

Algorithm for complex network diameter based on distance matrix

CHEN Bin^{1,2,*}, ZHU Weixing¹, and LIU Ying³

1. Institute of Command Information System, PLA University of Science and Technology, Nanjing 210007, China;

2. College of Computer Science and Engineering, Sanjiang University, Nanjing 210012, China;

3. Institute of Field Engineering, PLA University of Science and Technology, Nanjing 210007, China

Abstract: The network diameter is an important characteristic parameter of a complex network. Calculation for a large-scale complex network's diameter has been an important subject in the study of complex networks. If the network diameter is calculated directly, the problem mainly exists in efficiency for searching and counting the shortest paths. If the network diameter is calculated indirectly by studying the statistical function about the relationship between the network diameter and parameters affecting the diameter, the problems not only exist in the efficiency of statistic, but also exist in the function which may be not applicable to all kinds of networks. An algorithm for the complex network diameter based on the k order distance matrix is proposed with a matrix multiplication approach, and a mathematical proof for the algorithm correctness is given as well. Furthermore, some relevant propositions and deductions for reducing the complexity of this algorithm are put forward. With a good theoretical basis and a simple calculation process, this algorithm can be used to calculate the diameter of a large-scale complex network with small-world effect more accurately and efficiently. Two cases about the advanced research projects agency (ARPA) network model and the Chinese airline network model are adopted to verify the effect of this algorithm.

Keywords: complex network, network diameter, distance matrix.

DOI: 10.21629/JSEE.2018.02.14

1. Introduction

The average path length of a network, often called network diameter, is usually regarded as one of the important characteristic parameters [1,2] to distinguish network types (such as regular network, stochastic network and complex network) and to measure network performances (such as node importance and network survivability). The common formula for calculating the network diameter is shown as follows:

$$D = \frac{\sum_{i>j} d_{ij}}{n(n-1)/2}. \quad (1)$$

It can be seen from (1) that network diameter D is the ratio of the sum of the shortest path lengths for all node pairs to the number of node pairs. n is the number of node pairs, representing the network size. d_{ij} is the length of the shortest path between node i and node j . The network diameter describes the average minimum number of edges between any two nodes in a network, which represents the closeness of network nodes on the whole [3,4].

If a large-scale complex network is considered with a special connection way, difficulty in direct calculation of network diameter lies in how to search and count the shortest paths efficiently [5]. Searching the shortest paths has always been an important topic in the study of complex networks [6–8]. Typical searching algorithms include Floyd, Dijkstra, Warshall [9], A* algorithm, ant colony algorithm, parallel computing [10–12] and other intelligent algorithms. The advantage of using these algorithms lies in the generality for different network types. However, the disadvantage lies in the efficiency. For example, the time complexity of the Floyd algorithm or the Warshall algorithm is $O(n^3)$, which is applicable for small-scale networks. However, for large-scale networks, the speed of the current computer is often insufficient for completing these algorithms execution within an acceptable time.

Except the direct way, people also try to use indirect way to calculate the network diameter of a complex network. There are two tendencies for the indirect way. One is to study the statistical function about the relationship between network diameter D and network size n so as to calculate D through the simple and intuitive parameter n . The other is to study parameters influencing D in a specific network type so as to calculate D through these parameters. For example, the network diameter of a stochastic

Manuscript received January 03, 2017.

*Corresponding author.

This work was supported by the National Natural Science Foundation of China (61273210).

network is generally expressed as $D \sim \ln(n)$; Albert et al. [13] proposed a prediction formula $0.35 + 2.06 \lg(n)$ for calculating the diameter for the WWW network; Lu et al. [14] proposed calculation formulas for different complex network types based on the network size, but it is difficult to explain the complicated relationship between the network topology and the network diameter by calculating the network diameter through the network size. Networks even with the same size may have different total numbers of edges and different connection ways between nodes. As a result, these networks may have different diameters. Through (1), it also can be intuitionistically analyzed that the network diameter shows a non-monotonic functional relationship with the network size because the numerator and the denominator of (1) correlate positively with the network size. In fact, just because of this relationship, complex networks show some amazing features, such as small world, high concentration and free scale. For example, when the size of some complex networks increases rapidly, such as exponential network and scale-free network, the diameter does not increase significantly. Two persons in a social network consisted of billions of people can interrelate each other by six jumps on the average. The diameter of the WWW network is only 11.2 jumps but it has 325 729 nodes. And the scale-free network will show “robust and fragile” characteristics when it is attacked [15]. In addition, for networks with the same type such as two WWW networks, even they are all conforming to power-law degree distribution $p(k) \sim k^{-\alpha}$, the exponential α may be different. A formula with only one parameter n cannot describe the influence of different α . Considering this, Li et al. [16] built a distribution $p(l)$ about the lengths of the shortest paths which only have parameter α to represent the node connection mode and the total number of edges in Internet with power-law degree distribution $p(k) \sim k^{-\alpha}$. That means if the exponential α of Internet is calculated statistically, the distribution $p(l)$ about the lengths of the shortest paths will be defined. If the network size n is also gotten, diameter D of Internet can be calculated finally. However, it is the same difficult to indirectly calculate the exponential α as to directly calculate the number of the shortest paths in a network. Meanwhile, this approach cannot apply to all types of networks.

To solve these problems, an algorithm based on the k order distance matrix is proposed as well as mathematical justification for its correctness. Furthermore, some relevant propositions and deductions for reducing complexity of the algorithm are put forward. Finally, effect of this algorithm is verified by two cases study.

2. Definition of k order distance matrix with its solution algorithm

2.1 Definition of k order distance matrix

Definition 1 The k ($k > 0$) order distance matrix of a

network with n nodes is $D_n^k = [d_{ij}^k]_{n \times n}$, where

$$d_{ij}^k = \begin{cases} k', & \text{If there is a shortest path} \\ & \text{between node } i \text{ and node } j \text{ with} \\ & \text{the length of } k' \text{ (} 0 < k' \leq k \text{)} \\ 0, & \text{Otherwise} \end{cases}.$$

2.2 Solution algorithm for k order distance matrix

(i) Assume the connection matrix of the network is $C_n = [c_{ij}]_{n \times n}$, where

$$c_{ij} = \begin{cases} 1, & \text{If there is an adjacent edge} \\ & \text{between node } i \text{ and node } j \\ 0, & \text{Otherwise} \end{cases}.$$

The first order distance matrix of the network equals the connection matrix, that is $D_n^1 = C_n$, where

$$d_{ij}^1 = c_{ij}. \quad (2)$$

(ii) Assume the k th ($k \geq 1$) power of the connection matrix C_n is C_n^k , the $(k+1)$ th power of C_n is

$$C_n^{k+1} = C_n^k \times C_n, \text{ where } c_{ij}^{k+1} = \sum_{q=1}^n c_{iq}^k \times c_{qj}, c_{ij}^{k+1}$$

equals the number of paths between node i and node j whose length is $k+1$ (there may be some repeated edges or nodes in a path). C_n^{k+1} is actually a number matrix of paths in the network whose length is $k+1$. If the k ($k \geq 1$) order distance matrix of the network is D_n^k , then the $k+1$ order distance matrix is $D_n^{k+1} = [d_{ij}^{k+1}]_{n \times n}$, where

$$d_{ij}^{k+1} = \begin{cases} d_{ij}^k, & d_{ij}^k > 0 \\ k+1, & d_{ij}^k = 0; c_{ij}^{k+1} > 0; i \neq j \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

(iii) The correctness of this algorithm is proved as follows:

i) If $d_{ij}^k > 0$: it can be inferred according to the definition of the distance matrix that there must be a shortest path between node i and node j whose length is less than or equal to k . Because the length of the shortest path is fixed, $d_{ij}^{k+1} = d_{ij}^k > 0$;

ii) If $d_{ij}^k = 0$ and $i \neq j$: assume $c_{ij}^{k+1} > 0$, there must be a path between node i and node j whose length is $k+1$. If this path is not the shortest path between node i and node j , there must be a shortest path between node i and node j whose length is less than $k+1$, its length can be assumed as k' ($k' \leq k$). Because $i \neq j$, $k' > 0$. It can be inferred according to the definition of the distance matrix that $d_{ij}^k = k' > 0$. This result is contradictory with the prerequisite of $d_{ij}^k = 0$. Therefore, that path must be the shortest path between node i and node j , and the length of that path is $k+1$, that is $d_{ij}^{k+1} = k+1$;

iii) If $d_{ij}^k = 0$ and $i \neq j$: assume $c_{ij}^{k+1} = 0$, there should not be a path between node i and node j whose length is

$k + 1$. If there is a shortest path between node i and node j whose length is less than $k + 1$, its length can be assumed as k' ($k' \leq k$). Because $i \neq j$, $k' > 0$, it can be inferred according to the definition of the distance matrix that $d_{ij}^k = k' > 0$. This result is contradictory with the prerequisite of $d_{ij}^k = 0$. Therefore, there is no shortest path between node i and node j whose length is less than $k + 1$. It can be inferred according to the definition of the distance matrix that $d_{ij}^{k+1} = 0$;

iv) If $i = j$: it is apparent that the distance between node i and node j is zero, that is $d_{ij}^{k+1} = 0$.

2.3 A solution algorithm for k order distance matrix

Assume the n order distance matrix of a network is $D_n^n = [d_{ij}^n]_{n \times n}$, the diameter of the network is

$$D = \frac{\sum_{i>j} d_{ij}^n}{n(n-1)/2}. \quad (4)$$

It is obvious that the distance between any two nodes will not exceed n in a network whose size is n , which can be reached and calculated before the n order distance matrix D_n^n is constructed. Thus the correctness of this algorithm can be proved according to (1).

3. Methods for improving efficiency of the solution algorithm for network diameter

The algorithm mentioned above calculates values of elements in the k order distance matrix by connection matrix multiplication. The time complexity of matrix multiplication [17] is generally $O(n^3)$. n matrix multiplications are adopted to calculate the n order distance matrix D_n^n , so the final time complexity of the algorithm is $O(n^4)$. Compared with the Floyd algorithm et al. with the time complexity of $O(n^3)$, efficiency of the algorithm above does not improve but reduces by an order of magnitude.

To improve efficiency of the algorithm so as to be applicable for large-scale networks, you should start with the optimization of the time complexity of the matrix multiplication. The time complexity of matrix multiplication should be optimized firstly. For a long time, nobody doubts matrix multiplication can be completed fewer than n^3 times. Until 1969, Strassen [18] reduced the time complexity of matrix multiplication from $O(n^3)$ to $O(n^{2.807\ 355})$. In 1981, Pan [19] reduced the time complexity to $O(n^{2.494})$. In 1987, Coppersmith and Winograd [20,21] reduced the time complexity to $O(n^{2.376})$. In 2010, Stothers [22] reduced the time complexity to $O(n^{2.374})$. In 2011, Williams [23] reduced the time complexity to $O(n^{2.372\ 864})$; in 2014, Gall [24] reduced the time complexity to $O(n^{2.372\ 863})$. For a large-scale network, the

complexity of $O(n^{2.372\ 863})$ is a very large increase than the complexity of $O(n^3)$, as is shown in Fig. 1 below. These work also laid a good foundation for the proposed algorithm to calculate the diameter of large-scale networks by matrix multiplications.

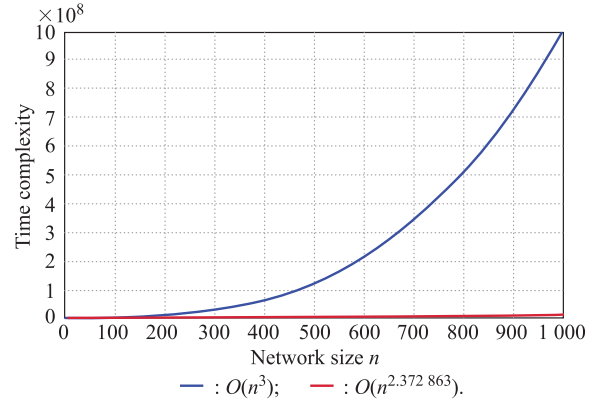


Fig. 1 Comparison on time complexity of matrix multiplication before and after optimization

The second way to improve efficiency of the proposed algorithm is to minimize the number of matrix multiplications. For most kinds of networks, especially for complex networks, to minimize the number of matrix multiplications is feasible because of their small-world characteristic. For example, on the average two persons can interrelate each other through six jumps in the social network consisted of more than seven billion people. Therefore, to calculate the social network diameter through the proposed algorithm does not actually need seven billion times of matrix multiplications. Some relevant propositions and deductions for minimizing the number of matrix multiplications are presented below for improving efficiency of the proposed algorithm.

3.1 Propositions and deductions for improving algorithm efficiency

Lemma 1 The lengths of the shortest paths in a network are consecutive natural numbers in range $[0, l]$, where l is the maximum length of the shortest paths in the network.

Proof (i) Assume that there is a shortest path $path1 : i \rightarrow \dots \rightarrow m \rightarrow j$ between node i and node j with length k in the network, where node m is the previous node of node j in the path and k ($k \geq 2$) is the number of edges in the path. There must be a shortest path between node i and node m . Assume this path is $path2 : i \rightarrow \dots \rightarrow m$. If the length of this path is greater than $k - 1$, node i can reach node m through $path1$ with length $k - 1$, which is contradictory with the second assumption that $path2$ is the shortest path between node i and node m . If the length of

this path is less than $k - 1$, node i can reach node j through $path3 : path2 \rightarrow j$ with the length less than k , which is contradictory with the first assumption that the length of the shortest path between node i and node j is k . Therefore, the length of the shortest path between node i and node m must be $k - 1$. It can be concluded that if there is a shortest path with length k in a network, there must be a shortest path with length $k - 1$.

(ii) If l ($l \geq 2$) is the maximum length of the shortest paths in a network, it can be inferred according to demonstration (i) that there must be a shortest path with length $l - 1$ in the network. The other lengths less than $l - 1$ can be inferred similarly. At last, it can be inferred that there must be a shortest path with length 1. Obviously, the length of the shortest path from node i to itself is 0. Therefore, the lengths of the shortest paths in a network are consecutive natural numbers in range $[0, l]$. $\square$

Proposition 1 Assume the k ($k \in (0, l]$) order distance matrix of a network with n nodes is $D_n^k = [d_{ij}^k]_{n \times n}$, where l ($l > 0$) is the maximum length of the shortest paths in the network, then $\max(d_{ij}^k) = k$ must be true.

Proof (i) Assume $k = 1$, it must be inferred according to the definition of the distance matrix that $\max(d_{ij}^1) \leq 1$. At this time, the distance matrix of the network equals the connection matrix of the network, namely $D_n^1 = C_n$. If $\max(d_{ij}^1) < 1$, all elements in distance matrix D_n^1 are zero, which means there is no adjacent edge between any two nodes in the network, there is no path between any two nodes and the maximum length of the shortest paths in the network is also zero. This conclusion is contradictory with prerequisite $l > 0$, so $\max(d_{ij}^1) = 1$ must be true. The proposition is thus demonstrated.

(ii) Assume $\max(d_{ij}^k) = k$ is true. There must be a shortest path with length k in the network. If $k + 1 \leq l$, it can be inferred according to Lemma 1 that the lengths of the shortest paths are consecutive natural numbers in range $[0, l]$, and there must be a shortest path with length $k + 1$ in the network. It can be inferred according to the definition of the distance matrix that $\max(d_{ij}^{k+1}) = k + 1$ is true.

(iii) It can be inferred that for $k \in (0, l]$, $\max(d_{ij}^k) = k$ must be true.

This proposition shows that the maximum value of the distance matrix is increasing progressively in range $(0, l]$ with order k of the distance matrix. $\square$

Proposition 2 Assume the k ($k > 0$) order distance matrix of a network with n nodes is D_n^k . If $D_n^{k-1} = D_n^k$, then $D_n^k = D_n^{k+1}$.

Proof (i) Assume l is the maximum length of the shortest paths in the network. According to Proposition 1, if $k \leq l$, $\max(d_{ij}^k) = k$ and $\max(d_{ij}^{k-1}) = k - 1$ must be true, then $D_n^{k-1} \neq D_n^k$ must be true, which is contradic-

tory with the prerequisite. Thus it can be inferred that if $D_n^{k-1} = D_n^k$, then $k > l$.

(ii) Because l is the maximum length of the shortest paths in the network, it can be inferred according to (3) that for $\forall d_{ij}^k \in D_n^k$:

i) If $d_{ij}^k > 0$, $d_{ij}^{k+1} = d_{ij}^k$ is true;

ii) If $d_{ij}^k = 0$ and $c_{ij}^{k+1} > 0$ and $i \neq j$, $d_{ij}^{k+1} = k + 1$.

It can be inferred that $l > k$, which is contradictory with proof (i). Therefore, the prerequisite is false;

iii) If $d_{ij}^k = 0$ and $i = j$ or $c_{ij}^{k+1} = 0$, $d_{ij}^{k+1} = 0$, and $d_{ij}^k = d_{ij}^{k+1}$ is true.

Therefore, it can be inferred that if $D_n^{k-1} = D_n^k$, $D_n^k = D_n^{k+1}$ is true. $\square$

Deduction 1 If $D_n^k = D_n^{k+1}$, D_n^k is the final distance matrix of the network.

Proof It can be inferred according to Proposition 2 that if $D_n^k = D_n^{k+1}$, $D_n^{k+1} = D_n^{k+2} = \dots = D_n^{k+i} = \dots$ is true ($i \geq 1$), namely the distance matrix will no longer change. Therefore, D_n^k is the final distance matrix of the network.

It can be inferred according to Deduction 1 that when $D_n^k = D_n^{k+1}$ appears, the derivation of the solution algorithm for the k order distance matrix of the network can be stopped, which can improve algorithm efficiency. Complex network, such as exponential network or scale-free network often has small-world effect. Thus the final distance matrix can be derived by several steps based on the 1 order distance matrix. $\square$

Deduction 2 If $D_n^k = D_n^{k+1}$, the minimum value of k is the maximum length of the shortest paths in a network.

Proof (i) If $0 < k < l$, it can be inferred according to Proposition 1 that the maximum value in the distance matrix increases progressively with order k in the range $(0, l]$. Thus $D_n^k \neq D_n^{k+1}$ is true, which is contradictory with the prerequisite $D_n^k = D_n^{k+1}$.

(ii) If $k = l$ and $D_n^l \neq D_n^{l+1}$, $d_{ij}^{l+1} \neq d_{ij}^l$ must be true.

i) If $d_{ij}^l > 0$, it can be inferred according to (3) that $d_{ij}^{l+1} = d_{ij}^l$, which is contradictory with the prerequisite $d_{ij}^{l+1} \neq d_{ij}^l$.

ii) If $d_{ij}^l = 0$, then $d_{ij}^{l+1} > 0$. It can be inferred according to (3) that if $c_{ij}^{l+1} > 0$ and $i \neq j$, then $d_{ij}^{l+1} = l + 1 > l$, which is contradictory with the prerequisite that l is the maximum length of the shortest paths in the network. If $i = j$ or $c_{ij}^{l+1} = 0$, $d_{ij}^{l+1} = 0$ must be true, which is contradictory with the prerequisite $d_{ij}^{l+1} \neq d_{ij}^l$.

iii) It can be inferred according to i) and ii) that $d_{ij}^{k+1} \neq d_{ij}^k$ cannot be true, so $D_n^l \neq D_n^{l+1}$ cannot be true. It can be inferred that if $k = l$ then $D_n^l = D_n^{l+1}$.

(iii) If $k > l$, it can be inferred according to Proposition 2 that if $D_n^k = D_n^{k+1}$, then $D_n^{k+1} = D_n^{k+2} = \dots$

must be true, namely the distance matrix no longer changes. Therefore, D_n^l is the final distance matrix of the network, and the minimum value of k equals the maximum length l of the shortest paths in the network.

This deduction shows that when $D_n^k = D_n^{k+1}$ appears for the first time in the solution algorithm of the k order distance matrix of a network, k is the maximum length of the shortest paths in the network, and D_n^k is the final distance matrix of the network. At this time, the derivation of the solution algorithm of the k order distance matrix of the network can be stopped. $\square$

It can be inferred according to Proposition 1 and Deduction 2 that the maximum element value $\max(d_{ij})$ of the k order distance matrix of a network increases progressively with order k , and when $k \geq l$, $\max(d_{ij})$ and D_n^k will no longer change, as is shown in Fig. 2.

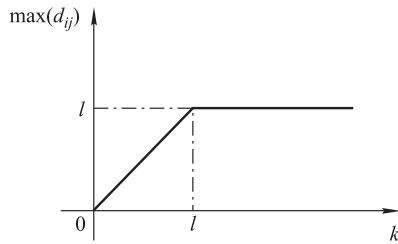


Fig. 2 Variation of maximum values $\max(d_{ij})$ with k

3.2 Analysis of algorithm efficiency

Propositions and deductions proposed above provide a simple, direct and effective judgment for calculating a large-scale network diameter by matrix multiplications with fewer orders, which also provide a good rationale for improving efficiency of the proposed algorithm. If the time complexity of matrix multiplication is $O(n^{2.372\ 863})$, the maximum length of the shortest paths in a network is a constant l , the time complexity of the proposed algorithm is $O(l * n^{2.372\ 863})$. When $1 < l < n^{0.627\ 137}$, the proposed algorithm is more efficient than the Floyd algorithm et al. with a time complexity of $O(n^3)$. The smaller l is when it is compared with n , the more efficient the proposed algorithm is, which means the larger size and the clearer small-world effect a network has, the more efficient the proposed algorithm is. For example, if the WWW network is considered and the maximum length of the shortest paths in it is assumed as 11.2 jumps, the relationship between the time complexity of the proposed algorithm and the network size n is shown in the figures below. From the figures, it can be seen that the efficiency advantage of the algorithm is unobvious when the network size is small. However, efficiency of the algorithm is greatly improved after optimization than before optimization when the network size reaches 1 000 (as is shown in Fig. 3). And efficiency of the algorithm

is also greatly improved if compared with the Floyd algorithm et al. with a time complexity of $O(n^3)$ (as shown in Fig. 4). The larger the network size is, the more obvious the efficiency advantage is. Thus the proposed algorithm is suitable for calculating diameters of large-scale complex networks with small-world effect.

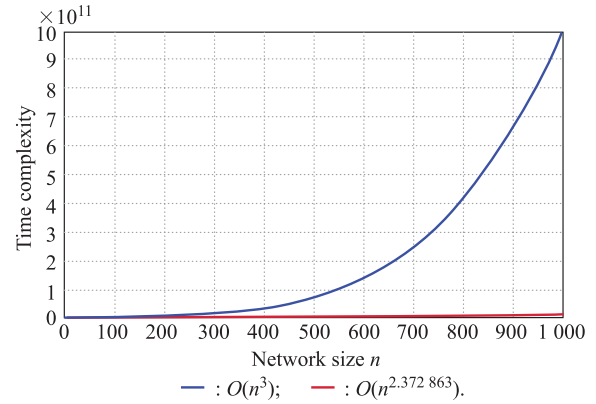


Fig. 3 Comparison on time complexity of the proposed algorithm before and after optimization

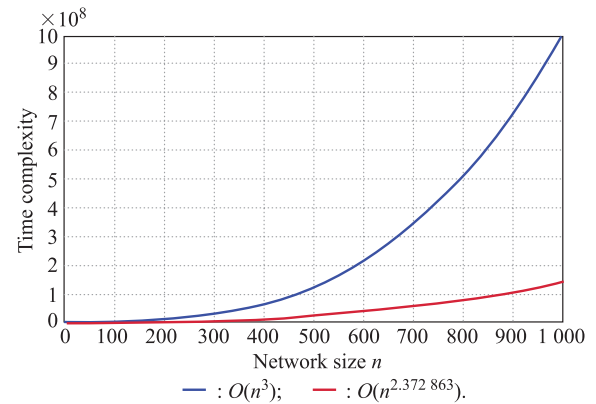


Fig. 4 Comparison on time complexity of Floyd algorithm et al. and the proposed algorithm

4. Cases study

To verify correctness and effect, diameters of the ARPA network and Chinese airline network are calculated and analyzed according to the proposed algorithm.

4.1 ARPA network

The ARPA network shown in Fig. 5 is a trunk network topology, which consists of 21 nodes and 23 links.

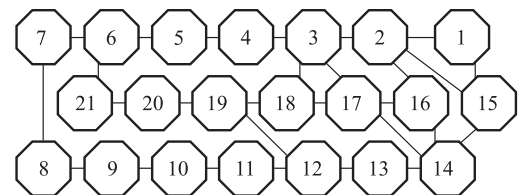


Fig. 5 ARPA network topology model

According to the proposed algorithm, a corresponding connection matrix is established for this network firstly. Then the network diameter is calculated. The distance matrix for this network does not change when it reaches seven order. It shows that the maximum length of the shortest path in this network is 7, which means any two nodes can be linked to each other through at most seven jumps. And the network diameter is 3.467 through calculation, which means any two nodes can be linked to each other through fewer than four jumps on average.

For the ARPA network, time complexity of the algorithm is $O(7 * 21^{2.372\ 863})$. Because $7 > 21^{0.627\ 137} = 6.748\ 6$, the proposed algorithm has no efficiency advantage for calculating the network diameter if compared with the Floyd algorithm et al., whose time complexity is $O(21^3)$. In turn, it can be inferred that the ARPA network is not a typical complex network with an obvious small-world effect. Through the calculation result discussed above, the inference can be confirmed. Because the network size is not very large, but the longest length of the shortest path is not as small as the network diameter, the complex network characteristic of ARPA network is not strong.

4.2 Chinese airline network

Chinese airline network topology shown in Fig. 6 consists of 161 urban airport nodes and 1 185 undirected edges (2 370 airlines). The data samples adopted are collected from airport flights in the summer and autumn of 2010, provided by Civil Aviation Resource Net of China. The samples do not include data about Hongkong, Macao and Taiwan. In addition, irregular flights are not included in the statistics. The reason for using these samples is that it is convenient to analyze and compare the conclusions with the conclusions in [25].



Fig. 6 Chinese airline network topology model

For this network, a corresponding connection matrix can be established, and the algorithm proposed is adopted to

calculate the network diameter. Besides, it can be found that the distance matrix for this network will no longer change when the order of the distance matrix reaches 3, namely matrix multiplications are performed for three times. It shows that the maximum length of the shortest path in the network is 3, and the maximum number of station transfers from one airport to another airport is up to three times. And the network diameter is 2.137 5 by calculation, which shows that any two airports can be linked to each other by about two transfers on average. This result is basically approximate with the network diameter calculated by the Warshall-Floyd algorithm adopted in [25]. Chinese airline network has a very large size. However, the average distance between two nodes is very small. Thus it has a clear small-world effect. For this network, time complexity of the proposed algorithm is $O(3 * 161^{2.372\ 863})$. Compared with the Warshall-Floyd algorithm et al. with time complexity $O(161^3)$, the proposed algorithm has an obvious efficiency advantage.

5. Conclusions

Because network topology is equivalent to its connection matrix, the network diameter calculated according to the algorithm proposed in this paper is accurate but not statistical. In addition, the algorithm is universal rather than for specific types of networks. The values of elements in the distance matrix are revised constantly through matrix multiplications until all of the shortest paths between two nodes are counted. Differences between the proposed algorithm and other algorithms like Floyd lie in that: (i) definition of the distance matrix is different from that proposed in other algorithms; (ii) the values of elements in the distance matrix are not calculated directly by iterated distance matrix multiplications, but indirectly by iterated connection matrix multiplications. These differences are not helpful to improve algorithm complexity. However, adoption of this definition and algorithm can set up relevant propositions and deductions with strict mathematical justification basis. The propositions and deductions are concise, simple and easy for use, which can greatly reduce time complexity of the solution algorithm. From verifications of two examples, it can be seen that the proposed algorithm based on the k order distance matrix is correct for calculating the network diameter, which can greatly improve solution efficiency of large-scale networks diameters, especially of complex networks with small-world effect.

References

- [1] DA F, COSTA L, EVSUKOFF A, et al. Communications in computer and information. Berlin: Springer Verlag, 2011.
- [2] GU L, HUANG H L, ZHANG X D. The clustering coefficient and the diameter of small-world networks. Acta Mathematica

- Sinica, 2013, 29(1): 199–208.
- [3] HUNG J, YEN N, LI K C. Frontier computing. Singapore: Springer Verlag, 2016.
 - [4] SQUILLERO G, SIM K. Applications of evolutionary computation. Zug: Springer Verlag, 2017.
 - [5] GAO J, EFRAT A, FEKETE S, et al. Algorithms for sensor systems. Berlin: Springer Verlag, 2015.
 - [6] YANG J, FANG F, SUN C. Intelligent science and intelligent data engineering. Berlin: Springer Verlag, 2013.
 - [7] SU X, HE T. Chinese lexical semantics. Zug: Springer Verlag, 2014.
 - [8] HERNÁNDEZ C B, GITLER I, KLAPP J. High performance computing. Zug: Springer Verlag, 2017.
 - [9] HOFNER P, MOLLER B. Dijkstra Floyd and Warshall meet Kleene. Formal Aspects of Computing, 2012, 24(4/6): 459–476.
 - [10] CORMEN T H, LEISERSON C E. Introduction to algorithm. 3rd ed. London: The MIT Press, 2009.
 - [11] COLONI A, DORIGO M, MANIEZZO V. Distributed optimization by ant colonies. Proc. of CAL91-European Conference on Artificial Life, 1991: 134–142.
 - [12] NI X J, ZHANG N, WANG M J. An MPI based parallel algorithm to calculate the shortest path of CERNET. Computer Engineering and Applications, 2006, 42(12): 135–137. (in Chinese)
 - [13] ALBERT R, JEONG H, BARABASI A L. Internet: diameter of the world-wide web. Nature, 1999, 401(6749): 130–131.
 - [14] FAN C, LU L Y. The diameter of sparse random graphs. <http://math.ucsd.edu/~fan/dia.pdf>.
 - [15] ALBERT R, JEONG H, BARABASI A L. Error and attack tolerance of complex networks. Nature, 2000, 406(6794): 378–382.
 - [16] LI Y, SHAN X M, REN Y. Average path length of internet with power law degree distribution. Acta Physica Sinica, 2004, 53(11): 3695–3700. (in Chinese)
 - [17] HASSANA S A, HEMEIDAB A M, MAHMOUDA M M M. Performance evaluation of matrix-matrix multiplications using Intel's advanced vector extensions (AVX). Microprocessors and Microsystems, 2016, 47(B): 369–374.
 - [18] STRASSEN V. Gaussian elimination is not optimal. Numerische Mathematik, 1969, 13(4): 354–356.
 - [19] PAN V Y. New combinations of methods for the acceleration of matrix multiplications. Computers & Mathematics with Applications, 1987, 7(1): 73–125.
 - [20] COPPERSMITH D, WINOGRAD S. Matrix multiplication via arithmetic progressions. Journal of Symbolic Computation, 1987, 9(3): 251–280.
 - [21] KHAN A H, AL-MOUHAMED M, FATAYER A, et al. Optimizing the matrix multiplication using Strassen and Winograd algorithms with limited recursions on many-core. International Journal of Parallel Programming, 2016, 44(4): 801–830.
 - [22] STOTHERS A J. On the complexity of matrix multiplication. Edinburgh, UK: University of Edinburgh, 2010.
 - [23] WILLIAMS V V. Multiplying matrices faster than

coppersmith-winograd. Proc. of the 44th Annual ACM Symposium on Theory of Computing, 2012: 887–898.

- [24] GALL L F. Powers of tensors and fast matrix multiplication. arXiv preprint arXiv, 2014, 1401: 7714.
- [25] ZENG X Z. Empirical study of Chinese airline network structure based on complex network theory. Nanjing, China: Nanjing University of Aeronautics and Astronautics, 2011. (in Chinese)

Biographies



CHEN Bin was born in 1979. He is a Ph.D. and an associate professor in People's Liberation Army University of Science and Technology. In 2008, he graduated from this university. After graduation, he has been engaged in related scientific research and teaching work. He participated in the National High Technology Research and Development Program ("863" Program) of China (capabilities requirements analysis technology about large-scale complex system, 2007AA01Z126), the National Natural Science Foundation of China (analysis and verification for command and control system capability requirements model, 61273210) and other army key projects.

He published more than 30 academic papers in this research field on authoritative domestic or foreign journals or international academic conferences. His research interests are complex system and complex network.

E-mail: 156735905@qq.com



ZHU Weixing was born in 1978. He is a doctor and a lecturer in PLA University of Science and Technology. His research interest is algorithm design. In 2012, he graduated from this university. After graduation, he has been engaged in related scientific research. He has published more than ten academic papers in this research field on authoritative domestic or foreign journals or international academic conferences.

E-mail: fly_fox_2008@139.com



LIU Ying was born in 1977. She is a doctor and a lecturer in PLA University of Science and Technology. Her research interest is complex system. In 2008, she graduated from this university. After graduation, she has been engaged in related scientific research and teaching work. She has published more than 20 academic papers in this research field on authoritative domestic or foreign journals or international academic conferences.

E-mail: 13913867236@139.com