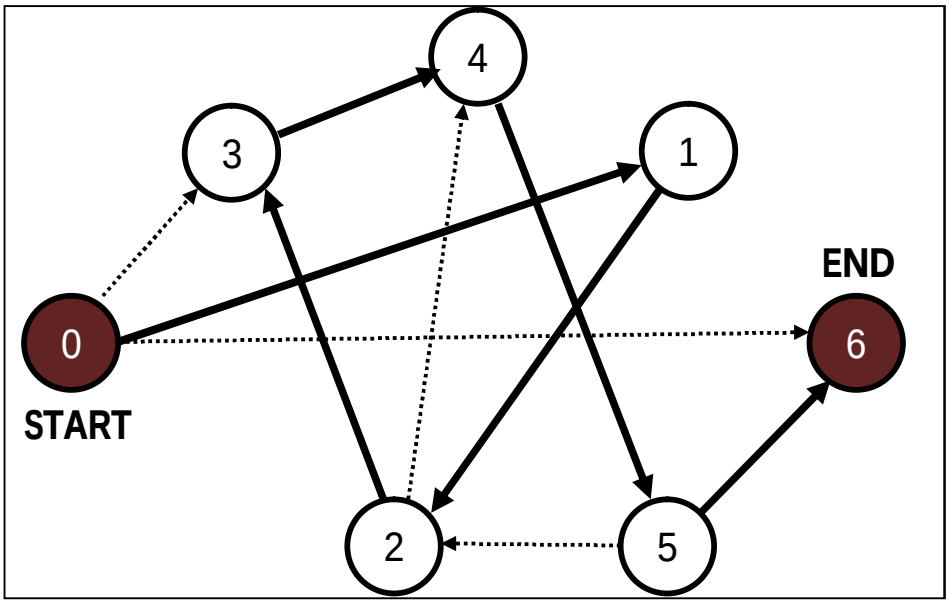

Figure 1.3: Hamiltonian Path Problem. The bold lines represent the only correct path that is 0→1, 1→2, 2→3, 3→4, 4→5, 5→6

The area of DNA computing was initiated by Dr. Adleman in 1994 when he discover a method of solving hard combinatorial problem using DNA. Adleman used a method of manipulating DNA to solve seven-node Hamiltonian Path Problem (HPP). The goal of Adleman's experiment is to determine the existence of a path which commence at the start city, finish at the end city and pass through each of the remaining cities exactly once. In DNA computation, each city is assigned a DNA sequence (Adleman, 1998). HPP is a directed graph with designated input and output vertices, v_{in} and v_{out}. A path from v_{in} to v_{out} is termed *Hamiltonian* if it involves every vertex exactly one. This implies that $v_{in} \neq vout$, because $v_{in} = v_{out}$ would be in the path twice. For example the graph depicted in Figure 1.3 has the designated input vertex 0 and output vertex 6. The path consisting of the directed edges 0→1, 1→2, 2→3, 3→4, 4→5, 5→6 is the only Hamiltonian path in this figure, which is shown bold arrows. In general, a Hamiltonian path is present for a given graph with directed edges and a specified start vertex and end vertex, if and only if there is a path that starts at the start vertex, ends at the end vertex and passes though each remaining vertex exactly once.

For Each vertex *i* in the graph shown in Figure 1.3, a random 20-mer DNA is generated and denoted as O_i. Table 1.1 shows the encoding O_2, O_3 and O_4 for vertices 2,3, and 4, respectively. For edge i→j in the graph, DNA $O_{i\to j}$ is derived from the 3' 10-**mer** of O_i and from the 5' 1 0-mer of O_j. as shown in Table 1.2 for edges $O_{2\to 3}$, $O_{3\to 4}$ and $O_{2\to 4}$.

Table 1.1: Encoding of the vertices of Hamiltonian Path Problem in DNA

Vertices	Sequences
O_2	5' – TATCGGATCG**GTATATCCGA** – 3'
O_3	5' – **GCTATTCGAGCTTAAAGCTA** – 3'
O_4	5' – **GGCTAGGTAC**CAGCATGCTT – 3'

Table 1.2: Encoding of the edges of Hamiltonian Path Problem in DNA

Vertices	Sequences
$O_{2\to 3}$	**GTATATCCGAGCTATTCGAG**
$O_{3\to 4}$	**CTTAAAGCTAGGCTAGGTAC**
$O_{2\to 4}$	**GTATATCCGAGGCTAGGTAC**

The encoded DNA was amplified by polymerase chain reaction (PCR) using O_0 and complementary of O_6 as primers. The two primers worked together in signalling the PCR process. The first alerted DNA polymerase to copy complements of sequence that had the right start city and the second initiated the duplication of molecule that encoded the right start city. Thus, only those molecules encoding paths that begin with vertex 0 and end with vertex 6 were amplified. Then, electrophoresis and affinity separation is used to ensure that the molecules have the right length and to find the shortest path.

Based on Adleman's success, researchers around the world are currently working to exploit the the extremely dense information storage and massive parallelism properties of DNA in hopes of one day producing a DNA computer which have a better performance compare to the conventional electronics computer.

1.4 ACS Based DNA Sequence Design

The reliability of DNA computation is highly dependable and influenced by the information represents on the DNA strand and the strand reaction. But due to technological difficulties and the nature of chemical characteristics of the molecules, DNA reactions may result in inaccuracies of the computation. One of the main approaches to overcome the possibilities of illegal reactions and consequently removing the potential error due to biochemical reaction in advance is to focus on designing a good set of independent DNA sequence. The independent DNA sequence set means a set of DNA sequences which have minimal tendency of cross-hybridization and maximal difference among them. In addition, they must have the similar physical conditions such as length and melting temperature. By removing the error before hand, no DNA is wasted due to illegal reaction, reliability of computation is improved and consequently ensuring high computation accuracy.

ACS algorithm in general should not limit the number of ants used in the system and always allow a new ant to be introduced into the system in hope of finding the optimum solution. Current method of solving DNA sequence using ACS restricts the number of ants used because the number of ant in the system dependent on the solution. Therefore, a new algorithm need to be developed to improve the current method for DNA sequence design using ACS which allow flexibility in terms of number of ants and to eliminate the dependability of solution produced by the system and the number of ants.

1.5 Objective

The objectives of the study are:

(i) To derive DNA computational model based on ACS algorithm that can be efficiently used for DNA sequence design and improved the limitation of the original computational model.

(ii) To assess the performance of the extended computational model over original computational model

(iii) To come out with the suitable number of ants for application of ACS algorithm in DNA sequence design

1.6 Scope of work

The boundary of the research is defined as follows:

(i) The proposed approach is developed by considering four objective functions namely: $H_{measure}$, *similarity*, *continuity*, and *hairpin* and two constraints, which are $GC_{content}$ and *melting temperature* (T_m).

(ii) The number of ants used for the ACS algorithm will be varied until the optimum solution have been found

(iii) The performance of the system will be assessed by comparing the sequences generated by the proposed ACS model with the sequences generated using GA, SA, MOEA, PSO, P-ACO as well as the original ACS model

1.7 Thesis organization

The thesis is organized as follows. First, a brief review of the previous work DNA sequence design as well as several applications of ACS algorithms in solving optimization problem in Chapter 2. Previous work on DNA sequence design includes the algorithm, objective function and constraints that were employed in solving the problem and the ACS algorithm application will discussed the type of problem that was solved and the number of agent/ants used in solving the problem.

In chapter 3, the design criteria of DNA sequence design problem will be outlined and the formulations of the objective functions and constraints will be explained in detail. An introductory overview of the ACS algorithm, which includes background and history of the algorithm can be found in Chapter 4. Chapter 4 also

discuss in detail the construction steps in generating a good set of DNA sequence as well as the improved ACS model in generating the sequences.

Chapter 5 will review the realization of DNA sequence based on ACS algorithm, and will present all the results obtained in this study. Validation of the model as well as the overall performance of the proposed ACS model is also discussed in this chapter. The overall performance is analyzed and discussed based on the comparison with other established algorithm such as GA, SA, PSO etc. The conclusion of the thesis as well as the research direction of the work will be concluded in Chapter 6.

REFERENCES

Adleman, L. (1994). Molecular computation of solutions to combinatorial problems, *Science*, 266: 1021-1024.

Ahmmed, A., Rana, M. A. A., Mahmudul Haque, A. A. M., Al Mamun, M,. (2008), A Multiple Ant Colony System for Dynamic Vehicle Routing Problem with Time Window, In *Proceeding of 3rd International Conference on Convergence and Hybrid Information Technology (ICCIT 08).* 28-30 Aug. 2008, Daejeon, Korea.

Alba, E.; Leguizamon, G.; Ordonez, G. (2005). Analyzing the behavior of parallel ant colony systems for large instances of the task scheduling problem. Proceeding of 19th IEEE International Parallel and Distributed Processing Symposium, 4-8 April 2005, Denver, Colorado

Arita, M., and Kobayashi, S. (2002). DNA sequence design using templates. *New Generation Computing*, 20: 263-277.

Ayob, M., and Jaradat, G., (2009), Hybrid Ant Colony Systems For Course Timetabling Problems, In *Proceeding of 2nd Conference on Data Mining and Optimization*, 27-28 October 2009, Selangor, Malaysia

Cui, G., Cao, X., Zhou, J., and Wang, Y. (2007a). The Optimization of DNA Encoding Sequence Based on Improved AFS Algorithms. In *Proceesings of the IEEE International Conference on Automation and Logistics*, 1141- 1144.

Cui, G., Niu, Y., Wang, Y., Zhang, X., and Pan, Liangqiang. (2007b). A new approach based on PSO algorithms to find good computational encoding sequences. *Progress in Natural Science,* 17(6):712-716.

Engelbrecht, A. P. (2005). *Fundamental of Computational Swarm Intelligence.* England: Wiley.

Feldkamp, U., Saghafi, S., Banzhaf, W., and Rauhe, H. (2001). DNA sequence generator–A program for the construction of DNA sequences. In *Proc. 7th Int. Workshop DNA Based Computer*, 179–188.

Deaton, R., Chen, J., Bi, H., Garzon, M., Rubin, H., and Wood, D. H. (2002b). A PCR-based protocol for in vitro selection of non-crosshybridizing oligonucleotides. In *Proceedings of 8th International Workshop on DNA Based Computers*, 196-204.

Deroussi, L., Barahona da Fonseca, J. (2009). A Hybrid Ant Colony System For Machine Assignment Problem In Flexible Manufacturing Systems. In *Proceedings of International Conference on Computers & Industrial Engineering, CIE 2009.* 6-9 July 2009, University of Technology of Troyes, France

Dorigo, M., and Gambardella, M. (1997). Ant Colony System: A cooperative earning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1(1): 53–66.

Dorigo, M., and Stutzle, T. (2004). *Ant Colony Optimization*. Massachuset: Massachusets Institute of Technology.

Dugardin, F., Amodeo, L., Yalaoui, F. (2011). Fuzzy Lorenz Ant Colony System to solve multiobjective reentrant hybride flowshop scheduling problem, *International Conference on Communication, Computing and Control Application (CCCA 2011),* 3 – 5 March 2011, Hammamet, Tunisia

Frutos, A. G., Liu, Q., Thiel, A. J., Sanner, A. M. W., Condon, A. E., Simith,L. M., and Corn, R. M. (1997a). *Demonstration of a word design strategy for DNA computing on surfaces.* Nucleic Acids Research, 25(23):4748–4757.

Garzon, M. H., and Deaton, R. J. (2004). Codeword design and information encoding. In *DNA ensembles. Natural Computing*, 3:253-292.

Garzon, M., Neathery, P., Deaton, R., Murphy, R. C., Franceschetti, D. R., and Stevens Jr., S. E. (1997). A new metric for DNA computing. In *Proceedings of Genetic Programming (GP97)*, 472–478.

Goss, S., Aron, S., Deneubourg, J. L., & Pasteels, J. M. (1989). Self-organized shortcuts in the Argentine ant. Naturwissenschaften, 76:579–581.

Hartemink, A. J., Gifford, D. K., and Khodor, J. (1998). Automated Constraint-based Nucleotide Sequence Selection for DNA Computation, In *Proc. 4th DIMACS Workshop DNA Based Computer*, 227–235.

Hussini, S., Kari, L., and Konstantinidis, S. (2003). Coding properties of DNA languages. *Theoretical Computer Science*, 290: 1557-1579.

Ibrahim, Z., Kurniawan, T.B., Khalid, N.K., Sudin, S., Khalid, M., (2009) Implementation of an ant colony system for DNA sequence optimization, *Artifial Life and Robotics*, (ISAROB 2009), 14:293–296

Lee, K.Y.; Vlachogiannis, J.G.; (2005). Optimization of Power Systems based on Ant Colony System Algorithms: An Overview. Proceedings of the 13th International Conference on Intelligent Systems Application to Power Systems, 2005. 6-10 Nov. 2005, Arlington, Virginia

Liu. J., Fang, Y., and Liu, Y., (2007) Ant Colony System Algorithm for Path Routing of Urban Traffic Vehicles, In *Proceedings of the IEEE International Conference on Automation and Logistics,* August 18 - 21, 2007, Jinan, China

Marathe, A., Condon, A. E., and Corn, R. M. (1999). On Combinatorial DNA Word Design, In *Proceedings of the 5th International Meeting on DNA Based Computers.*

Mauri, G., and Ferretti, C. (2004). Word design for molecular computing: A survey. In *Proceeding of 9th DIMACS Workshop on DNA Based Computers*, 37-46.

Ono, H., and Mori, Y., (2007). The Optimal Design of Vehicle Routing Problem, with Time Windows by Ant Colony System, In *Proceedings of The SICE Annual Conference 2007*, Sept. 17-20, 2007. Kagawa University, Japan

Penchovsky, R., and Ackermann, J. (2003). DNA library design for molecular computation. *Journal Computer Bio*, 10(2): 215–229.

Ramkumar, A. S., and Ponnambalam, S. G., (2006). Hybrid Ant colony System for solving Quadratic Assignment Formulation of Machine Layout Problems, *IEEE Conference on Cybernetics and Intelligent System,* 7-9 June 2006, Bangkok

She Yang, X. (2010). *Engineering Optimization.* University of Cambridge, United Kingdom

Shin, S. Y., Lee, I. H., Kim, D., and Zhang, B. T. (2005). Multiobjective evolutionary optimization of DNA sequences for reliable DNA computing. *IEEE Transaction on Evolutionary Computation*, 9(2): 143-158.

Tanaka, F., Nakatsugawa, M., Yamamoto, M., Shiba, T., and Ohuchi, A. (2001). Developing support system for sequence design in DNA computing. In *Proceedings of The 7th International Workshop on DNA Based Computers*, 340–349.

Vlachogiannis, J.G.; Hatziargyriou, N.D.; Lee, K.Y.; (2005). Ant Colony System-Based Algorithm for Constrained Load Flow Problem. *IEEE Transaction on Power Systems*, 20:1242-1249

Zeng, Z., and Wang, J. (2010). *Advances in Neural Network Research Applications: Lecture Notes in Electrical Engineering*. The Chinese University of Hong Kong

Zhang, K., Xu, J., Geng, X., Xiao, J., and Pan, L. (2008). Improved taboo search algorithm for designing DNA sequences. *Science Direct Progress in Natura1 Science*, 18: 623-627.

Zhang, B. T., and Shin, S. Y. (1998). Molecular algorithms for efficient and reliable DNA computing. In *Proceeding of Genetic Programming (GP98)*, 735-742.

Zhou, S., Zhang, Q., Zhao, J., and Li, J. (2007). DNA Encodings Based on Multi-Objective Particle Swarm. *Journal of Computational and Theoretical Nanoscience*, 4: 1249-1252.