

Modeling of Anti-tracking Network based on Convex-polytope Topology^{*}

Changbo Tian^{1,2}, Yongzheng Zhang^{1,2}(✉), and Tao Yin^{1,2}

¹ Institute of Information Engineering, Chinese Academy of Sciences,
Beijing 100093, China

² School of Cyber Security, University of Chinese Academy of Sciences,
Beijing 100049, China
{tianchangbo, zhangyongzheng, yintao}@iie.ac.cn

Abstract. Anti-tracking network plays an important role in protection of network users' identities and communication privacy. Confronted with the frequent network attacks or infiltration to anti-tracking network, a robust and destroy-resistant network topology is an important prerequisite to maintain the stability and security of anti-tracking network. From the aspects of network stability, network resilience and destroy-resistance, we propose the convex-polytope topology(CPT) applied in the anti-tracking network. CPT has three main advantages: (1) CPT can easily avoid the threat of key nodes and cut vertices to network structure; (2) Even the nodes could randomly join in or quit the network, CPT can easily keep the network topology in stable structure without the global view of network; (3) CPT can easily achieve the self-optimization of network topology. Anti-tracking network based on CPT can achieve the self-maintenance and self-optimization of its network topology. We compare CPT with other methods of topology optimization. From the experimental results, CPT has better robustness, resilience and destroy-resistance confronted with dynamically changed topology, and performs better in the efficiency of network self-optimization.

Keywords: Anti-tracking network · Topology management · Network topology · Self-organization · Cyber security.

1 Introduction

The rapid growth of Internet access has made communication privacy an increasingly important security requirement. And the low threshold and convenience of the techniques about network monitoring and tracing also have posed a great threat on the online privacy network users [1–3]. Anti-tracking network is proposed as the countermeasure to fight against the network monitoring and censorship [4–6]. So, the attacks on anti-tracking network increase rapidly in recent years, such as network paralysis, network infiltration, communication tracking and so on. Especially, for the P2P-based anti-tracking network which takes

^{*} Supported by the national natural science foundation of China under grant No.U1736218 and No.61572496.

the advantage of the wide distribution of nodes for tracking-resistant communication, it is important to keep a stable, secure and destroy-resistant topology structure. Also, considered that the nodes in P2P-based anti-tracking network can join in or quit the network freely and randomly, it increases the uncertainty and insecurity for anti-tracking network to maintain a robust network topology.

Much research has been done regarding the management or optimization of network topology. But as for the aspect of anti-tracking network, current research have some limitations which are concluded as follows:

- (1) **The cost of network maintenance.** Some of current research needs the global view of network to optimize the network topology [7–9]. Confronted with dynamically changed network topology, the cost of network maintenance would be very high.
- (2) **Weak destroy-resistance.** P2P-based anti-tracking network confronts not only the problem of communication tracing, but also the threats of network attacks [10–12]. Especially, the attack to cut vertices and key nodes will destroy the network structure tremendously.
- (3) **Weak tracking-resistance.** Network infiltration is another threat on the security of anti-tracking network [13–15]. Malicious nodes can be deployed in the network to measure the network topology and traceback the communication.

To address the above problems, we propose the convex-polytope topology (CPT) which can be applied in anti-tracking network. The anti-tracking network based on CPT achieves the self-optimization of its network. The novelty and advantages of our proposal are summarized as follows:

- (1) We apply the convex-polytope topology in the anti-tracking network. The anti-tracking network based on CPT has better performance in network stability and resilience.
- (2) We propose a topology maintenance mechanism based on CPT. Each node in network only needs to maintain its local topology to keep the whole network in stable convex-polytope structure.
- (3) We propose a network self-optimization mechanism based on CPT. With the self-optimization mechanism, anti-tracking network can optimize its network topology as the network topology changes.

2 Convex-polytope Topology

Convex-polytope topology (CPT) is a structured topology in which all nodes are constructed into the shape of convex-polytope, as illustrated in Fig. 1. CPT has the following advantages to maintain the robust and destroy-resistant topology of anti-tracking network.

- **Stability:** The structure of CPT has the advantage in avoiding the cut vertices, except the extreme cases in which the connectivity of CPT is too sparse, such as ring structure. So, the maintenance of network topology is just to keep the convex-polytope structure.

- Resilience: Confronted with the dynamically changed topology, each node in CPT only needs to maintain its local topology to accord with the properties of convex-polytope, then the whole network will be in the convex-polytope structure.
- Self-optimization: The network stability, robustness and destroy-resistance benefit from the balanced distribution of nodes' connectivity. CPT can achieve the self-optimization of the whole network through the optimization of each node's local topology.

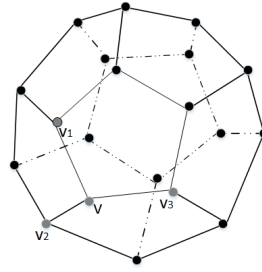


Fig. 1. An example of convex-polytope topology.

2.1 The CPT Model

Consider the topology structure of CPT illustrated in Fig. 1, the whole network is constructed in a convex-polytope structure logically. To maintain this structure, each node has to record both of the neighbouring nodes and the corresponding surfaces in the convex-polytope.

Let v denotes a node, CN denotes its neighbouring node collection, and CS denotes its surface collection. We take the example of node v shown in Fig. 1, then $CN = \{v_1, v_2, v_3\}$ and $CS = \{(v_1, v_2), (v_1, v_3), (v_2, v_3)\}$. So, if one neighbouring node of v disconnects, then node v knows which surfaces are affected by the lost neighbouring node, and adjusts its local topology accordingly.

As we can see from the Fig. 1, each surface of convex-polytope can be different polygons. For a fixed number of nodes, if each surface has more edges, the connectivity of the whole network becomes more sparse. On the contrary, each surface has less edges, the connectivity of the whole network becomes more dense. If all surfaces have three edges, then the connectivity of the network reaches the maximum. So, in our work, we construct CPT of which each surface is triangle to provide better stability and resilience of anti-tracking network.

Assume there are n nodes to construct CPT with triangle surfaces, then the number of surfaces and edges in this kind of CPT is fixed. Theorem 1 gives the details about the calculation of the number of this CPT's surfaces and edges.

Theorem 1. *If a convex-polytope has n nodes, and each surface is triangle, then the number of surfaces and edges of this convex-polytope is fixed. The number of edges is: $l = 3 \times (n - 2)$, the number of surfaces is: $m = 2 \times (n - 2)$.*

Proof. Because each surface of convex-polytope with n nodes is triangle, and each edge is shared by two surfaces, assume the number of edges is l , and the number of surfaces is m , then $3 \times m = 2 \times l$. According to Euler Theorem of Polyhedron, we have the equation $n + m - l = 2$.

$$\begin{cases} 3 \times m = 2 \times l \\ n + m - l = 2 \end{cases} \Rightarrow \begin{cases} l = 3 \times (n - 2) \\ m = 2 \times (n - 2) \end{cases} \quad (1)$$

2.2 The Construction of CPT

Even though CPT has the advantages in the maintenance of network topology, in some extreme cases shown in Fig. 2, CPT still has the potential threat in topology structure, such as key nodes. Key nodes would have big influence on the stability, robustness and security of network topology.

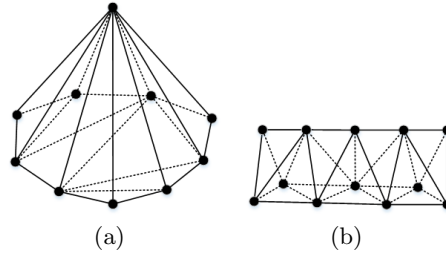


Fig. 2. Two examples of extreme convex-polytope topology.

In the construction process of CPT, the balanced distribution of each node's connectivity is the priority to construct a stable and robust anti-tracking network. We propose the construction algorithm to construct a CPT with balanced distribution of nodes' connectivity. The construction algorithm takes two main steps shown as follows:

- (1) **Constructing a circuit:** All nodes construct a circuit.
- (2) **Adding edges iteratively:** Each node takes turns at selecting a node which has the span of 2 and allow new connections along with one direction of the circuit, and adding an edge with it to generate a triangle surface. If all surfaces of one node are triangles, it will refuse new connection with it in order to keep the whole topology in convex-polytope structure. After all nodes have triangle surfaces, the CPT with triangle surfaces has been constructed.

Algorithm 1 Construction Algorithm

Input: $C = [v_1, v_2, \dots, v_n]$: node collection.

```

1: BuildNodeCircuit( $C$ ). ▷ All nodes construct a circuit.
2:  $N \leftarrow n$ .
3: while True do
4:    $v \leftarrow \text{GetOneNode}(C)$ .
5:    $u \leftarrow \text{GetLinkNode}(v)$  ▷ Get a node  $u$  which has the span of 2 with  $v$ .
6:   Connect( $v, u$ ).
7:   if The surfaces of  $v$  are all triangles then
8:      $N \leftarrow N - 1$ .
9:   end if
10:  if  $N \leq 0$  then
11:    Break.
12:  end if
13: end while

```

Algorithm 1 gives the pseudocode for the construction algorithm of CPT. The time complexity of the construction algorithm mainly depends on the executions of the process of building node circuit (line 1) and while loop (line 3 ~ 6). The time complexity of building node circuit is $O(n)$. Because the edges of CPT is fixed which has been proved in Theorem 1, and the node circuit has formed n edges. Then, the while loop in Algorithm 1 only needs to be executed $(2 \times n - 6)$ times. So, the time complexity of Algorithm 1 is $O(n)$.

To make straightforward sense of CPT construction algorithm, Fig. 3 illustrates the construction process of CPT with 9 nodes, labeled v_1, v_2, \dots, v_9 . Firstly, these 9 nodes construct a circuit. We take v_1 as the first node to add an edge with v_3 , then v_1 has a triangle surface (v_1, v_2, v_3) . Iteratively, v_2 adds an edge with v_4 , and v_3 adds an edge with v_5 , until the surfaces of all nodes are triangles. In Fig. 3(f), v_2 needs to select a node with the span of 2 to add an edge, there are three nodes, v_5, v_6, v_8 . Because all the surfaces of v_5 and v_8 are triangles, the extra edges may break the structure of convex-polytope. So, v_2 can only add an edge with v_6 .

2.3 The Maintenance of CPT

As the nodes can join in or quit the anti-tracking network freely, the frequent change of network topology inevitably has a big influence on the stability and security of network structure. So, an effective maintenance mechanism of network topology guarantees the robustness and usability of anti-tracking network. In this section, we will discuss the maintenance mechanism in which the relevant nodes only update its local topology to keep the anti-tracking network in convex-polytope structure when confronted with the dynamically changed topology.

In the maintenance mechanism of CPT, two cases are taken into consideration: that of the nodes join in the network and that of the nodes quit the network.

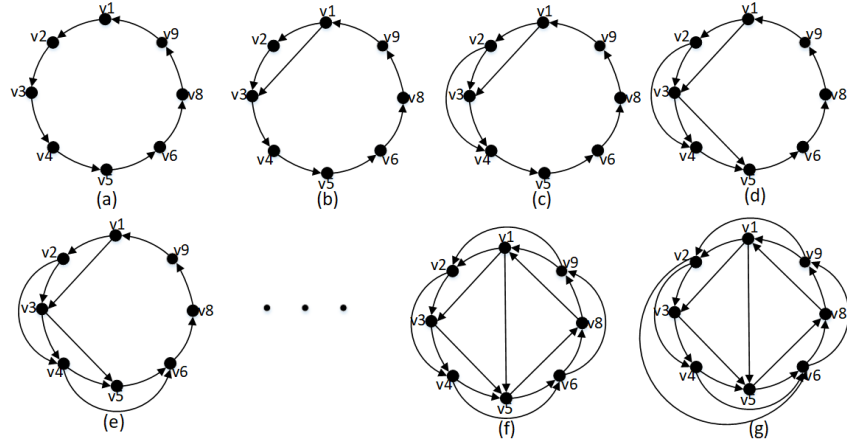
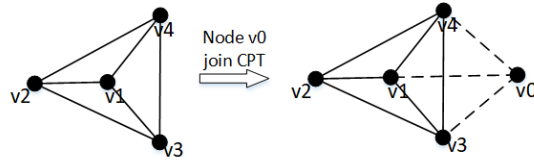


Fig. 3. Construction process of CPT

Nodes join in CPT. All surfaces of CPT are triangles. If new nodes want to join in CPT, they only need to select one surface of CPT and connect each node of this surface.

Fig. 4. Node v_0 join the network with CPT

As illustrated in Fig. 4, node v_1 has three surfaces, (v_2, v_3) , (v_2, v_4) and (v_3, v_4) . A new node v_0 connects v_1 and selects the surface (v_3, v_4) to join in the network. Because of the join of v_0 , v_1 breaks the surface (v_3, v_4) to generates two new surfaces (v_0, v_3) and (v_0, v_4) . The new node v_0 generates three surfaces, (v_1, v_3) , (v_1, v_4) and (v_3, v_4) . So, the node join of CPT only changes the topology of one surface. The whole topology can be easily maintained in the convex-polytope structure.

Nodes quit CPT. If a node quits CPT, all the surfaces of this node will merge into one surface. If the quitting node has a high degree, CPT will generate a large surface which is detrimental to the stable and robust network topology.

For example, Fig. 5(a) shows a node with degree of 3 quits CPT. When v_1 quits CPT, its three surfaces (v_2, v_3) , (v_2, v_4) and (v_3, v_4) merge into one surface which is still a triangle. In this case, CPT need no additional actions to maintain the topology.

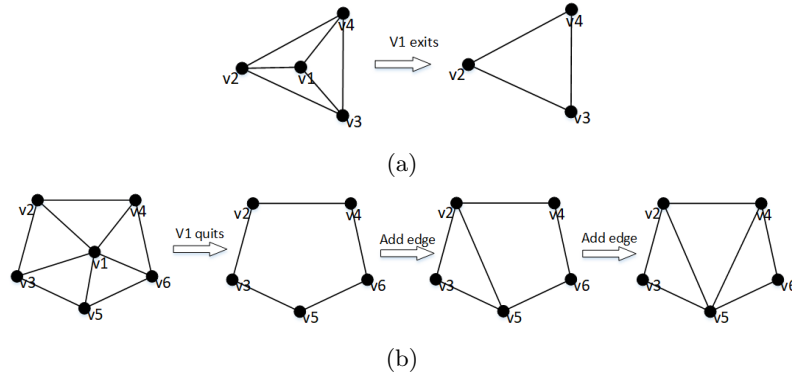


Fig. 5. CPT maintains its structure when some nodes exit

But in Fig. 5(b), node v_1 with degree of 5 quits CPT, the surface generated by quitting v_1 is pentagon. In this case, the relevant nodes have to update its local topology to keep convex-polytope structure with all triangle surfaces. The maintenance process only exists in the nodes of the generated pentagon surface. Through the connection between these nodes, the pentagon surface can be divided into different triangle surfaces.

The maintenance of the generated pentagon surface is similar with the construction algorithm of CPT. If all surfaces of a node are triangles, it will refuse any connections. Each node in the pentagon surface connects with a node which has the span of 2 and allow new connection iteratively until all nodes in this surface has triangle surfaces. For example shown in Fig. 5(b), after v_1 quits CPT, v_2 firstly establishes a connection with v_5 to generate a triangle surface (v_3 , v_5). Then v_3 has all triangle surfaces and refuse any connection with it. Next, v_5 establishes a connection with v_4 to generate a triangle. Then v_6 has all triangle surfaces and refuse any connections. At last, the maintenance of CPT is finished.

3 Self-optimization Mechanism based on CPT

As we have discussed above, CPT can keep the network topology in the stable and resilient convex-polytope structure. But the convex-polytope structure still has some extreme cases shown in Fig. 2 which may pose a potential threat on the structure stability and communication efficiency of anti-tracking network because of the key nodes. According to the properties of CPT, we can achieve the self-optimization of anti-tracking network to balance the distribution of nodes' connectivity.

As we have mentioned in Theorem 1 that the number of edges in convex-polytope of which all surfaces are triangles is: $l = 3 \times (n - 2)$, and n denotes the number of the nodes. Then, we can calculate the average degree of CPT as shown in Equ. 2. When n is very large, the average degree of CPT \bar{d} is close to 6. So, we usually set the upper limit of nodes' degree as 10 in CPT. The nodes of

which the degree is bigger than 10 should adjust its local topology to reduce its connectivity. Every node in network has the ability to control its connectivity, then CPT has the ability to optimize the network topology.

$$\bar{d} = \lim_{n \rightarrow \infty} \frac{2 \times l}{n} = 6 - \lim_{n \rightarrow \infty} \frac{12}{n} = 6 \quad (2)$$

Theorem 1 has proved that the number of edges are fixed for a CPT of which all surfaces are triangles, if the degree of some nodes are reduced, the degree of other nodes should be increased to keep the convex-polytope structure. So, the self-optimization of CPT can be viewed as a transfer process of node degree from the high-degree nodes to the low-degree nodes.

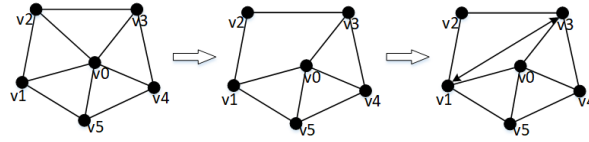


Fig. 6. The self-optimization process of CPT

Assume that node v_0 has n neighbouring nodes which are labeled in sequence from v_1 to v_n . Because all the neighbouring nodes of v_0 form a circuit, we use v_{i-1} and v_{i+1} respectively represent the adjacent nodes of v_i in the node circuit. If node v_0 needs to reduce its degree, firstly node v_0 has to request for the degree d_i of its each neighbouring node v_i ($1 \leq i \leq n$). According to the degree of its neighbouring nodes, node v_0 selects a suitable neighbouring node to disconnect. To maintain the convex-polytope structure, the corresponding two neighbouring nodes of node v_0 have to connect with each other. As illustrated in Fig. 6, node v_0 disconnects with node v_2 , a quadrilateral surface (v_0, v_1, v_2, v_3) is formed. After v_1 connects with v_3 , the degree of node v_0 and v_2 is reduced and the degree of v_1 and v_3 is increased.

To balance the distribution of nodes' connectivity, it is better to guarantee that two disconnected nodes are with high-degree and the two connected nodes are with low-degree. So, before node v_0 decides the disconnected node, it needs to calculate which node is more suitable to disconnect according to the degree of each neighbouring node.

For each neighbouring node v_i of node v_0 , we calculate the fitness value c_i of node v_i as shown in Equ. 3. If the degree of node v_i is smaller than one of its two adjacent nodes, the fitness value of node v_i is 0. Because after node v_0 disconnects with this candidate node, the degree of its adjacent nodes will become bigger. If the degree of node v_i is bigger than both of its adjacent nodes, node v_0 calculate the fitness value by the multiplication of $(d_i - d_{i-1})$ and $(d_i - d_{i+1})$. At last, node v_0 selects the candidate node with the biggest fitness value to disconnect. Then, node v_0 instructs the two adjacent nodes of candidate node to connect with each other to maintain the topology stable.

$$c_i = \begin{cases} 0 & \text{if } d_i \leq d_{i-1} \text{ or } d_i \leq d_{i+1} \\ (d_i - d_{i-1}) \times (d_i - d_{i+1}) & \text{if } d_i > d_{i-1} \text{ and } d_i > d_{i+1} \end{cases} \quad (3)$$

Algorithm 2 gives the pseudocode for the self-optimization algorithm of each

Algorithm 2 Self-optimization Algorithm

Input: v_0 : current node; d_0 : the degree of v_0 .

```

1: while  $d_0 > DegreeLimit$  do
2:    $(d_1, d_2, \dots, d_n) \leftarrow GetDegree()$ .  $\triangleright$  Get the degree of each neighbouring node of
      node  $v_0$ .
3:    $(c_1, c_2, \dots, c_n) \leftarrow CalculateFitnessValue()$ .  $\triangleright$  Calculate the fitness value for
      each neighbouring node.
4:    $v_i \leftarrow GetMaxValueNode()$ .  $\triangleright$  Get the node with the highest fitness value.
5:    $Disconnect(v_0, v_i)$ .
6:    $Connect(v_{i-1}, v_{i+1})$ .
7:    $Update(v_0)$ .  $\triangleright$  Update the edges, surfaces of node  $v_0$ .
8: end while

```

node. Once the degree of each node is bigger than the upper limit of degree (We usually set the upper limit of degree is 10), the current node begins the self-optimization process to adjust its local topology. And the basic idea of self-optimization is to transfer the degree from the high-degree node to low-degree node. With the collaboration of all nodes in anti-tracking network, CPT can achieve the self-optimization of network topology.

4 Experiments and Analysis

In this section, we compare CPT with two self-organizing methods of network topology, one is based on the neural network (NN) [16] and the other is based on distributed hash tables (DHT) [17]. NN achieves the self-optimization of topology by the neural network algorithm deployed in each node. DHT achieves the self-organizing network based on the hierarchical aggregation structure. We evaluate the three methods from the following aspects: (1) network robustness, (2) maintenance efficiency, and (3) communication efficiency.

4.1 Evaluation of Network Robustness

Firstly, we separately simulate three networks with 5000 nodes by CPT, NN and DHT, calculate the distribution of nodes' degree D_d and the distribution of nodes' density D_s to compare the difference of the three network topologies. We use the number of nodes which have the span not more than 2 with v_i to represent the density of v_i .

As illustrated in Fig. 7(a), D_d of CPT is mainly distributed in the interval $[3, 10]$, because of the convex-polytope structure limitation. D_d of NN is

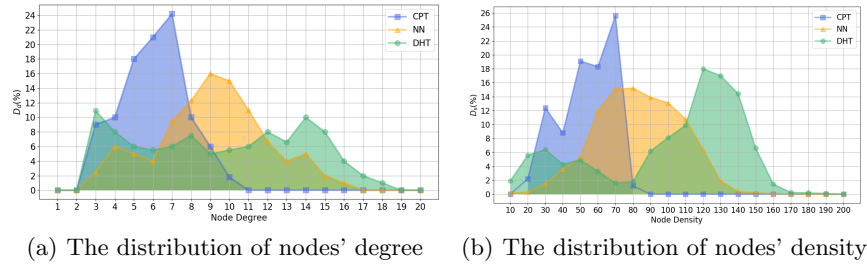


Fig. 7. The evaluation of the distribution of nodes' degree and density

mainly distributed in the interval $[3, 15]$. D_d of DHT is distributed widely. The distribution of nodes' density has the similar tendency with the distribution of nodes' degree. In general, the local density of nodes is in direct proportion to their degree, and the nodes with high density becomes the key nodes easily. In Fig. 7(b), D_s of CPT is mainly distributed in the low density area which means the topology structure of CPT is more stable than NN and DHT.

To evaluate the network resilience, we firstly give a metric to quantify the performance of network resilience. As shown in Equ. 4, we use a graph G to represent the original network, $G(p)$ denotes the subgraph after p percent of nodes is removed from G , $MCS(G(p))$ denotes the maximum connected subgraph of $G(p)$, $Num(g)$ denotes the node number of a graph g , N denotes the node number of G . The metric β measures the maximum connectivity of the network after some nodes are removed from the network. The β is higher, the network resilience is better.

$$\beta = \frac{Num(MCS(G(p)))}{N} \quad (4)$$

Based on the above simulated networks constructed by CPT, NN and DHT, we remove p percent nodes from the three networks each time until all nodes are removed. In each round, we calculate the β to evaluate network resilience. In the experiments, we use two different ways to remove nodes shown as follows.

- **Random-p Removal.** In each round of nodes removal, we remove p percent of nodes from the network randomly.
- **Top-p Removal.** In each round of nodes removal, we remove p percent of nodes with the highest degree.

From the experimental results shown in Fig. 8, CPT has a better performance in network resilience and β keeps higher than NN and DHT in both random-p removal and top-p removal. In random-p removal, even β of CPT is always higher than NN and DHT, the difference of the three methods is not very high. The performance of the three methods in network resilience begins to decrease when β is less than 40%.

But in top-p removal, the performance of DHT degrades sharply. The performance of NN in network resilience begins decrease when β is less than 20%.

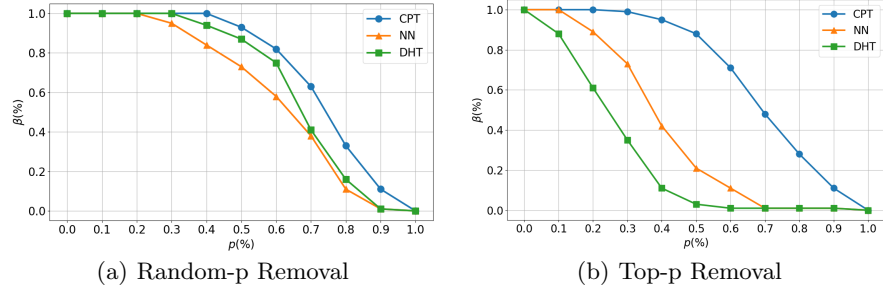


Fig. 8. The evaluation of network resilience

But CPT still maintains good performance in network resilience under top-p removal. The experiments show that top-p removal causes bigger damage to the network of NN and DHT than random-p removal. But CPT maintains good performance of network resilience in both random-p removal and top-p removal.

4.2 Evaluation of Maintenance Efficiency

The real network environment is complex and changeable, the maintenance efficiency has a direct influence on the performance of network self-optimization. If the network maintenance process costs too much time or computing power, it is ineffective in the practical application. In this section, we evaluate the maintenance efficiency of CPT, NN and DHT from two aspects.

- **Efficiency of network construction.** According to the network construction methods of CPT, NN and DHT, we construct a network in simulation environment and count the time requirement for comparison.
- **Efficiency of network self-optimization.** For each network constructed by CPT, NN and DHT, we randomly remove p percent of nodes and count the time requirement of network self-optimization as the metric. By increasing the percentage of removed nodes, we compare the change of time requirement to evaluate the efficiency of network self-optimization.

Fig. 9(a) illustrates the time requirement of network construction with CPT, NN and DHT. The value of X-axis denotes the node number of the constructing network. With the increase of the network node number, the time requirement of CPT increases slower than NN and DHT. Limited by the computation of neural network, the time requirement of NN increases sharply. DHT relies too much on the agent nodes to construct and manage the network topology. Benefited from the convex-polytope structure, the whole network of CPT keeps the balanced distribution of nodes' connectivity. Without the too much computation like NN and too many connections like DHT, CPT can achieve better efficiency of network construction.

Fig. 9(b) illustrates the time requirement of network self-optimization after p percent of nodes are removed from the network randomly. When more than 50%

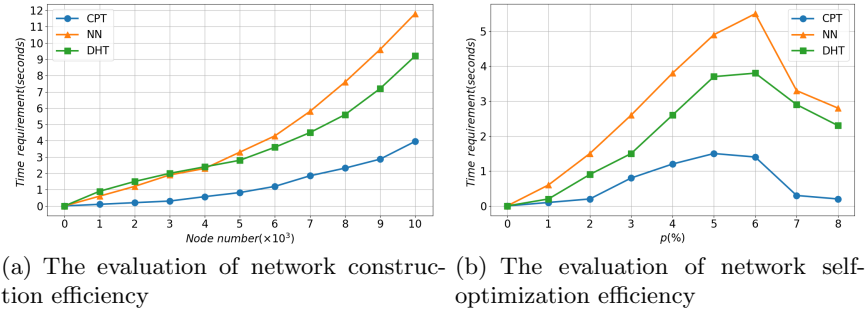


Fig. 9. The evaluation of network maintenance efficiency

nodes are removed randomly, the network will be split into different subgraphs. In this case, we need to connect the different subgraphs into one whole network, then continue the network self-optimization process and count the time requirement. As we can see in Fig. 9(b), the curves of three methods increase at the beginning, and then decrease when p is more than 50%. Because along with the increase of the percentage of removed nodes, the network size becomes smaller. So, when the network size reduces to a certain degree, the time requirement of network self-optimization will decrease.

In this experiment, CPT performs better in both network construction and network self-optimization. The simple structured topology of CPT improves the efficiency of network construction, maintenance and self-optimization.

4.3 Evaluation of Communication Efficiency

To evaluate the communication efficiency, we randomly select two nodes to communicate in each test and count the average time requirement after 100 round tests. With the increase of network size, we compare the communication efficiency of CPT, NN and DHT through the average time requirement.

The network diameter and average path length increase with the increase of network size. The randomly selected two nodes would spend more time to communicate in each network. As illustrated in Fig. 10, the average time requirement of CPT is always higher than NN and DHT. But, the difference of communication efficiency between CPT, NN and DHT is not very big and it is also acceptable for anti-tracking network.

The reason of CPT's low communication efficiency blames for the convex-polytope structure. To keep the convex-polytope structure, CPT improves the network robustness and destroy-resistance at the cost of communication efficiency. And the research has proved that anti-tracking network can not achieve all the three properties: strong tracking-resistance, low bandwidth overhead, and low latency overhead [18]. But from the tracking-resistant standpoint, the weakness in communication efficiency does not impact the overall efficiency of anti-tracking network. And also, we can add extra connections between different

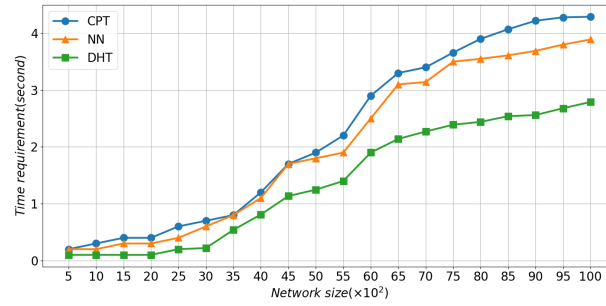


Fig. 10. Evaluation of communication efficiency between CPT, NN and DHT

nodes along the diagonal inside the convex-polytope topology to reduce the network diameter, then the communication efficiency of CPT would be improved greatly.

5 Conclusion

In this paper, we propose convex-polytope topology (CPT) which can be applied in anti-tracking network. We construct the anti-tracking network according to the properties of convex-polytope, and maintain the network topology in the convex-polytope structure. When the node join in or quit the network, CPT can still maintain the convex-polytope topology to keep a stable and resilient network. Based on the convex-polytope topology, we design a self-optimization mechanism for anti-tracking network. The experimental results show that CPT has better performance in network robustness, maintenance efficiency than current works.

Acknowledgements We thank the anonymous reviewers for their insightful comments. This research was supported in part by the national natural science foundation of China under grant No.U1736218 and No.61572496. Yongzheng Zhang is the corresponding author.

References

1. Fatemeh S , Milivoj S , Rizwan A M , et al. A Survey on Routing in Anonymous Communication Protocols[J]. ACM Computing Surveys, 2018, 51(3):1-39.
2. Sun Y, Edmundson A, Vanbever L, et al. RAPTOR: Routing Attacks on Privacy in Tor[C]//24th USENIX Security Symposium (USENIX Security 15). 2015: 271-286.
3. Buchanan T, Paine C, Joinson A N, et al. Development of measures of online privacy concern and protection for use on the Internet[J]. Journal of the American Society for Information Science and Technology, 2007, 58(2): 157-165.
4. Tian C, Zhang Y, Yin T, et al. Achieving Dynamic Communication Path for Anti-Tracking Network[C]// 2019 IEEE Global Communications Conference (GLOBE-COM). IEEE, 2019: 1-6.

5. Jan M A, Nanda P, He X, et al. A Sybil attack detection scheme for a forest wildfire monitoring application[J]. *Future Generation Computer Systems*, 2018, 80: 613-626.
6. Tiwari R, Saxena T. A Review on Sybil and Sinkhole of Service Attack in VANET[J]. *Recent Trends in Electronics and Communication Systems*, 2018, 5(1): 7-11.
7. Khan S A, Engelbrecht A P. A fuzzy particle swarm optimization algorithm for computer communication network topology design[J]. *Applied Intelligence*, 2012, 36(1): 161-177.
8. Auvinen A, Keltanen T, Vapa M. Topology management in unstructured P2P networks using neural networks[C]//2007 IEEE Congress on Evolutionary Computation. IEEE, 2007: 2358-2365.
9. Nakano T, Suda T. Self-organizing network services with evolutionary adaptation[J]. *IEEE Transactions on Neural Networks*, 2005, 16(5): 1269-1278.
10. Tian C, Zhang Y Z, Yin T, et al. A Loss-Tolerant Mechanism of Message Segmentation and Reconstruction in Multi-path Communication of Anti-tracking Network[C]//International Conference on Security and Privacy in Communication Systems. Springer, Cham, 2019: 490-508.
11. Yang M, Luo J, Ling Z, et al. De-anonymizing and countermeasures in anonymous communication networks[J]. *IEEE Communications Magazine*, 2015, 53(4): 60-66.
12. Yang M, Luo J, Zhang L, et al. How to block Tor's hidden bridges: detecting methods and countermeasures[J]. *The Journal of Supercomputing*, 2013, 66(3): 1285-1305.
13. Mittal P, Borisov N. Information Leaks in Structured Peer-to-Peer Anonymous Communication Systems[J]. *ACM Transactions on Information and System Security*, 2012, 15(1):1-28.
14. Zang W, Zhang P, Wang X, et al. Detecting Sybil Nodes in Anonymous Communication Systems[J]. *Procedia Computer Science*, 2013, 17: 861-869.
15. Wang J, Wang T, Yang Z, et al. Seina: A stealthy and effective internal attack in hadoop systems[C]//2017 International Conference on Computing, Networking and Communications (ICNC). IEEE, 2017: 525-530.
16. Tian C, Zhang Y Z, Yin T, et al. A Smart Topology Construction Method for Anti-tracking Network Based on the Neural Network[C]//International Conference on Collaborative Computing: Networking, Applications and Worksharing. Springer, Cham, 2019: 439-454.
17. Li Z P, Huang J H, Tang H. A P2P computing based self-organizing network routing model[J]. *Journal of Software*, 2005, 16(5): 916-930.
18. Das D, Meiser S, Mohammadi E, Kate A. Anonymity Trilemma: Strong Anonymity, Low Bandwidth Overhead, Low Latency-Choose Two[C]//2018 IEEE Symposium on Security and Privacy (SP). IEEE, 2018: 108-126.