frontiers | Frontiers in Genetics

TYPE Methods
PUBLISHED 21 November 2022
DOI 10.3389/fgene.2022.1023092

A depth-first search algorithm for oligonucleotide design in gene assembly

Hanjie Liang, Zengrui Chen and Gang Fang*

Institute of Computing Science and Technology, Guangzhou University, Guangzhou, China

Check for updates

OPEN ACCESS

EDITED BY
Quan Zou,
University of Electronic Science and Technology of China, China

REVIEWED BY
Xiangxiang Zeng,
Hunan University, China
Leyi Wei,
Shandong University, China

*CORRESPONDENCE
Gang Fang,
gangf@gzhu.edu.cn

SPECIALTY SECTION
This article was submitted to Computational Genomics, a section of the journal Frontiers in Genetics

RECEIVED 19 August 2022
ACCEPTED 28 October 2022
PUBLISHED 21 November 2022

CITATION
Liang H, Chen Z and Fang G (2022), A depth-first search algorithm for oligonucleotide design in gene assembly.
Front. Genet. 13:1023092.
doi: 10.3389/fgene.2022.1023092

When synthesizing a gene with a long DNA sequence, it is usually necessary to divide it into several fragments. Based on these fragments, a set of oligonucleotides for gene assembly is produced. Each oligonucleotide is synthesized separately by the chemical reaction, and then the obtained oligonucleotides are assembled into the full gene sequence, in a specific environment, by polymerase chain reaction (PCR) or ligase chain reaction (LCR). In this paper, an effective and efficient algorithm to divide long genes into oligonucleotide sets is presented. First, according to the length of the overlapping oligonucleotide region, the long DNA sequence to be synthesized is divided into fragments of approximately equal length. Second, the length of these fragments is iterated to dynamically optimize the length of the overlapping regions to reduce melting temperature fluctuations. Then, the improved depth-first search algorithm is used according to the design principle of pruning optimization to obtain a uniform set of oligonucleotides with very close melting temperatures. This will decrease the errors in gene assembly with PCR or LCR. Lastly, the oligonucleotides that have homologous melting temperatures needed for PCR-based synthesis and two-step assembly of the target gene are deduced and outputted.

KEYWORDS

gene assembly, depth-first search, algorithm, oligonucleotide design, melting temperature

Introduction

Gene synthesis now mainly utilizes overlapping oligonucleotides to assemble large genes (>1000 bp) by polymerase chain reaction (PCR) or ligase chain reaction (LCR) (Stemmer et al., 1995; Au et al., 1998). To optimize the PCR or LCR process and minimize errors in assembly, gene synthesis computer programs have been developed to aid in designing the oligonucleotides. The program's algorithm automates and streamlines the oligonucleotide design process, so that errors in assembly are minimized and large genes can be synthesized effectively. In gapless PCR assembly, some web-based applications, for example, TmPrime and DNAWorks which use an iterative algorithm, have been developed (David and DNAWorks, 2002; Marcus et al., 2009). In gapped PCR assembly, applications such as Gene2Oligo, Assembly PCR Oligo Maker, and GeneDesign that mainly rely on an iterative algorithm have been composed as well

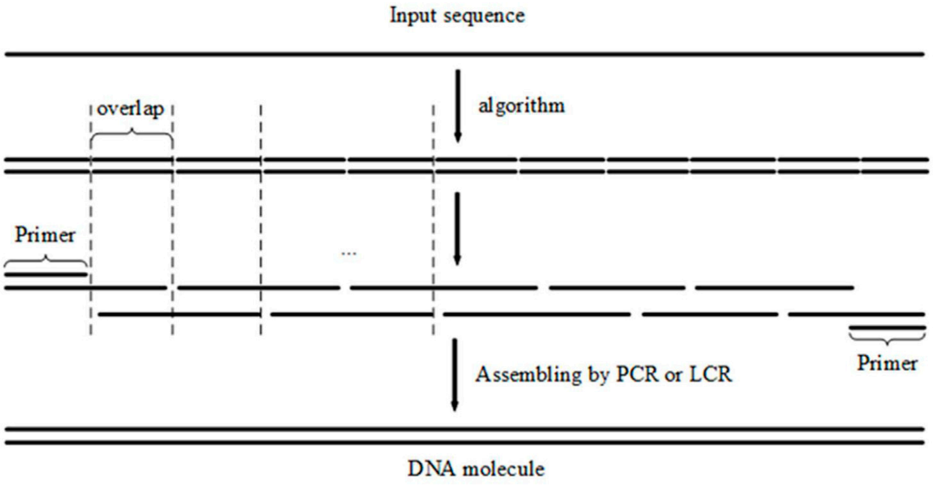


FIGURE 1
When assembling oligonucleotides with PCR or LCR, the overlapped regions of oligonucleotides are connected sequentially by the input sequence. The 'Primer' in the figure indicates that the primers for PCR share the same sequence with first and last overlap.

(Jean-Marie et al., 2004; Roman et al., 2005; Sarah et al., 2006). In the key step of oligonucleotide design, the input gene sequences are optimally split into oligonucleotides by the algorithm, so as to have approximately the same melting temperatures, and the overlapping regions of these oligonucleotides possess homologous melting temperatures. The best result is a deviation in melting temperature of less than 1°C in the overlapped region, which is attained with the use of a dynamic programming algorithm (Fang and Liang, 2022). However, its time complexity is too high to be used as a web-based application. Here, we present a depth-first search (DFS) algorithm with lower time complexity, for a web-based gene synthesis application, and to minimize errors in PCR assembly.

In gene synthesis, when assembling oligonucleotides with gapless PCR or LCR, all oligonucleotides are ligated tightly together with no gaps between adjacent oligonucleotides. The overlapped regions are connected sequentially according to the input sequence. Based on the aforementioned findings, the problem of splitting the input sequence into oligonucleotides with approximately the same melting temperature in overlapping regions can be treated as a problem of segmenting the input sequence into fragments with homologous melting temperatures; each fragment explicitly represents an overlapped region (Marcus et al., 2009) (Figure 1).

When assembling oligonucleotides using gapped PCR, the DNA fragments are contiguous with few base deletions. Compared with gapless PCR or LCR, gapped PCR assembly can lead to more assembly errors, but these errors are insignificant and can be ignored (Xiong et al., 2000). Gapped PCR assembly is more flexible and economical (Xiong et al., 2000). In this work, an algorithm that makes use of overlapping regions with homologous melting temperatures to output oligonucleotides for gapped PCR assembly is proposed.

Methods

According to the simple observation depicted in Figure 1, the proposed algorithm with an iteration step is introduced. First, according to the length of the overlapped regions, the long input DNA sequence to be synthesized is divided into fragments of approximately equal length. Second, we optimized the result of the initial segmentation step to reduce fluctuations in the melting temperatures of the overlapped regions (Figure 2). The input DNA sequence is processed by the iteration algorithm to split it into segments with similar melting temperatures. By iteratively adjusting the boundary between segments obtained from the initial segmented result, the melting temperature of the segments is first calculated by the algorithm and then the segments with the closest melting temperatures are selected for combination. In Table 1, the iteration algorithm is given by detailed pseudocode.

A nearest neighbor model is used to calculate melting temperatures along with Santa Lucia's thermodynamic parameters (Santa Lucia and Hicks, 2004), the salt and oligonucleotide concentrations, and the totality of phosphates in the duplex (Owczarzy et al., 2008). The equations and procedures used to calculate DNA melting temperature are described in detail in the Supplementary Material. After processing with the iteration algorithm, the oligonucleotide set can be deduced from the segmentation results. These oligonucleotides can be used for LCR or gapless PCR assembly; however, this step may not provide the best

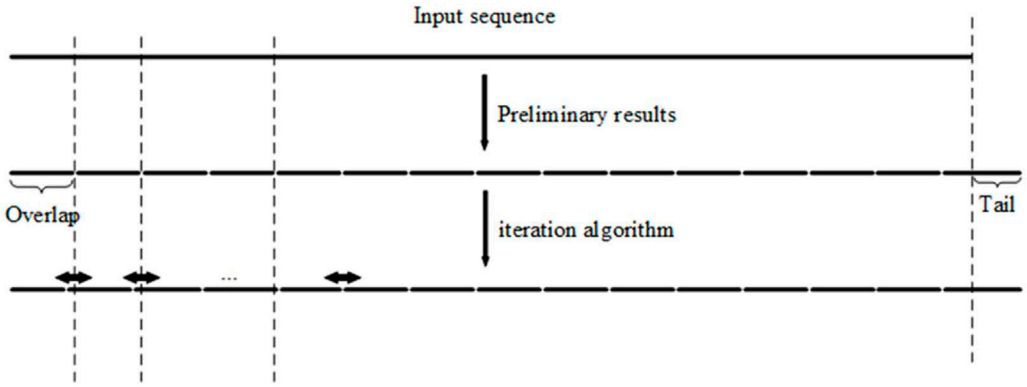
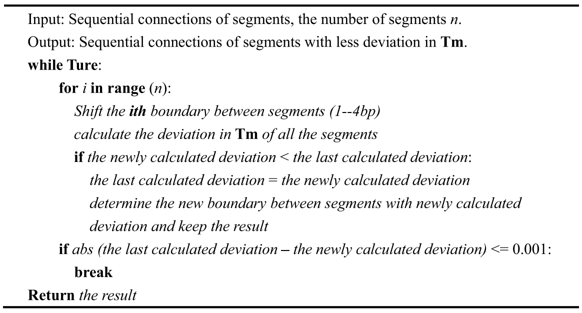


FIGURE 2
Iteration algorithm is carried out by shifting the boundary between overlaps (segments) iteratively until convergence.

TABLE 1 Details of the iteration algorithm. The iteration algorithm is an initial algorithm that roughly segments the input sequence.



```
Input: Sequential connections of segments, the number of segments n.
Output: Sequential connections of segments with less deviation in Tm.
while Ture:
    for i in range (n):
        Shift the ith boundary between segments (1--4bp)
        calculate the deviation in Tm of all the segments
        if the newly calculated deviation < the last calculated deviation:
            the last calculated deviation = the newly calculated deviation
            determine the new boundary between segments with newly calculated
            deviation and keep the result
    if abs (the last calculated deviation – the newly calculated deviation) <= 0.001:
        break
Return the result
```

solution to the problem. In an attempt to further reduce the melting temperature fluctuation of overlapped regions, we present a DFS algorithm, which serves as the foundation for a large number of graphic applications that utilize tree-based search algorithms. Given a vertex, the DFS algorithm can find all reachable vertices as well as directed and undirected graphs containing the shortest paths from one vertex to others (Cong, 2010). As an important graph-search algorithm, the DFS algorithm plays an important role in solving issues in computer science like planarity testing, scheduling problems, inspecting network structuring, and topological sorting (Palanisamy and Vijayanathan, 2020). In our work, the DFS algorithm was adapted to solve the oligonucleotides design

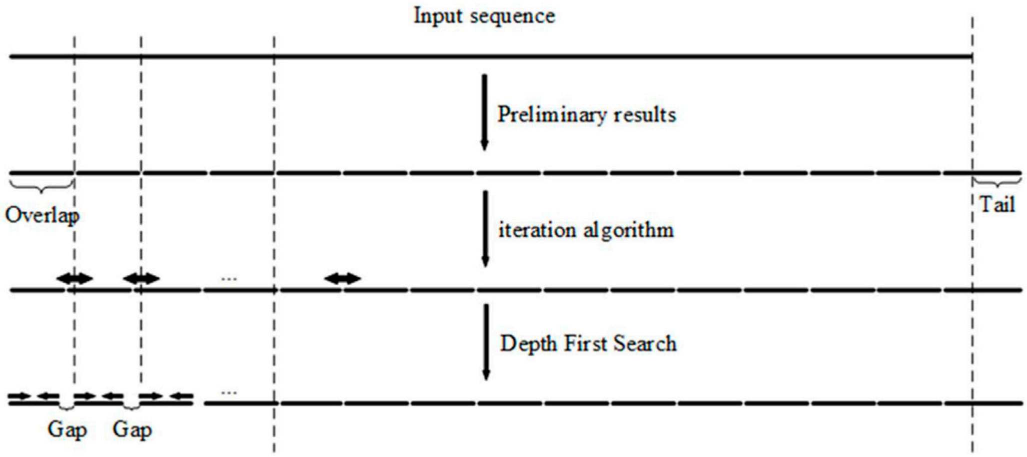
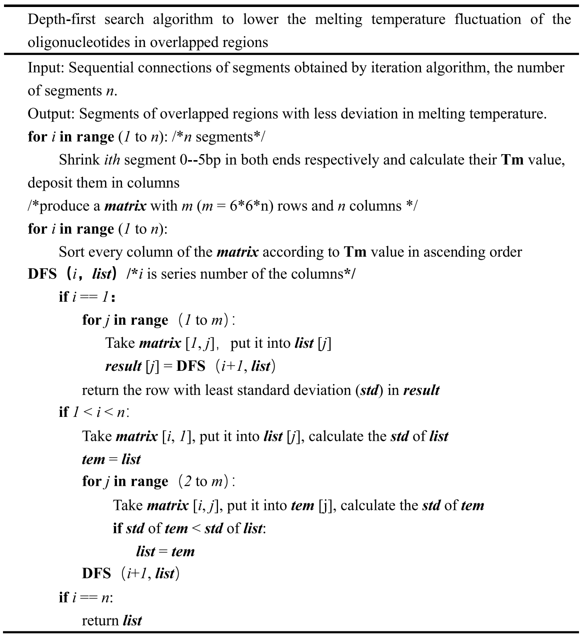


FIGURE 3
Depth-first search algorithm takes the results from the iteration algorithm, shrinks each end of an overlap, and produces a gap between segments. This reduces the standard deviation of the overlap melting temperatures.

TABLE 2 Depth-first search algorithm (DFS).



```
Depth-first search algorithm to lower the melting temperature fluctuation of the
oligonucleotides in overlapped regions

Input: Sequential connections of segments obtained by iteration algorithm, the number
of segments n.
Output: Segments of overlapped regions with less deviation in melting temperature.
for i in range (1 to n): /*n segments*/
    Shrink ith segment 0--5bp in both ends respectively and calculate their Tm value,
deposit them in columns
/*produce a matrix with m (m = 6*6*n) rows and n columns */
for i in range (1 to n):
    Sort every column of the matrix according to Tm value in ascending order
DFS（i， list） /*i is series number of the columns*/
    if i == 1：
        for j in range （1 to m）:
            Take matrix [1, j]，  put it into list [j]
            result [j] = DFS （i+1, list）
        return the row with least standard deviation (std) in result
    if 1 < i < n:
        Take matrix [i, 1], put it into list [j], calculate the std of list
        tem = list
        for j in range （2 to m）:
            Take matrix [i, j], put it into tem [j], calculate the std of tem
            if std of tem < std of list:
                list = tem
        DFS （i+1, list）
    if i == n:
        return list
```

problem in gene assembly (Figure 3). In Table 2, the DFS algorithm adapted to minimize the fluctuation of melting temperatures of oligonucleotide overlapped region is shown in detailed pseudocode.

Oligonucleotides with close, uniform melting temperatures can be generated using results computed by the DFS algorithm for gapped PCR assembly. A short tail may be appended to the 3' end of an input sequence (Figures 2, 3), which will guarantee the imbricated structure of the oligonucleotides output in the PCR reaction diagrammed in Figure 1. The added tail can be removed by PCR using specific primers.

Results

It is crucial to ensure uniformity of melting temperature, especially in the overlapped oligo regions, when designing oligonucleotides for gene synthesis. It will reduce mis-hybridization between oligonucleotides and decrease errors in assembly. The oligonucleotide sets produced by the presented algorithm cannot achieve a smaller standard deviation in overlap melting temperatures than the result produced by the integrated algorithm, but the result is acceptable for gapped PCR assembly and results in a lower SD of Tm in the overlapped regions than TmPrime (Table 3). In fact, there are few base deletions (gaps) and frequently no gaps between adjacent oligonucleotides such that the final oligonucleotides are contiguous in gapped assembly. This phenomenon is caused by the algorithm's intrinsic nature. The number of bases to be shrunk in the first *for* loop in the DFS algorithm (Table 2) can be changed to modify the number of gaps. Readers are referred to Supplementary Material or https://github.com/Jacka03/oligoOptimizer for more information. *Python* 3.7 was adopted for the algorithm implementation. On a desktop computer with dual 3.3-GHz Intel Xeons and 4 GB RAM, it takes less than 2 s to design a set of oligonucleotides for a multi-kilobase (<3 kb) gene. Under the same conditions, the integrated algorithm takes 10s (Fang and Liang, 2022). The depth-first search algorithm was developed for web-based application, which needs to return a result with no unacceptable delay.

Discussion

In this paper, we presented an effective and efficient depth-first search algorithm for oligonucleotide design in gene synthesis based on its intrinsic nature. The average melting temperature of the final oligonucleotide set can be used to set the annealing temperature of assembly. It is difficult to calculate the time complexity of an iteration algorithm, but the variance threshold between the last calculated deviation and the newly calculated deviation can be adjusted to achieve rapid convergence. Empirically, a variance threshold of 0.001 often yields a rapid convergence. Given the number of sequential connections of segments, *L*, along with the number of possible column segments, *N*, the time complexity is $O(LN^3)$ for the dynamic programming algorithm and $O(N^L)$ for the exhaustive

TABLE 3 Oligonucleotide set designed by three algorithms. Compared to other algorithms, the depth-first search algorithm does not surpass the integrated algorithm, but it takes less time to process. For a fair comparison, the algorithms were tested under the same conditions.

Algorithm	Gene		
	S100A4 (752 bp) overlap Tm std	PKB2 (1446 bp) overlap Tm std	GFPuv (760 bp) overlap Tm std
TmPrime	1.14	1.23	0.93
Depth-first search	0.84	0.52	0.40
Integrated algorithm (gapped)	0.27	0.49	0.17

algorithm (Fang and Liang, 2022). In contrast, the time complexity is $O(N^2)$ for the DFS algorithm, which will guarantee the technical feasibility of this algorithm. Furthermore, it only requires a computer of average speed and RAM for running, which is more suitable for a web-based application needed to return a result with little delay. Based on the simple observation, gapless assembly can use an iteration algorithm to generate the oligonucleotides, regardless of its elementary results. Theoretically, this kind of optimization problem is not guaranteed to have the best solution; and even if there is one, it is difficult to acquire (Pevzner and Waterman, 1995; Berman et al., 2002). With regard to this problem, the presented depth-first algorithm is a type of approximation algorithm, which means that it runs faster than an integrative algorithm. When it is carried out after an iteration algorithm, an acceptable result is acquired. The oligonucleotides produced have a greater uniformity of melting temperature than TmPrime, which reduces assembly error; but these oligonucleotides can only be used for gapped PCR assembly, which can result in a higher assembly error rate than gapless assembly. The oligonucleotides produced by the DFS algorithm are very close in Tm, which can diminish the effect caused by gaps between adjacent oligonucleotides. Synchronization of the melting temperatures of the overlapped region is the vital factor in gene assembly, and it is important to take into account factors like the appearance of repeated regions, high CG content regions, etc. These factors can cause mis-hybridization between oligonucleotides with the formation of unwanted secondary structures that can increase errors in the PCR reaction. In this paper, we stress the importance of Tm uniformity of overlapped regions as a way to compensate for potential mismatch problems and facilitate hybridization between overlapping oligonucleotides. The reason for this is that oligonucleotides with uniform Tm are more likely to simultaneously overlap and hybridize correctly under the same temperature.

The position and number of bases to be shrunk in the first *for* loop in the DFS algorithm (Table 2) can be changed to modify the position and number of gaps in the algorithm. The size of segments to be split before using the iteration algorithm can also be changed to modify the length of oligonucleotides (in general, a base number of 20–30 bp to be split will generate an overlap of 20–30 bp and oligonucleotides of 40–60 bp in length). The DFS algorithm is the core of this project. While the input sequence is only approximately segmented by iteration algorithms, the uniformity of melting temperature of oligonucleotides for gene synthesis is further achieved by the DFS algorithm, a strategy that is widely adopted in the optimization theory (Hidayatullah et al., 2017; Zheng et al., 2021). It is expected to experience an explosive increase in application due to its practicability and universality in biology, especially in synthetic biology. The integrated algorithm in our previous study was complex, consisting of three main sub-algorithms (the greedy algorithm, iteration algorithm, and dynamic programming algorithm) (Fang and Liang, 2022). In theory, its time complexity is $O(LN^3)$, but in practice it is variable. When processing longer sequences and for more accurate results, the time complexity always exceeded $O(LN^3)$ and required a more powerful computer to run; furthermore, it is not open-source. Although DFS for this application is an approximation algorithm, it is open-source. It was not only developed for web-based application, but also for people in related fields to modify and develop related algorithms, and it is easy to carry out on an ordinary desktop computer. Oligo design is very important in gene synthesis. The web-based applications, TmPrime, DNAWorks, Gene2Oligo, Assembly PCR Oligo Maker etc., are no longer available and their source code is not always open-source. The DFS algorithm has been developed for oligo design and is open-source for all users. This project has been written into the computer program to facilitate gene synthesis.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

Author contributions

HL wrote the code, ZC carried out validation, and GF designed the algorithm and the study.

Funding

The study was supported by the National Natural Science Foundation of China under grant number 61972107.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fgene.2022.1023092/full#supplementary-material

References

Au, L. C., Yang, F. Y., Yang, W. J., Lo, S. H., and Kao, C. F. (1998). Gene synthesis by a LCR-based approach: High-level production of leptin-L54 using synthetic gene in *Escherichia coli*. *Biochem. Biophys. Res. Commun.* 248, 200–203. doi:10.1006/bbrc.1998.8929

Berman, P., Hannenhalli, S., and Karpinski, M. (2002). "1.375-approximation algorithm for sorting by reversals," in European Symposium on Algorithms, volume 2461 of Lecture Notes in Computer Science, 200–210.

Cong, G. G. A. a. V. S. (2010). "Fast PGAS implementation of distributed graph algorithms," in Proceedings of the International Conference for High Performance Computing NetworkingStorage and Analysis, 1–11.

David, M. H., and Dnaworks, J. L. (2002). DNAWorks: An automated method for designing oligonucleotides for PCR-based gene synthesis. *Nucleic Acids Res.* 30, e43. doi:10.1093/nar/30.10.e43

Fang, G., and Liang, H. (2022). An integrated algorithm for designing oligodeoxynucleotides for gene synthesis. *Front. Genet.* 13, 836108. doi:10.3389/fgene.2022.836108

Hidayatullah, A. S., Jati, A. N., and Setianingsih, C. (2017). "Realization of depth first search algorithm on line maze solver robot," in 2017 International Conference on Control, Electronics, Renewable Energy and Communications (ICCREC), 247–251.

Jean-Marie, R., Woonghee, L., Gilles, T., Xiaolian, G., and Xiaochuan, Z. (2004). Gene2Oligo: Oligonucleotide design for *in vitro* gene synthesis. *Nucleic Acids Res.* 32, W176–W180. doi:10.1093/nar/gkh401

Marcus, B., Samuel, K., Hongye, Y., Mo-Huang, L., and JackieTmPrime, Y. Y. (2009). TmPrime: Fast, flexible oligonucleotide design software for gene synthesis. *Nucleic Acids Res.* 37, W214–W221. doi:10.1093/nar/gkp461

Owczarzy, R., Moreira, B. G., You, Y., Behlke, M. A., and Walder, J. A. (2008). Predicting stability of DNA duplexes in solutions containing magnesium and monovalent cations. *Biochemistry* 47, 5336–5353. doi:10.1021/bi702363u

Palanisamy, V., and Vijayanathan, S. (2020). "A novel agent based depth first search algorithm," in 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA), 443–448.

Pevzner, P. A., and Waterman, M. S. (1995). Multiple filtration and approximate pattern matching. *Algorithmica* 13, 135–154. doi:10.1007/bf01188584

Roman, R., Sharon, X. Z., and Philip, E. J. (2005). Assembly PCR oligo maker: A tool for designing oligodeoxynucleotides for constructing long DNA molecules for RNA production. *Nucleic Acids Res.* 33, W521–W525. doi:10.1093/nar/gki380

SantaLucia, J., Jr., and Hicks, D. (2004). The thermodynamics of DNA structural motifs. *Annu. Rev. Biophys. Biomol. Struct.* 33, 415–440. doi:10.1146/annurev.biophys.32.110601.141800

Sarah, M. R., Sarah, J. W., Robert, M. Y., and GeneDesign, J. (2006). GeneDesign: Rapid, automated design of multikilobase synthetic genes. *Genome Res.* 16, 550–556. doi:10.1101/gr.4431306

Stemmer, W. P., Crameri, A., Ha, K. D., Brennan, T. M., and Heyneker, H. L. (1995). Single-step assembly of a gene and entire plasmid from large numbers of oligodeoxyribonucleotides. *Gene* 164, 49–53. doi:10.1016/0378-1119(95)00511-4

Xiong, A., Yao, Q., Peng, R., Li, X., Fan, H., Cheng, Z., et al. (2000). A simple, rapid, high-fidelity and cost-effective PCR-based two-step DNA synthesis method for long gene sequences. *Nucleic Acids Res.* 32, e98. doi:10.1093/nar/gnh094

Zheng, K., Huang, Y., Shen, A., Kosari, S., Liu, X., and Qiang, X. (2021). Prediction of pandemic risk for animal-origin coronavirus using a deep learning method. *Infect. Dis. Poverty* 10, 128. doi:10.1186/s40249-021-00912-6