journal homepage: www.intl.elsevierhealth.com/journals/cmpb

Improvements on a privacy-protection algorithm for DNA sequences with generalization lattices

*Guang Li**, *Yadong Wang, Xiaohong Su*

School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, People's Republic of China

ARTICLE INFO

Article history:
Received 29 April 2010
Received in revised form
9 February 2011
Accepted 21 February 2011

Keywords:
Personal DNA database
Privacy protection
k-Anonymity

ABSTRACT

When developing personal DNA databases, there must be an appropriate guarantee of anonymity, which means that the data cannot be related back to individuals. DNA lattice anonymization (DNALA) is a successful method for making personal DNA sequences anonymous. However, it uses time-consuming multiple sequence alignment and a low-accuracy greedy clustering algorithm. Furthermore, DNALA is not an online algorithm, and so it cannot quickly return results when the database is updated. This study improves the DNALA method. Specifically, we replaced the multiple sequence alignment in DNALA with global pairwise sequence alignment to save time, and we designed a hybrid clustering algorithm comprised of a maximum weight matching (MWM)-based algorithm and an online algorithm. The MWM-based algorithm is more accurate than the greedy algorithm in DNALA and has the same time complexity. The online algorithm can process data quickly when the database is updated.

1. Introduction

With the development of genotyping technology, DNA sequences are increasingly becoming a part of the patient medical record [1], and an increasing amount of personal DNA sequences are being collected. Databases of personal DNA sequences are also being developed. The collection of DNA occurs at many different kinds of institutions: at research sites for clinical trials and basic research, at hospitals for diagnostic testing, and at commercial companies, where gene discovery is of high commercial value.

At the same time, genomic data pose complex privacy problems. The genetic information of an individual is as personally revealing as a fingerprint, if not more revealing [2]. Many people fear that information gleaned from their genomic data, such as a health situation or a family member relationship, will be misused or abused to influence their employment and insurance status or simply to cause social stigma [3,4]. In addition to social pressures, there are legal mechanisms for protecting genomic data privacy, such as the Privacy Rule of the Health Insurance Portability and Accountability Act in the United States and the Data Protection Act of 1998 in the European Union. Thus, without an appropriate guarantee of anonymity, not only will patients be less willing to provide data but also many data collectors will be unable to share genomic data for worthwhile endeavors. Given this situation, genomic privacy is considered one of the major challenges for the biomedical community [5,6].

Unfortunately, contrary to popular belief, the protection of a patient's anonymity in genomic data is not as simple as removing or encrypting explicit identifying attributes, such as name or social security number [7–10]. To resolve this problem, a DNA sequence anonymity method called DNA lattice anonymization (DNALA) has been presented [11]. It is based on the *k*-anonymity principle.

* *Corresponding author at*: Room 434, the A17 Student Dormitory, Science Park, Harbin Institute of Technology, No. 2 Yikuang Street, Nangang District, Harbin, Heilongjiang 150001, People's Republic of China. Tel.: +86 15846591735.
E-mail address: hit6006@126.com (G. Li).

0169-2607/$ – see front matter © 2011 Elsevier Ireland Ltd. All rights reserved.
doi:10.1016/j.cmpb.2011.02.013

Although DNALA can protect the privacy of personal DNA data, it has some shortcomings. Specifically, it uses the multiple sequence alignment, which is time-consuming, together with a low-accuracy greedy clustering algorithm. Furthermore, DNALA is not an online algorithm and, thus, cannot quickly return results when the database is updated.

Li et al. [12] improved the original DNALA method by replacing the multiple sequence alignment with global pairwise sequence alignment to save time; they also replaced the greedy clustering algorithm with stochastic hill-climbing method to improve the precision of the clustering. However, the stochastic hill-climbing method is not an online algorithm; moreover, there is still potential for improvements in precision.

Our study in this paper focused on overcoming the shortcomings in DNALA. When calculating distance matrix, this study used the method in Li et al. [12] by replacing the multiple sequence alignment in DNALA with a global pairwise sequence alignment. For clustering, it replaced the greedy algorithm in DNALA with a hybrid algorithm consisting of a maximum weight matching (MWM)-based algorithm and an online algorithm. Compared with DNALA, our new method is faster, especially when the database is updated, and it can achieve more accurate results.

The rest of this paper is organized in the following manner. Section 2 introduces some related work, while Section 3 provides details on the original DNALA method. Section 4 explains the principal ideas behind our method, and Section 5 presents the improvements for calculating the distance matrix. Section 6 discusses the MWM-based clustering algorithm, and Section 7 discusses the online clustering algorithm. Finally, Section 8 presents the results of the experiments, and Section 9 presents the conclusions.

2. Related work

2.1. *k*-Anonymity

The *k*-anonymity principle is a privacy protection principle proposed by Samarati and Sweeney [13–15]. It has been extensively studied in recent years [16–19].

The key idea behind *k*-anonymity is to make individuals indistinguishable in a released table. A data set complies with *k*-anonymity protection if each individual's record stored in the released data set cannot be distinguished from at least $k-1$ other individuals, whose data also appear in the data set. This protection guarantees that the probability of identifying an individual based on the released data in the data set does not exceed $1/k$. The larger the value of *k* is, the better the protection.

Generalization and suppression are the two main methods for achieving *k*-anonymity. Generalization means replacing a value by a less specific, more general value that is faithful to the original one, for example, by using "Europe" to replace "Netherlands" or using "[20–30]" to replace "27". Suppression means removing data from a table so that they are not released. Suppression can be done for tuples [13] or attributes [15,19]. Generalization and suppression can be used alone or in combination.

Recently, some improvements over *k*-anonymity have been proposed [20], including *p*-sensitive *k*-anonymity [21], (α, *k*)-anonymity [22], *l*-diversity [23] and *t*-closeness [24]. They all can be achieved by similar methods for achieving *k*-anonymity.

2.2. Genomic data privacy protection

For the research needs of and potential benefits to health care [25], personal genomic data should be shared. But as we showed above, because of social concerns and public policy, person-specific genomic records must be shared in a manner that preserves the anonymity of data subjects.

There are three main ways for protecting privacy for personal genomic data. They are data deidentification, data augmentation and methods based on cryptology.

The deidentification method protects privacy by removing or encrypting person-specific identifiers, such as name and social security number, which are initially associated with genomic records. Recent studies have shown that data deidentification is not enough to protect privacy [7–10].

Data augmentation is often achieved by generalization. It protects privacy by making each record be indistinguishable from some other shared records [11,26,27]. DNALA is a data augmentation method that performs generalization on DNA sequences [11]. The study by Zhen et al. [26] mainly focused on single nucleotide polymorphisms but not DNA sequences. Loukides et al. [27] performed generalization on diagnoses codes but not on DNA sequences.

Cryptology-based methods do not access the original data. They maintain data utility by using privacy-preserving data querying methods that can be applied to genomic sequences [28,29].

2.3. Sequence alignment

In bioinformatics, sequence alignment is a way of arranging the sequences of DNA, RNA or protein to identify regions of similarity that may be a consequence of functional, structural or evolutionary relationships between the sequences. This paper only considers global sequence alignment.

From the perspective of computer science, global sequence alignment is the process of adding gaps to sequences and making them as similar as possible, not least by giving them the same length. The added gaps should be as few as possible.

We assume that sequence alignment is done for *n* sequences. If $n=2$, it is called pairwise sequence alignment, and if $n>2$, it is called multiple sequence alignment.

For example, if using "#" to represent a gap, then the alignment for "AAACGTTT" and "AAACGCTTT" is as follows.

```
A   A   A   C   G   #   T   T   T
A   A   A   C   G   C   T   T   T
```

In addition, the alignment for "AAACCGCTTT", "AAACGTTT" and "AAACGCTTT" is the following.

```
A   A   A   C   C   G   C   T   T   T
A   A   A   #   C   G   #   T   T   T
A   A   A   #   C   G   C   T   T   T
```

For pairwise sequence alignment, the ordinary dynamic programming algorithm has time complexity $O(nm)$, where

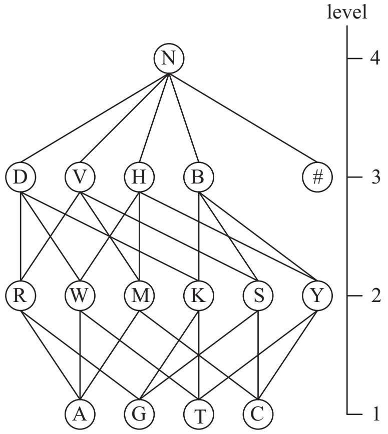

Fig. 1 – DNA generalization lattice used in DNALA.

n and *m* are the lengths of the two sequences. For multiple sequence alignment, the dynamic programming algorithm has exponential time complexity, and heuristic methods are often used [30].

CLUSTALW used by DNALA is a famous algorithm for multiple sequence alignment [31]. CLUSTALW has three steps. In the first step, it performs pairwise sequence alignment for all pairs of sequences. In the second step, it builds a guide tree. In the third step, the sequences are progressively aligned according to the branching order in the guide tree.

3. The DNALA algorithm

DNALA reduces data precision and enhances privacy. It uses the *k*-anonymity principle with $k = 2$. DNALA changes DNA sequences to ensure that every sequence has at least one other sequence that is the same as it. To retain the usability of the data, this change should be as slight as possible, and the loss of information should be minimal. DNALA has two steps: (1) generate a distance matrix and (2) generalize the DNA sequences.

3.1. *The definition of generalization and distance*

In DNALA, the generalization of characters is defined as the lowest common predecessor of these characters in the graph shown in Fig. 1 (Fig. 1 and Table 1 are adapted from Malin [11]). We assume that g (a, b) is the generalization of characters a and b, and s and t are two sequences with the same length n. The generalization of s and t is the string composed by $g(s[1], t[1])$, $g(s[2], t[2]), \ldots$, and $g(s[n], t[n])$, where $s[i]$ and $t[i]$ are the ith characters of s and t, respectively.

In DNALA, the distance is defined to measure the information loss in generalization. The distance between two characters x and y is defined as $d\ (x, y) = 2lev\ (z) - lev\ (x) - lev\ (y)$, where z is the generalization of x and y, and $lev\ (a)$ is the level of character a in Fig. 1.

If s and t are two sequences with the same length n, the distance between them is thesum of the distance of all characters at the same position, as given by the following formula.

$$d(s, t) = \sum_{i=1}^{n} d(s[i], t[i]).$$

For example, "M" is the value to which "A" and "C" generalize; thus, d (A, C) = 2 × 2 − 1 − 1 = 2, and d (ACC, CAA) = 2 + 2 + 2 = 6. The generalization of "ACC" and "CAA" is "MMM".

3.2. *The distance matrix*

If there are n sequences, the distance matrix A is a matrix with dimensions $n \times n$. We assume a_{ij} is the entry in the ith row and jth column of A. If $i = j$, a_{ij} does not have any noteworthy meaning; else, a_{ij} equals the distance between the ith sequence and the jth sequence.

Original sequences are often not aligned. DNALA uses the CLUSTALW algorithm to perform multiple sequence alignment for all sequences. After that, all sequences have the same length, and the distance matrix can be derived.

3.3. *Clustering and generalization*

After establishing the distance matrix, clustering is performed to divide the sequences into groups. Each group has at least two sequences. Two groups cannot include the same sequence, and the distance between the sequences in one group should be as small as possible. After clustering, all sequences in one group are replaced with their generalization.

The DNALA method uses a greedy clustering algorithm. Two sequences a and b are selected randomly. If there is no sequence c that satisfies $d\ (c, a) < d\ (b, a)$ or $d\ (c, b) < d\ (a, b)$, let a and b be in one group, and delete them from the sequence set. Repeat this process until there are no sequences left. If there is an odd number of sequences, the last three sequences will be in one group.

4. The main concept behind the proposed algorithm

The DNALA algorithm has the following limitations.

(1) Multiple sequence alignment, which is used for calculating the distance matrix, is a time-consuming procedure [30].
(2) The precision of the greedy clustering algorithm is not so high, and it can be improved.
(3) The greedy clustering algorithm is not an online algorithm. When the database is updated, the entire algorithm must be run again to obtain the new result. The new result cannot be quickly obtained by simply adjusting the original result, even if the database is changed only slightly.

This study improved the DNALA algorithm in the following respects.

Table 1 – International Union of Biochemists (IUB) code for DNA and associated ambiguities.

A	Adenine	G	Guanine
T	Thymine	C	Cytosine
M	A or C	R	A or G
W	A or T	S	C or G
Y	C or T	K	G or T
V	A or G or C	H	A or T or C
D	A or G or T	B	G or T or C
#	Gap (added in sequence alignment)	N	Indeterminate (A or G or T or C or gap)

Table 2 – A comparison of clustering algorithms.

Method	Greedy algorithm in DNALA	MWM-based algorithm	Online algorithm	Stochastic hill-climbing algorithm
Time complexity to process *n* sequences	$O(n^3)$	$O(n^3)$	$O(n^2)$	This is a randomized algorithm, and so the running time and result precision are uncertain. The longer the algorithm is run, the more likely it obtains better results.
Time complexity to add or delete one sequence within a database of *n* sequences	$O(n^3)$	$O(n^3)$	$O(n)$	
Result precision	In-between	High	Low	
Online algorithm	No	No	Yes	No

(1) We used the method in Li et al. [12] to calculate the distance matrix by using global pairwise sequence alignment to replace the multiple sequence alignment in DNALA.

(2) The DNALA greedy clustering algorithm was replaced by a hybrid algorithm comprised of a MWM-based algorithm and an online algorithm.

The MWM-based algorithm has the same time complexity as the greedy algorithm in the DNALA, and it has higher precision.

The online clustering algorithm can obtain a new result quickly by simply adjusting the old result when the database is updated. However, it is less precise.

In the hybrid algorithm, the MWM-based algorithm and the online algorithm are used together. When the database is updated, the online algorithm can be used to quickly obtain results. The MWM-based algorithm is used periodically to improve accuracy. The frequency of implementing the MWM-based algorithm is determined by users. If necessary, we can exclusively use the MWM-based algorithm.

Table 2 shows a comparison of some clustering algorithms, including the MWM-based algorithm, the online algorithm, the greedy algorithm in DNALA and the stochastic hill-climbing algorithm in Li et al. [12].

5. Improvements in calculating the distance matrix

In our method, the multiple sequence alignment in DNALA was replaced by global pairwise sequence alignment for all pairs of sequences to calculate the distance matrix, as in Li et al. [12].

Essentially, the difficulty of DNA sequence generalization is that the sequences are not aligned. We do not know what a character in one sequence corresponds to in other sequences. For comparison, in an ordinary relational database, every value is clearly demarcated with the attributes it has. A value in one tuple corresponds to values in the same attribute in other tuples. Distance and generalization as defined by DNALA cannot be calculated directly for unaligned sequences.

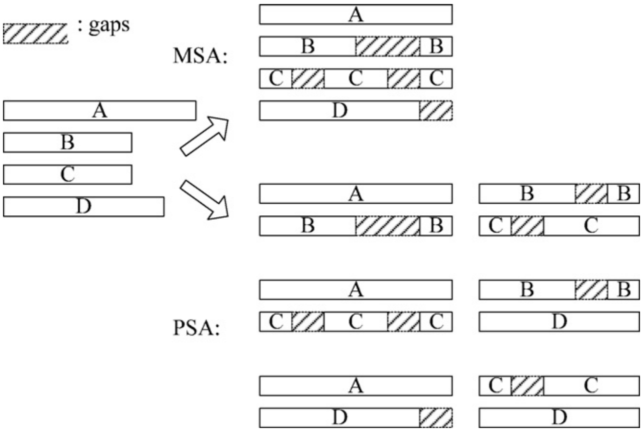

Fig. 2 – Schematic diagram of the pairwise sequence alignment (PSA) and multiple sequence alignment (MSA). The results from PSA and MSA can be the same (for example, A and B, A and C, A and D in this figure), or be the same but require deletion of gaps in the same location (for example, B and C in this figure), or be different (for example, B and D, C and D in this figure).

DNALA uses multiple sequence alignment to solve this problem. As Fig. 2 shows, after multiple sequence alignment, all sequences are aligned. Some gaps are added into the sequences, and all sequences have the same length.

After analysis, we noted that the distance is defined for two sequences, while clusters always include two sequences in DNALA. As such, generalization is always performed for two sequences. That means that we do not have to align all sequences but only every two sequences. Thus, pairwise sequence alignment is adequate for our objectives. As Fig. 2 shows, after pairwise sequence alignment for all pairs of sequences, every two sequences are aligned, and thus, distance and generalization can be calculated.

```
MWM-based clustering algorithm for an odd number of sequences.

Input: Original sequence set $S$, which has an odd
       number of sequences.
Output: $S$'s clustering result $M$.

Step 1: Find sequences $a$ and $b$ that have the smallest
        distance.
Step 2: Assuming $f$ is the generalization of $a$ and $b$, let
        $S' = S + f - a - b$. Then $|S'|$ is even. Find the
        MWM $M'$ of $S'$. $M'$ is the best clustering result
        for $S'$.
Step 3: If $c$ and $f$ are in the same group in $M'$, obtain
        the final result $M$ by deleting the group $\{c, f\}$ and
        adding the group $\{a, b, c\}$. That means $M = M' -$
        $\{c, f\} + \{a, b, c\}$ is the clustering result for $S$.
        Stop.
```

Fig. 3 – The workflow of the MWM-based clustering algorithm for an odd number of sequences.

```
Online algorithm for adding a sequence.
$M' = Insert\ (S, M, a)$

Input: Sequence set $S$, the added sequence $a$ and $S$'s
       clustering result $M = \{m_1, m_2, \ldots, m_k\}$, $|m_i \cap m_j| =$
       $0\ (i \neq j)$, with $|m_i| = 2$ or $3$ and $m_1 \cup m_2 \cup \ldots \cup m_k =$
       $S$.
Output: $(S + a)$'s clustering result $M'$.

Step 1: Find $b$, which is the nearest sequence of $a$ in $S$.
        $m_i$ is the group including $b$.
Step 2: Let $m = m_i + a$. If $|m| = 3$, go to Step 3. If $|m| =$
        $4$, go to Step 4.
Step 3: Let $M' = M - m_i + m$. Stop.
Step 4: Divide $m$ into two groups $p$ and $q$ using
        enumeration, where $|p| = |q| = 2$. If $p = \{e_1, e_2\}$
        and $q = \{e_3, e_4\}$, the sum $d\ (e_1, e_2) + d\ (e_3, e_4)$ is
        the smallest, where $d\ (e_1, e_2)$ is the distance
        between $e_1$ and $e_2$. Let $M' = M - m_i + p + q$. Stop.
```

Fig. 4 – The workflow of the online clustering algorithm when adding a sequence.

Because our method only completes pairwise sequence alignment of all pairs of sequences, which is the first step of the CLUSTALW algorithm used in DNALA, it clearly saves time. The experiments confirmed this as well. The experiments also showed that our method does not reduce the precision of the final result. Furthermore, our method works for online processing. When adding a new sequence, our method needs only to align it to each previous sequence. The alignment for the other sequences does not change because of the added sequences. If multiple sequence alignment is used when adding a sequence, we must align all sequences again, as the alignment for all sequences may be changed.

Because $k = 2$ for k-anonymity principle in DNALA and in our method, we try to make every cluster include two sequences. If there is an even number of sequences, a cluster of three sequences will appear. Furthermore, our online algorithm may generate some clusters with three sequences. Unlike under multiple sequence alignment, our method only aligns each pair of sequences. Thus, it cannot directly perform generalization for three sequences. To solve this problem, multiple sequence alignment is performed only for these three sequences to derive their generalization after clustering.

6. MWM-based clustering

We developed an MWM-based clustering algorithm that can obtain more precise results with the same time complexity as the greedy algorithm in DNALA. In this proposed algorithm, the problem of clustering is translated into finding the MWM on a graph.

(1) We assume that there is an even number of sequences, and M is a number larger than the distance between any two sequences. G is a complete graph, and each node of G corresponds to a sequence. The weight of the edge between two of G's nodes corresponding to sequences a and b is $M - d\ (a, b)$, where $d\ (a, b)$ is the distance between a and b. The best clustering result then corresponds to the MWM of G. The problem of clustering on the distance matrix is thus equivalent to the problem of finding the MWM on a graph.

(2) If there is an odd number of sequences, we convert the clustering problem to the situation which has an even number of sequences. Fig. 3 shows the details of it.

In the algorithm presented in this paper, "$A + a$" means "add the element a into set A," and "$A - a$" means "remove the element a from set A."

Finding the MWM is a classic problem in graph theory. We used the accurate algorithm, which has time complexity $O(n(m + n \log n))$, where n is the number of nodes in the graph, and m is the number of edges in the graph [32]. In our algorithm, the MWM is found in a complete graph. Thus, $m = n(n - 1)/2$, and the time complexity of our algorithm is $O(n^3)$, which is the same as the greedy algorithm in DNALA. Our experiments showed that the MWM-based algorithm can obtain better results than the greedy algorithm in DNALA.

The MWM-based algorithm is also better than the stochastic hill-climbing method presented in Li et al. [12]. The stochastic hill-climbing method is a randomized algorithm. Running it twice under the same conditions may result in different results. The longer the algorithm is run, the more likely it will obtain better results. The MWM-based algorithm is a deterministic algorithm, and so its performance is steady. The experiments show that even with much more computational time, the results from the stochastic hill-climbing method are still slightly worse than those of the MWM-based algorithm.

7. Online clustering

We designed an online classification algorithm that can quickly obtain results when the database is updated.

We assumed A and B are two different sequence sets; B is the update of A, and A's clustering result is known. The online algorithm calculates B's clustering result by adjusting the clustering result of A. If only the sequence set B is given,

Table 3 – Details on the experimental data sets.

	Data set	
	I	II
GenBank IDs	AF387914–AF387969	AF392063–AF392434
Origin	Melanocortin gene promoter	Human mtDNA sequences
Sequences number	56	372
Sequences length	6.6 kb	0.5 kb
Average levenshtein distance	29.56	29.91

```
Online algorithm for deleting a sequence.
$M' = Delete(S, M, a)$

Input: Sequence set $S$, deleting sequence $a$, which is in
    $S$, and $S$'s clustering result $M = \{m_1, m_2, \ldots, m_k\}$,
    $|m_i \cap m_j| = 0$ $(i \neq j)$, with $|m_i| = 2$ or 3 and $m_1 \cup m_2$
    $\cup \ldots \cup m_k = S$.
Output: $(S - a)$'s clustering result $M'$.

Step 1: The group including $a$ is $m_i$. If $|m_i| = 3$, go to
    Step 2. If $|m_i| = 2$, go to Step 3.
Step 2: Let $m = m_i - a$ and $M' = M - m_i + m$. Stop.
Step 3: If $m_i = \{a, b\}$, let $M = M - m_i$, $S = S - a - b$ and
    $M' = Insert(S, M, b)$. Stop.
```

Fig. 5 – The workflow of the online clustering algorithm when deleting a sequence.

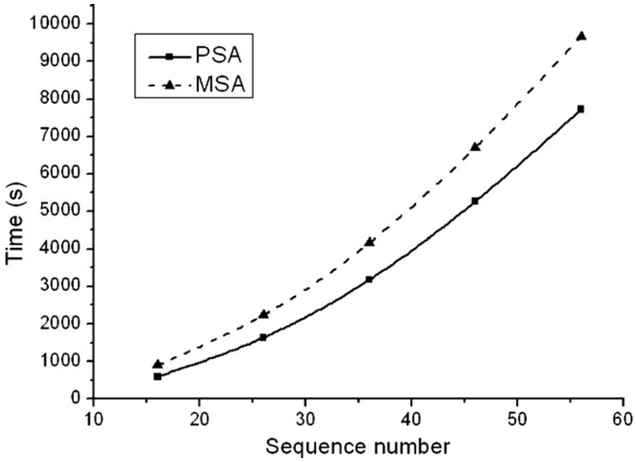

Fig. 6 – The comparison of running time using different methods to generate distance matrices with data set I. PSA means the pairwise sequence alignment and MSA means multiple sequence alignment.

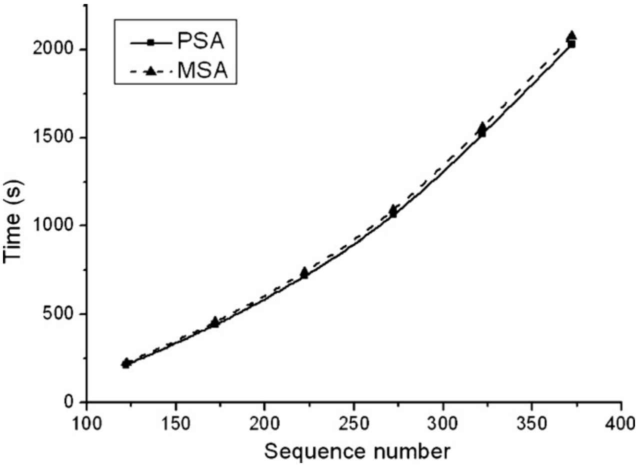

Fig. 7 – The comparison of running time using different methods to generate distance matrices with data set II. PSA means the pairwise sequence alignment and MSA means multiple sequence alignment.

we let A be the set of two sequences selected randomly from B.

The updating process can be split into two steps. First, add $(B \setminus A)$ into A; then, delete $(A \setminus B)$ from the new A. This split can be expressed as the formula: $B = A \cup (B \setminus A) \setminus (A \setminus B)$.

Noting that adding or deleting a sequence set is equivalent to adding or deleting the sequences in this set one-by-one, our online clustering algorithm has two parts that correspond to adding and deleting a sequence. Figs. 4 and 5 show details of them, respectively.

As Table 2 showed, the online algorithm requires less time than the greedy algorithm in DNALA, especially for changing database. If there are n sequences in the database, adding or deleting a sequence using the online algorithm takes $O(n)$ time; using the greedy algorithm, this takes $O(n^3)$ time.

8. Experimental results and analysis

The experiments used two real data sets [33,34]. Table 3 provides details on them.

Experiment 1: This experiment aimed to demonstrate that our distance matrix calculating method is better than the original method in DNALA.

In this experiment, we derive two distance matrices for each data set. One is calculated according to our method using pairwise sequence alignment, and the other is calculated by the DNALA method using multiple sequence alignment.

Figs. 6 and 7 show the running time of these two algorithms for increasingly larger subsets of the data sets. The sequences in the subsets are selected in ascending order of Genbank IDs. Considering that this experiment examined the actual computer running time, which can differ by a small amount even when running the same program on the same computer, we repeated these calculations of the two distance matrices three times and averaged the running time to obtain the data shown in Figs. 6 and 7.

Based on Figs. 6 and 7, our method using pairwise sequence alignment requires less time than the DNALA method using multiple sequence alignment. This phenomenon is not obvi-

Table 4 – The average distances between sequences and their generalizations using different clustering algorithms on different distance matrices. PSA means the pairwise sequence alignment and MSA means multiple sequence alignment.

Clustering method	Data set	MSA	PSA
DNALA's greedy algorithm	I	13.79	13.57
	II	3.33	3.35
MWM-based algorithm	I	13.39	13.18
	II	2.99	2.98
Online algorithm	I	16.93	16.81
	II	3.79	3.80
Stochastic hill-climbing algorithm	I	13.39	13.18
	II	3.13	3.11

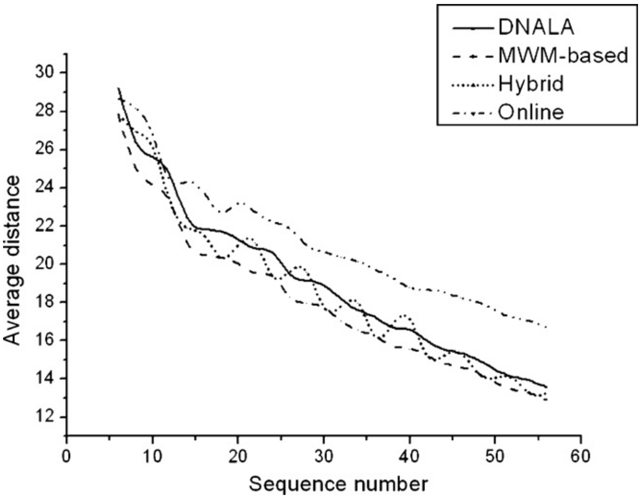

Fig. 8 – The average distances between sequences and their generalizations for data set I.

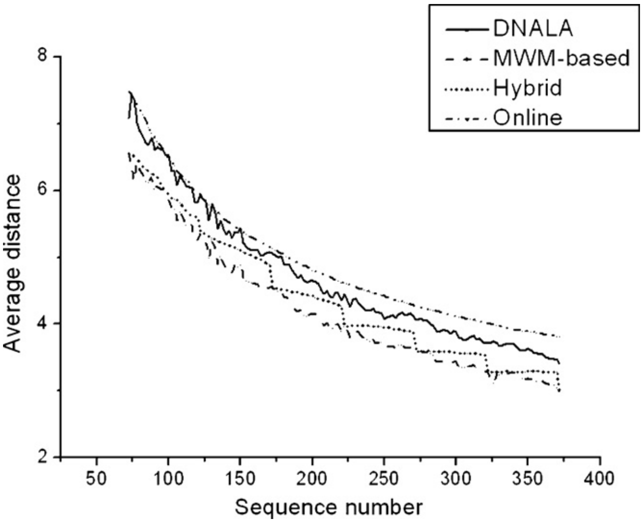

Fig. 9 – The average distances between sequences and their generalizations for data set II.

ous in data set II (the pairwise sequence alignment is about 1.2 times faster than the multiple sequence alignment for data set I, and about 1.02 times faster for the data set II), which we believe is due to the characteristics of this data set. Data set II has much more sequences than data set I, and these sequences are much shorter than the sequences in data set I. With respect to sequence length, the sequences in data set II are less similar than those in data set I.

Table 4 shows the average distances of the original sequences and their generalizations. The clustering results are obtained by using four different clustering algorithms on two distance matrices. These four clustering algorithms are the greedy algorithm in DNALA, the stochastic hill-climbing algorithm [12], our proposed online algorithm and the MWM-based algorithm. The two distance matrices are separately calculated by our method and the DNALA method. In the stochastic hill-climbing algorithm, the number of hill climbs is set to 100,000. The stochastic hill-climbing algorithm was repeated three times and averaged to obtain the data shown in Table 4. In the online algorithm, the sequences are added in a random order. The online algorithm was repeated 100 times and averaged to obtain the data shown in Table 4.

Table 4 shows that for all four clustering algorithms, the use of a particular distance matrix is not important to the precision of results. That is, each clustering algorithm achieves similar results for the two different distance matrices. Hence, replacing the multiple sequence alignment with the pairwise sequence alignment while calculating the distance matrix does not reduce the precision of the result.

Experiment 2: This experiment compared the DNALA clustering algorithm and our clustering algorithms.

Figs. 8 and 9 show the average distances of the original sequences and their generalizations based on the greedy clustering algorithm in DNALA, the online clustering algorithm, the MWM-based algorithm and the hybrid algorithm using increasingly larger subsets of the data sets. At the beginning, there were 6 sequences for data set I and 72 sequences for data set II. The sequences were added two-by-two in a random order. In the hybrid algorithm, the MWM-based algorithm was run after every 6 added sequences for data set I and after every 50 added sequences for data set II. This experiment used the distance matrix calculated by our method using pairwise sequence alignment. The experiment was repeated 100 times and averaged to obtain the data shown in Figs. 8 and 9.

As we showed before in Table 2, the online algorithm is faster than the DNALA greedy algorithm, especially when updating the database. In addition, the MWM-based algorithm has the same time complexity as the DNALA greedy algorithm.

From Figs. 8 and 9, note that as compared with the DNALA greedy method, the online algorithm obtains a lower precision result, and the MWM-based algorithm obtains a higher precision result.

In the hybrid algorithm, the online algorithm and MWM-based algorithm are alternately used. Figs. 8 and 9 showed that with the online algorithm, result precision drops. In addition, the MWM-based algorithm effectively improves the results. The peaks in the figures for hybrid algorithm are formed because the average distance of the original sequences and their generalizations continuously increased under the online algorithm but sharply decreased under the MWM-based algorithm.

The hybrid algorithm provides flexibility for users, who can choose how often to run the MWM-based algorithm, according to their needs. The more frequently the MWM-based algorithm is used, the better the result will be; however, it will require more computing time. In an extreme case, the online algorithm may never be used; in this case, the hybrid algorithm is simply the MWM-based algorithm, which has the same time complexity as the DNALA greedy algorithm but can achieve better results.

9. Conclusion

This study improved the DNALA, which is an algorithm that makes personal DNA sequences anonymous, in the following respects.

(1) When calculating the distance matrix, we use the method presented in Li et al. [12]. Multiple sequence alignment in DNALA was replaced by pairwise sequence alignment for all pairs of sequences to improve efficiency.
(2) The greedy clustering algorithm in DNALA is replaced by a hybrid clustering algorithm, which is comprised of an MWM-based algorithm and an online algorithm.

The MWM-based algorithm can achieve better clustering results with the same time complexity as the greedy algorithm in DNALA. The online algorithm has an efficiency advantage, especially when the database is updated. However, the accuracy of the results is not high. The hybrid algorithm was designed to take advantage of these two algorithms. The online algorithm is used when the database is updated to quickly obtain results, while the MWM-based algorithm is run periodically or when there are abundant computing resources available to improve the results. The frequency of running the MWM-based algorithm is determined by users. If necessary, the MWM-based algorithm can be used alone.

Our method and the DNALA method are both based on the *k*-anonymity principle, and both fix *k* at 2 according to the definition of distance [11]. This finding means that users cannot decide on the level of privacy protection, and sometimes, $k=2$ may not be adequate for privacy. In the future, further research should be conducted to address this issue.

In recent years, some improvements over the *k*-anonymity principle have been proposed [20–24]. In the future research, we aim to develop a DNA sequence anonymity method based on these improvements.

Although *k*-anonymity is a successful privacy protection principle, it has mainly been studied for a single database. If we can collect multiple databases that all satisfy *k*-anonymity, privacy information on the elements of these databases may be leaked [35]. In future research, we plan to study privacy protection in the context of multiple databases.

Conflict of interest

None declared.

Acknowledgement

The authors would like to thank National High-Tech Research and Development Plan of China, under Grant No. 2007AA02Z329, for funding this research.

REFERENCES

[1] R.B. Altman, Bioinformatics in support of molecular medicine, in: Proceedings of 1998 AMIA Symposium, 1998, pp. 53–61.
[2] C. Leonard, G. Chase, G. Childs, Genetic counseling: a consumers view, New England Journal of Medicine 287 (9) (1972) 433–439.
[3] M.A. Hall, S.S. Rich, Patients' fear of genetic discrimination by health insurers: the impact of legal protections, Genetics in Medicine 2 (4) (2000) 214–221.
[4] E.W. Clayton, Ethical, legal, and social implications of genomic medicine, New England Journal of Medicine 349 (6) (2003) 562–569.
[5] L.T Vaszar, M.K. Cho, T.A. Raffin, Privacy issues in personalized medicine, Pharmacogenomics 4 (2) (2003) 107–112.
[6] R.B. Altman, T.E. Klein, Challenges for biomedical informatics and pharmacogenomics, Annual Review of Pharmacology and Toxicology 42 (2002) 113–133.
[7] B. Malin, L. Sweeney, Determining the identifiability of DNA database entries, in: Proceedings of the 2000 AMIA Symposium, 2000, pp. 537–541.
[8] B. Malin, L. Sweeney, How (not) to protect genomic data privacy in a distributed network: using trail re-identification to evaluate and design anonymity protection systems, Journal of Biomedical Informatics 37 (3) (2004) 179–192.
[9] B. Malin, L. Sweeney, Re-identification of DNA through an automated linkage process, in: Proceedings of the 2001 AMIA Symposium, 2001, pp. 423–427.
[10] B. Malin, An evaluation of the current state of genomic data privacy protection technology and a roadmap for the future, Journal of the American Medical Informatics Association 12 (1) (2005) 28–34.
[11] B. Malin, Protecting genomic sequence anonymity with generalization lattices, Methods of Information in Medicine 44 (5) (2005) 687–692.
[12] G. Li, Y. Wang, X. Su, X. Li, Improvement of a method of privacy protection for personal DNA data, China Journal of Bioinformatics 2 (2007) 78–81.
[13] P. Samarati, Protecting respondents' identities in microdata release, IEEE TKDE 13 (6) (2001) 1010–1027.
[14] L. Sweeney, k-Anonymity: a model for protecting privacy, International Journal on Uncertainty, Fuzziness and Knowledge-based Systems 10 (5) (2002) 557–570.
[15] L. Sweeney, Achieving k-anonymity privacy protection using generalization and suppression, International Journal on Uncertainty, Fuzziness and Knowledge-based Systems 10 (5) (2002) 571–588.
[16] LeFevre Kristen, David F DeWitt, Ramakrishnan Raghu, Incognito: efficient full-domain k-anonymity, in: Proceedings

of the 2005 ACM SIGMOD International Conference on Management of Data (SIGMOD'05), 2005, pp. 49–60.

[17] LeFevre Kristen, David F DeWitt, Ramakrishnan Raghu, Mondrian multidimensional k-anonymity, in: Proceedings of the 22nd International Conference on Data Engineering (ICDE'06), 2006, p. 25.

[18] Jiuyong Li, Raymond Chi-Wing Wong, Ada Wai-Chee Fu, Jian Pei, Anonymization by local recoding in data with attribute hierarchical taxonomies, IEEE TKDE 20 (9) (2008) 1181–1194.

[19] Slava Kisilevich, Lior Rokach, Yuval Elovici, Bracha Shapira, Efficient multidimensional suppression for k-anonymity, IEEE TKDE 22 (3) (2010) 334–347.

[20] Josep Domingo-Ferrer, Vicenc Torra, A critique of *k*-anonymity and some of its enhancements, in: Proceedings of the Third International Conference on Availability, Reliability and Security (ARES'08), 2008, pp. 990–993.

[21] Traian Marius Truta, Bindu Vinay, Privacy protection: p-sensitive k-anonymity property, in: Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06), 2006, p. 94.

[22] Raymond Chi-Wing Wong, Jiuyong Li, Ada Wai-Chee Fu, Ke Wang, (α, k)-Anonymity: an enhanced k-anonymity model for privacy-preserving data publishing, in: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'06), 2006, pp. 754–759.

[23] M. Ashwin, Daniel Kifer, Johannes Gehrke, V. Muthuramakrishnan, *L*-diversity: privacy beyond *k*-anonymity, ACM Transactions on Knowledge Discovery from Data (TKDD) 1 (1) (2007) 1–55.

[24] Ninghui Li, Tiancheng Li, V. Suresh, t-Closeness: privacy beyond k-anonymity and l-diversity, in: Proceedings of the IEEE 23rd International Conference on Data Engineering (ICDE'07), 2007, pp. 106–115.

[25] Nils Homer, et al., Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays, PLoS Genetics 4 (8) (2008) e1000167.

[26] Zhen Lin, Micheal Hewett, Russ B. Altman, Using binning to maintain confidentiality of medical data, in: Proceedings of the American Medical Informatics Association Annual Symposium, 2002, pp. 454–458.

[27] Grigorios Loukides, Aris Gkoulalas Divanis, Bradley Malin, Anonymization of electronic medical records for validating genome-wide association studies, PNAS 107 (17) (2010) 7898–7903.

[28] Murat Kantarcioglu, Wei Jiang, Ying Liu, Bradley Malin, A cryptographic approach to securely share and query genomic sequences, IEEE Transactions on Information Technology in Biomedicine 12 (5) (2008) 606–617.

[29] Michael T. Goodrich, The mastermind attack on genomic data, in: The 2009 30th IEEE Symposium on Security and Privacy, 2009, pp. 204–218.

[30] Joao Setubal, Joao Meidanis, Introduction to Computational Molecular Biology, Scientist Publishing Company, Beijing, China, 2003, translated by Zhu Hao.

[31] Julie D. Thompson, Desmond G. Higgins, Toby J. Gibson, Clustal W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice, Nucleic Acids Research 22 (22) (1994) 4673–4680.

[32] Xie Zheng, Network Algorithm and Theory of Complexity, Publishing Company of the National University of Defense Technology, Changsha, China, 2004.

[33] K.D Makova, M. Ramsay, T. Jenkins, W.H. Li, Human DNA sequence variation in a 6.6-kb region containing the melanocortin 1 receptor promoter, Genetics 158 (3) (2001) 1253–1268.

[34] Y.G. Yao, L. Nie, H. Harpending, Y.X. Fu, Z.G. Yuan, Y.P. Zhang, Genetic relationship of Chinese ethnic populations revealed by mtDNA sequence diversity, American Journal of Physical Anthropology 118 (1) (2002) 63–76.

[35] Ji-Won Byun, Yonglak Sohn, Elisa Bertino, Ninghui Li, Secure anonymization for incremental datasets, Lecture Notes in Computer Science (Secure Data Management) 4165 (2006) 48–63.