

A Survey on Network Coordinate Systems

Ruosi Cheng¹, Yu Wang^{2,a}

¹Luoyang Electronic Equipment Test Center of China, Luoyang 471003, China

²Henan University of Engineering, Xinzheng 451191, China

Abstract. Network coordinate systems play an important role in applications such as P2P file sharing, online network games, and IP positioning. Many network applications require a large amount of network delay data. However, plenty of network delay measurements will cause the network heavily loaded and unreachable problems between network nodes. The proposed network coordinate system provides an important means to solve these problems. Firstly the concept and application of the network coordinate system are introduced. Secondly, the existing representative network coordinate systems are elaborated in detail, such as system based on Euclidean distance and matrix decomposition, and then the advantages of each system are compared. Finally, the factors affecting the accuracy of network coordinate system prediction are analyzed, and the problems are pointed out which the network coordinate system is worth researching in the next step.

1 INTRODUCTION

At present, with the rapid development of Internet technology, the scale of IP networks has grown exponentially, and various large-scale distributed network services and applications have emerged. Therefore, users' requirements for network service quality are also constantly improving. Whether for a network service provider or a network user, it is urgent to grasp the IP network performance and status parameters, as well as the network's operational performance and behavior characteristics in order to optimize network applications and improve service efficiency. Inter-node delay is one of the key parameters that can reflect network performance and state, and many Internet applications rely on accurate delay measurement, such as distributed overlay network multicast [1,2], content addressable overlay network [3], P2P file sharing [4], online games and IP positioning based on network coordinate system [5,6] and so forth. However, due to the large-scale, high-dynamic, self-organizing complexity of the network, it is extremely difficult to accurately and completely measure the inter-node delay. The most straightforward way to obtain latency is to make end-to-end measurements between network nodes. This approach is the most accurate one, but it is only suitable for small-scale systems, typically dozens of nodes. In an N -node network, there are $N(N-1)$ end-to-end links between the two nodes, and the overhead for measurement is $O(N^2)$, so measuring these N nodes takes a long time. Therefore, in the case of a large-scale network, such an end-to-end measurement brings a large amount of measurement overhead to the network, and there is also a problem that some inter-node delay cannot be measured in the network. These problems reduce the

usability of the end-to-end direct measurement of the delay.

Using the Network Coordinate System to predict the delay (also known as the network distance) is one of the efficient approaches to obtain the inter-node delay [7]. Using the coordinates in the space to represent the network nodes, the approach embeds the network space into a certain measurement space, and converts the measurement of the network delay into the calculation of the distance between the nodes in the space, which can greatly reduce the measurement overhead caused by the actual network measurement.

Most network coordinate systems use a calculation model based on Euclidean distance. In this calculation model, the network coordinate system maps N nodes into a D -dimensional space by $O(N)$ measurements, and defines the network coordinates of any node i as $x_i = r_1^i + r_2^i + \dots + r_d^i$, then the predicted network distance of any two nodes i and j is:

$$D^E(i, j) = \|x_i - x_j\| \quad (1)$$

Research on network coordinate system starts from 2001, and many typical systems have emerged, such as GNP [8], VL [9], ICS [10] and other central network coordinate systems. The common characteristic of these coordinate systems is about that: A small number of fixed nodes are first arranged as reference nodes, and the delays are measured between each other. The coordinates of these reference nodes are obtained by a global optimization algorithm. Then each common node calculates the coordinates of the reference node using the coordinates of the reference node and the measurement of the distance from itself to the reference node. Since 2004, distributed coordinate systems represented by

* Corresponding author: ^aYu Wang: stonchor@163.com

Vivaldi [11], PIC [12], NPS [13], WDCS [14] and other systems have been proposed. These coordinate systems are characterized by a fully distributed implementation and do not require the placement of reference nodes; thus, the scale of the coordinate system can be expanded. Beginning in 2006, a network coordinate system based on matrix decomposition is proposed. The characteristic of this system is to predict the distance between nodes by decomposing the delay matrix. This approach avoids the Triangle Inequality Violation (TIV) problem [15, 16] based on the Euclidean distance coordinate system.

2 Existing network coordinate system

There are a variety of network coordinate systems nowadays, which can be roughly divided into two categories, namely: a network coordinate system based on Euclidean distance and a network coordinate system based on matrix decomposition. The former represented by GNP and Vivaldi maps network nodes to a geometric space. Each node has corresponding coordinates in geometric space. The network distance between nodes can be calculated by corresponding node coordinates. And the latter represented by IDES [17] obtains the incoming vector and the outgoing vector of the node by decomposing the delay matrix, and then obtains network distance (delay) between nodes according to a certain calculation rule with the incoming vector and the outgoing vector.

2.1 Network coordinate system based on Euclidean distance

2.1.1 GNP

GNP is the first network coordinate system based on Euclidean distance proposed by Eugene Ng [8] of Carnegie Mellon University in 2002. In GNP, the nodes in the network are mapped to coordinate points in the Euclidean space, and each node has a coordinate in the space, thereby transforming the network distance problem into a distance problem between space nodes. And the distance between any two nodes can be calculated from coordinate values with the Euclidean space distance formula.

GNP divides the nodes in the system into two parts, *landmarks* (reference nodes) and *ordinary nodes*. Among them, the landmarks are some representative nodes selected in the network according to a certain criteria. They measure the network distance (delay) between each other, and use the Downhill Simplex method [18] to solve the coordinates of the landmarks, its objective function is:

$$E = \min \sum_{i,j} [D(L_i, L_j) - D^E(L_i, L_j)]^2 \quad (2)$$

With this objective function, the landmark coordinates can be obtained, and all the landmark coordinates play an auxiliary role in the calculation of other ordinary node coordinates.

2.1.2 Vivaldi

Vivaldi regards the entire Internet as a spring system, simulating the space embedding problem as a spring elastic potential minimization problem. The connection between the network nodes is equivalent to a spring. Each node has force with each other by springs. The magnitude of the force is proportional to the difference between the current predicted distance and the actual distance. When the coordinates of each node in the network tend to be the optimal value, the elastic potential of the whole system is the smallest and the system tends to be stable. Suppose x_i represents the network coordinates of node i , and the network coordinates are the values of the minimization error calculation formula E . By this minimization process, the elastic potential energy of the entire spring system is minimized.

$$E = \sum_i \sum_j (D(i, j) - \|x_i - x_j\|)^2 \quad (3)$$

And $\|x_i - x_j\|$ is the distance between node i and j in the Euclidean space.

In the Vivaldi system, each node has its own network coordinates x_i and local error e_i . All nodes periodically update their network coordinates and local errors according to their own measurements of other nodes in the network, combined with the network coordinates of the nodes being measured. c_e and c_c are tunable parameters in the system.

Algorithm 1 *vivaldi*(rtt, x_i, e_i)

- 1: $w = e_i / (e_i + e_j)$
 - 2: $e_s = \|x_i - x_j\| - rtt / rtt$
 - 3: $e_i = e_s \times c_e \times w + e_i \times (1 - c_e \times w)$
 - 4: $\delta = c_c \times w$
 - 5: $x_i = x_i + \delta \times (rtt - \|x_i - x_j\|) \times u(x_i - x_j)$
-

Algorithm 1 describes the process of one coordinate update of a Vivaldi node. Assuming that the node is i , the node referenced by node i in this coordinate update is node j (usually randomly selected from the list of neighbor nodes in each update cycle). Firstly, a weight value w for the current update process can be calculated based on the local error e_i of the node i and the local error e_j of the node j . Secondly, by measuring the distance between the node i and j , a temporary relative error e_s can be calculated; Then with w and e_s , a new local error e_i can be calculated. Finally, it can recalculate the own network coordinates of node i .

Compared with GNP, Vivaldi's biggest advantage lies in its distributed structure, which does not require a fixed anchor node, so the system's scalability is enhanced. However, Vivaldi is a network coordinate system essentially based on Euclidean distance. The asymmetry of the network distance and the characteristics of violating the triangle inequality constraint make its prediction accuracy affected. At the same time, the Vivaldi distributed computing feature also makes the algorithm's convergence time long.

2.1.3 Pharos

Pharos [19] is a hierarchical network coordinate system extended on the basis of Vivaldi. It improves the accuracy of network distance prediction by giving multiple different network coordinates to a same node. One group is the network coordinates of the global scope (the entire Internet), while the other coordinates represent a relatively small scope. In Pharos, both the global and local networks run a separate Vivaldi system. A node can calculate its global or local coordinates by joining the global network or one of the local networks, so as to accurately predict the links of different length scopes.

A significant advantage of Pharos system is its compatibility with the Vivaldi system. For a network node, it first deploys its Vivaldi system, then join the global network and local network defined by the Pharos system, and finally calculate its coordinates.

2.2 Network coordinate system based on matrix decomposition

2.2.1 IDES

IDES is the coordinate system which first proposed the matrix decomposition calculation model. Through a small amount of measurement, the distance matrix D of $N \times N$ is decomposed into two $N \times d$ matrixes X and Y , so that $D \approx X \times Y^T$.

$$D_{N \times N} \approx \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{pmatrix}_{N \times d} \times \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{pmatrix}_{N \times d}^T \quad (4)$$

$$= \begin{pmatrix} \alpha_1^1 & \alpha_1^2 & \dots & \alpha_1^d \\ \alpha_2^1 & \alpha_2^2 & \dots & \alpha_2^d \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_N^1 & \alpha_N^2 & \dots & \alpha_N^d \end{pmatrix} \times \begin{pmatrix} \beta_1^1 & \beta_1^2 & \dots & \beta_1^d \\ \beta_2^1 & \beta_2^2 & \dots & \beta_2^d \\ \vdots & \vdots & \ddots & \vdots \\ \beta_N^1 & \beta_N^2 & \dots & \beta_N^d \end{pmatrix}^T$$

The coordinates of node i include two d -dimensional vectors, which are:

The d -dimensional outgoing vector:

$$X_i = (\alpha_1^i, \alpha_2^i, \dots, \alpha_d^i), \alpha_k^i \in \mathbb{R}, 1 \leq k \leq d \quad (5)$$

The d -dimensional incoming vector:

$$Y_i = (\beta_1^i, \beta_2^i, \dots, \beta_d^i), \beta_k^i \in \mathbb{R}, 1 \leq k \leq d \quad (6)$$

The predicted distance from node i to node j can be calculated by the dot product of the outgoing vector of the node i and the incoming vector of the node j :

$$D_{i,j}^e = X_i \times Y_j = \sum_{k=1}^d X_{ik} Y_{jk} \quad (7)$$

In the network coordinate system based on Euclidean distance, the principle of triangle inequality should be satisfied for any three nodes i, j, k . In the computational model based on matrix decomposition, there is no such limitation, so the computational model based on matrix decomposition can better model the Internet where TIV

is ubiquitous. However, IDES also has two main problems:

First, IDES cannot ensure that the predicted network distance is non-negative. The delay in the network will not be negative, and a negative value will seriously affect other participating nodes of the entire network. Secondly its scalability is not strong because the initial nodes used to construct the delay matrix are fixed.

2.2.2 Phoenix

Phoenix [20] is a network coordinate system built on the basis of IDES. Phoenix specifies that all elements of each node's outgoing and incoming vector must be non-negative, which guarantees that the distance predicted by the Phoenix system cannot be negative. At the same time, the matrix decomposition based Phoenix system does not provide a fixed reference node for the coordinate calculation of the entire network, and introduces the concept of vector weights. Any node that has calculated its own network coordinates can be the reference node for the newly added node coordinate calculation. Suppose there are N such nodes in the system. When a new node H joins the system, it will randomly select any m nodes as its reference nodes and start the periodic coordinate update process. In each round of updates, H will take the coordinates of m points and measure its distances to the m nodes, then update its own coordinates.

For the newly added node H , its m reference nodes are respectively defined as R_1, R_2, \dots, R_m . The outgoing vectors of these m nodes are defined as X_1, X_2, \dots, X_m , and the incoming vectors are defined as Y_1, Y_2, \dots, Y_m . Each ordinary node evaluates the accuracy of its individual reference vectors. The more accurate the reference vector is, the higher the weight value is obtained. Conversely, the reference vector with a large error is given a small weight value. The introduction of vector weights can effectively avoid the propagation of errors [21]. For each reference node R_i of a new node H , the weight of X_i is defined as w_{X_i} , while the weight of Y_i is w_{Y_i} . After introducing vector weights, X_{new} and Y_{new} can be calculated using the following formula:

$$X_{new} = \arg \min_{U \in \mathbb{R}^{+d}} \sum_{i=1}^m w_{Y_i} [D(H, R_i) - U \cdot Y_i^T]^2 \quad (8)$$

$$Y_{new} = \arg \min_{U \in \mathbb{R}^{+d}} \sum_{i=1}^m w_{X_i} [D(R_i, H) - X_i \cdot U^T]^2 \quad (9)$$

According to the calculated X_{new} and Y_{new} , the predicted distance between the node H and each reference node R_i satisfies the following formula:

$$D^E(H, R_i) = X_{new} \cdot Y_i^T \quad (10)$$

$$D^E(R_i, H) = X_i \cdot Y_{new}^T \quad (11)$$

If H is a newly joined node, for each of its reference nodes R_i , set $w_{X_i} = w_{Y_i} = 1$ and directly calculate its initial coordinates. The calculation process of its weight is as follows:

$$w_{X_i} = \begin{cases} 1, & \text{if } E_{X_i, Y_{new}} \leq M_X \\ (M_X / E_{X_i, Y_{new}})^2, & \text{if } M_X < E_{X_i, Y_{new}} < C \cdot M_X \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

$$w_{Y_i} = \begin{cases} 1, & \text{if } E_{X_{new}, Y_i} \leq M_Y \\ (M_Y / E_{X_{new}, Y_i})^2, & \text{if } M_Y < E_{X_{new}, Y_i} < C \cdot M_Y \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

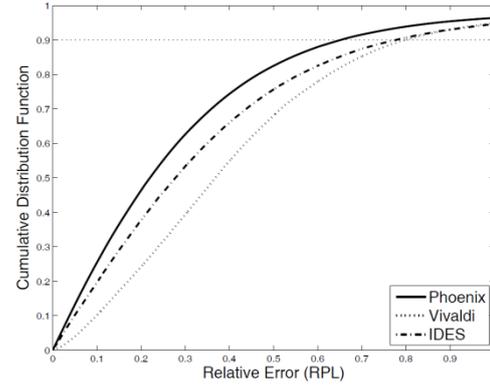
While considering reference host $R_i (1 \leq i \leq m)$, we look at the absolute prediction error $E_{X_i, Y_{new}}$ from R_i to H . Likewise, we also obtain the absolute prediction error E_{X_{new}, Y_i} from H to R_i .

$$E_{X_i, Y_{new}} = \|X_i \cdot Y_{new}^T - D(R_i, H)\| \quad (14)$$

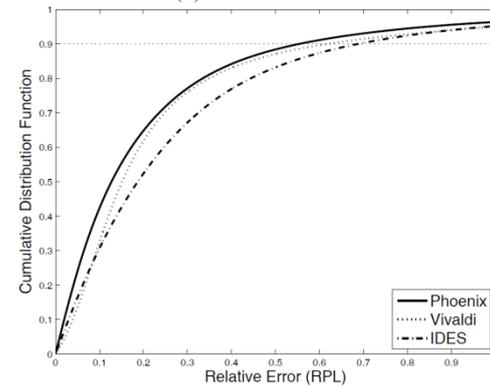
$$E_{X_{new}, Y_i} = \|X_{new} \cdot Y_i^T - D(H, R_i)\| \quad (15)$$

M_X and M_Y are the median of $E_{X_i, Y_{new}}$ and E_{X_{new}, Y_i} , and C is a constant coefficient. For a reference vector whose absolute error is less than or equal to the corresponding median, its weight value is set to 1. For a reference vector whose absolute error is greater than the median but less than the median C times, it is assumed to be k times the corresponding median value and its weight value is $(1/k)^2$. The reference vector whose absolute error is greater than the corresponding median C times is set to a weight value of 0.

By introducing reference vector weights, the Phoenix system can effectively avoid error propagation in coordinate calculations. Moreover, the occurrence of a negative prediction distance is avoided. From the perspective of prediction error, Phoenix has significantly improved compared to Vivaldi and IDES, as shown in Figure 1.



(a) PL data set



(b) King data set

Figure 1. CDF of relative error (RPL) [20]

3 Comprehensive analysis of network coordinate system

The prediction performance of network coordinate system is affected by many factors, and the affected factors vary from different coordinate systems.

3.1 TIV of network distances

In the network coordinate system based on Euclidean distance, the predicted network distance must satisfy the triangle inequality rule. And TIV is any three nodes A, B, C in the network, assuming $D(A, B) \geq D(B, C)$ and $D(A, B) \geq D(A, C)$, if $D(A, B) > D(A, C) + D(B, C)$, then the triangle inequality violation (TIV) occurs for nodes A, B , and C . Any three nodes that have violated the triangle inequality rule cannot be represented by the Euclidean distance-based network coordinate system without error, because distances must satisfy the triangle inequality rule in the Euclidean space. However in the actual network, the triangle inequality rule is not always true. Therefore, TIV will greatly affect the accuracy of the network coordinate system based on Euclidean distance. In the network coordinate system based on matrix decomposition, it will not be affected by TIV because it uses the matrix decomposition calculation model.

3.2 Reference nodes in network coordinate system

The selection of reference nodes in the network coordinate system also affects the prediction accuracy. According to whether the reference nodes are fixed, the network coordinate systems can be divided into the central network coordinate systems (like GNP, IDES) and the non-central network coordinate systems (Vivaldi, Phoenix, Pharos). The central network coordinate system has high requirements on the reference nodes, which usually causes high load problems in the reference nodes. Compared with the central network coordinate system, the non-central type do not need a fixed reference node, so its system scalability is enhanced, the load capacity is high, and the problem that prediction tends to be local minimization caused by the fixed reference node can be effectively solved. So the selection of reference nodes also has a great impact on the accuracy of prediction [22].

3.3 Network coordinate system security

Most network coordinate systems assume that participating nodes can be trusted. If malicious nodes in the network provide false distance information [23], the accuracy of distance prediction will be seriously reduced or even the failure of the entire prediction system. Such threats are also encountered in systems based on fixed reference nodes.

4 Further research on network coordinate system

Currently, all network coordinate systems are focused on predicting network latency, but there are other features in the network, such as bandwidth, hop count, and so on. These features could be added to network coordinate systems [7] and provide more support for the prediction accuracy of network coordinate systems by finding the correlation between these features and the network delay. On the other hand, in the network coordinate system based on matrix decomposition, SVD [24] and NMF [25] decomposition algorithms are mostly used, and the quality of the decomposition approach will also affect the distance prediction. If a new decomposition approach can be proposed, it will help predict the network delay. In terms of application, further research is needed on whether the network coordinate system can be better applied to IP positioning.

5 Conclusion

A new approach, network coordinate system, is discussed for predicting network delay. The proposed network coordinate system provides a new mode for the measurement of network delay, that is, an approach for obtaining the network distance between nodes without requiring plenty of network measurements. The two types of network coordinate systems based on Euclidean distance and matrix decomposition are introduced. The

problems of each system are discussed, and the factors affecting the prediction of network coordinate system are analyzed. Aiming at the problems existing in the network coordinate system in delay prediction, some improved ideas are proposed.

Sponsoerd by :

1. National Natural Science Foundation of China, No. 61802115;
2. Key scientific research projects of Henan Province Education Department, No.18A520004;
3. Henan province science and technology projects, No. 182102310925

References

1. Zhang, R., Tang, C., Hu, Y. C., Fahmy, S., & Lin, X. (2006, April). Impact of the inaccuracy of distance prediction algorithms on internet applications-an analytical and comparative study. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings* (pp. 1-12). IEEE.
2. Lua, E. K., Zhou, X., Crowcroft, J., & Van Mieghem, P. (2008). Scalable multicasting with network-aware geometric overlay. *Computer Communications*, **31**(3), 464-488.
3. Zhao, B. Y., Huang, L., Stribling, J., Rhea, S. C., Joseph, A. D., & Kubiawicz, J. D. (2004). Tapestry: A resilient global-scale overlay for service deployment. *IEEE Journal on selected areas in communications*, **22**(1), 41-53.
4. Lian, Q., Zhang, Z., Yang, M., Zhao, B. Y., Dai, Y., & Li, X. (2007, June). An empirical study of collusion behavior in the Maze P2P file-sharing system. In *Distributed Computing Systems, 2007. ICDCS'07. 27th International Conference on* (pp. 56-56). IEEE.
5. KOMOSNY D, SIMEK M, KATHIRAVELU G. (2013) . Can Vivaldi Help in IP Geolocation?. *Przegląd Elektrotechniczny*, 2013, **89**(5): 100-106.
6. Moravek, P., Komosny, D., Burget, R., Sveda, J., Handl, T., & Jarosova, L. (2011). Study and performance of localization methods in IP based networks: Vivaldi algorithm. *Journal of Network and Computer Applications*, **34**(1), 351-367.
7. Donnet, B., Gueye, B., & Kaafar, M. A. (2010). A survey on network coordinates systems, design, and security. *IEEE Communications Surveys & Tutorials*, **12**(4), 488-503.
8. Ng, T. E., & Zhang, H. (2002). Predicting Internet network distance with coordinates-based approaches. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE* (Vol. 1, pp. 170-179). IEEE.
9. Tang, L., & Crovella, M. (2003, October). Virtual landmarks for the internet. In *Proceedings of the 3rd*

- ACM SIGCOMM conference on Internet measurement* (pp. 143-152). ACM.
10. Lim, H., Hou, J. C., & Choi, C. H. (2005). Constructing Internet coordinate system based on delay measurement. *IEEE/ACM Transactions on Networking (TON)*, **13**(3), 513-525.
 11. Dabek, F., Cox, R., Kaashoek, F., & Morris, R. (2004, August). Vivaldi: A decentralized network coordinate system. In *ACM SIGCOMM Computer Communication Review* (Vol. **34**, No. 4, pp. 15-26). ACM.
 12. Costa, M., Castro, M., Rowstron, R., & Key, P. (2004). PIC: Practical Internet coordinates for distance estimation. In *Distributed Computing Systems, 2004. Proceedings. 24th International Conference on* (pp. 178-187). IEEE.
 13. Ng, T. E., & Zhang, H. (2004, June). A Network Positioning System for the Internet. In *USENIX Annual Technical Conference, General Track* (pp. 141-154).
 14. Liu, Y., Du, H., & Ye, Q. (2015). WDCCS: A Weight-Based Distributed Coordinate System. In *Combinatorial optimization and application* (pp. 251-260). Springer, Cham.
 15. Zheng, H., Lua, E. K., Pias, M., & Griffin, T. G. (2005, March). Internet routing policies and round-trip-times. In *International Workshop on Passive and Active Network Measurement* (pp. 236-250). Springer, Berlin, Heidelberg.
 16. Cheng, J., Liu, Y., Ye, Q., Du, H., & Vasilakos, A. V. (2017). DISCS: A distributed coordinate system based on robust nonnegative matrix completion. *IEEE/ACM Transactions on Networking (TON)*, **25**(2), 934-947. [J]. *IEEE/ACM Transactions on Networking (TON)*, 2017, **25**(2): 934-947.
 17. Mao, Y., Saul, L. K., & Smith, J. M. (2006). Ides: An internet distance estimation service for large networks. *IEEE Journal on Selected Areas in Communications*, **24**(12), 2273-2284.
 18. Nelder, J. A., & Mead, R. (1965). A simplex method for function minimization. *The computer journal*, **7**(4), 308-313
 19. Chen, Y., Xiong, Y., Shi, X., Deng, B., & Li, X. (2007, November). Pharos: A decentralized and hierarchical network coordinate system for internet distance prediction. In *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE* (pp. 421-426). IEEE.
 20. Chen, Y., Wang, X., Shi, C., Lua, E. K., Fu, X., Deng, B., & Li, X. (2011). Phoenix: A weight-based network coordinate system using matrix factorization. *IEEE Transactions on Network and Service Management*, **8**(4), 334-347.
 21. Chen, Y., Wang, X., Song, X., Lua, E. K., Shi, C., Zhao, X & Li, X. (2009, May). Phoenix: Towards an accurate, practical and decentralized network coordinate system. In *International Conference on Research in Networking* (pp. 313-325). Springer, Berlin, Heidelberg.
 22. Zhao F, Chen J, Ye D & Luo X. (2017, June). A Network Coordinate System Constructing Algorithm Based on Optimal Neighbor Nodes. In *Data Science in Cyberspace (DSC), 2017 IEEE Second International Conference on* (pp. 140-144). IEEE.
 23. Kaafar, M. A., Mathy, L., Turletti, T., & Dabbous, W. (2006, December). Virtual networks under attack: Disrupting internet coordinate systems. In *Proceedings of the 2006 ACM CoNEXT conference* (p. 12). ACM.
 24. Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, **401**(6755), 788.
 25. Boutsidis, C., & Gallopoulos, E. (2008). SVD based initialization: A head start for nonnegative matrix factorization. *Pattern Recognition*, **41**(4), 1350-1362.