Reference Point Based Multi-objective Evolutionary Algorithm for DNA Sequence Design

Haozhi Zhao[1](✉), Zhiwei Xu[1](✉), and Kai Zhang[1,2](✉)

[1] School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan 430065, Hubei, China
873674474@qq.com, {xuzhiwei,zhangkai}@wust.edu.cn

[2] Hubei Province Key Laboratory of Intelligent Information Processing and Real-Time Industrial System, Wuhan 430065, Hubei, China

Abstract. DNA computing is a parallel computing model based on DNA molecules. High-quality DNA sequences can prevent unwanted hybridization errors in the computation process. The design of DNA molecules can be regarded as a multi-objective optimization problem, which needs to satisfy a variety of conflicting DNA encoding constraints and objectives. In this paper, a novel reference point based multi-objective optimization algorithm is proposed for designing reliable DNA sequences. In order to obtain balance Similarity and H-measure objective values, the reference point strategy is adapted to searching for idea solutions. Firstly, every individual should be assigned a rank value by the non-dominated sort algorithm. Secondly, the crowding distance is replaced by the distance to the reference point for each individual. Lastly, the proposed algorithm is compared with some state-of-the-art DNA sequence design algorithms. The experimental results show our algorithm can provide more reliability DNA sequences than existing sequence design techniques.

Keywords: DNA encoding · Multi-objective optimization · Reference point

1 Introduction

DNA computing is a new computational paradigm, which has shown great potential to solve NP-complete problems, such as Hamiltonian path problem (HPP) [1], satisfaction problem (SAT) [2], traveling salesman problem (TSP) [3] and graph coloring problem (GCP) [4]. High-quality DNA sequences can improve the efficiency and reliability. Therefore, the design of DNA molecules should be carefully designed to prevent unwanted hybridization errors. The design of DNA

Supported by the National Natural Science Foundation of China (Grant Nos. U1803262, 61702383, 61602350).

L. Pan et al. (Eds.): BIC-TA 2019, CCIS 1160, pp. 178–188, 2020.
https://doi.org/10.1007/978-981-15-3415-7_14

molecules can be regarded as a multi-objective optimization [22–25] problem, which need satisfy a variety of conflicting DNA encoding constraints and objectives [5].

In the past few decades, a lot of efficient algorithms have been proposed to solve DNA sequences design problem. Frutos et al. [6] proposed the Template-Map method, but it is difficult to derive templates and mappings that satisfy the combined constraints when there are many constraints. Hartemink et al. [7] implemented an exhaustive search algorithm to design DNA sequences, which satisfy the constraints, but the algorithm has a high time complexity. Feldkamp [8] uses directed trees to design DNA encoding that the fixed length subsequences are only allowed to appear once, but the length of the sub-sequences requires a lot of testing to determine in the actual design. Recent years, evolutionary algorithms are widely adapted for DNA encoding design, such as genetic algorithm [9,11,14,20], particle swarm optimization [11,15,17,18], ant colony algorithm [12], simulated annealing [16], and multi-objective evolutionary algorithms [10,13,19]. However, existing algorithms often obtain the DNA sequences set with bias Similarity or H-measure values, which is easy to introduce errors during the DNA computing process.

In this paper, a reference point based multi-objective optimization evolutionary algorithm is proposed for designing DNA sequences. Firstly, the non-dominated sort algorithm is adapted to select convergent solutions rank by rank. Secondly, the crowding distance sort algorithm is adapted to choose the solutions that are closer to the reference point. The algorithm can provide a set of DNA sequences near idea point which are more reliable and efficient for DNA computing. Finally, to validate the proposed algorithm, we compare our algorithm with some state-of-art techniques. The experimental results confirm the performance of our algorithm in designing high-quality DNA sequences set efficiently.

In the following chapters, Sect. 2 introduces the relevant basis of DNA sequence design problem. The proposed reference-based evolution algorithm for designing DNA sequences is then detailed in Sect. 3. In Sect. 4, we present the experiment results of the proposed algorithm and compare them with the other literatures. Finally, conclusions are drawn in Sect. 5 along with pertinent observations identified.

2 Problem Formulation

During the process of DNA computing, single-strand DNA molecules dismissed randomly in the vitro, therefore four kinds of molecules exist simultaneity, including DNA molecular X, corresponding complementary sequence X C, reverse sequence X R and reverse complementary sequence X RC. Most existing DNA computing models are based on the specific hybridization between a given molecular X and it's unique Watson-Crick complement X C. In fact, the non-specific hybridization often occurs because unwanted mismatches maybe take place between random molecules, as shown in Fig. 1.

However, the non-specific hybridization could introduce errors, such as false positives and negatives, and degrade efficiency. Obviously, it is important to

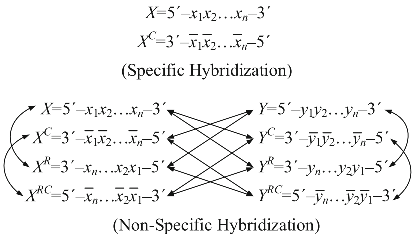
X=5′–$x_1x_2\ldots x_n$–3′
X^C=3′–$\bar{x}_1\bar{x}_2\ldots\bar{x}_n$–5′
(Specific Hybridization)
X=5′–$x_1x_2\ldots x_n$–3′
X^C=3′–$\bar{x}_1\bar{x}_2\ldots\bar{x}_n$–5′
X^R=3′–$x_n\ldots x_2x_1$–5′
X^{RC}=5′–$\bar{x}_n\ldots\bar{x}_2\bar{x}_1$–3′
Y=5′–$y_1y_2\ldots y_n$–3′
Y^C=3′–$\bar{y}_1\bar{y}_2\ldots\bar{y}_n$–5′
Y^R=3′–$y_n\ldots y_2y_1$–5′
Y^{RC}=5′–$\bar{y}_n\ldots\bar{y}_2\bar{y}_1$–3′
(Non-Specific Hybridization)

Fig. 1. Specific and non-specific hybridizations.

design reliable DNA sequences for DNA computing, and the key is to avoid the non-specific hybridization. The design of reliable DNA sequences involves several conflicting design constraints which have to be considered simultaneously. In mathematical terms, DNA sequence design problem can be formulated as a multi-objective optimization problem as Eq. (1).

$$\min f(X) = \min \left[f_1(X), f_2(X), \cdots, f_M(X)\right]^T, f(X) \in R^M \tag{1}$$

where DNA sequence $X = [x_1, x_2, \cdots, x_N]^T \in \Omega$ consists of N bases $x_i \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$, and the search space Ω is 4^N. $f(X)$ consists of M objective functions $f_m(x)$, $m = 1, \ldots, M$. R^M denotes the objective space. Several typical biochemical design criteria are chosen that other relevant authors use to evaluate and generate reliable DNA libraries. The formal definition for each design criteria is provided in the following subsections.

2.1 Similarity Criterion

Let X_i and X_j be two different DNA sequences, the similarity criterion refers to the degree of similarity in base composition between X_i and X_j. By controlling the similarity, non-specific hybridization between X_i and the complementary of X_j, (i.e. X_j^C). The calculation of Similarity is shown in Eq. (2).

$$\begin{aligned} f_{\text{Similariy}}(X) &= \sum_{i=1}^{n}\sum_{j=1}^{n} \text{ Similarity } (X_i, X_j) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{n} \text{Max}_{g,i}\left(Si_{dis}(X_i, X_j, s) + Si_{con}(X_i, X_j, s)\right) \end{aligned} \tag{2}$$

where the function $Max_{g,i}$ represents traversing all possible values of g and i and taking the maximum value as a result. The function $Si_{dis}(X_i, X_j, s)$ represents the number of identical bases in which the DNA sequence X_i is shifted to the right by the s position compared with the sequence X_j. The function $Si_{con}(X_i, X_j, s)$ represents that the DNA sequence X_i shifts to the right by the s bit and the base compared with X_j is continuously the same penalty value.

2.2 H-Measure Criterion

For DNA sequences X and Y, the H-measure constraint is to limit non-specific hybridization between X and reverse Y. The calculation of H-measure is as shown in Eq. (3).

$$\begin{aligned} f_{H\text{-measioe}}(X) &= \sum_{i=1}^{n} \sum_{j=1}^{n} H\text{-measure}\left(X_i, X_j\right) \\ &= \sum_{i=1}^{n} \sum_{j=1}^{n} \mathrm{Max}_{g,i}\left(h_{dis}\left(X_i, X_j^{\mathrm{R}}, s\right) + h_{con}\left(X_i, X_j^{\mathrm{R}}, s\right)\right) \end{aligned} \tag{3}$$

where the function $Max_{g,i}$ represents traversing all possible values of g and i and taking the maximum value as a result. The function $h_{ds}\left(X_i, X_j^{\mathrm{R}}, s\right)$ represents the number of base complements in which the DNA sequence X_i shifts to the right by the s-bit compared with the sequence X_j. The function $h_{con}\left(X_i, X_j^{\mathrm{R}}, s\right)$ represents a base continuous pairing penalty value in which the DNA sequence X_i is shifted to the right by the s-bit compared with X_j.

Similarity Criterion describes the degree of similarity between DNA sequences, and H-measure Criterion describes the degree of complementary hybridization between DNA sequences. Similarity Criterion and H-measure Criterion are two conflicting objectives, and they are difficult to optimize at the same time. Shin et al. [10] had proved that Similarity Criterion and H-measure Criterion are conflicting, and they are both discontinuous functions and have many locally optimal solutions.

2.3 Continuity Criterion

Continuity constraint means that in the single strand of DNA, the same base appears continuously, and an undesired secondary structure occurs under the hydrogen bonding force of the base molecule. The calculation of Continuity is as shown in Eq. (4).

$$\begin{aligned} f_{\text{Continuity}}(X) &= \sum_{i=1}^{n} \text{ Continuity }\ \left(X_i\right) \\ &= \sum_{i=1}^{n} \sum_{i=1}^{l-t+1} T\left(c_a(x, i), t^2\right) \end{aligned} \tag{4}$$

2.4 Hairpin Structure Criterion

The hairpin structure constraint refers to a single-stranded DNA molecule formed by reverse folding of itself, resulting in a secondary structure of a hairpin shape. The calculation of Hairpin is as shown in Eq. (5).

$$\begin{aligned} f_{\text{Hairpin}}(X) &= \sum_{i=1}^{n} \text{ Hairpin } \left(X_i\right) \\ &= \sum_{i=1}^{n} \sum_{s=s_{mn}}^{(l/R_{mn})/2} \sum_{r=R_{mn}}^{l-2s} \sum_{i=1}^{l-2s-r} T\left(\sum_{j=1}^{s} bp\left(x_{s+i-j}, x_{s+i+r+j}\right), \frac{s}{2}\right) \end{aligned} \tag{5}$$

2.5 GC Content Criterion

DNA computing prefer the DNA molecules with uniform GC content. The GC content refers to the number or percentage of bases G and bases C in the DNA sequence. The calculation of GC% is as shown in Eq. (6).

$$\begin{aligned} & f_{GC}(X) = \max_i \left\{\text{GC}\left(X_i\right)\right\} - \min_j \left\{\text{GC}\left(X_j\right)\right\} \\ & \text{GC} = \textstyle\sum_{i=1}^{n} \sum_{i=1}^{l} gc\left(x_i\right), \text{gc}\left(x_i\right) = \begin{cases} 1, x_i = G \text{ or } x_i = C \\ 0, x_i = A \text{ or } x_i = T \end{cases} \end{aligned} \tag{6}$$

2.6 Melting Temperature Criterion

The melting temperature is the temperature at which 50% of the DNA molecules open the double strand into a single strand during the warming denaturation of the double-stranded DNA molecule. The melting temperature is an important parameter for evaluating the thermodynamic stability of DNA molecules. The calculation of Tm is as shown in Eq. (7).

$$\begin{aligned} & f_{Tm}(X) = \max_i \left\{\text{Tm}\left(X_i\right)\right\} - \min_j \left\{\text{Tm}\left(X_j\right)\right\} \\ & \text{Tm}\left(X_i\right) = \textstyle\sum_{i=1}^{n} \frac{\Delta \text{H}^{\circ}}{\Delta S^{\circ} + R \ln\left(\left|C_T\right|/4\right)} \end{aligned} \tag{7}$$

ΔH° is the total enthalpy of the adjacent base, ΔS° is the total entropy of the adjacent base, R is the gas constant (1.987 cal/Kmol), and C_T is the DNA molecule concentration.

3 Problem Formulation

Because Similarity and H-measure are two conflict objectives, we would obtain a set of non-dominated solutions using MOEA. However, the DNA sequences which have high Similarity values will lead to non-specific hybridization between X and the complementary strand Y^C. Moreover, the DNA sequences which have high H-measure values will lead to non-specific hybridization between X and the reverse strand Y^R. Among the whole PF, only the nonbiased point is the idea solutions for DNA sequences design problem, as shown in Fig. 2.

In response to the above problems, we adopt R-NSGA-II [21] to search for DNA sequences, in which the crowding distance is replaced by the distance to the reference point. In our algorithm, the reference distance (RD) can be calculated as shown in Eq. (8).

$$RD = \sqrt{\sum_{i=1}^{m}\left(\frac{f_i(x) - R_i}{f_i^{max} - f_i^{min}}\right)^2} \tag{8}$$

where f_i^{max} and f_i^{min} are the global maximum and minimum function values of the i-th objective function, and R_i is the reference value of the i-th objective. In our algorithm, the reference point is set to $R = (f_1, f_2, f_3, f_4, f_5, f_6) = \left(0, 0, 0, 0, \frac{N}{2}, 50\right)$.

Most of the algorithms calculate the objective functions on the entire population P_t. However, two objective functions Similarity an H-measure are the full correlation with all the individuals. If k DNA sequences with best fitness values are selected for DNA computing, they may not remain optimal. In our algorithm, we re-evaluate the individuals with the population P_{t+1}, and update the fitness values in P_t one by one. Three main procedures are iteratively run in our algorithm, specifically the non-dominated sort, the reference crowding distance sort, and full correlation fitness update. The algorithm procedure is also shown in Fig. 3.

Firstly, tournament selection is adapted on Pt, and the winner of two randomly selected individuals should be added into mating pool Qt. Then, crossover and mutation operators are adapted to generate new offspring, and replace the individuals in mating pool Qt. Secondly, the non-dominated sorting is applied to the union set PtQt, and the non-dominated fronts are copied to parent population rank by rank. Thirdly, the reference distance should be calculated for every individual, and the individual with minimum reference distance could be added into the new population $\mathrm{Pt} + 1$ until the population size N. Moreover, in order to select the individuals with optimal full correlation objective values, we re-evaluate the population Pt when individual is selected and added into $\mathrm{Pt} + 1$. The pseudocode is shown as Algorithm 1.

Algorithm 1. *Proposed Algorithm*

```
Initialization $P_0$
while (stopping criterion is not satisfied) do
    $Q_t$ = Tournament Selection ($P_t$)
    $Q_t$ = Crossover and Mutation ($P_t$)
    $R_t = P_t \cup Q_t$
    EvaluatePopulation on $R_t$
    Non-dominated Sort ($R_t$)
    for $i = 0$ to $P_t$ do
        Reference Distance calculate ($R_t$)
        $P_{t+1} = P_{t+1}$ + Nearest Individual with min($R_t$)
        Re-EvaluatePopulation on $P_{t+1}$
    end for
end while
```

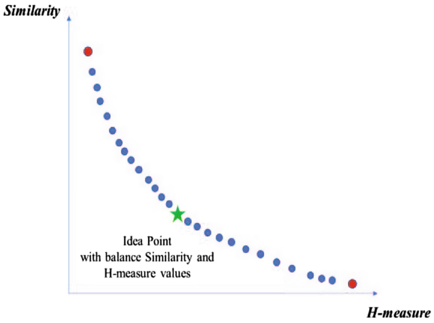

Fig. 2. Idea solution within the non-dominated solutions.

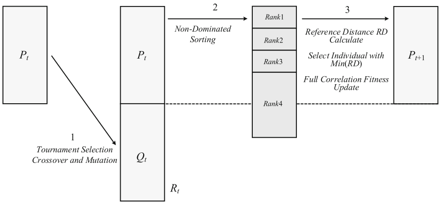

Fig. 3. The procedure of our algorithm.

4 Result and Discussion

In order to verify the effectiveness of the proposed algorithm, we compare the obtained results with various known algorithms. In our comparison, the population size is set to 200, the DNA length is set to 20, and the maximum number of iterations is set to 1000. The algorithm is implemented in Eclipse Java and tested on a PC (running environment intel® $Core^{TM}$ i5-8400 CPU @ 2.802 GHz, 8G RAM, Windows 10).

Table 1 shows the obtained sequences generated by MGA [20], NACST/Seq [10], and our algorithm. As can be seen from Table 1, all the algorithms obtain same Hairpin and GC content values. The sequences of MGA and our algorithm have same Continuity values, which are better than the sequences of NACST/Seq. The MGA has most uniform melting temperature values fluctuated within one degree Celsius. The temperature fluctuation range of our sequences is ±1.2526 °C, which is better than NACST/Seq.

The Similarity value of our algorithm is 290, which is much smaller than MGA(444) and NACST/Seq(374). In addition, the H-measure value of our algorithm is 284, which is also much smaller than MGA(438) and NACST/Seq(338). Moreover, the balanced Similarity and H-measure values imply that our sequences are more reliable and have a lower probability of unwanted non-hybridization.

Table 2 shows obtained larger group of DNA sequences by three compared algorithms. As can be seen from Table 2, all the algorithms obtain same Continuity and GC content values. Our algorithm obtains best Hairpin objective value, however, ten sequences in MGA set have poor Hairpin objective values. The sequences of our algorithm have most uniform melting temperature values fluctuated within ±0.9002 °C. The melting temperature of MGA and NACST/Seq fluctuate in range ±1.7500 and ±2.9574 respectively. The H-measure and Similarity values of the sequences designed by our algorithm are balance and much lower than compared algorithms, which means the mismatch and non-hybridization between the coding sequences can be greatly reduced.

Table 1. Comparison results of the obtained seven sequences with 20 bases.

Sequence	Continuity	Hairpin	H-measure	Similarity	Tm	GC%
MGA [20]						
TAGACCACTGTTGCACATGG	0	0	58	52	56.0900	50
ATTCGGTCAGACTTGCTGTG	0	0	64	52	56.2400	50
ATAGTGCGGACAGTAGTTCC	0	0	66	59	54.9200	50
AATACGCGGAACGTAACCTC	0	0	61	85	55.8300	50
AATACGCGGAACGTAACCTC	0	0	61	85	55.4000	50
ACAGCCTTAAGCCTAACTCC	0	0	65	54	56.0641	50
ATGCTTCCGACATGGAATGG	0	0	63	57	55.8500	50
Objective values	0	0	438	444	55.5800 (±0.6600)	50 (±0)
NACST/Seq [10]						
CTCTTCATCCACCTCTTCTC	0	0	43	58	46.6803	50
CTCTCATCTCTCCGTTCTTC	0	0	37	58	46.9393	50
TATCCTGTGGTGTCCTTCCT	0	0	45	57	49.1066	50
ATTCTGTTCCGTTGCGTGTC	0	0	52	56	51.1380	50
TCTCTTACGTTGGTTGGCTG	0	0	51	53	49.9252	50
GTATTCCAAGCGTCCGTGTT	0	0	55	49	50.7224	50
AAACCTCCACCAACACACCA	9	0	55	43	51.4735	50
Objective values	9	0	338	374	49.0769 (±2.3966)	50 (±0)
Our algorithm						
ACAACAACCACCACCACCAA	0	0	37	45	50.2236	50
CCAAGGAAGGAAGGAAGGAA	0	0	54	33	49.0486	50
CCTCTCCTCTTCTTATCTCC	0	0	34	49	49.6556	50
GTGTGTGTGTGTGTGTGTGT	0	0	48	25	50.9244	50
CCAACCAACCAACCAACCAA	0	0	34	45	51.3054	50
CTTCTTCCTCCTTCTTCTCC	0	0	36	45	48.8003	50
CTCTCGCTCTATATCTCTCC	0	0	41	48	49.4115	50
Objective values	0	0	284	290	50.0529 (±1.2526)	50 (±0)

Table 2. Comparison results of the obtained fourteen sequences with 20 bases.

Sequence	Continuity	Hairpin	H-measure	Similarity	Tm	GC%
MGA [20]						
CTCATCTAATCAGCCTCGCA	0	0	135	114	55.2900	50
CTAATAGTGACAGCTGCGTG	0	3	131	119	53.9200	50
GCATCGTTAGAGACACCTAC	0	3	134	124	53.1000	50
GCATCAATATGCGCGACTAC	0	0	131	125	54.8700	50
CATTAAGTAGACGCTGTCGG	0	3	132	114	53.6100	50
TATGGATGAGGAGGACCTAG	0	3	133	117	53.2300	50
CAGAGATGTTCTGTACCACC	0	3	128	117	53.2000	50
CGTCGAGAATTCGTAGCTCA	0	0	137	119	55.1300	50
TCTGTTACCGTATCGGATCG	0	3	129	115	54.4900	50
AGAAGAGTTCGACTTGCTGG	0	3	134	121	55.6300	50
GCAAGGAATTCACCGTCTGT	0	3	133	129	56.6000	50
CGTGTGAAGAGAGTGGTTCA	0	0	127	123	55.5000	50
CGACTGAATCATGGACCTGT	0	3	134	126	55.5300	50
TACCGAGAAGTAGGACTGCA	0	3	134	124	56.0100	50
Objective values	0	30	1852	1687	54.8500 (±1.7500)	50 (±0)
NACST/Seq [10]						
GTGACTTGAGGTAGGTAGGA	0	3	129	115	47.2490	50
ATCATACTCCGGAGACTACC	0	3	132	121	47.2304	50
CACGTCCTACTACCTTCAAC	0	0	128	121	47.4589	50
ACACGCGTGCATATAGGCAA	0	3	141	117	52.5401	50
AAGTCTGCACGGATTCCTGA	0	3	132	115	50.5497	50
AGGCCGAAGTTGACGTAAGA	0	0	132	116	51.0482	50
CGACACTTGTAGCACACCTT	0	0	132	123	50.2683	50
TGGCGCTCTACCGTTGAATT	0	0	135	116	52.0565	50
CTAGAAGGATAGGCGATACG	0	0	134	117	46.6253	50
CTTGGTGCGTTCTGTGTACA	0	0	140	116	50.5774	50
TGCCAACGGTCTCAACATGA	0	0	132	121	51.8587	50
TTATCTCCATAGCTCCAGGC	0	0	136	117	48.1017	50
TGAACGAGCATCACCAACTC	0	0	121	121	50.3351	50
CTAGATTAGCGGCCATAACC	0	0	127	119	47.6383	50
Objective values	0	12	1851	1655	49.2420 (±2.9574)	50 (±0)
Our algorithm						
GAGAATAGAGAAGGAGGAGG	0	0	84	115	49.6556	50
TGTTGTGGTGTGGTGTGGTT	0	0	124	80	50.1562	50
GAAGGAAGGAAGGAAGGAAG	0	0	77	106	49.4336	50
GAGAGTGAGAGGATAAGAGG	0	0	91	112	49.5929	50
TTGTTCTGGTGGTGGTGGTT	0	0	116	82	49.6702	50
GTTGGTTGGTTGGCTTGGTT	0	0	113	84	50.1442	50
CACACGCACAGACATACACA	0	0	99	98	50.2702	50
GGAAGAGCAATAGCAGAAGG	0	0	88	116	49.0941	50
CAACGACCAAGAACGACCAA	0	0	95	109	49.6784	50
AACACATCACACAGCACACC	0	0	103	102	49.9036	50
ACACACCTCACACTCAACAC	0	0	105	97	49.9141	50
CCACACGACACACTACACAA	0	0	102	104	50.8945	50
AACCAGCAACTACCAGCAAC	0	0	103	104	49.2441	50
AATGGAATGGAATGGCGAGG	0	0	100	111	49.8795	50
Objective values	0	0	1400	1420	49.9943 (±0.9002)	50 (±0)

5 Conclusion

In this study, a multi-objective DNA sequence design algorithm had been successfully implemented for reliable DNA computation. The algorithm was based on ideal reference point, which could guide the population to search for balance Similarity and H-measure objective values efficiently. The algorithm was compared with some state-of-the-art approaches. The experimental results showed our algorithm can generate high quality DNA sequences set which satisfied various conflict DNA encoding criterions.

References

1. Yang, R., Zhang, C., Gao, R.: A new bionic method inspired by DNA computation to solve the hamiltonian path problem. In: IEEE International Conference on Information and Automation (ICIA), pp. 219–225. IEEE (2017)
2. Song, B., Pérez-Jiménez, M.J., Pan, L.: An efficient time-free solution to SAT problem by P systems with proteins on membranes. J. Comput. Syst. Sci. **82**(6), 1090–1099 (2016)
3. Wang, X.: Research on solution of TSP based on improved genetic algorithm. In: International Conference on Engineering Simulation and Intelligent Control (ESAIC), pp. 78–82. IEEE (2018)
4. Jafarzadeh, N., Iranmanesh, A.: A new graph theoretical method for analyzing DNA sequences based on genetic codes. MATCH-Commun. Math. Comput. Chem. **75**(3), 731–742 (2016)
5. Chaves-González, J.M., Vega-Rodrgíuez, M.A.: A multiobjective approach based on the behavior of fireflies to generate reliable DNA sequences for molecular computing. Appl. Math. Comput. **227**, 291–308 (2014)
6. Frutos, A.G., Liu, Q., Thiel, A.J., et al.: Demonstration of a word design strategy for DNA computing on surfaces. Nucleic Acids Res. **25**(23), 4748–4757 (1997)
7. Hartemink, A.J., Gifford, D.K., Khodor, J.: Automated constraint-based nucleotide sequence selection for DNA computation. Biosystems **52**(1–3), 227–235 (1999)
8. Feldkamp, U., Saghafi, S., Banzhaf, W., Rauhe, H.: DNASequenceGenerator: a program for the construction of DNA sequences. In: Jonoska, N., Seeman, N.C. (eds.) DNA 2001. LNCS, vol. 2340, pp. 23–32. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-48017-X_3
9. Arita, M., Nishikawa, A., Hagiya, M., et al.: Improving sequence design for DNA computing. In: Conference on Genetic and Evolutionary Computation, pp. 875–882. Morgan Kaufmann Publishers Inc. (2000)
10. Shin, S.Y., Kim, D.M., Lee, I.H., et al.: Evolutionary sequence generation for reliable DNA computing. In: 2002 Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002, pp. 79–84. IEEE (2002)
11. Xu, C., Zhang, Q., Wang, B., et al.: Research on the DNA sequence design based on GA/PSO algorithms. In: The International Conference on Bioinformatics and Biomedical Engineering, pp. 816–819. IEEE (2008)
12. Kurniawan, T.B., Khalid, N.K., Ibrahim, Z., et al.: Sequence design for direct-proportional length-based DNA computing using population-based ant colony optimization. In: ICCAS-SICE, pp. 1486–1491. IEEE (2009)

13. Wang, Y., Shen, Y., Zhang, X., et al.: An improved non-dominated sorting genetic algorithm-II (INSGA-II) applied to the design of DNA codewords. Math. Comput. Simul. **151**, 131–139 (2018)
14. Zhang, Q., Wang, B., Wei, X., et al.: DNA word set design based on minimum free energy. IEEE Trans. Nanobioscience **9**(4), 273–277 (2010)
15. Muhammad, M.S., Selvan, K.V., Masra, S.M.W., et al.: An improved binary particle swarm optimization algorithm for DNA encoding enhancement. In: Swarm Intelligence, pp. 1–8. IEEE (2011)
16. Mantha, A., Purdy, G., Purdy, C.: Improving reliability in DNA-based computations. In: IEEE International Midwest Symposium on Circuits and Systems, pp. 1047–1050. IEEE (2013)
17. Ibrahim, Z., Khalid, N.K., Lim, K.S., et al.: A binary vector evaluated particle swarm optimization based method for DNA sequence design problem. In: Research and Development, pp. 160–164. IEEE (2012)
18. Kurniawan, T.B., Khalid, N.K., Ibrahim, Z., et al.: Evaluation of ordering methods for DNA sequence design based on ant colony system. In: Second Asia International Conference on Modelling and Simulation, pp. 905–910. IEEE Computer Society (2008)
19. Jeong, K.S., Kim, M.H., Jo, H., et al.: Search of optimal locations for species- or group-specific primer design in DNA sequences: non-dominated sorting genetic algorithm II (NSGA-II). Ecol. Inform. **29**, 214–220 (2015)
20. Peng, X., Zheng, X., Wang, B., et al.: A micro-genetic algorithm for DNA encoding sequences design. In: International Conference on Control Science and Systems Engineering, pp. 10–14. IEEE (2016)
21. Deb, K., Sundar, J.: Reference point based multi-objective optimization using evolutionary algorithms. In: Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation, pp. 635–642. ACM (2006)
22. Pan, L., He, C., Tian, Y., Su, Y., Zhang, X.: A region division based diversity maintaining approach for many-objective optimization. Integr. Comput.-Aided Eng. **24**(3), 279–296 (2017)
23. He, C., Tian, Y., Jin, Y., Zhang, X., Pan, L.: A radial space division based evolutionary algorithm for many-objective optimization. Appl. Soft Comput. **61**, 603–621 (2017)
24. Pan, L., He, C., Tian, Y., Wang, H., Zhang, X., Jin, Y.: A classification-based surrogate-assisted evolutionary algorithm for expensive many-objective optimization. IEEE Trans. Evol. Comput. **23**(1), 74–88 (2018)
25. Pan, L., Li, L., He, C., Tan, K.C.: A subregion division-based evolutionary algorithm with effective mating selection for many-objective optimization. IEEE Trans. Cybern. (2019). https://doi.org/10.1109/TCYB.2019.2906679