ICIC Express Letters
Volume **3**, Number **1**, March **2009**

 ISSN 1881-803X
pp. **27–32**

FUNCTION MINIMIZATION IN DNA SEQUENCE DESIGN BASED ON CONTINUOUS PARTICLE SWARM OPTIMIZATION

Noor Khafifah Khalid[1], Zuwairie Ibrahim[1], Tri Basuki Kurniawan[1]
Marzuki Khalid[1], Nor Haniza Sarmin[2] and Andries P. Engelbrecht[3]

[1]Center for Artificial Intelligence and Robotics (CAIRO)
Faculty of Electrical Engineering
[2]Faculty of Science
Universiti Teknologi Malaysia
81310 Skudai, Johor, Malaysia
ifah.khalid@gmail.com; zuwairiee@fke.utm.my; tribasukikurniawan@yahoo.com
marzuki.khalid@gmail.com; nhs@fs.utm.my

[3]Department of Computer Science
University of Pretoria, South Africa
engel@cs.up.ac.za

Received August 2008; accepted November 2008

Abstract. *In DNA based computation and DNA nanotechnology, the design of good DNA sequences has turned out to be an essential problem and one of the most practical and important research topics. Basically, the DNA sequence design problem is a multi-objective problem, and it can be evaluated using four objective functions, namely, $H_{measure}$, similarity, continuity, and hairpin. In this paper, particle swarm optimization (PSO) is proposed to minimize those objective functions, individually, subjected to two constraints: melting temperature, T_m, and $GC_{content}$. A model is presented in order to minimize the objective functions using PSO. An implementation of the optimization process is presented using 20 particles. The results obtained verified that PSO can be used to minimize each objective individually.*
Keywords: DNA computing, DNA sequence design, Particle swarm optimization

1. **Introduction.** Deoxyribonucleic Acid (DNA) has certain unique properties such as self-assembly and self-complementary, which makes it able to save an enormous amount of data and perform massive parallel reactions. With the view of the utilization of such attractive features for computation, DNA computation research field has been initiated [1]. Usually, in DNA computing, the calculation process consists of several chemical reactions, where the successful wet lab experiment depends on DNA sequences we used. Thus, DNA sequence design turns out to be one of the approaches to achieve high computation accuracy and become one of the most practical and important research topics in DNA computing.

The necessity of DNA sequence design appears not only in DNA computation, but also in other biotechnology fields, such as the design of DNA chips for mutational analysis and for sequencing. For these approaches, sequences are designed such that each element uniquely hybridizes to its complementary sequence, but not to any other sequence. Due to the differences in experimental requirements, however, it seems impossible to establish an all-purpose library of sequences that effectively caters for the requirements of all laboratory experiments [2]. Since the design of DNA sequences is dependent on the protocol of biological experiments, a method for the systematically design of DNA sequences is highly required.

Various kinds of methods and strategies have been proposed to date to obtain good DNA sequences. These methods are exhaustive search method [3], random search algorithm [4], simulated annealing [5], dynamic programming approach [6], graph method [7], template-map strategy [8,9], genetic algorithms [10,11], and multi-objective evolutionary optimization [12].

2. **Objectives and Constraints in DNA Sequence Design.** The objective of the DNA sequence design problem is basically to obtain a set of DNA sequences where each sequence is unique or cannot be hybridized with other sequences in the set. In this work, two objective functions, namely $H_{measure}$, and *similarity* are chosen to estimate the uniqueness of each DNA sequence. Another two additional objective functions, *hairpin* and *continuity*, are used to prevent the secondary structure of a DNA sequence.

DNA sequence design is actually a multi-objective optimization problem, where four objective functions have to be minimized simultaneously. However, in this paper, a model that is suitable for PSO computation is presented to minimize these four objective functions individually. Those objective functions are $H_{measure}$, *similarity*, *hairpin*, and *continuity*. *Melting temperature* and $GC_{content}$ are included as constraints in the PSO computation. The formulations for all objective functions and constraints can be referred to [13].

This paper is divided into five sections. Section 2 gives an overview of the problem in DNA sequence. Section 3 introduces basic continuous PSO. Section 4 presents the proposed approach of DNA sequence design based on PSO. Experimental results obtained from the proposed approach are shown and discussed in Section 5. Section 6 concludes the paper.

3. **Particle Swarm Optimization.** Particle swarm optimization (PSO) is a population-based stochastic optimization technique developed by Kennedy and Eberhart in 1995 [14]. This method finds an optimal solution by simulating social behavior of bird flocking. The PSO algorithm consist of a group of individuals named "particles". Each particle is a potential solution to an n-dimensional problem.

The group can achieve the solution effectively by using the common information of the group and the information owned by the particle itself. The particles change their state by "flying" around in an n-dimensional search space based on the velocity updated until a relatively unchanging state has been encountered, or until computational limitations are exceeded.

The velocity of each particle is calculated using

$$\boldsymbol{v}_i^{k+1} = \omega \boldsymbol{v}_i^k + c_1 r_1 (\boldsymbol{pbest_i} - \boldsymbol{s}_i^k) + c_2 r_2 (\boldsymbol{gbest^k} - \boldsymbol{s}_i^k) \tag{1}$$

where $\boldsymbol{v}_i^k$, $\boldsymbol{v}_i^{k+1}$ and $\boldsymbol{s}_i^k$, are the velocity vector, modified velocity vector, and positioning vector of particle i at generation k, respectively. $\boldsymbol{pbest_i^k}$ is the best position found by particle i and $\boldsymbol{gbest}^k$ is the best position found by the particle's neighborhood or the entire swarm. Parameter c_1 and c_2 are the cognitive and social coefficients, respectively, and w is the inertia weight, which is employed to control the impact of the previous history of velocities on the current velocity of each particle. r_1 and r_2 are two different random parameters. The modified position vector, $\boldsymbol{s}_i^{k+1}$, is obtained using

$$\boldsymbol{s}_i^{k+1} = \boldsymbol{s}_i^k + \boldsymbol{v}_i^{k+1} \tag{2}$$

4. **Optimization.** For DNA sequence design application, the proposed approach is based on basic PSO algorithm. A DNA sequence can be represented in binary, where A, C, G, and T, are encoded as 00_2, 01_2, 10_2, and 11_2, respectively. A sequence represents one dimension in PSO and the length of sequence, l, is defined based on the number of bases in the sequence. For example, a sequence of length l is normally referred as an l-mer

sequence. Thus, the search space is $4^l - 1$. As such, for 5-mer nucleic acid sequence, the size of the search space is $4^5 - 1$, which is from 0 to 1023, and contains all sequences from AAAAA to TTTTT.

In this study, basic PSO is employed, which has been designed for continuous-valued search spaces. In order to eliminate the floating point in the computation, caused by the random values and coefficient factors, the floating values are approximated to the nearest decimal numbers. The decimal numbers are converted to binary, and binary representations are converted into sequences. For example, the decimal number of 908.8 is approximated to 909_{10}, which is equals to 1110001101_2, in binary, and is then translated into "TGATC" DNA sequence.

TABLE 1. The value of PSO control parameters

Parameter	**Value**
Cognitive factor, c_1	1.4
Social factor, c_2	1.4
Inertia weight, w	$0.9 \sim 0.4$
Random values: r_1, r_2	[0,1]
No. of particles	20
Max iteration	400

5. **Experimental Results and Discussions.** The proposed approach has been implemented using Visual Basic 6.0. Table 1 shows the value of PSO parameters used in the experiments. A set of 10 DNA sequences with the length, $l = 20$, is chosen as a case study. The constraints are $30\% \sim 80\%$ for $GC_{content}$ and 50° C $\sim 80^\circ$ C for T_m. In this study, a decreasing inertia weight is used, where

$$\omega^{iteration} = \omega_{\max} - \left(\frac{\omega_{\max} - \omega_{\min}}{iteration_{\max}} \times iteration \right) \tag{3}$$

TABLE 2. The results obtained for $H_{measure}$, the standard deviations are shown in parentheses.

DNA Sequences	$\boldsymbol{H}_{measure}$	***Similarity***	***Continuity***	***Hairpin***
1. CTAAGTCAACCAAGCACACA				
2. AAACACATATCGACTGGGAG				
3. GACAACGTGGCGACTGTGCC				
4. ATTCTGACGCCTTTAAGTTA				
5. TCTCCGCAATTAAAAGGAGT	4.7	286.08	311.6	3.23
6. CATAGCCCTAGGTCCAGTAA	(6.842)	(80.316)	(69.069)	(3.494)
7. ATAAAAACGAACCAAAACCC				
8. GCGGCTGGCAAACAAAATGC				
9. CCCATTCAACTACCAGTTTT				
10.TCCTCGCCGCAATACGCAAG				

The results for each objective in DNA sequence design are shown in Table 2, 3, 4, and 5. For each objective, the function minimization is applied for 10 times and the average values and the standard deviations obtained are calculated. However, in this paper, only one sample of the results is shown for each objective.

Table 2 shows the results for $H_{measure}$ function, where the fitness value is the average value for 10 sequences. Although PSO can find the minimum value for $H_{measure}$, the values for other objectives, which are *similarity* and *continuity,* are high, except for *hairpin.* For *similarity* function, the results are shown in Table 3. The minimized value for *similarity* has been obtained but the values for other objectives are not minimized. In Table 4, the sample of the results of *continuity* function is shown. Even though PSO is able to find the minimum value for *continuity* function, the values for other objectives are not minimized.

TABLE 3. The results obtained for *similarity*, the standard deviations are shown in parentheses.

DNA Sequences	$\boldsymbol{H}_{measure}$	***Similarity***	***Continuity***	***Hairpin***
1. CTCTGCGCTATACACTGTTG				
2. TCTTTTCGCTCCTGAAAACG				
3. CGCGGAAAAGGTATGGCTCA				
4. TGCGTTTTGCCCAGGAAAAG				
5. CTCAACACGCGTTTTGGACA	97.38	65.47	17.55	70.06
6. CAACGATAGATTCTACATTA	(0.3944)	(20.127)	(4.455)	(4.872)
7. GTCGTTCGAGCCACCCAGAC				
8. AGTCTGATGTAAAAGTTCCC				
9. TCTATTGGGAAGCACGGAGG				
10.CGCCGTCAAGTCGGCCCAGG				

Table 5 shows the results of *hairpin* function. The fitness value obtained is the lowest among the fitness functions. Although PSO has found the minimum value for *hairpin*, the values for other objective functions are high.

TABLE 4. The results obtained for *continuity*, the standard deviations are shown in parentheses.

DNA Sequences	$\boldsymbol{H}_{measure}$	***Similarity***	***Continuity***	***Hairpin***
1. TGATAATGGAGCATGACACC				
2. GTAGCGGATGATTCTCGCGT				
3. ATGGATCTAGCTCATCTTCA				
4. GTACCAAGTTGCGCTAGCAA				
5. ATATAACATTCACTGTCCAT	75.02	19.44	0.45	107.9
6. GGTGCGCCCAGGAACGGAGC	(1.369)	(3.264)	(0.45)	(2.373)
7. CCAACACTCTCCGACAGTCG				
8. TTCAGCCAGAAGCTCGCCAT				
9. CTGTGACTATGAGTTGGCAC				
10.TTACATAGCGTATGCGCATA				

6. **Conclusions.** This paper presented an application of PSO in DNA sequence design. A model that is suitable for PSO computation is presented to minimize four objective functions individually. These objective functions are $H_{measure}$, *similarity, hairpin*, and

TABLE 5. The results obtained for *hairpin*, the standard deviations are shown in parentheses.

DNA Sequences	$H_{measure}$	*Similarity*	*Continuity*	*Hairpin*
1. CGACTTCTGGCAATGCGAAT				
2. TCTTGTGAGTTAGTGTGCTG				
3. CGAATAGTTGATCTGGGGAT				
4. CAACGGGAGCTAATCGAACT				
5. GGTCTCCCTCACTCCTAGAC	203.1	39.15	168.84	2.82
6. CATGTGTCGCTCAGGAGCGC	(79.414)	(13.176)	(72.853)	(2.033)
7. TTATTGCGGTTAAATTCTGC				
8. CTCTAGGATCTGCATGACTT				
9. TTAGGAAAAACGCGCACAGT				
10.CGCCATATGCGGGAGGATGT				

continuity. $GC_{content}$ and T_m constraints were embedded in the computation to ensure that the DNA sequences obtained are within the acceptable range.

The results obtained prove that the minimum value of each objective function in DNA sequence design can be obtained using the proposed approach. However, DNA sequence design is multi objective problem, and the objectives should be optimized simultaneously. Therefore, future works will include the solution of multi-objective problem in DNA sequence design using Pareto optimality concept and Vector Evaluated PSO (VEPSO).

Acknowledgment. This research is supported financially by the Ministry of Science, Technology, and Innovation (MOSTI), Malaysia, under eScienceFund Research Funding (Vot 79034).

REFERENCES

[1] M. Arita, A. Nishikawa, M. Hagiya, K. Komiya, H. Gouzu and K. Sakamoto, Improving sequence design for DNA computing, *Proc. of the Genetic Evol. Comput. Conf.*, pp.875-882, 2000.

[2] L. Adleman, Molecular computation of solutions to combinatorial problems, *Science*, vol.266, pp.1021-1024, 1998.

[3] A. J. Hartemink, D. K. Gifford and J. Khodor, Automated constraint based nucleotide sequence selection for DNA computation, *Proc. of the 4th DIMACS Workshop DNA Based Computer*, pp.227-235, 1998.

[4] R. Penchovsky and J. Ackermann, DNA library design for molecular computation, *J. Comput. Bio.,* vol.10, no.2, pp.215-229, 2003.

[5] F. Tanaka, M. Naktsugawa, M. Yamamoto, T. Shiba, and A. Ohuchi, Toward a general-purpose sequence design system in DNA computing, *Proc. of the Congr. Evil. Comput.*, pp.73-78, 2002.

[6] A. Marathe, A. E. Condon, R. M. Corn, On combinatorial DNA word design, *Proc. of the 5th International Meeting on DNA Based Computers*, 1999.

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

[8] M. Arita and S. Kobayashi, DNA sequence design using templates, *New Generation Comput.*, vol.20, pp.263-277, 2002.

[9] A. G. Frutos, A. J. Thiel, A. E. Condon, L. M. Smith and R. M. Corn, DNA computing at surfaces: Four base mismatch word design, *Proc. of the 3rd DIMACS Workshop DNA Based Computer*, pp.238, 1997.

[10] R. Deaton, R. C. Murphy, M. Garzon, D. T. Franceschetti, S. E. Stevens Jr., Good encodings for DNA-based solutions to combinatorial problems, *Proc. of the Second Annual Meeting on DNA Based Computers*, Princeton University, pp.159-171, 1996.

[11] R. Deaton, R. C. Murphy, J. A. Rose, M. Garzon, D. T. Franceschetti, S. E. Stevens Jr., Genetic search for reliable encodings for DNA-based computation, *Proc. of the First Conference on Genetic Programming*, 1996.

[12] S. Y. Shin, I. H. Lee, D. Kim and B. T. Zhang, Multi-objective evolutionary optimization of DNA sequences for reliable DNA computing, *IEEE Transaction on Evolutionary Computation*, pp.143-158, 2005.

[13] T. B. Kurniawan, N. K. Khalid, Z. Ibrahim, M. Khalid and M. Middendorf, An ant colony system for DNA sequence design based on thermodynamics, *Proc. of the Fourth IASTED International Conference Advances in Computer Science and Technology*, Langkawi, Malaysia, pp.144-149, 2008.

[14] J. Kennedy and R. C. Eberhart, Particle swarm optimization, *Proc. of the IEEE International Conference on Neural Networks*, Perth, Australia, pp.1942-1948, 1995.