A Study of the Contribution of Nearest-Neighbour Thermodynamic Parameters to the DNA Sequences Generated by Ant Colony Optimisation

Zuwairie Ibrahim, Ismail Ibrahim, Mohd Falfazli Mat Jusof, Faradila Naim,
Mohd Zaidi Mohd Tumari, Muhammad Salihin Saealal, Nurul Wahidah Binti Arshad,
Kamarul Hawari Ghazali, Zulkifli Md. Yusof, Kamal Khalil, Muhammad Arif Abdul Rahim,
Norhaliza Abdul Wahab, Sophan Wahyudi Nawawi

***Abstract*—The process of designing a set of good DNA sequences is an essential problem and one of the most practical and important research topics in DNA-based computing and the DNA nanotechnology area. In this field of research, a DNA sequence design problem is defined as a multi-objective problem, and it is evaluated using four objective functions, h-measure, similarity, continuity and hairpin. In addition, two constraints, GC content and melting temperature (Tm), are used to maintain uniform chemical characteristics of the sequences. In the authors' previous research, an ant colony system (ACS) was proposed to solve the DNA sequence design problem based on nearest neighbour. The Watson-Crick base pair ΔGo37 was used as the distance between nodes for the thermodynamic parameters in the problem models for the heuristic approach in the ACS algorithms. In the current study, a non-heuristic approach and four new models using the heuristic approach are proposed, and results from the models are compared.**

***Index Terms*—Ant colony optimization, DNA sequence design, nearest-neighbour thermodynamic.**

I. Introduction

In DNA computing [1], single-strand DNA must hybridise correctly to produce a good solution. Otherwise, DNA computing fails to generate identical results for the same problem and algorithm. Additionally, DNA molecules could be wasted if they are used in an undesirable reaction. Usually, in DNA computing, the calculation process consists of several chemical reactions, where the successful lab experiment depends on DNA sequences that have been used. Thus, DNA sequence design is an approach for achieving high computation accuracy and is one of the most practical and important research topics in DNA computing.

In the authors' previous research [2], an ant colony system (ACS) was proposed to solve the DNA sequence design problem based on nearest neighbour thermodynamic parameters using the Watson-Crick base pair ΔGo37 as the distance between nodes in the problem model for a heuristic approach in the ACS algorithm. In this study, a non-heuristic approach and four new models with heuristic approaches are proposed, and results from the models are compared.

This work was supported by the UMP-MOHE RAGS Fund (RDU 121405).

Zuwairie Ibrahim, Ismail Ibrahim, Mohd Falfazli Mat Jusof, Faradila Naim, Mohd Zaidi Mohd Tumari, Muhammad Salihin Saelal, Nurul Wahidah Arshad, and Kamarul Hawari Ghazali are with the Faculty of Electrical & Electronic Engineering, Universiti Malaysia Pahang, 26600 Pekan, Malaysia (e-mail: zuwairie@ump.edu.my).

Zulkifli Md. Yusof, Kamal Khalil, Muhammad Arif Abdul Rahim, Norhaliza Abdul Wahab and Sophan Wahyudi Nawawi are with the Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 Johor Bahru, Malaysia.

II. DNA Sequence Design

In this study, the objective functions and constraints, which have been employed by Shin et al. [3], are chosen, but the formulations have been modified for *h-measure* and *similarity* based on Garzon's formulation [4]. Two objective functions, *h-measure* and *similarity*, are chosen to estimate the uniqueness of each DNA sequence. The function *H-measure* checks the possibility of unintended DNA base pairing based on the hamming distance [4], and *similarity* is defined as an inverse hamming distance between two given DNA sequences [3]. Two additional objective functions, *hairpin* and *continuity*, are used to prevent the formation of a secondary structure of the DNA sequence. Two constraints, *GC-content* and *melting temperature* (T_m), are used to maintain uniform chemical characteristics of the sequences.

DNA sequence design is a multi-objective optimisation problem. There are several ways to solve a multi-objective problem, such as the value function method, the weighted sum method, and evolutionary algorithms. In this study, a common method is used, the weighted sum method, to convert the problem into a single-objective problem, which can be formulated as follows:

$$\min f_{DNA} = \sum_i \omega_i f_i \tag{1}$$

as subjected to T_m and *GC-content* constraints, where f_{DNA} is the objective function for each i {*h-measure*, *similarity*, *hairpin*, *continuity*}, and ω_i is the weight for each f_i. In this problem, ω is typically set by the decision maker such that and ω>0. If all the weights are committed or set to 1, then all objectives are treated equally.

The basic notations are defined as shown in Table 1. The following notations are used:

$$bp(a,b) = \begin{cases} 1 & a = \bar{b} \\ 0 & otherwise \end{cases} \tag{2}$$

TABLE I
BASIC DEFINITIONS

Notation	Description
Λ	{A,C,G, T,−}
Λ_{nb}	{A,C,G, T}
Λ^*	Λ and Λ_{nb}
$a, b \in \Lambda$	a, b = {A, C,G, T, − } (with blank)
$X, y \in \Lambda^*$	x, y = {A, C,G, T} and {A, C, G, T, − }
$\|x\|$	length of x
x_i $(1 \leqslant i \leqslant \|x\|)$	ith nucleotide from 5'-end of sequence x
Σ	A set of n sequences with the same length l
Σ_i	ith member of Σ
$\bar{a}$	complementary base of a

$$eq(a,b) = \begin{cases} 1 & a = b \\ 0 & otherwise \end{cases} \tag{3}$$

$$T(i,j) = \begin{cases} i & i > j \\ 0 & otherwise \end{cases} \tag{4}$$

For a given sequence $x \in \Lambda^*$, the number of non-blank nucleotides is defined as follows:

$$length_{nb}(x) = \sum_{i=1}^{|x|} nb(x_i) \tag{5}$$

where

$$nb(a) = \begin{cases} 1 & a \in \Lambda_{nb} \\ 0 & otherwise \end{cases} \tag{6}$$

and a shift of sequence x by i bases is denoted as follows:

$$shift(x,i) = \begin{cases} (-)^i x_1 \cdots x_{l-1} & i \geq 0 \\ x_{i+1} \cdots x_l (-)^i & i < 0 \end{cases} \tag{7}$$

A. h-measure

The *h-measure* computes how many nucleotides are complementary to prevent cross-hybridisation of two sequences including position shift. The fitness function is as follows:

$$f_{h-measure}(\sum\nolimits_i) = \sum_{j=1}^{n} h-measure(\sum\nolimits_i, \sum\nolimits_j) \tag{8}$$

where Σ_i and Σ_j are anti-parallel to each other. Anti-parallel means the sequences have different directions. The first sequence has a 5' →3' direction, and the second sequence has a 3' →5' direction. *h-measure*(x,y) is also divided into two terms. One term is for the overall complementary, and the other is the penalty term for the continuous complementary region.

$$h-measure(x,y) = \max_{|i|<l-1} \begin{pmatrix} h_{dis}(x, shift(rev(y),i) + \\ h_{con}(x, shift(rev(y),i) \end{pmatrix} \tag{9}$$

where

$$0 \leq g \leq l-3, \qquad |x| \leq l-3 \tag{10}$$

$$h_{dis}(x,y) = T\left(\sum_{i=1}^{l} bp(x_i, y_i), h_{dis} \times length_{nb}(y) \right) \tag{11}$$

$$h_{con}(x,y) = \sum_{i=1}^{l} T\left(cbp(x,y,i), h_{con} \right) \tag{12}$$

$$cbp(x,y,i) = \begin{cases} c & if\ ^{\exists}c, s.t. \quad bp(x_i, y) = 0, \\ & bp(x_{i+j}, y_{i+j}) = 1\ for\ 1 \leq j \leq c, \\ & bp(x_{i+c+1}, y_{i+c+1}) = 0 \\ 0 & otherwise \end{cases} \tag{13}$$

h_{dis} is a real value between 0 and 1, and h_{con} is an integer between 1 and l. Both values are set by the user [3].

B. Similarity

The *similarity* measure, $f_{Similarity}(x,y)$, computes the *similarity* in the same direction of two given sequences to keep each sequence as unique as possible including position shift [4]. *Similarity* uses a fitness function as follows:

$$f_{Similarity}(\sum\nolimits_i) = \sum_{j=1,\, j \neq i}^{n} Similarity(\sum\nolimits_i, \sum\nolimits_j) \tag{14}$$

where Σ_i and Σ_j are parallel to each other. *Similarity*(x, y) is also divided into two terms. One term is for the overall complementary and the other is the penalty term for the continuous complementary region.

$$similarity(x,y) = \max_{|i|<l-1} \left(s_{dis}(x, shift(y,i) + s_{con}(x, shift(y,i)) \right) \tag{15}$$

where

$$0 \leq g \leq l-3, \qquad |x| \leq l-3 \tag{16}$$

$$s_{dis}(x,y) = T\left(\sum_{i=1}^{l} eq(x_i, y_i), s_{dis} \times length_{nb}(y) \right) \tag{17}$$

$$s_{con}(x,y) = \sum_{i=1}^{l} T\left(ceq(x,y,i), s_{con} \right) \tag{18}$$

$$ceq(x,y,i) = \begin{cases} c & if\ ^{\exists}c, s.t. \quad eq(x_i, y_i) = 0, \\ & eq(x_{i+j}, y_{i+j}) = 1 \\ & for\ 1 \leq j \leq c, eq(x_{i+c+1}, y_{i+c+1}) = 0 \\ 0 & otherwise \end{cases} \tag{19}$$

s_{dis} is a real value between 0 and 1, and s_{con} is an integer between 1 and l. Both values are set by the user.

C. Hairpin

The *hairpin* measure calculates the probability of forming a secondary structure. *Hairpin* uses a fitness function as follows:

$$f_{hairpin}(\sum) = \sum_{i=1}^{n} hairpin(\sum\nolimits_i) \tag{20}$$

TABLE II
ΔH AND ΔS VALUES OF FOUR NEAREST-NEIGHBOUR PARAMETERS FOR THE HEURISTIC MODEL

Pair	Breslauer		SantaLucia		Unified		Sugimoto	
	ΔH_x	ΔS_x	ΔH_x	ΔS_x	ΔH_x	ΔS_x	ΔH_x	ΔS_x
AA/TT	-9.1	-24.0	-8.4	-23.6	-7.9	-22.2	-8.0	-21.9
AG/CT	-7.8	-20.8	-6.1	-16.1	-7.8	-21.0	-6.6	-16.4
AT/AT	-8.6	-23.9	-6.5	-18.8	-7.2	-20.4	-5.6	-15.2
AC/GT	-6.5	-17.3	-8.6	-23.0	-8.4	-22.4	-9.4	-25.5
GA/TC	-5.6	-13.5	-7.7	-20.3	-8.2	-22.2	-8.8	-23.5
GG/CC	-11.0	-26.6	-6.7	-15.6	-8.0	-19.9	-10.9	-28.4
GC/GC	-11.1	-26.7	-11.1	-28.4	-9.8	-24.4	-10.5	-26.4
TA/TA	-6.0	-16.9	-6.3	-18.5	-7.2	-21.3	-6.6	-18.4
TG/CA	-5.8	-12.9	-7.4	-19.3	-8.5	-22.7	-8.2	-21.0
CG/CG	-11.9	-27.8	-10.1	-25.5	-10.6	-27.2	-11.8	-29.0

$$hairpin(x) = \sum_{p=P\min}^{(l-R\min)/2} \sum_{r=R\min}^{l-2p} \sum_{i=1}^{l-2p-r} T\left(\sum_{j=1}^{pinlen(p,r,i)} bp(x_{p+i+j}, x_{p+i+r+j}), \right) \quad (21)$$
$$pinlen(p,r,i)/2$$

where

$$pinlen(p,r,i) = \min(p+i, l-r-i-p) \quad (22)$$

D. Continuity

If the same bases occur continuously in a sequence, the sequence might exhibit unexpected structures. $f_{Continuity}(x)$ calculates the degree of successive occurrence of the same bases [4]. *Continuity* for a set of sequences Σ is defined as follows:

$$f_{continuity}(\sum) = \sum_{i=1}^{n} continuity(\sum_i) \quad (23)$$

where

$$continuity(x) = \max_{1 \le i \le l} \left(\sum_{a \in \Lambda nb} c(a,i) \right) \quad (24)$$

$$c(a,i) = \begin{cases} n & if\ ^{\exists} n, s.t.\ eq(a_i, a_{i+j}) = 1\ for\ 1 \le j < n, \\ & \quad eq(a_i, a_{i+n}) = 0 \\ 0 & otherwise \end{cases} \quad (25)$$

E. GC content

GC content is the percentage of G and C in a sequence. For example, the 12-mer DNA sequence 5'-ATGGTTGCATGC-3' has four Gs and two Cs. Thus, by using Eq. 26, the GC content for this DNA sequence is 50%.

$$\text{GCcontent} = (yG + zC) / (wA + xT + yG + zC) \quad (26)$$

F. Melting temperature

Melting temperature is one of the most important features for a laboratory experiment. It is defined as the temperature where half of the double-stranded DNA starts to break into its single-stranded form. The nearest-neighbour formulation for *melting temperature* is defined as follows:

$$T_m(x) = \frac{\Delta H}{\Delta S + R \ln C_T} + 16.6 \log(Na^+) \quad (27)$$

where ΔH and ΔS are enthalpy and entropy changes of the annealing reaction. R denotes the universal gas constant (Boltzmann's constant) and C_T is the total oligonucleotide strand concentration. For non-self-complementary molecules, C_T is replaced by $C_T/4$. Na^+ is the salt concentration for salt adjustment.

III. ANT COLONY SYSTEM (ACS)

Ant colony optimisation (ACO) is a population-based metaheuristic for combinatorial optimisation problems. ACO is inspired by the ability of ants to find the shortest path between their nest and a source of food. Marco Dorigo first introduced ACO in his PhD thesis [5] and applied it to the travelling salesman problem (TSP). Since then, ACO has been applied to the quadratic assignment problem [6], the vehicle routing problem [7], bin packing, stock cutting [8], and RNA secondary structure prediction [9].

The ant colony system (ACS) is an improved ant system (AS) [10] in three main aspects: the state transition rule, the global updating rule, and the local pheromone updating rule.

1. State transition rule

In ACS, the state transition rule is the following. An ant that is positioned at node r chooses the city s to move to by applying the rule given by Eq. 28 as follows:

$$s = \begin{cases} \arg \max_{u \in J_k(r)} \left\{ [\tau(r,u)][\eta(r,u)^\beta] \right\} & if\ q \le q_o \\ S & otherwise \end{cases} \quad (28)$$

where q is a random number uniformly distributed in [0 .. 1], q_0 is a parameter ($0 \leqslant q_0 \leqslant 1$), S is a random variable selected according to a probability distribution, $\tau(r,u)$ is the pheromone intensity and $\eta(r,u)$ is heuristic information.

2. Global updating rule

In ACS, only the globally best ant is allowed to deposit pheromone. Global updating is performed after all ants have

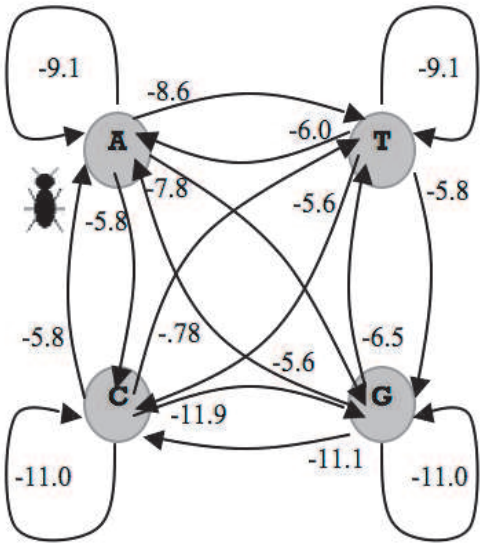

Fig. 1. ACS modelling for DNA sequence design problem with the thermodynamic values.

completed their tours. The pheromone level is updated by applying the global updating rule of Eqs. (29-30) as follows:

$$\tau(r,s) \leftarrow (1-\alpha).\tau(r,s) + \alpha.\Delta\tau(r,s) \tag{29}$$

$$\Delta\tau(r,s) = \begin{cases} \left(L_{gb}\right)^{-1} & if\ (r,s) \in global-best-tour \\ 0 & otherwise \end{cases} \tag{30}$$

where $0<\alpha<1$ is the pheromone decay parameter for global updating, and L_{gb} is the length of the globally best tour from the beginning of the trial.

3. Local updating rule

While building a solution (i.e., a tour), ants visit edges and change their pheromone level by applying the local updating rule of Eq. (31) as follows:

$$\tau(r,s) \leftarrow (1-\rho).\tau(r,s) + \rho.\Delta\tau(r,s) \tag{31}$$

where $0<\rho<1$ is the pheromone decay parameter for local updating.

IV. Ant Colony System for DNA Sequence Design

Since ACS algorithms are normally applied to path finding problems, a model similar to a finite state machine, which has four nodes, is presented to solve the DNA sequence design problem. In this model, the nodes represent A, C, G, and T of the DNA bases. Every node is connected to every other node, including itself.

TABLE III
DNA SEQUENCE PARAMETERS

Parameter		Value
h-measure	h_{con}	6
	h_{dis}	0.17
similarity	s_{con}	6
	s_{dis}	0.17
continuity threshold	t	2
hairpin	R_{min}	6
	P_{min}	6
GC content	*Min*	0
	Max	100
Tm	*Min*	0°C
	Max	150°C
Na^+		1 Moll

TABLE IV
ANT COLONY SYSTEM PARAMETERS

Parameter	Value
B	2
ζ	0.1
ρ	0.1
q_0	0.9
N-remove/new	1
Number of sequences	7 (no. of ants-n_k)
Length of DNA sequence	20 (no. of tours)
Maximum number of iteration (t_{max})	100

In this study, two types of models are proposed: an approach without heuristic [11] and an approach with heuristic [2]. For the model using the heuristic approach, four nearest neighbour thermodynamic parameters are used: Breslauer [12], SantaLucia [13], Unified [14], and Sugimoto [15], as shown in Table 2. Figure 1 illustrates how these values are used as distances in the model using a heuristic approach in the ACS algorithm.

At first, every ant is placed randomly at any node. Next, every ant moves from one node to the other nodes to construct the DNA sequence. During the tour, the ant chooses the next node by applying the state transition rule, as formulated in Eq. (28).

Because the required solution is a set of DNA sequences, a mechanism is needed to store the DNA sequence in an archive to be analysed. The archive storing process is started with the calculation of all objective functions for each DNA sequence. Next, the total values are sorted by ascending order, and the DNA sequences are placed in the archive, starting with the smallest value of the total objective and only if the range of *GC content* and *melting temperature* constraints are satisfied. The storing archive process continues until the number of DNA sequences in the archive is equal to number of ants, *n*.

Next, in the archive updating process, the DNA sequences in the archive are sorted in descending order based on the total objective values. The *N*-first worst DNA sequences will be selected and removed. Those DNA sequences will be replaced by *N*-new DNA sequences. Figure 2 shows the pseudo-code for the ACS algorithm.

TABLE V
COMPARISON AMONG THE BEST RESULTS FROM EACH OF THE PROPOSED APPROACHES

Approaches	*h-measure*	similarity	Continuity	Hairpin	Total
without Heuristic	54.00	43.71	0.00	0.00	97.71
with Breslauer ΔH	39.86	54.29	0.00	0.00	94.14
with Breslauer ΔS	42.86	52.57	0.00	0.00	95.43
with SantaLucia ΔH	54.57	44.00	0.00	0.00	98.57
with SantaLucia ΔS	46.86	49.71	0.00	0.00	96.57
with Unified ΔH	55.29	44.86	0.00	0.00	100.14
with Unified ΔS	49.57	45.71	0.00	1.86	97.14
with Sugimoto ΔH	51.71	43.71	0.00	3.00	98.43
with Sugimoto ΔS	53.14	41.71	0.00	1.86	96.71

V. RESULTS AND DISCUSSION

The experiments consist of nine different cases. The first one involved a computational model without heuristic values. The rest involved computational model with heuristic values (ΔH or ΔS), which are taken from Breslauer, SantaLucia, Unified, and Sugimoto nearest-neighbor thermodynamic parameters. The experiments were conducted for 100 runs, and the experimental results were collected for further analysis. The parameters for the calculation of objectives and constraints are shown in Table 3 and the parameters for the ACS are shown in Table 4. In this experiment, the value of weight for each fitness is equal to one and the length of each sequence is fixed to 20 nucleobases.

The comparison among the best results obtained from each of the proposed approaches is shown in Table 5. It was found that the ΔH and ΔS of Breslauer gives the best results. The ΔS of SantaLucia and the ΔS of Unified show better results than without the heuristic, but the ΔH of SantaLucia and the ΔH of Unified show the worst results.

VI. CONCLUSIONS

This research presents an implementation of the ant colony system in a DNA sequence design problem using two models, an approach without a heuristic and an approach with a heuristic. For the modelling approach with the heuristic, four nearest neighbour thermodynamic parameters are used, which are Breslauer, SantaLucia, Unified, and Sugimoto. The ACS algorithm is implemented with four objectives, *H-measure*, *similarity*, *continuity*, and *hairpin*, and subjected to two constraints, *GC content* and T_m. In summary, the Breslauer nearest-neighbor thermodynamic parameter is the most suitable values in DNA sequence design application.

REFERENCES

[1] Adleman, L.M, "Molecular computation of solutions to combinatorial problem", *Science-266*, 1994, 1021-1024.

[2] T.B. Kurniawan, N.K. Khalid, Z. Ibrahim, M. Khalid, and M. Middendorf, "An Ant Colony System for DNA Sequence Design bases on Thermodynamic", 2008, 144-149.

[3] S.Y. Shin, I.H. Lee, D. Kim, B.T. Zhang, "Multiobjective evolutionary optimization of DNA sequences for reliable DNA computing", *IEEE Transaction on Evolutionary Computation*, Vol. 9, No. 2, 2005, pp. 143-158.

[4] M. Garzon, P. Neathery, R. Deaton, R.C Murphy, D.R. Franceschetti, and S.E. Stevens Jr. "A new metric for DNA computing", *Proceedings of Genetic Programming*, 1997, 472-478.

[5] M. Dorigo, "Optimization, learning and natural algorithms, *PhD Thesis," Dipartimento di Elettronica, Politechico di Milano, Italy*, 1992.

[6] V. Maniezzo, A. Colorni, and M. Dorigo, "The ant system applied to the quadratic assignment problem," *Universite Libre de Bruxelles, Belgium, Tech. Rep. IRIDIA*/94-28, 1994.

[7] B. Bullnheimer, R.F. Hartl, C. Strauss, "An improved Ant System algorithm for the Vehicle Routing Problem", *Annals of Operations Research, Volume 89, Issue 0*, Jan 1999, 319 – 328.

[8] F. Ducatelle and J. Levine, "Ant Colony Optimisation for Bin Packing and Cutting Stock Problems," *presented at UK Workshop on Computational Intelligence (UKCI-01), Edinburgh*, 2001.

[9] McMellan, N, "RNA Secondary Structure Prediction using Ant Colony Optimisation, *Master Thesis", School of Informatics, University of Edinburgh*, 2006.

[10] M. Dorigo, Luca. M.G, Ant Colony System: A cooperative learning approach to the Traveling Salesman Problem, *IEEE Trans. Evol, Comput, vol. 1*, 1997.

[11] Z. Ibrahim, T.B. Kurniawan, N. K. Khalid, S. Sudin, M. Khalid, "Implementation of an ant colony system for DNA sequence optimization", *Artif Life Robotic*, vol. 14, 2009, 293-296.

[12] K.J, Breslauer, R. Frank, H. Blocker, and L.A. Marky, "Predictiong DNA duplex stability from base sequence", *Proceeding Natl Acid Sci USA*, 1998, 1746-3750.

[13] J. SantaLucia Jr, H. T. Allawi, and P. A. Seneviratne, "Improved Nearest Neighbor Parameter for Predicting DNA Duplex Stability", *Biochemistry* vol.35, 1996, 3555-2562.

[14] J. SantaLucia, Jr. "A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics". *Proceedings. Natl Acad. Sci. USA, 95*, 1998, 1460–1465.

[15] N. Sugimoto, S. Nakano, M. Yoneyama, and K. Honda, "Improved thermodynamic parameters and helix initiation factor to predict stability of DNA duplexes", NucleicAcids Res. Vol. 24, 1996, 4501-4505.

// --- Initialisation step
Initialise parameter t, α, ρ, q_0, n_k, N, *and all DNA parameters, such as* h_{con}, h_{dis}, s_{con}, s_{dis};
Calculate τ_o;
For each link(i, j) **do**
    τ (i, j) = τ_o;   // --- Pheromone initialise
end
t= 0;   // --- initialise no of iteration.
// --- Main Process
Repeat
    Repeat
        Place all ants, k =1, …, n_k;   // --- (n_k = number of ants)
        For each ant k =1, …, n_k **do**
            Repeat
// --- State Transition Rule
                Each ant applies a ***state transition rule*** (Eq28) to incrementally build a DNA sequence;
// --- Local Pheromone Updating
                A ***local pheromone updating rule*** (Eq31) is applied;
            Until all ants have built a complete a DNA sequence;   // --- (no. of tours)
        Next
// --- Archive update
        If no. of DNA sequences in archive are equal with the number of ants (full) **Then**
            Calculate all DNA sequences in archive bases on four objective functions and then sort in descending order;
// --- N-first(s) DNA sequences in archive
            N-DNA sequences which have higher value in archive is removed;
        EndIf
// --- Storing the archive process
        Calculate all DNA sequences bases on four objective functions and then sort in ascending order;
        For each DNA sequence **do**
            Checked for $GC_{content}$ and T_m constraints;
            If passed and archive is not full **Then** stored DNA sequence to archive;
        Next
// --- Global Pheromone Updating for the best DNA sequence produced by ants;
        A ***global pheromone updating rule*** (Eq. 29) is applied;
    Until number of DNA sequences in archive are equal with the number of ants;
    t = t + 1;
Until End_Condition // --- (maximum no. of iteration is reached)

Figure 2 : ACS algorithms for DNA sequence design problem