Mathematical Biosciences and Engineering

MBE, 20(4): 6174–6190.
DOI: 10.3934/mbe.2023266
Received: 12 December 2022
Revised: 16 January 2023
Accepted: 17 January 2023
Published: 31 January 2023

http://www.aimspress.com/journal/MBE

Research article

Antibody sequences assembly method based on weighted de Bruijn graph

Yi Lu[1], Cheng Ge[2], Biao Cai[1], Qing Xu[1], Ren Kong[1,*] and Shan Chang[1,*]

[1] Institute of Bioinformatics and Medical Engineering, School of Electrical and Information Engineering, Jiangsu University of Technology, Changzhou 213001, China
[2] Key Laboratory of Marine Drugs, Chinese Ministry of Education, School of Medicine and Pharmacy, Ocean University of China, Qingdao 266003, China

* **Correspondence:** Email: rkong@jsut.edu.cn, schang@jsut.edu.cn.

Abstract: With the development of next-generation protein sequencing technologies, sequence assembly algorithm has become a key technology for de novo sequencing process. At present, the existing methods can address the assembly of an unknown single protein chain. However, for monoclonal antibodies with light and heavy chains, the assembly is still an unsolved question. To address this problem, we propose a new assembly method, DBAS, which integrates the quality scores and sequence alignment scores from de novo sequencing peptides into a weighted de Bruijn graph to assemble the final protein sequences. The established method is used to assembling sequences from two datasets with mixed light and heavy chains from antibodies. The results show that the DBAS can assemble long antibody sequences for both mixed light and heavy chains and single chains. In addition, DBAS is able to distinguish the light and heavy chains by using BLAST sequence alignment. The results show that the algorithm has good performance for both target sequence coverage and contig assembly accuracy.

Keywords: de novo sequencing; de Bruijn graph; sequence alignment; monoclonal antibody; sequence assembly

1. Introduction

Antibodies are a type of globulin with immune function that can bind specifically to the corresponding antigen. Generally, antibody is composed by light chain and heavy chain, and the composition and arrangement of amino acids in the variable region determines the antigen-binding

specificity of the antibody. The variability of monoclonal antibodies plays an important role in therapeutic strategies. However, we have not been able to sequence them systematically so far due to the mutation mechanism. Each monoclonal antibody (mAb) sequence is a new protein that needs to be sequenced. There are no similar proteins in the database. Therefore, it is a difficult problem to solve for antibody sequence assembly.

Edman degradation is a tool for de novo sequence analysis of unknown protein species [1]. From low-throughput sequencing methods to high-throughput sequencing methods, protein sequencing technology has made significant advances in the last few decades. In particular, liquid chromatography-tandem mass spectrometry (LC-MS/MS) has become a routine technique for protein identification. High-throughput sequencing requires computational analysis and statistical extrapolation of data. This includes de novo sequencing directly from tandem mass spectrometry and database search methods [2–7]. Shotgun protein sequencing (SPS) combines electron transfer dissociation (ETD), collision-induced dissociation (CID), and high-energy collision-induced dissociation (HCD) fragmentation techniques to increase sequencing coverage [8]. It tries to make some progress in the full-length sequencing for proteins, especially antibodies. Other existing methods are sequenced on the basis of having similar proteins or combining top-down methods. Despite the effectiveness of these methods, de novo sequencing from tandem mass spectra of unknown proteins remains a challenging problem [9–11]. It is not feasible to assemble the mass spectrometry data directly. We need to sequence the mass spectrometry data and the set of sequences obtained is the reads set. The de novo assembly for reads set comprises three steps: 1) contig assembly, 2) scaffolding, and 3) gap filling. In the contig assembly step, the reads are assembled as long consensus sequences without gaps. Then, the contigs are connected by large-insert reads in the scaffolding step. Once the contigs are scaffolded, spaces called 'gaps' remain between the contigs if there is no overlap between the contigs. The gaps are carefully filled by using other independent reads to complete the assembly [12].

At present, there have been many studies on sequence assembly methods. According to the different assembly strategies, the sequence assembly methods can be divided into three types: 1) algorithms based on Greedy strategy, 2) algorithms based on Over-lap-Layout-Consensus strategy, 3) algorithms based on de Bruijn Graph strategy [13]. ISEA is an iterative seed expansion algorithm proposed by Professor Jianxin Wang's team [14]. To ensure the accuracy of sequence assembly, it combines scoring function and seed expansion strategy based on de Bruijn Graph. The SSVAGE assembly is an algorithm for de novo assembly of viral quasispecies based on an overlap graph strategy [6]. The algorithm uses three different overlap graph construction methods to identify sequence errors by calculating clusters in the graph. To address the high repetition rate of short reads, the ALLPATHS algorithm is proposed by the Broad Institute of MIT and Harvard Research Center. It introduces k-mers numbering algorithm, so that k-mers can be numbered in sequence of reads [15]. This algorithm improves the accuracy of sequence assembly, but preserves the inherent ambiguity caused by antibody sequence variants.

Although these assembly methods have achieved good results, they can only be applied to assemble for pure light chains or heavy chains. The peptide sequences obtained from mass spectrometry are mixed light and heavy chain data. Therefore, the accuracy of assembling peptide sequences without making a distinction between light and heavy chain is greatly reduced. ALPS is a greedy algorithm-based de novo assembly algorithm [16]. It uses de novo peptides, sequence confidence scores, and database homology search information to build a weighted de Bruijn graph to assemble complete monoclonal antibody sequences. This method also cannot distinguish between light

and heavy chains from the mixed data, and the accuracy still needs to be improved. In this paper, the de Bruijn graph algorithm is used to assemble the sequences. The method splits the reads into shorter k-mers, and when the k-mer is small, the probability of error is reduced. The steps of the method are as follows: 1) Reads are decomposed into k-mers by sliding a window of size k across the reads, 2) The k-mers become vertices in the graph, with edges connecting overlapping k-mers, 3) Contigs are built by walking the graph from edge nodes [17]. The process of the de Bruijn graph assembly is shown in Figure 1. A de Bruijn graph is a mathematical construct, where each vertex (or node) in a graph represents a k-mer. Nodes in the graph are connected where the last *k*-1 amino acids of one node overlap with the first *k*-1 amino acids of another node. A graph is built by considering each read in turn, decomposing it into individual overlapping k-mers and creating nodes for new k-mers, updating coverage for existing k-mers and adding vertices for new transitions. In an ideal case, the de Bruijn graph would form a single line, in which each node, apart from the two edges, is connected to one other node in the forward orientation and one in the reverse orientation.

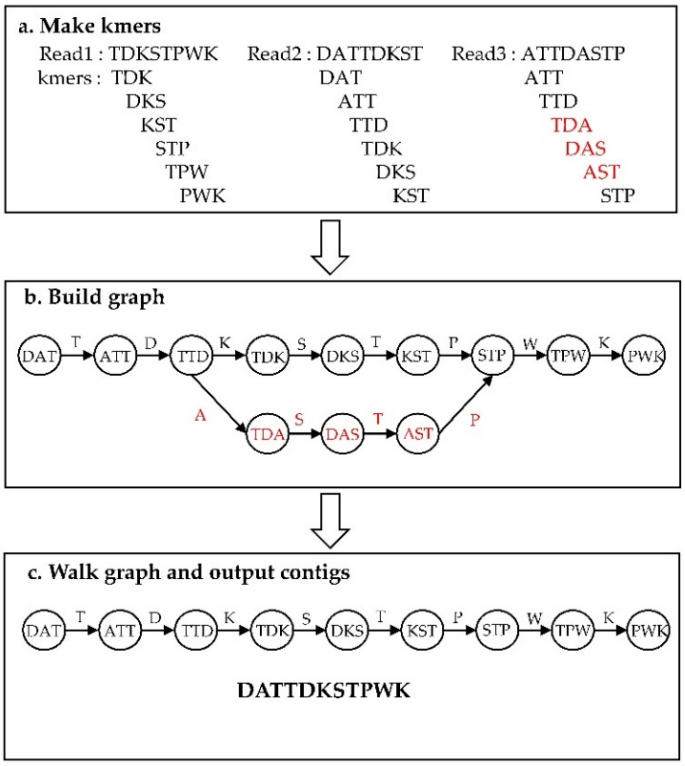

Figure 1. The process of the de Bruijn graph assembly.

In this study, we propose an antibody sequence assembly method that combines de Bruijn graph and BLAST sequence comparison [18], DBAS. It applies peptide sequences from de novo sequencing, sequence confidence scores, and BLAST sequence alignment scores to construct a weighted de Bruijn graph. In addition, we create BLAST localization database involving the use of sequences from the international Immunogenetics information system (IMGT) and the Antibody Research System (abYsis) [19]. The method can discriminate and assemble light or heavy chains based on the de novo sequencing data with mixed light and heavy chains, as well as the data with only single chains. We compared the method with existing sequence assembly methods, such as ALPS. The

experimental results show that DBAS performs better than ALPS in the sequence coverage and accuracy for antibodies [16].

2. Materials and methods

2.1. Data source and formats

Here we report the data sources and formats of two monoclonal antibody datasets. Antibody sequence datasets were generated from purified human antibody samples. The first dataset, Human_denovo, was generated from pre-processed raw LC-MS/MS data by PEAKS de novo sequencing [20,21]. The other dataset, Human_spider, was generated by searching the PEAKS de novo sequencing data in the corresponding antibody database using PEAKS SPIDER for the dataset. To evaluate our method, amino acid sequences were manually calculated from LC-MS/MS data by using PEAKS 7.5. Therefore, the following criteria were required:

1) The false discovery (FDR) at the peptide-spectrum matching (PSM) level was below 0.1%.
2) At least 20 PSMs supported an amino acid.
3) Each amino acid had a pair of its ion peaks with a relative intensity of more than 5%.

The data information mainly includes the sequence number, peptide sequence, peptide confidence scores, and peptide intensity (feature region), which are saved in csv file format. The peptide confidence scores and sequence alignment scores are used to form the weight information which plays a crucial role in selecting the appropriate expansion path and improving the assembly quality.

2.2. Blast sequence alignment

The sequence alignment algorithm obtains the optimal alignment of two or more sequences based on a given scoring function [22]. In the process of matching two or more sequences strings, this algorithm displays or deletes matched characters or "-" to obtain the most similar arrangement between sequences. The sequence alignment algorithms include global alignment and local alignment algorithms. The BLAST used in this study is the local alignment algorithm.

The BLAST algorithm, proposed by Altschul et al, uses a combination of short fragment matching algorithm and a statistical model to find the best local alignment results between the target sequence and the database [23]. It uses the set scoring matrix to score all possible enhancement points for the input. It also retains those that score higher than the set value T and increases the speed of good quality but low number of enhancement points generated. The significance of each partial matching result is counted. In this way, the probability of that partial matching result is calculated. The smaller the probability, the more meaningful the result is. The final similarity fragments of a certain length are obtained. These are called high score fragment pairs [24].

This study required the use of BLAST localization software to build the antibody database. All possible k-mers were extracted from the set of peptide sequence reads. Each k-mer is put into the antibody database for BLAST sequence alignment. According to the matching results, the sequence of each k-mer matched on the antibody database and the corresponding matching score can be obtained. Meanwhile, the database sequences successfully matched are marked with a light-heavy chain marker. The light and heavy chains are distinguished by this marker for each k-mer. The process of obtaining sequence alignment scores in our method is shown in Figure 2.

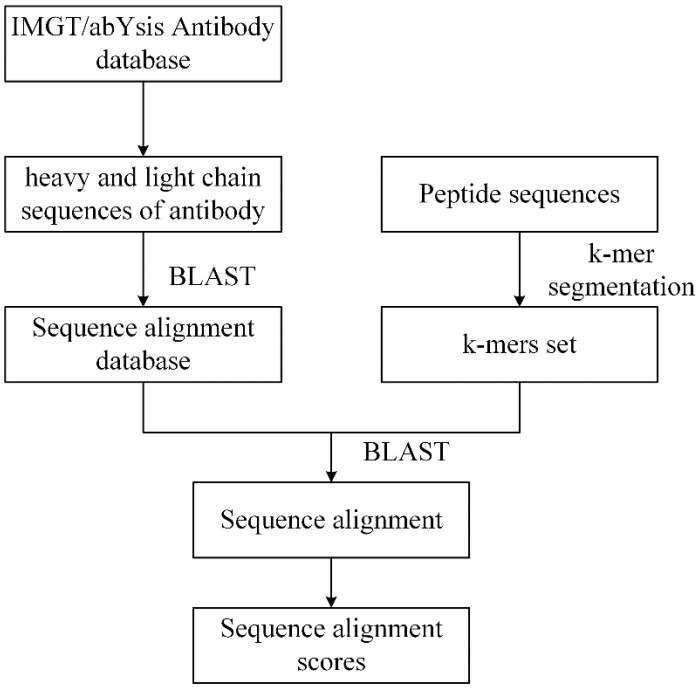

Figure 2. The process of obtaining sequence alignment scores in this method.

2.3. De Bruijn graph

The de Bruijn graph is used to solve the problem of short sequence assembly of genomes. It is able to handle the high information redundancy caused by the high coverage of the dataset with low memory usage. Nevertheless, the assembly is achieved by determining the Eulerian paths in the de Bruijn graph, which are easier to handle compared to the Hamiltonian paths [25–28].

In this study, the antibody peptide sequences obtained at the sequencing stage are assumed to be the set of reads, $X = \{X_1, X_2, \ldots, X_n\}$. We divide each read into the set of k-mers consisting of k consecutive bases, $K = \{K_1, K_2, \ldots, K_n\}$. Each k-mer is further split into two overlapping substrings of length k-1, called left and right (k-1)-mers. The left and right (k-1)-mer represent nodes in the de Bruijn graph, while the k-mer corresponds to a directed edge in the graph pointing from left to right (k-1)-mer. Then, we assign weight values (α, β) to each (k-1)-mer. The node weights (α_1, β_1) of the light chain de Bruijn graph and the node weights (α_2, β_2) of the heavy chain de Bruijn graph are defined in the following equations.

$$\alpha_1 = \sum_{i=0}^{i-k+1} \left(e^{\frac{s\cdot\lambda_1 + (b[0])(\ln(a[i]))}{(t+b[0])}} \right) \tag{1}$$

$$\beta_1 = \sum_{i=0}^{l-k+1} \left(e^{\frac{s\cdot\lambda_1 + (b[k-1])(\ln(a[i+k-1]))}{(t+b[k-1])}} \right) \tag{2}$$

$$\alpha_2 = \sum_{i=0}^{i-k+1} \left(e^{\frac{s\cdot\lambda_2 + (b[0])(\ln(a[i]))}{(t+b[0])\cdot\lambda_2}} \right) \tag{3}$$

$$\beta_2 = \sum_{i=0}^{l-k+1} \left(e^{\frac{s\cdot\lambda_2 + (b[k-1])(\ln(a[i+k-1]))}{(t+b[k-1])\cdot\lambda_2}} \right) \tag{4}$$

In the above equations, s and t are defined in the following equations.

$$s = \sum_{i=0}^{l-k+1} \sum_{j=i+1}^{i+k-1} (b[j-i])(\ln(a[i])) \tag{5}$$

$$t = \sum_{i=0}^{l-k+1} \sum_{j=i+1}^{i+k-1} (b[j-i]) \tag{6}$$

$$s.t.\ \ b[k] = \{k-1, 1, \ldots, 1, k-1\}$$

The array a[i] is the set of confidence scores of the corresponding peptide sequence. In order to keep the difference between the weight values within a small range of values, we take the natural logarithm approach on a[i]. The first and last values of the array b are the lengths of (k-1)-mer and the middle is filled with the value 1. l is the length of a peptide sequence obtained by sequencing. λ_1 is the percentage of the corresponding light chain BLAST alignment score and λ_2 is the percentage of the corresponding heavy chain BLAST alignment score. k is the value of the input k-mer, here we generally input $k = 6$. We define the confidence score of each (k-1)-mer by the weighted geometric mean of the amino acid confidence scores. The weight of each (k-1)-mer is expressed as the product of the confidence score of the (k-1)-mer and the intensity of its extracted peptide. In the case where (k-1)-mer may occur in multiple peptides, its weight was accumulated gradually as it is processed in these peptides. The equation in the previous articles was without λ_1 and λ_2 coefficients. The weighted equation in our paper adds λ_1 and λ_2 coefficients, which are the light and heavy chain sequence comparison scores. Using these coefficients can help to detect heavy chain peptide sequences and light chain peptide sequences, which can have better results in sequence assembly.

After the de Bruijn graph is constructed, contigs are assembled by performing greedy walks through the graph as the following.

1) First, the (k-1)-mer with the largest weight is selected as the new contig seed.
2) Second, this step is repeated by selecting the neighboring amino acid with the greatest weight and connecting the new amino acid to the current contig. This operation is extended forward or backward to the seed. Then, this step is repeated until no extension is possible. When a (k-1)-mer is used for the expansion of a seed, the seed is removed from the graph.
3) Finally, the above two steps are repeated until the required number of contigs are generated or the graph becomes empty. At this point, the output is a list of assembled contig.

The weighted de Bruijn graph is shown in Figure 3.

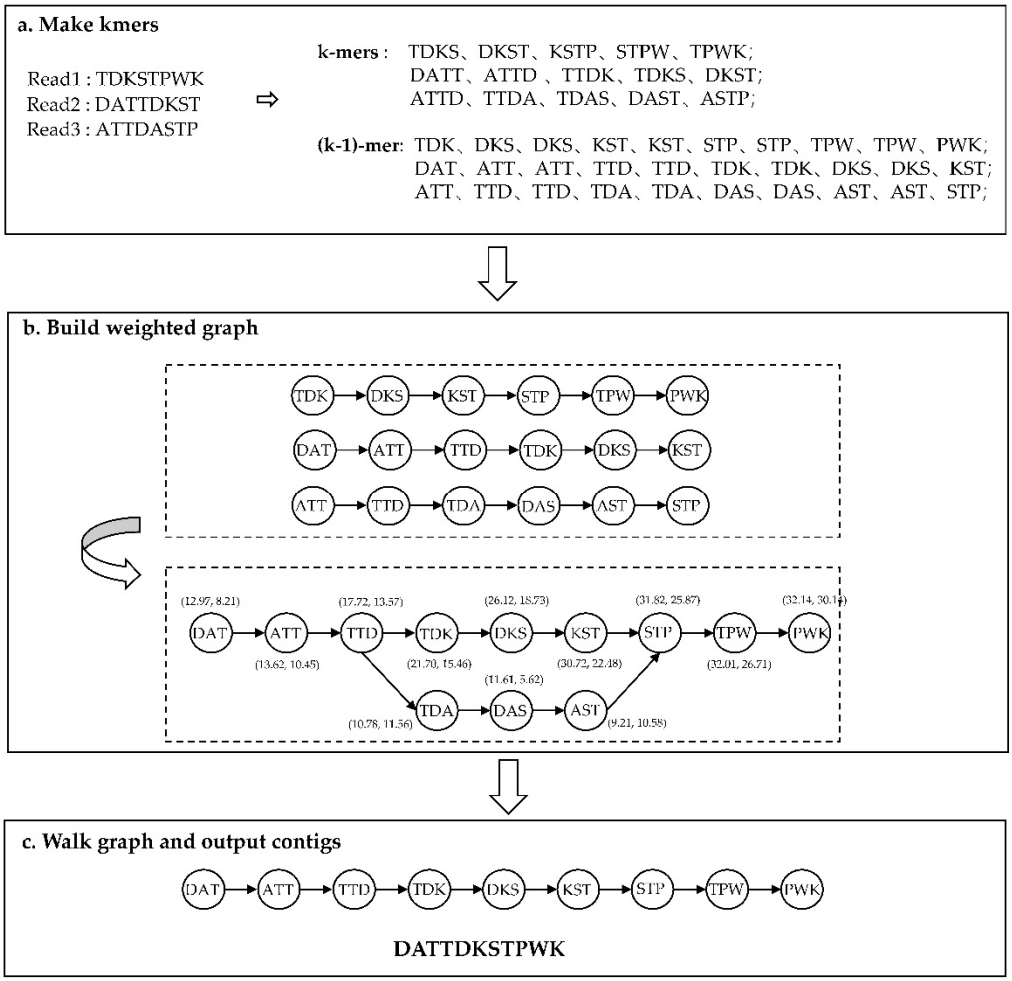

Figure 3. The construction process of weighted de Bruijn graph.

2.4. Algorithm process

The flow of the algorithm proposed in this paper is shown in Figure 4. First, the database was searched in the UniProt (Universal Protein) database to identify the species. Next, the antibody light-heavy chain sequences of the corresponding species were achieved from the IMGT and abYsis databases. We used these data to create the BLAST localized antibody sequence database. Here the human light-heavy chain sequence alignment database is created. Then, the sequence database created above was compared with all possible k-mers which were extracted from the peptides by BLAST software to obtain the alignment scores. If the k-mer sequence set is judged to be mixed light and heavy chain, the k-mer set and the corresponding alignment scores are classified into two categories: light chain and heavy chain. The weighted de Bruijn graphs were created according to two types of data separately. The previous section describes the construction and traversal of the weighted de Bruijn graph. Therefore, light chain contigs and heavy chain contigs are assembled simultaneously. If the k-mer sequence set is judged to be heavy or light chains, the corresponding alignment scores of k-mer need not be classified. The de Bruijn graph is directly constructed to assemble light or heavy chain contigs. As for the set of mixed light and heavy chain sequences, the BLAST alignment of peptide sequences and the construction of de Bruijn graph took a long time. Its average calculation time was 6

hours. Because of the small number of light or heavy chain peptide sequences, the alignment and assembly of single chain peptide sequences takes less time. Its average calculation time is three hours.

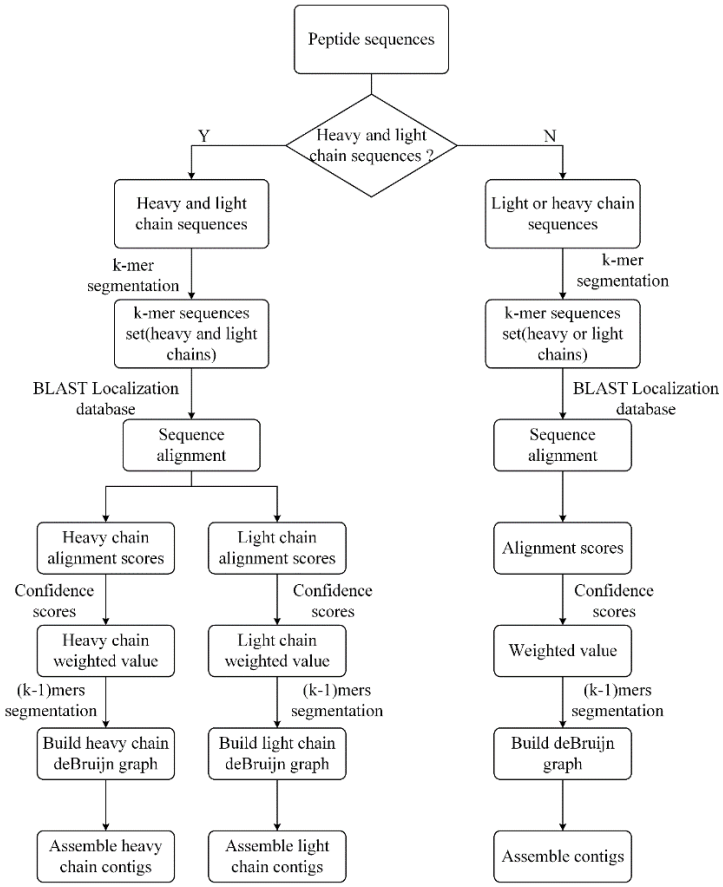

Figure 4. Flow of assembly method, DBAS.

The powerful performance of de Bruijn graph has been demonstrated in the major genome. It is also commonly used in de novo peptide sequencing assembly based on mass spectrometry data. However, the accuracy of peptide assembly is very low when using the mixed data of light and heave chains. Therefore, we used BLAST sequence alignment tool to distinguish between light and heavy chains from the mixed data before assembling the peptide sequences. When the test data are mixed light and heavy chain sequences, the method DBAS by constructing weighted de Bruijn graph can have better performances in assembling light and heavy chains separately.

2.5. Evaluation methods

In this study, we propose an antibody sequence assembly method that combines de Bruijn graph and BLAST sequence alignment, DBAS. To evaluate the assembly results, we used the online BLAST alignment system of NCBI (National Center for Biotechnology Information) to compare the assembled contigs with the corresponding target sequences. For comparison, we used the similar evaluation metrics and related data published by Tran et al. [16]. The data was integrated into two test data, including the human light-heavy chain data from de novo sequencing and the data generated by searching the PEAKS de novo sequencing data in the corresponding antibody database using PEAKS

SPIDER for the dataset. The evaluation metrics included target sequence coverage (coverage, Cov) and contig assembly accuracy (accuracy, Acc). len1 represents the length of the best assembled contig. len2 represents the number of amino acids recovered (AA). target_len indicates the length of the target sequence. The target sequence coverage (Cov) represents that the percentage of amino acids of the target sequence were covered by the respective best contig [16]. The corresponding formula is $Cov = len2 / target\_len$. The contig assembly accuracy (Acc) represents that the percentage of correct amino acids of the best contig were aligned to the respective target sequence [16]. The corresponding formula is $Acc = len2 / len1$.

3. Results

The purpose of this paper is that our DBAS method can assemble long antibody sequences for both mixed light and heavy chains and single chains. The DBAS method uses a combination of de Bruijn graph algorithm and BLAST sequence alignment for sequence assembly. First, we describe the data and software that we used. In addition, we also highlight the differences between IMGT, VBASE, and our program and state reasons for using our program. Next, we describe the effect of k-mer on the sequence assembly and pick the most suitable k-mer value. Finally, we compare the DBAS method with the ALPS method to highlight the advantages of the DBAS method in terms of target sequence coverage and contig assembly accuracy. The following sections provide the details.

3.1. Data and software

We use de Bruijn graph and BLAST sequence comparison algorithm to assemble the peptide sequences. The program of de Bruijn graph algorithm is written in Java using the software Intellij IDEA. The installation package of the Basic Local Alignment Search Tool (BLAST) is available from the National Center for Biotechnology Information website (NCBI). Two peptide sequence datasets, Human_denovo and Human_spider published by Tran et al, are used as test set to evaluate the assembly methods [16].

As shown in Table 1, we compare IMGT, VBASE, and our program on different functions. They are both able to distinguish between light and heavy chains and perform sequence alignment online. Neither IMGT nor VBASE can batch process and download the result files. IMGT and VBASE methods are highly effective in distinguishing between light and heavy chains [29–32]. However, they both analyze and compare antibody sequences online. Our program has tens of thousands of data to process. If it is online, it will take a longer time to deal with the data because of network and other problems. Our program requires sequence alignment result files. IMGT and VBASE cannot provide the function of batching process and download the result files after the sequence alignment is finished, so it is very difficult to get the result files. Therefore, we have improved BLAST and used it to build a local database. All variable gene sequences in VBASE are extracted from the EMBL database. The database SGSM used by IMGT consists of sequence databases, genome database, structure database, and monoclonal antibodies database. To obtain more antibody data, we collected antibody sequence data from the SGSM and abYsis database.

Next, we introduce the advantages of VBASE and IMGT methods. VBASE serves as a platform for interconnection between several existing self-contained data systems and it can be accessed either by a web-based text query or by a sequence similarity search with the DNAPLOT software [29,30]. In contrast to VBASE, VBASE2 provides new information and sequences as it implements the current

knowledge derived from the genome sequencing projects by linking to the Ensembl Genome Browser. Then, IMGT provides a high-quality and integrated system for analysis of the genomic and expressed IG and TR repertoire of the adaptive immune responses, including NGS high-throughput data [31,32]. IMGT is the acknowledged high-quality integrated knowledge resource in immunogenetics for exploring immune functional genomics. Finally, we present the advantages of our program. Our program uses BLAST localization software to differentiate between light and heavy chains of peptide sequences and then constructs a weighted de Bruijn graph to assemble the peptide sequences. For mixed light and heavy chain data, our program is able to implement light and heavy chain sequence assembly separately. For single-chain data, our program can also achieve the corresponding assembly. Taking the human_denovo dataset mentioned in Table 4 of the article as an example, contig assembly accuracy of the light chain is as high as 97.14% and contig assembly accuracy of the heavy chain is as high as 78.40% by using our program. These data sets are mainly provided on the web by Tran et al. [16]. In terms of these assembly results in Table 4, our procedure has better contig assembly accuracy than the ALPS method. Therefore, VBASE, IMGT, and our program have their own advantages in some ways.

Table 1. Comparison of IMGT, VBASE, and Our Program in terms of some functions.

Function	light and heavy chain differentiation	Online web server	Local batch processing	Source of dataset
IMGT	✓	✓	×	SGSM
VBASE	✓	✓	×	EMBL
Our Program	✓	×	✓	SGSM + abYsis

In the sequence alignment section, we need to collect human light-heavy chain sequences in the standard database. Thousands of human light-heavy chain sequences from the IMGT database and abYsis database are used to build BLAST localization database. We distinguish between light and heavy chains from mixed data by using BLAST sequence alignment.

We are performing peptide sequence assembly on two peptide datasets, Human_denovo and Human_spider. Human_denovo is the human peptide sequence dataset processed by de novo sequencing. It has 23,248 light chain peptide sequences and 26,047 heavy chain peptide sequences. The other dataset, Human_spider is obtained by using PEAKS SPIDER to deal with Human_denovo. SPIDER tries to match de novo sequence tags with the database proteins and reconstructs a true sequence to minimize the sum of de novo errors and homology mutations between the true sequence and the one recorded in the database when a significant similarity is found. It has 13,176 light chain peptide sequences and 14,742 heavy chain peptide sequences. Each sequence dataset consists of light and heavy chain peptide sequences. Then, we use the corresponding human antibody full length sequence containing both heavy and light chains. The full length of human light chain is 216 AA and the full length of heavy chain is 446 AA.

3.2. The effect of k-mer

In the k-mer processing of the peptide sequences, we consider that a too small segmentation length could lead to incorrect sequences on the match. Therefore, we choose three conditions as following, k = 5, k = 6 and k = 7 in peptide length, to compare the obtained experimental results and select the most suitable k-value. The k-value is the length of k-mer. The test data of Human_denovo is taken

as an example. The assembly results of the Human_denovo and Human_spider datasets at k = 5 or k = 6 are shown in Table 2. When k = 5 is chosen, the target sequence coverage of the light chain result is about 5.09% lower than that of k = 6. Its assembly accuracy is about 34.29% lower than that of k = 6. The target sequence coverage of the heavy chain result is about 3.81% lower than that of k = 6. Its assembly accuracy is about 19.11% lower than that of k = 6. Similarly, using the test data Human_spider as an example, the coverage of each target and assembly accuracy for k = 5 are also lower than those for k = 6. At this point, the same method is used to test k = 7. When the conditions are k = 6 or k = 7, the assembly results of DBAS method are greatly improved.

Table 2. Sequence assembly results for the two datasets at k = 5 or k = 6.

DBAS	Human-Light (216 AA)				Human-Heavy (446 AA)			
	len1[1]	len2[2]	Cov[3]	Acc[4]	len1	len2	Cov	Acc
Human_denovo k = 5	253	159	73.61	62.85	253	150	33.63	59.29
Human_denovo k = 6	175	170	78.70	97.14	213	167	37.44	78.40
Human_spider k = 5	285	182	84.26	63.86	285	171	38.34	60.00
Human_spider k = 6	216	216	100.0	100.0	212	205	45.96	96.70

Note: [1]The length of the best assembled contig. [2]Number of Amino Acids Recovered (AA). [3]The target sequence coverage was calculated as the percentage of amino acids of the target sequence that were covered by the respective best contig. [4]The contig assembly accuracy was calculated as the percentage of correct amino acids of the best contig that were aligned to the respective target sequence.

The results show that k = 6 or k = 7 can achieve the optimal antibody sequence assembly. Therefore, we use k = 6. In addition, we set four other parameters. l represents the input data as a collection of light chain sequences. h represents the input data as a collection of heavy chain sequences. lh represents the input data as a collection of mixed light and heavy chain sequences. A k-mer is a consecutive sequence of k amino acids, and k is the length of the k-mer. contigs_num represents the number of output results, which is used to set the number of assembled contigs. A contig represents a long peptide sequence that is formed by merging multiple reads. The parameters are described as shown in Table 3.

Table 3. Parameter description of the DBAS method.

Parameter	Description
l	light chain
h	heavy chain
lh	the mixed light and heavy chain
k	k-mer parameter setting, k = 6
contigs_num	total number of assembled contigs

3.3. The compared performance between DBAS and ALPS

The light chain and heavy chain of these datasets have 216 and 446 amino acids, respectively. In this paper, we use an assembly method that combines de Bruijn graph and BLAST sequence comparison algorithm. From the above, the value of k and the number of sequences in the sequence alignment database both have a certain degree of influence on the result of the assembly. The optimal

value of k is 6. After assembling peptide sequences by using our assembly method, a light chain assembly result file and a heavy chain assembly result file are generated. Each result file has 10 contigs. The assembled contigs are subjected to BLAST sequence alignment with the target sequences. Some alignment details are shown in Figure 5. As shown in Figure 5(a), the full length of a light chain contig assembled by DBAS is 216 AA, which can mostly match with the full length of human target light chain. The target sequence coverage and contig assembly accuracy are very high. As shown in Figure 5(b), the full length of a heavy chain contig assembled by DBAS is 205 AA, and it can match the human target heavy chain from No. 241 to No. 445 completely. The sequence from No. 1 to No. 240 of the target heavy chain can match with the other contigs which our method assembled.

```
Query  1    ELVLTQSPASLSLSPGERATLSCRASQSVSSYLAWYQHKPGQAPRLLLYDASTRATGLPA  60
            ELVLTQSPASLS+SPGERATLSCRASQSVSSYLAWYQHKPGQAPR+L+YDASTRATG+PA
Sbjct  1    ELVLTQSPASLSISPGERATLSCRASQSVSSYLAWYQHKPGQAPRILIYDASTRATGIPA  60

Query  61   RFSGSGSGTDFTLTLSSLEPEDFALYYCQQRSNWPPSFTFGPGTRVDLKRTVAAPSVFLF  120
            RFSGSGSGTDFTLT+SSLEPEDFA+YYCQQRSNWPPSFTFGPGTRVDLKRTVAAPSVF+F
Sbjct  61   RFSGSGSGTDFTLTISSLEPEDFAIYYCQQRSNWPPSFTFGPGTRVDLKRTVAAPSVFIF  120

Query  121  PPSDEQLKSGTASVVCLLNNFYPREAKVQWKVDNALQSGNSQESVTEQDSKDSTYSLSST  180
            PPSDEQLKSGTASVVCLLNNFYPREAKVQWKVDNA+QSGNSQESVTEQDSKDSTYS+SST
Sbjct  121  PPSDEQLKSGTASVVCLLNNFYPREAKVQWKVDNAIQSGNSQESVTEQDSKDSTYSISST  180

Query  181  LTLSKADYEKHKVYACEVTHQGLSSPVTKSFNRGEC  216
            LTLSKADYEKHKVYACEVTHQGLSSPVTKSFNRGEC
Sbjct  181  LTLSKADYEKHKVYACEVTHQGLSSPVTKSFNRGEC  216
```

(a)

```
Query  1    FLFPPKPKDTLMLSRTPEVTCVVVDVSHEDPEVKFNWYVDGVEVHNAKTKPREEQYNSTY  60
            FLFPPKPKDTLMLSRTPEVTCVVVDVSHEDPEVKFNWYVDGVEVHNAKTKPREEQYNSTY
Sbjct  241  FLFPPKPKDTLMLSRTPEVTCVVVDVSHEDPEVKFNWYVDGVEVHNAKTKPREEQYNSTY  300

Query  61   RVVSVLTVLHQDWLNGKEYKCKVSNKALPAPLEKTLSKAKGQPREPQVYTLPPSREEMTK  120
            RVVSVLTVLHQDWLNGKEYKCKVSNKALPAPLEKTLSKAKGQPREPQVYTLPPSREEMTK
Sbjct  301  RVVSVLTVLHQDWLNGKEYKCKVSNKALPAPLEKTLSKAKGQPREPQVYTLPPSREEMTK  360

Query  121  NQVSLTCLVKGFYPSDLAVEWESNGQPENNYKTTPPVLDSDGSFFLYSKLTVDKSRWQQG  180
            NQVSLTCLVKGFYPSDLAVEWESNGQPENNYKTTPPVLDSDGSFFLYSKLTVDKSRWQQG
Sbjct  361  NQVSLTCLVKGFYPSDLAVEWESNGQPENNYKTTPPVLDSDGSFFLYSKLTVDKSRWQQG  420

Query  181  NVFSCSVMHEALHNHYTQKSLSLSP  205
            NVFSCSVMHEALHNHYTQKSLSLSP
Sbjct  421  NVFSCSVMHEALHNHYTQKSLSLSP  445
```

(b)

Figure 5. (a) BLAST alignment of the best contig assembled from Human_spider against the target light chain. (b) BLAST alignment of the best contig assembled from Human_spider against the target heavy chain. The Query is contig and the Sbjct is target sequence in Figure 5.

The results of sequence assembly method on the datasets Human_denovo and Human_spider are shown in Table 4. For comparison, we also show the results of the ALPS method. Since the ALPS method cannot distinguish the light and heavy chains, the results are compared with the original human light and heavy chain sequences separately. From the table data, taking the dataset Human_denovo as an example, the light chain target sequence coverage of DBAS is 5.09% higher than ALPS. Its assembly accuracy is 34.29% higher than ALPS. The heavy chain target sequence coverage of DBAS is 6.72%

higher than ALPS. Its assembly accuracy is 24.25% higher than ALPS. Taking the dataset Human_spider as an example, the light chain target sequence coverage of DBAS is 15.74% higher than ALPS. Its assembly accuracy is 36.14% higher than ALPS. The heavy chain target sequence coverage of DBAS is 10.09% higher than ALPS. Its assembly accuracy is 40.56% higher than ALPS.

Table 4. Sequence assembly results for the two datasets at ALPS or DBAS methods (k = 6).

k = 6	Human-Light (216 AA)				Human-Heavy (446 AA)			
	len1[1]	len2[2]	Cov[3]	Acc[4]	len1	len2	Cov	Acc
Human_denovo ALPS	253	159	73.61	62.85	253	137	30.72	54.15
Human_spider ALPS	285	182	84.26	63.86	285	160	35.87	56.14
Human_denovo DBAS	175	170	78.70	97.14	213	167	37.44	78.40
Human_spider DBAS	216	216	100.0	100.0	212	205	45.96	96.70

Note: [1]The length of the best assembled contig. [2]Number of Amino Acids Recovered (AA). [3]The target sequence coverage was calculated as the percentage of amino acids of the target sequence that were covered by the respective best contig. [4]The contig assembly accuracy was calculated as the percentage of correct amino acids of the best contig that were aligned to the respective target sequence.

The DBAS method is mainly applied to mixed light and heavy chain datasets. However, DBAS is also able to perform sequence assembly on a single light chain dataset or heavy chain dataset. Its assembly results have better target sequence coverage and contig assembly accuracy compared with the ALPS method. The results of the DBAS method on the heavy chain datasets HumanH_denovo and HumanH_spider are shown in Table 5. The heavy chain dataset HumanH_denovo has 26,074 heavy chain peptide sequences. The target sequence coverage of DBAS on this dataset reached 32.06% and contig assembly accuracy reached 96.62%. The heavy chain dataset HumanH_spider has 14,742 heavy chain peptide sequences. The target sequence coverage of DBAS on this dataset reached 76.91% and contig assembly accuracy reached 99.13%.

Table 5. Sequence assembly results for other datasets at ALPS or DBAS methods (k = 6).

k = 6	Human-Heavy (446 AA)			
	len1[1]	len2[2]	Cov[3]	Acc[4]
HumanH_denovo ALPS	154	121	27.13	78.57
HumanH_spider ALPS	346	343	76.91	99.13
HumanH_denovo DBAS	148	143	32.06	96.62
HumanH_spider DBAS	346	343	76.91	99.13

Note: [1]The length of the best assembled contig. [2]Number of Amino Acids Recovered (AA). [3]The target sequence coverage was calculated as the percentage of amino acids of the target sequence that were covered by the respective best contig. [4]The contig assembly accuracy was calculated as the percentage of correct amino acids of the best contig that were aligned to the respective target sequence.

4. Discussion

From the above, the value of k and the number of sequences in the sequence alignment database both have a certain degree of influence on the result of the assembly. In the Human_denovo dataset, the contig assembly accuracy of the human heavy chain reached 78.40% and the contig assembly

accuracy of the human light chain reached 97.14%. In the Human_spider dataset, the contig assembly accuracy of the human heavy chain reached 96.70% and the contig assembly accuracy of the human light chain reached 100%. The human light chain sequence assembled by DBAS method can almost match the target sequence, while the human heavy chain sequence can only match half of the target sequence. Compared with results of ALPS, the target sequence coverage and contig assembly accuracy of DBAS are significantly improved.

The human heavy chain sequence assembled by DBAS cannot all match the target sequence. It is possible that the sequence data in BLAST localization database is not rich enough, for which we can collect more antibody sequences. It is also possible that some k-mers match with multiple sequences in the database during sequence alignment, for which we can screen the sequence alignment results and perform the alignment again. These are some guesses about the problems in DBAS.

Compared with other methods, the powerful performance of de Bruijn graph has been demonstrated in major genome and transcriptome assemblers such as Velvet [4], Trinity [33], and others. In the field of de novo protein sequencing, the idea of de Bruijn graph has been used for spectral alignment (A-Bruijn) [34]. The de Bruijn graph shows superiority in the above literature. In addition, the established method is used to assembling sequences from two datasets with mixed light and heavy chains from antibodies. The key advantage of the assembly method proposed in this paper is that DBAS can distinguish and assemble long antibody sequences of light and heavy chains separately. Our method obtains better performance in sequence coverage and accuracy for light chain and heavy chain assembly compared to current state of art technology, ALPS.

5. Conclusions

In this paper, a sequence assembly method, DBAS, combining weighted de Bruijn graph and BLAST sequence comparison was designed to perform sequence assembly of antibody with mixed light and heavy chain peptide sequences. The assembly sequences show high sequence coverage and contig assembly accuracy, which are very close to the target sequences. Compared with other assembly methods, DBAS can discriminate light and heavy chain sequences based on the mixed peptide de novo sequencing data, and then assemble light and heavy chains separately from mixed data. The determination of light and heavy chains for mixed data is based on some sequence alignment tools. In addition, this method has better target sequence coverage and contig assembly accuracy compared to ALPS. It is also able to obtain longer and more continuous contigs. Adding the sequence alignment scores to the weighted de Bruijn graph is an innovative point. With the rapid development of mass spectrometry, the coverage and accuracy of peptide sequencing will continue to be improved. This method provides a solution for sequence assembly for mixed antibody light and heavy chains.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (81603152) and the fund of Changzhou Sci. and Tech. Program (CE20205033).

Conflict of interest

The authors declare no conflict of interest. The funders had no role in the design of the study; in

the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. V. Pham, W. J. Henzel, D. Arnott, S. Hymowitz, W. N. Sandoval, B. T. Truong, et al., De novo proteomic sequencing of a monoclonal antibody raised against OX40 ligand, *Anal. Biochem.*, **352** (2006), 77–86. https://doi.org/10.1016/j.ab.2006.02.001
2. C. S. Pareek, R. Smoczynski, A. Tretyn, Sequencing technologies and genome sequencing, *J. Appl. Genet.*, **52** (2011), 413–435. https://doi.org/10.1007/s13353-011-0057-x
3. X. Liao, M. Li, Y. Zou, F. X. Wu, Y. Pan, J. Wang, Current challenges and solutions of de novo assembly, *Quant. Biol.*, **7** (2019), 90–109. https://doi.org/10.1007/s40484-019-0166-9
4. D. R. Zerbino, E. Birney, Velvet: Algorithms for de novo short read assembly using de Bruijn graphs, *Genome Res.*, **18** (2008), 821–829. https://doi.org/10.1101/gr.074492.107
5. N. Bandeira, H. Tang, V. Bafna, P. Pevzner, Shotgun protein sequencing by tandem mass spectra assembly, *Anal. Chem.*, **76** (2004), 7221–7233. https://doi.org/10.1021/ac0489162
6. J. A. Baaijens, A. Z. E. Aabidine, E. Rivals, A. Schönhuth, De novo assembly of viral quasispecies using overlap graphs, *Genome Res.*, **27** (2017), 835–848. https://doi.org/10.1101/gr.215038.116
7. C. Ge, Y. Lu, J. Qu, L. Xie, F. Wang, H. Zhang, et al., DePS: An improved deep learning model for de novo peptide sequencing, preprint, arXiv:2203.08820.
8. A. Guthals, K. R. Clauser, A. M. Frank, N. Bandeira, Sequencing-grade de novo analysis of MS/MS triplets (CID/HCD/ETD) from overlapping peptides, *J. Proteome Res.*, **12** (2013), 2846–2857. https://doi.org/10.1021/pr400173d
9. B. Ma, K. Zhang, C. Hendrie, C. Liang, M. Li, A. Doherty-Kirby, et al., PEAKS: Powerful software for peptide de novo sequencing by tandem mass spectrometry, *Rapid Commun. Mass Spectrom.*, **17** (2003), 2337–2342. https://doi.org/10.1002/rcm.1196
10. M. M. Rahman, R. Sharker, S. Biswas, M. S. Rahman, HaVec: An efficient de Bruijn graph construction algorithm for genome assembly, *Int. J. Genomics*, **2017** (2017), 1–12. https://doi.org/10.1155/2017/6120980
11. J. Zhang, L. Xin, B. Shan, W. Chen, M. Xie, D. Yuen, et al., PEAKS DB: De novo sequencing assisted database search for sensitive and accurate peptide identification, *Mol. Cell. Proteomics*, **11** (2012). https://doi.org/10.1074/mcp.M111.010587
12. J. Sohn, J. W. Nam, The present and future of de novo whole-genome assembly, *Briefings Bioinf.*, **19** (2018), 23–40. https://doi.org/10.1093/bib/bbw096
13. R. E. Green, A. S. Malaspinas, J. Krause, A. W. Briggs, P. L. Johnson, C. Uhler, et al., A complete Neandertal mitochondrial genome sequence determined by high-throughput sequencing, *Cell*, **134** (2008), 416–426. https://doi.org/10.1016/j.cell.2008.06.021
14. M. Li, Z. Liao, Y. He, J. Wang, J. Luo, Y. Pan, ISEA: Iterative seed-extension algorithm for de novo assembly using paired-end information and insert size distribution, *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **14** (2017), 916–925. https://doi.org/10.1109/TCBB.2016.2550433
15. J. Butler, I. MacCallum, M. Kleber, I. A. Shlyakhter, M. K. Belmonte, E. S. Lander, et al., ALLPATHS: De novo assembly of whole-genome shotgun microreads, *Genome Res.*, **18** (2008), 810–820. https://doi.org/10.1101/gr.7337908

16. N. H. Tran, M. Z. Rahman, L. He, L. Xin, B. Shan, M. Li, Complete de novo assembly of monoclonal antibody sequences, *Sci. Rep.*, **6** (2016), 1–10. https://doi.org/10.1038/srep31730
17. M. Ayling, M. D. Clark, R. M. Leggett, New approaches for metagenome assembly with short reads, *Briefings Bioinf.*, **21** (2020), 584–594. https://doi.org/10.1093/bib/bbz020
18. S. F. Altschul, W. Gish, W. Miller, E. W. Myers, D. J. Lipman, Basic local alignment search tool, *J. Mol. Biol.*, **215** (1990), 403–410. https://doi.org/10.1016/S0022-2836(05)80360-2
19. O. S. Upasani, M. M. Vaidya, A. N. Bhisey, Database on monoclonal antibodies to cytokeratins, *Oral Oncol.*, **40** (2004), 236–256. https://doi.org/10.1016/j.oraloncology.2003.08.022
20. W. Li, R. Li, H. Liu, X. Guo, A. S. Shaikh, P. Li, et al., A comparison of liquid chromatography-tandem mass spectrometry (LC-MS/MS) and enzyme-multiplied immunoassay technique (EMIT) for the determination of the cyclosporin A concentration in whole blood from Chinese patients, *BioSci. Trends*, **11** (2017), 475–482. https://doi.org/10.5582/bst.2017.01121
21. A. Guthals, Y. Gan, L. Murray, Y. Chen, J. Stinson, G. Nakamura, et al., De novo MS/MS sequencing of native human antibodies, *J. Proteome Res.*, **16** (2017), 45–54. https://doi.org/10.1021/acs.jproteome.6b00608
22. R. B. Batista, A. Boukerche, A. C. M. A. de Melo, A parallel strategy for biological sequence alignment in restricted memory space, *J. Parallel Distrib. Comput.*, **68** (2008), 548–561. https://doi.org/10.1016/j.jpdc.2007.08.007
23. K. Katoh, J. Rozewicki, K. D. Yamada, MAFFT online service: Multiple sequence alignment, interactive sequence choice and visualization, *Briefings Bioinf.*, **20** (2019), 1160–1166. https://doi.org/10.1093/bib/bbx108
24. P. Pandey, M. A. Bender, R. Johnson, R. Patro, deBGR: An efficient and near-exact representation of the weighted de Bruijn graph, *Bioinformatics*, **33** (2017), i133–i141. https://doi.org/10.1093/bioinformatics/btx261
25. J. Liu, Q. Lian, Y. Chen, J. Qi, Amino acid based de Bruijn graph algorithm for identifying complete coding genes from metagenomic and metatranscriptomic short reads, *Nucleic Acids Res.*, **47** (2019), e30. https://doi.org/10.1093/nar/gkz017
26. G. Peng, P. Ji, F. Zhao, A novel codon-based de Bruijn graph algorithm for gene construction from unassembled transcriptomes, *Genome Biol.*, **17** (2016), 1–12. https://doi.org/10.1186/s13059-016-1094-x
27. R. Rizzi, S. Beretta, M. Patterson, Y. Pirola, M. Previtali, G. D. Vedova, et al., Overlap graphs and de Bruijn graphs: Data structures for de novo genome assembly in the big data era, *Quant. Biol.*, **7** (2019), 278–292. https://doi.org/10.1007/s40484-019-0181-x
28. A. Bankevich, A. V. Bzikadze, M. Kolmogorov, D. Antipov, P. A. Pevzner, Multiplex de Bruijn graphs enable genome assembly from long, high-fidelity reads, *Nat. Biotechnol.*, **40** (2022), 1075–1081. https://doi.org/10.1038/s41587-022-01220-6
29. I. Retter, H. H. Althaus, R. Münch, W. Müller, VBASE2, an integrative V gene database, *Nucleic Acids Res.*, **33** (2005), D671–D674. https://doi.org/10.1093/nar/gki088
30. S. Mollova, I. Retter, W. Müller, Visualising the immune repertoire, *BMC Syst. Biol.*, **1** (2007), 1. https://doi.org/10.1186/1752-0509-1-S1-P30
31. M. P. Lefranc, V. Giudicelli, C. Ginestoux, J. J. Michaloud, G. Folch, F. Bellahcene, et al., IMGT®, the international ImMunoGeneTics information system®, *Nucleic Acids Res.*, **37** (2009), D1006–D1012. https://doi.org/10.1093/nar/gkn838

32. M. P. Lefranc, V. Giudicelli, P. Duroux, J. J. Michaloud, G. Folch, S. Aouinti, et al., IMGT®, the international ImMunoGeneTics information system® 25 years on, *Nucleic Acids Res.*, **43** (2015), D413–D422. https://doi.org/10.1093/nar/gku1056
33. M. G. Grabherr, B. J. Haas, M. Yassour, J. Z. Levin, D. A. Thompson, I. Amit, et al., Full-length transcriptome assembly from RNA-Seq data without a reference genome, *Nat. Biotechnol.*, **29** (2011), 644–652. https://doi.org/10.1038/nbt.1883
34. N. Bandeira, K. R. Clauser, P. A. Pevzner, Shotgun protein sequencing: Assembly of peptide tandem mass spectra from mixtures of modified proteins, *Mol. Cell. Proteomics*, **6** (2007), 1123–1134. https://doi.org/10.1074/mcp.M700001-MCP200

AIMS Press