Available online at www.sciencedirect.com

ScienceDirect

Progress in Natural Science 18 (2008) 623–627

Progress in Natural Science

www.elsevier.com/locate/pnsc

Special report

Improved taboo search algorithm for designing DNA sequences

Kai Zhang [a], Jin Xu [a,b], Xiutang Geng [a], Jianhua Xiao [a], Linqiang Pan [a,*]

[a] *The Key Laboratory of Image Processing and Intelligent Control, Department of Control Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China*

[b] *School of Electronic Engineering and Computer Science, Peking University, Beijing 100871, China*

Received 21 December 2007; received in revised form 10 January 2008; accepted 14 January 2008

Abstract

The design of DNA sequences is one of the most practical and important research topics in DNA computing. We adopt taboo search algorithm and improve the method for the systematic design of equal-length DNA sequences, which can satisfy certain combinatorial and thermodynamic constraints. Using taboo search algorithm, our method can avoid trapping into local optimization and can find a set of good DNA sequences satisfying required constraints.

Keywords: DNA computing; Taboo search algorithm; DNA sequence design

1. Introduction

Since Adleman [1] presented the experiment of using molecular biology to solve a 7-vertex instance of Hamiltonian path problem in 1994, DNA computing shows a great potential to solve the NP-complete problems. DNA sequences design is one of the most practical and important research topics in DNA computing. In order to obtain successful results of biological experiments, effective DNA sequences must be designed for target computational problem. There has been a great deal of previous work in designing DNA sequences [2–5]. In particular, Frutos et al. [6,7] proposed the template-map method for DNA word design. Feldkamp [8] demonstrated a DNA sequence compiler algorithm for designing DNA sequences. Deaton et al. [9–11] presented a genetic algorithm for generating DNA strands. However, the obvious disadvantage of the current DNA generator algorithm is the possibility of being trapped into local optimization which may be far from the global optimal solution.

Taboo search is a general technique proposed by Glover [12,13] for obtaining approximate solutions to combinatorial optimization problems. Taboo search avoids being trapped into local minimum by allowing the temporal acceptance of worse solution. And it has been successfully applied to a wide range of combinatorial optimization problems such as job shop scheduling problem [14], graph coloring problem [15], and maximum independent set problem [16–18]. It provides encouraging optimization performance.

In this paper, we adopt taboo search algorithm to solve the DNA-encoding problem, and improve the algorithm to integrate the required combinatorial and thermodynamic constraints for DNA computing.

2. Problem description

Let $W = 5'\text{-}w_1 w_2 \cdots w_n\text{-}3'$ be a DNA word, where w_i belongs to the alphabet set $\{A, C, G, T\}$. The goal of the problem is to design a set of DNA words with equal-length

[*] Corresponding author. Tel.: +86 27 87556070; fax: +86 27 87543130. *E-mail address:* lqpan@mail.hust.edu.cn (L. Pan).

1002-0071/$ - see front matter © 2008 National Natural Science Foundation of China and Chinese Academy of Sciences. Published by Elsevier Limited and Science in China Press. All rights reserved.
doi:10.1016/j.pnsc.2008.01.005

n, satisfying certain combinatorial constraints and thermodynamic constraints. Six kinds of common constraints are considered in this study.

2.1. Hamming distance constraint

The Hamming distance between two binary strings is the number of corresponding places where two characters differ. In DNA coding, Hamming distance is used to describe the non-similar degree between two DNA sequences, with the greater the Hamming distance, the less similar the degree of two base pairs and the less likely for mismatch hybridization.

For every pair of distinct words $W1$, $W2$ in the set, $H(W1, W2) \geqslant d$. Here, $H(W1, W2)$ represents the Hamming distance between words $W1$ and $W2$, namely, the number of positions i at which the ith letter in $W1$ differs from the ith letter in $W2$. $H(W1, W2)$ is given by

$$H(W_1, W_2) = \sum_{i=1}^{n} h(w_{1i}, w_{2i})$$
$$h(w_{1i}, w_{2i}) = \begin{cases} 0, & \text{if } w_{1i} = w_{2i} \\ 1, & \text{else} \end{cases} \tag{1}$$

2.2. Similarity constraint

The similarity constraint is used to describe a similar degree between two DNA sequences. The similarity constraint computes the similarity in the same direction to keep each sequence as unique as possible including the position shift. Similarity between two binary strings is the number of the corresponding places where two characters are the same.

Similarity between two DNA words $W1$, $W2$ is given by

$$S(W_1, W_2) = \min_{-n<k<n} H(W_1, \sigma^k(W_2)) \tag{2}$$

where σ is the (right-) left-shift and $H(^*,^*)$ is the ordinary Hamming distance.

2.3. GC content constraint

The GC content is the percentage of bases in any word $W \in S$ which is either G or C. The GC content affects the thermodynamic properties of a DNA molecule. Therefore, if all these words will ensure similar GC content, all DNA sequences must have similar thermodynamic characteristics which can effectively reduce the probability of the occurrence of non-specific hybridization.

2.4. H-measure constraint

The H-measure constraint considers two sequences as complementary ones. H-measure computes how many nucleotides are complementary between the given sequences to prevent cross-hybridization of two sequences. H-measure takes the minimum of all the Hamming distances obtained by successively shifting and lining up the Watson–Crick complement of $W2$ against $W1$.

H-measure between two DNA words $W1$, $W2$ is given by

$$H_{\text{measure}}(W_1, W_2) = \min_{-n<k<n} H(W_1, \sigma^k(\overline{W_2})) \tag{3}$$

where σ is the (right-) left-shift and $H(^*,^*)$ is the ordinary Hamming distance.

2.5. Forbidden subsequence constraint

For any word $W \in S$, the word must not contain any given undesired subsequences through the whole strand. In some biological experiments, some subsequences are reserved for special purposes. For example, special DNA subsequences such as the restriction enzyme site should be controlled. In addition, most biological experiments must avoid the hairpin structure and the dimer structure, and such subsequences as ‘AAA’, ‘TTT’, ‘CCC’, ‘GGG’, ‘CATG’, ‘TCGA’, ‘AGCT’, and ‘GTAC’ are forbidden.

2.6. Hairpin constraint

Hairpin is an undesirable DNA secondary structure, because it can hybridize itself. In order to make the hybridization between a DNA word and its Watson–Crick complement more efficient, the single DNA word should be hairpin structure-free. Hairpin structure consists of the ring part and the stem part. The length of the typical minimum hairpin stem is 3.

3. Taboo search algorithm for DNA encoding

The taboo search has been successfully applied to a large number of combinatorial optimization problems. The algorithm is a well-known hill-climbing heuristic, which uses a memory function to avoid being trapped at a local minimum.

The algorithm procedures are generally simple. The procedure starts with a feasible initial DNA word with required length and stores the candidate solution as the current seed. Then the neighbor DNA words of the current seed are produced by the neighborhood structure. These neighbor DNA words are candidate solutions. These DNA words are evaluated by certain combinatorial constraints and thermodynamic constraints, and a candidate which satisfies the aspiration criterion is selected as a new seed solution. This selection is called a move and added to the taboo list. Iterations are repeated until a stop criterion has been satisfied. Fig. 1 shows the procedure of taboo search algorithm.

3.1. Initial solution

The initial solution is a nucleotide word string generated by a random method. Then we use the GC content constraint

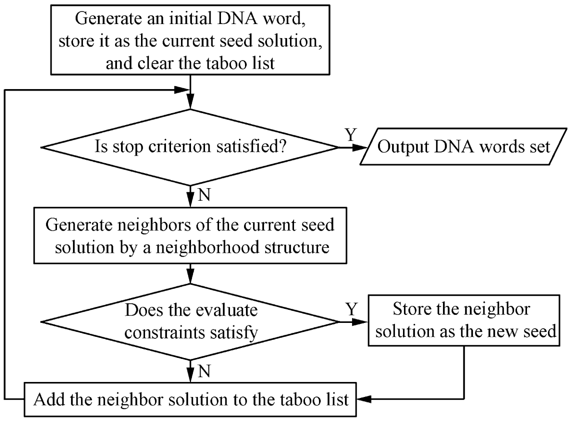
Generate an initial DNA word, store it as the current seed solution, and clear the taboo list
Is stop criterion satisfied?
Y
Output DNA words set
N
Generate neighbors of the current seed solution by a neighborhood structure
Does the evaluate constraints satisfy
Y
Store the neighbor solution as the new seed
N
Add the neighbor solution to the taboo list

Fig. 1. The flow diagram of the taboo search algorithm.

and the forbidden subsequence constraint to test the initialization. If the initial solution fits all constraints, it will be added to the taboo list and stored as the current seed solution.

3.2. The neighborhood structure

A neighborhood structure is a mechanism which can obtain a set of new neighbor solutions by applying a diversification strategy to a given solution. Each neighbor solution is reached immediately from a seed solution by a move. The neighborhood structure is directly effective on the efficiency of TS algorithm.

In this study, the neighborhood solutions are generated by exchanging the sequence nucleotide bases. The whole neighborhood representation is about a permutation of all DNA words which fits Hamming distance constraint. This significantly facilitates the data structure, since every solution may be stored by means of a permutation of the nucleotide alphabet $\{A, C, G, T\}$ with the length of n.

For a given solution word $W = 5'\text{-}w_1w_2\cdots w_i\ldots w_n\text{-}3'$ replaces w_i by another nucleotide base k_i, $w_i \neq k_i$, $k_i \in \{A, C, G, T\}$. The new word is $W = 5'\text{-}w_1w_2\cdots ki\cdots w_n\text{-}3'$ and the Hamming distance $H(W, W')$ is 1. If the required Hamming distance is h, replace $(w_{i1}, w_{i2}, \ldots, w_{ih})$ with $(k_{i1}, k_{i2}, \ldots, k_{ih})$ at different positions; the Hamming distance between two words $H(W, W)$ is h. If the required DNA word length is n and the required Hamming distance is h, we change h different nuclide bases of seed word W randomly, and such neighborhood structure can construct almost C_n^h neighbor candidate solutions.

3.3. Evaluating neighbor solution

Each generated DNA words must be evaluated by combinatorial constraints and thermodynamic constraints. If the word violates any constraint, the word will be added to taboo list.

For each solution $W \in S$, let $f_{\text{H-measure}}(W)$, $f_{\text{GC}}(W)$, $f_S(W)$, $f_{\text{FS}}(W)$, $f_{HP}(W)$ be the H-measure constraint function, the GC content constraint function, the similarity constraint function, the forbidden subsequence constraint function, and the hairpin constraint function, respectively. When moving from a solution W to another solution $W' \in N(W)$, the new solution is evaluated by the following functions:

(i) $f_{\text{H-measure}}(W) = \sum_{i=1}^{n}\sum_{j=1}^{n} H_{\text{measure}}(W_i, W_j)$ computes all the H-measure (Wi, Wj) values. If the function $f_{\text{H-measure}}(W)$ does not satisfy the target value, new DNA sequence W should be neglected and added to the taboo list.

(ii) $f_{GC}(W) = (GC_{\text{target}} - \text{GC}(W_i))^2$ calculates the GC content of the word W. If $f_{GC}(W)$ does not satisfy the required GC content, the tested sequence should be added to the taboo list.

(iii) $f_S(W) = \sum_{i=1}^{n}\sum_{j=1}^{n} S(W_i, W_j)$ computes all $S(Wi, Wj)$ values. If the function $f_S(W)$ does not satisfy target value, new DNA sequence W should be neglected and added to the taboo list.

(iv) $f_{FS}(W) = \sum_{i=1}^{n}\sum_{j=1}^{n} S(W, Subsequence)$ searches for all forbidden subsequences. If the function $f_{FS}(W)$ does not satisfy target value, the candidate sequence should be added to the taboo list.

(v) $f_{HP}(W)$ tests the Hairpin constraint. If $f_{HP}(W)$ does not satisfy the minimum stem length, the tested sequence should be added to the taboo list. Fig. 2 shows the hairpin secondary structure parameters.

$$f_{HP}(W) = HP_{\text{target}} - \sum_{i=0}^{s+k-1} bp(w_i, w_{k+2s+r-1-i}),$$

$$bp(x, y) = \begin{cases} 1, x = \bar{y} \\ 0, x \neq \bar{y} \end{cases}$$

$$3 < r < n - 2s, 0 < k < n - 2s - r$$

Because all constraints are parallel, the five terms listed above are logical multiplication. Solutions are evaluated using a function

$$F(W) = f_S(W)f_H(W)f_{GC}(W)f_{FS}(W)f_{HP}(W) \tag{4}$$

3.4. Taboo list and termination criterion

Because evaluating a neighbor solution will take much time, if a neighbor solution violates any combina-

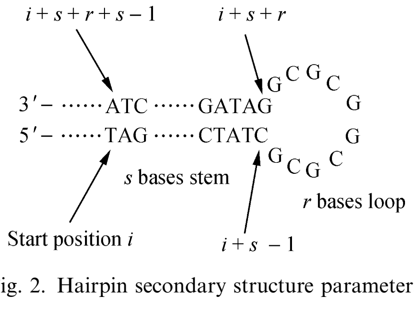

Fig. 2. Hairpin secondary structure parameters.

Table 1
Comparison results of the sequences by our algorithm and the sequences in Refs. [10,19]

Sequence (5′–3′)	Continuity	Hairpin	H-measure	Similarity	GC%
Our algorithm					
CCACCACCACCACCAATAAT	0	0	37	55	50
ACCTCACTCACTCACTCAAC	0	0	44	64	50
TAACAGAACAGAACAGGCCG	0	0	53	48	50
CACACACACACACACACACA	0	0	33	54	50
AATCTCTCTCTCTCTCTGCC	0	0	46	55	50
CGCCAGCCAGCCTATATATA	0	0	59	54	50
TTGCATTCCTTCCTTCCTGG	0	0	54	46	50
Deaton et al. [10]					
ATAGAGTGGATAGTTCTGGG	9	3	55	64	45
CATTGGCGGCGCGTAGGCTT	0	0	69	51	65
CTTGTGACCGCTTCTGGGGA	16	0	60	63	60
GAAAAAGGACCAAAAGAGAG	41	0	58	45	40
GATGGTGCTTAGAGAAGTGG	0	0	58	54	50
TGTATCTCGTTTTAACATCC	16	4	61	50	35
TTGTAAGCCTACTGCGTGAC	0	3	75	55	50
Shin et al. [19]					
CTCTTCATCCACCTCTTCTC	0	0	43	58	50
CTCTCATCTCTCCGTTCTTC	0	0	37	58	50
TATCCTGTGGTGTCCTTCCT	0	0	45	57	50
ATTCTGTTCCGTTGCGTGTC	0	0	52	56	50
TCTCTTACGTTGGTTGGCTG	0	0	51	53	50
GTATTCCAAGCGTCCGTGTT	0	0	55	49	50
AAACCTCCACCAACACACCA	9	0	55	43	50

torial or thermodynamic constraint, it will be added to the taboo list to avoid repeat testing. The algorithm ends when the number of iterations reaches a maximum value, or the solution DNA word set has got enough DNA words.

4. Simulation results

In the above section, the taboo search algorithm was improved to design good DNA sequences. The algorithm has been implemented on Pascal language compiler of Borland Delphi 7. In the simulation, DNA sequences of length 20-mer are considered. The neighborhood structure Hamming distance is 9. The subsequences ‘AAA’, ‘CCC’, ‘GGG’, and ‘TTT’ could guarantee that each DNA sequence must satisfy the continuity constraint. We assumed that the hairpin formation requires at least a six-base loop and six-base pairings.

We choose the best DNA sequences set generated by our algorithm and compared the results with those of Deaton et al. [10] and Shin et al. [19] The comparison results are shown in Table 1. Our sequences show much lower similarity values and H-measure. This implies that the sequences made by our algorithm have much higher probability to hybridize with its complementary sequences. Moreover, the secondary structure is strictly prohibited due to the very low continuity and hairpin. *GC* content ensures that these DNA sequences have similar thermodynamic characteristics.

5. Conclusion

A new algorithm of DNA sequence design for DNA computing has been proposed. Because the neighbor structure in this algorithm can overlap the whole solution space, our algorithm can generate one of the greatest DNA sequence sets satisfying the required constraints. Because all constraints in our algorithm are parallel, additional constraints are supported by the algorithm and can be integrated into our model in a straightforward way.

Acknowledgements

This work was supported by the National High Technology Research and Development Program (863 Program) of China (Grant No. 2006AA01Z104), the National Natural Science Foundation of China (Grant Nos. 60373089, 60674106, 30570431, and 60533010), the Program for New Century Excellent Talents in University (Grant No. NCET-05-0612), the Ph.D. Programs Foundation of Ministry of Education of China (Grant No. 20060487014), the Chenguang Program of Wuhan (Grant No. 200750731262), and HUST-SRF (Grant No. 2007Z015A).

References

[1] Adelman LM. Molecular computation of solutions to combinatorial problems. Science 1994;266:1021–4.

[2] Brennerman A, Condon A. Strand design for bio-molecular computation. Theor Comput Sci 2001;287:39–58.

[3] Marathe A, Condon A, Corn RM. On combinatorial DNA word design. J Comput Biol 2001;8:201–19.

[4] Deaton R, Garzon M, Murphy R, et al. Genetic search of reliable encodings for DNA based computation. In: Proceedings of the 1st annual conference on genetic programming, vol. 21;1996. p. 9–15.

[5] Shoemaker D, Lashkari DA, Morris D, et al. Quantitative phenotypic analysis of yeast deletion mutants using a highly parallel molecular bar-coding strategy. Nature 1996;16:450–6.

[6] Frutos AG, Smith LM, Corn RM. Enzymatic ligation reactions of DNA "Word" on surfaces for DNA computing. J Am Chem Soc 1998;120(40):10277–82.

[7] Frutos AG, Liu QH, Thiel AT, et al. Demonstration of a word design strategy for DNA computing on surface. Nucleic Acids Res 1997;25:4748–57.

[8] Feldkamp U, Banzhaf W, Rauhe H. A DNA sequence compiler. In: Proceedings of the 6th DIMACS workshop on DNA based computers, 2000. p. 253.

[9] Deaton R, Garzon M, Murphy RC, et al. Genetic search of reliable encodings for DNA-based computation. In: 1ST Genetic Programming Conference, 1996. p. 9–15.

[10] Deaton R, Murphy RC, Garzon M, et al. Good encodings for DNA-based solutions to combinatorial problems. DIMACS Ser Discrete Math Theor Comput Sci 1999;44:247–58.

[11] Deaton R, Murphy RC, Rose JA, et al. A DNA based implementation of an evolutionary computation. In: Proceedings IEEE conference on evolutionary computation, 1997. p. 267–71.

[12] Glover F. Tabu search part I. ORSA J Comput 1989;190–206.

[13] Glover F. Tabu search part II. ORSA J Comput 1990;4–32.

[14] Hertz A, Widmer M. An improved tabu search approach for solving the job shop scheduling problem with tooling constraints. Discrete Appl Math 1996;65(3):319–45.

[15] Blochligera I, Zuffereyb N. A graph coloring heuristic using partial solutions and a reactive tabu scheme. Comput Oper Res 2006;14(5):1–16.

[16] Hansen MP. Tabu search for multi-objective optimization. MOTS. Paper Presented at the 13th International Conference on Multi Criteria Decision Making (MCDMA97) 1997;1:6–10.

[17] Kulturel-Konak S, Coit DW, Smith AE. Efficiently solving the redundancy allocation problem using tabu search. IIE Trans 2003;35(6):515–26.

[18] Friden C, Hertz A, Werra D. An exact algorithm based on tabu search for finding a maximum independent set in a graph. Comput Oper Res 1990;17(5):437–45.

[19] Shin SY, Lee IH, Kim DM, et al. Multi-objective evolutionary optimization of DNA sequences for reliable DNA computing. IEEE Trans Evolut Comput 2005;9(2):143–58.