International Journal of Future Generation Communication and Networking
Vol. 9, No. 6 (2016), pp. 113-128
http://dx.doi.org/10.14257/ijfgcn.2016.9.6.11

NSDA: A Novel Node Selecting Optimal Algorithm Based on Dijkstra

Tianbo Lu, Jiao Zhang, Lingling Zhao and Yang Li and Xiaoyan Zhang

School of Software Engineering，Beijing University of Posts and Telecommunications, 100876, Beijing, China
lutb@bupt.edu.cn, tianyishirly@163.com, wodepengyouzhao@163.com

Abstract

This paper first represent the basic conceptions of anonymous communication, then introduce the fundamentals in Anonymous Communication. Afterwards, we went deep to the classification of anonymous network topological, therefore we could get a better understanding on P2P anonymous communication system. We also systematically analyzes the existing node-select algorithms. Furthermore, we propose and implement a new node selection algorithm based on Dijkstra algorithm, named NSDA algorithm. We give a belief description about the backgrounds and design of NSDA Algorithm. The algorithm can combine the node properties with link properties to select node, which can adjust the system performancing and anonymity. In order to evaluate the NSDA algorithm, we have done experiments in Network Simulator 3, and we also explain the reason why we choose Network Simulator 3 as the simulate tool. In the end, we analyze the characteristics of NSDA algorithm according to the results of experiments.

Keywords*: Cyber Physical Systems, Security, Cyber-Physical Attack*

1. Introduction

Nowadays, Internet privacy becomes more and more important and sensitive, and has been one of the latest buzz words to hit the Internet world. As one of privacy enhancing technologies, anonymous communication has been extensively studied by researchers from various aspects. Existing research focuses on solving problems and challenges in these systems and architectures, especially the issues of anonymity and performance. The issue of node is the core part of anonymous P2P network, and routing algorithm is the core algorithm which affects the anonymous system performance. Therefore, this paper focuses on the study of node selection to find the more suitable for constructing an anonymous path. Firstly, this paper systematically analyzes the existing node selection algorithms. Then, we propose and implement a new node selection algorithm based on Dijkstra algorithm, named NSDA algorithm. The algorithm can combine the node properties with link properties to select node, which can adjust the system performance and anonymity. In order to evaluate the NSDA algorithm, we have done experiments in Network Simulator 3, and according to the results of experiments, we analyze the characteristics of the algorithm. Finally, some related problems are presented to be studied deeply in future.

2. Related Work

2.1. Anonymous Communication Fundamentals

In the early 1980s, the research of anonymous communication came out. Chaum puts forward the concept of multilayer encryption forward, aimed to solve the problem of email traffic analysis and published the groundbreaking papers [1]. Anonymous

ISSN: 2233-7857 IJFGCN

communication is hidden identity or relationship between the two parties, focuses on the confidentiality of the user's identity in network communication. Here, the user identity refers primarily host IP address, email address and other relative information [2].

Anonymous communication must satisfy three requirements: unable to identify the source node, the target node is not recognized, the data stream can't be tracked. These properties can be referred as having satisfied anonymity. Anonymity means indistinguishable communications entities from collective communication parties. Anonymous path is that the use of an ordered collection of nodes from the sender to the receiver transmission. The number of nodes used in the path is the length of anonymous path. In anonymous path, typically involves three relay nodes, respectively:

Entry node: First hop anonymous path, the node knows the sender of this communication.

Exit node: Last-hop anonymous path, the node knows the recipient of this communication.

Intermediate node: The node between the entry and exit nodes. Intermediate nodes only know their backward hop and forward hop and completely unknown about identity of the two parties in the communication.

2.2. Anonymous Communication based on Mix Algorithm

Anonymous communications based on Mix mechanisms [3] mostly use a single or multiple Mix nodes forwarding and processing data in a certain way, to achieve anonymity. The core idea of the Mix algorithm is: before sending a message, you need to use the Mix nodes' public key to encrypt the message in reverse sequence, and then encrypting the message to the next Mix node, when the Mix node has received the message, firstly decrypt the message with its own the private key, and then reorder the data, increasing the redundancy information, and then send the data to the next Mix node, and so, until a message is sent to the recipient. Currently, systems based on Mix algorithm include Babel [4], Mixminion [5], and Mixmaster [6], Tor etc. Tor is the most typical and most widely used open source anonymous system, which currently consists of more than 5,000 relay, most of the nodes are located in Germany, USA, France and the Netherlands [7]. The characteristics of communication Based on Mix algorithm are as follows:

➢Before communication, the sender need to determine the path of anonymous communication, then get the public key, address and other information of each Mix nodes in the path.

➢Mix node will decrypt the received message, disordering the message, adding redundant information, to secure high anonymity of the system.

The flaw of anonymous communication based on Mix algorithm is high latency, limited ability on anti-collusion attack.

2.3. P2P Anonymous Communication System

Peers make a portion of their resources, such as processing power, disk storage or network bandwidth, directly available to other network participants, without the need for central coordination by servers or stable hosts. Peers are both suppliers and consumers of resources, in contrast to the traditional client-server model in which the consumption and supply of resources is divided. Anonymous generally aid groups to hide their behavior, but a way to get a large-scale population is P2P network [13], so people began to study the large-scale anonymous P2P networks. Existing anonymous P2P networks including Tarzan [10], MorphMix [11], Crowds [14], I2P [15], designed to provide general network infrastructure anonymity. In these networks, each node not only as a server to provide services to other users; and can be used as a client, use the services provided by an anonymous system.

2.4. The Classification of Anonymous Network Topological

The virtual infrastructure of network is topology. Network topology can affect anonymity of anonymous communication systems and the overhead of link filling. Currently, there are two kinds of classifications on anonymous network topology. The first classification is structured topology (such as Chord, Pastry) and unstructured topologies (such as Tor). The second classification which is from the perspective of the complexity of the connected nodes can be divided into free, Stratified, Cascade [8, 9].

2.4.1. Free

In free anonymity network(as Figure 1), anonymous systems is unnecessary to construct or maintain anonymous path, nor impose restrictions on the construction of an anonymous path, but leave it to the sender to make the decision, the sender can choose to build their own node path of arbitrary length[]. Tor system for example, assuming that there are n nodes in the network, any three different nodes can be combined into a path. Given an entry node, intermediate nodes may be uniformly randomly selected from the remaining n-1 nodes, and then select the exit node uniform random from the remaining nodes of the n-2. A typical system freedom topology including: Tarzan [10], MorphMix [11], I2P [12] and Tor [7], etc. The virtues of free network topology are full of dynamic ability that can be adapted to changes in the node, and no costs to build and maintain anonymous path.

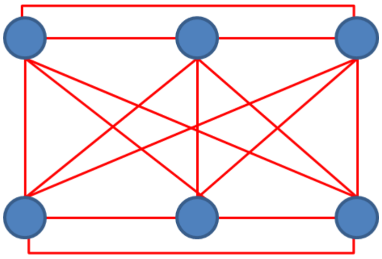

Figure 1. Free (n=6)

2.4.2. Stratified

In Stratified anonymous network (as Figure 2), the nodes will be divided into three sets, namely the entry nodes set, intermediate nodes set, exit nodes set. If the number of all nodes is n, the each set contains n / 3 nodes. The entry node might connect to any one of intermediate nodes, similarly, the intermediate nodes also might connect to any exit node. Given an entry node, pick one node from intermediate nodes set (n / 3), and then pick one from the exit nodes set (n / 3).

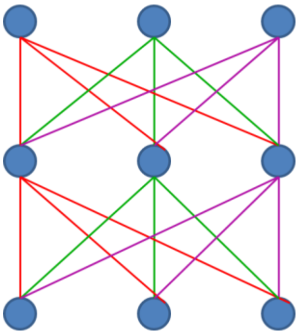

Figure 2. Stratified (n=9)

In addition, the stratified may limit the quantity of connections of per node, that the selected nodes set n＝$3K^2$ (K is an integer), the K＝$\sqrt{n/3}$. Each node connected K intermediate nodes. Similarly, each intermediate node is connected K exit nodes. Any entry node i (0 <= i <= n / 3-1) is connected the intermediate nodes n / 3 + [(i + j) mod n / 3], where j = 0 ... K-1. Any middle node i (n / 3 <= i <= 2n / 3-1) is connected to any of exit nodes 2n / 3 + [(i + j * K) mod n / 3]. When randomly selected a entry node from entry nodes set n / 3, the relationship between the entry node and exit node is constructed by one intermediate node.

2.4.3. Cascade

In cascade anonymous networks (as Figure 3), the sender unnecessary to construct their own anonymous path, and use the path which the anonymous system has built communicate with each other. In other words, the sender simply selects a fixed path from the system to send a message anonymously. *i.e.* given an entry node, intermediate nodes and exit nodes are fixed, such as a set of n nodes, there will be n / 3 fixed path. Falls topology is mainly applied to a central system, but the topology must have long-term stability of the nodes online, otherwise the path provided by the system is unstable, and once the size of the system is expanded, it is difficult to maintain the anonymity of the path system. In general, the cascaded topology is not as high anonymity as free, but its corresponding cost and complexity is also high.

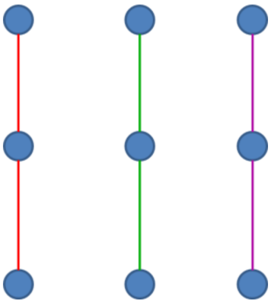

Figure 3. Cascade (n=9)

In the above three types of topologies, cascade can get the best trade-off anonymity and communication overhead. On communication overhead, the cascade superior to stratified, and in terms of anonymity, but it needs high cost, in other words, its extensibility is poor in anonymity. In the terms of anonymity and overhead, the free is better than stratified topology, and given free type require disproportionate get filled in when filling, free type is not reality.

2.5. Node Resource Selection Algorithm

Selection of node resources, namely finding the right node resources to construct communication path, is also finding out the specific node resources. Node resource selection can affect bandwidth utilization and throughput, mainly to solve the safe, rational and effectively select relay nodes. With the growth scale of the nodes, relay nodes may differ in terms of network performance, uptime, geographical distribution, etc. Therefore, the relay node will show different categories and levels. Currently, due to different considerations, lots of nodes resource selection algorithm have been put up. Broadly speaking, the nodes resource selection can be divided into two categories: based on nodes features (such as bandwidth) and based on link features (delay) algorithm. It is worth mentioning that the nodes resource selection algorithm is same as the path selection algorithm, but in a different order, nodes selection algorithm specifically pointed out how to select each node, thus establishing these nodes for anonymous path nodes. Path selection algorithm is based on certain principles elect a path, which in turn can establish a path containing nodes. Based on this, this paper does not strictly two options algorithms.

3. NSDA Algorithm Background & Design

3.1. NSDA Algorithm Background

In anonymous P2P network, the performance of all the nodes is not identical. According to the bucket theory, the performance is depend on the worst nodes in anonymous communication path, so when we choose to construct anonymous node path, given that it is not practical to obtain global topology in the P2P network. So according to local topology information, we choose nodes with relatively good performance, and complex and relatively close geographic. In this case, performance means the comprehensive of bandwidth, CPU time, and online resources. Although there are many nodes selection algorithm, as described in the previous section, but most of them are for Tor nodes selection algorithm system does not apply to anonymous P2P networks. Therefore, this paper selection algorithm in basic research on the existing node resources, inspired by the above algorithm, ant colony algorithm, the small world theory, topological theory, proposed a Dijkstra (Dijkstra) algorithm based on node selection algorithm (Node Selection based on Dijkstra Algorithm, NSDA).

3.1.1. Ant Colony Algorithm

Ant colony algorithm is a bionic evolutionary algorithm proposed in 1996 by Dorigo et al [16], which is based on population factors Optimization heuristic search algorithm. The algorithm is mainly origin from the exploration of the natural foraging behavior of ants. When foraging, ants will separate to find food. When ants found, then it is returned to tell ant colony nest and leave a pheromone along the way, which is to guide the ant colony pheromone found food again. But pheromones will continue to be volatile. If there are two ants also found a food and return to nest along different routes, then the smell will last lighter on the farther line, so the ant colony will choose food from closer route to the location. According to the principle of ant colony algorithm through - the more concentrated pheromone path, the more optimal route, select the best path.

Although initially ant colony algorithm was to solve the traveling salesman problem (Traveling Salesman Problem, TSP), but now has penetrated into many areas, especially in the network area, such as communication network load balancing and routing processing problems. Ant colony algorithm is a typical implementation of swarm intelligence, is well suited for decentralized P2P mechanism. Based on anonymous P2P system, in which we focus on the ant colony algorithm in two characteristics: First, the whole process of adaptive path algorithm, which does not depend on the initial point of selection, less affected by the initial point. Second, the positive feedback mechanism: the more pheromone on a path, the more ants choose the path; ant increases, the corresponding pheromone increases, so the convergence rate will accelerate as the positive feedback.

Inspired by ant colony algorithm, when constructed an anonymous path, each node in the network was set a parameter, just like the ant colony algorithm pheromone concentration parameters recording how many paths the node has been involved, in the case of two paths are involved, then the parameter is set to 2. So when you select an anonymous path, set up a multi-parameter selection criteria, the user can load and weigh anonymous efficiency, if the user selects a high load path, follow the ant colony algorithm, the high concentration of pheromone paths, thereby constructs the shortest path. If the user chooses the high efficiency of the anonymous path, choose low pheromone concentrations path because this path there are fewer other nodes involved, unlikely to cause an attacker's attention and interests, and select the network to avoid the "hot spots" will not overload conditions. In addition, if every node to choose low pheromone concentrations path, you can ensure a balanced load P2P network node resources, greatly increasing the utilization of the network.

3.1.2. Small World Theory

Small world theory (Six Degrees of Separation), also known as Six Degrees of Separation theory, theory of six degrees of separation, is that the interval between a man and a stranger is no more than six[14]. Watts and Strogatz in 1998 [12] proposed a "small-world model (WS model)," and noted that networks with six degrees of separation nature is the small-world network (Small World Network, SWN). Features of WS model: differences between the degree of network nodes are small, can be approximately equal; length of paths between any two nodes to establish is quite short. For the small-world networks, characteristic path length is an average of the length between any two nodes pair, clustering coefficient refers to a cross circle of friends between two adjacent nodes. In this network, characterized path length between nodes is small, and is close to random network, but the clustering coefficient is high, close to the regular network, which can reduce the communication overhead of the network and reduce the average path length of the network. Therefore, small-world network is the network between the random network and regular work.

P2P network also has the small world characteristic. In the view of from the topological, It can be considered that distance between any two points connected graph is not greater than some constant, which average path length of any two nodes is approximately constant. Being inspired by this, when select an anonymous path, do not find out all the neighbors of the node, but to pick some constant nodes, thereby reducing the amount of computation.

3.1.3. Shortest Path

Anonymous P2P network node is connected, not existing isolated. In order to achieve the purpose of anonymity requires forwarding node, and forwarding nodes are located in a circle, with a variable number of neighbors, and the neighbors may also be connected. Therefore, the two parties of communication and forwarding nodes and its neighbors may constitute a small range of local connectivity topology. Due to many practical applications to their roots are mathematics problems, to some extent the solution for anonymous path the problem can be considered as finding the shortest path problem. Although the performance is proportional to the bandwidth of the path, seemingly unable to use the shortest path to solve, but if the value of the node bandwidth reciprocal regarded vertices, so the shortest path is available.

In order to quantify the network topology, bandwidth, throughput, trust, and delay between nodes can be regarded as the edge weight, so that the topology can be abstracted as a weighted connected graph. Although this graph is an undirected graph, it also can be regarded as a two-way connection connected graph. In order to guarantee the performance of the anonymous path, finding path with good performance of anonymity can be seen as solving a request from the sender to the recipient of the shortest path. Dijkstra's algorithm is to solve directed graph shortest path problem, that is, find shortest path from one point to the other point algorithm to ensure that the path is simple and effective without loop.

Inspired by the above, this paper presents a high flexibility of the node based on Dijkstra's algorithm, and uses the first amplification and further reduced method, that is, firstly expand the scope of topology, select nodes in a wide range, then use Dijkstra's algorithm to narrow the nodes scope, optimizing the path from the sender to the receiver.

3.2. NSDA Algorithm Design

Given the performance and safety of anonymity systems can be affected by factors of node bandwidth, delay, the number of participated paths, geographic diversity and etc. Therefore this article has considered these factors on the basis, proposed a new node selection algorithm based on Dijkstra algorithm.

```
Input: Sender S, Receiver R, x, router_list
Output: path{n0,n1,..ni}
Procedure:
i = 0, f_randomNode();
for i=0 to (N-1) do
        path0[] = f_randomPath(S,R);
        nodeNeigh[] = f_getNeighbor(path0[]);
        nodeSet[] = path0[] +nodeNeigh[];
        f_getWeight()
        if x=0 then
               args[$n_{i-1}$][ $n_i$] = f_metricDelay($n_{i-1}$,$n_i$);
        else if x=1 then
        args[$n_{i-1}$][ $n_i$] = f_metricNode($n_{i-1}$,$n_i$);
else args[$n_{i-1}$][ $n_i$] = f_metricDelay($n_{i-1}$,$n_i$) + x* f_metricNode($n_{i-1}$,$n_i$ )
end if
end for
     path1[]= dijkstra(nodeSet[], args[$n_{i-1}$][ $n_i$])
     path2[]= dijkstra(nodeSet[], args[$n_{i-1}$][ $n_i$])
if path1[] != path2[] then
       numAs1 = f_getAsNum(path1[]);
       numAS2= f_getAsNum(path2[]);
       if numAs1 < numAs2 then
       path{n0,n1,...,ni } = path2[]
       end if
       end if
path{n0,n1,...,ni } = path1[]
return path{n0,n1,...,ni}
```

(1)Use f_randomPath (S, R) function, randomly selected a communicate path between A and B, recording nodes (such as C, D, E) on pathway, and labeled nodes set G0, as shown in Figure 4 selected G0 = {C, D, E}. All nodes contained for G1 = {A, B, C, D, E}. To ensure anonymity, the path length is longer than 1. Set the minimum is 2, because it is possibly to select a direct interconnection between A and B, we are unable to create an anonymous path, if only through one node forwarding, the probability of being attacked is quite high, so the path at least go through two intermediate nodes.

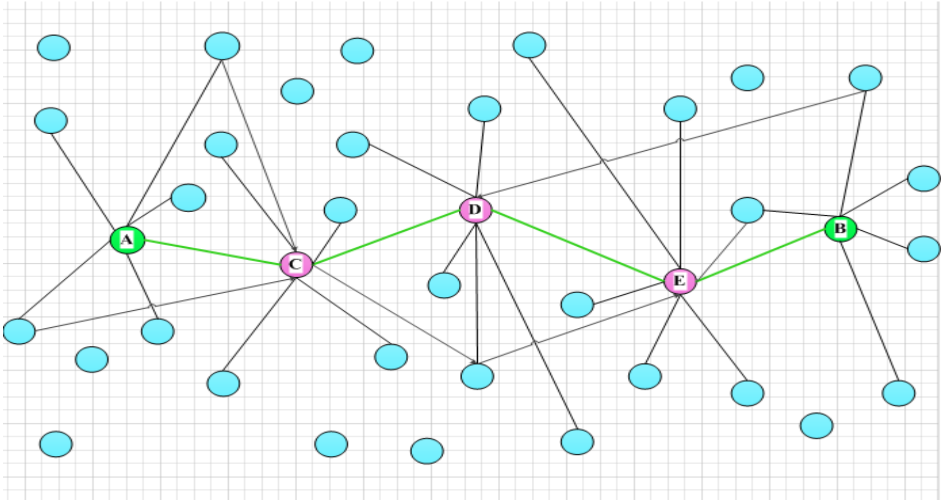

Figure 4. The Random Path between A and B

(2) Construct nodes set. Each node has neighborhood, chosen from G1 adjacent nodes in the network denoted as G2.Due to the node in network has more than one neighbor, the node N0 in the G0 set may be adjacent to node C in G1, may also be adjacent to the nodes D in G1. To ensure that the path to receive node is available, reducing the neighbor nodes ofG2, only retaining the nodes which have more than two connected nodes. These nodes constitute nodes set G3. The total nodes set G4 contains communicate nodes A, B, nodes set G0 and its retaining neighborhood nodes set G3, as shown in Figure 5 selected (R1.R2, R3, R4, R5).

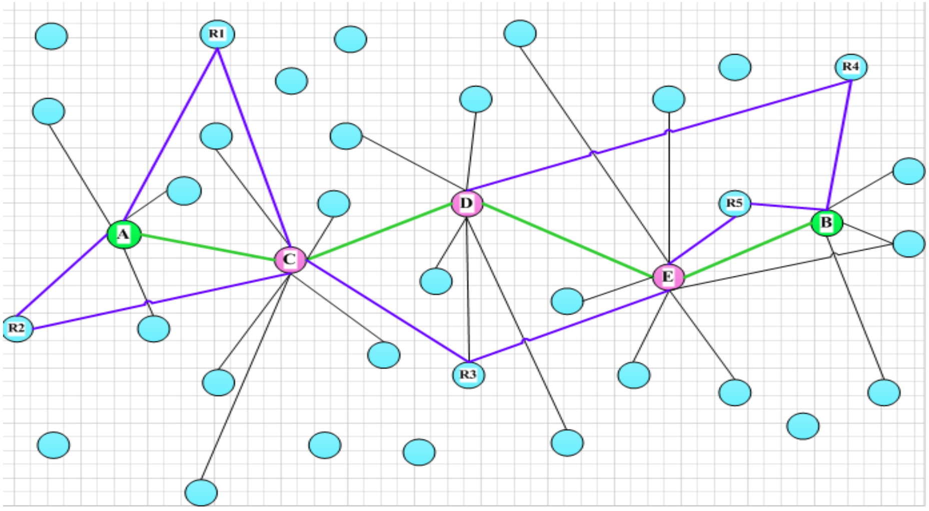

Figure 5. Selection Neighbor Nodes on Demand

(3)Construct a weighted connected graph. That simplifies the actual network connection (Figure 6), nodes of G_2 are regarded as point which has nothing to do with its size, shape, the path connected the nodes into can be abstracted to lines, therefore made of a connected graph (shown in Figure 6). The round-trip delay between two nodes is regarded as edge weights. Using GeoIP service according to the given IP information to deduce the country code and the number of autonomous systems, and mark the next autonomous systems.

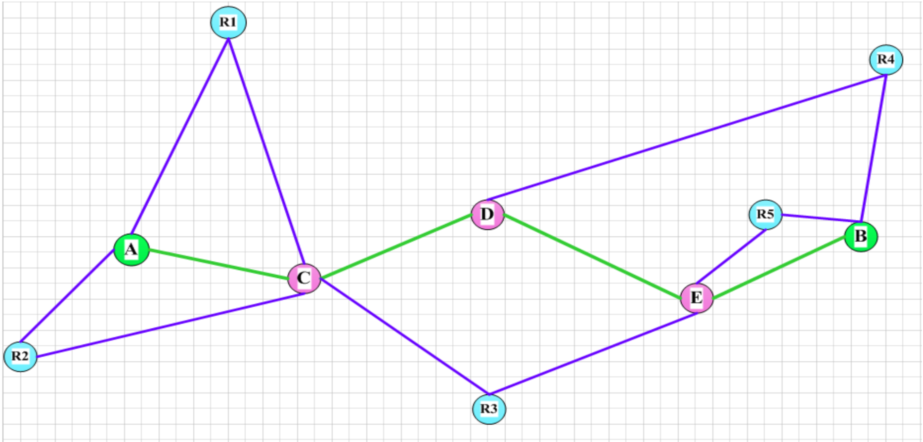

Figure 6. NSDA Algorithm Selected Local Topology

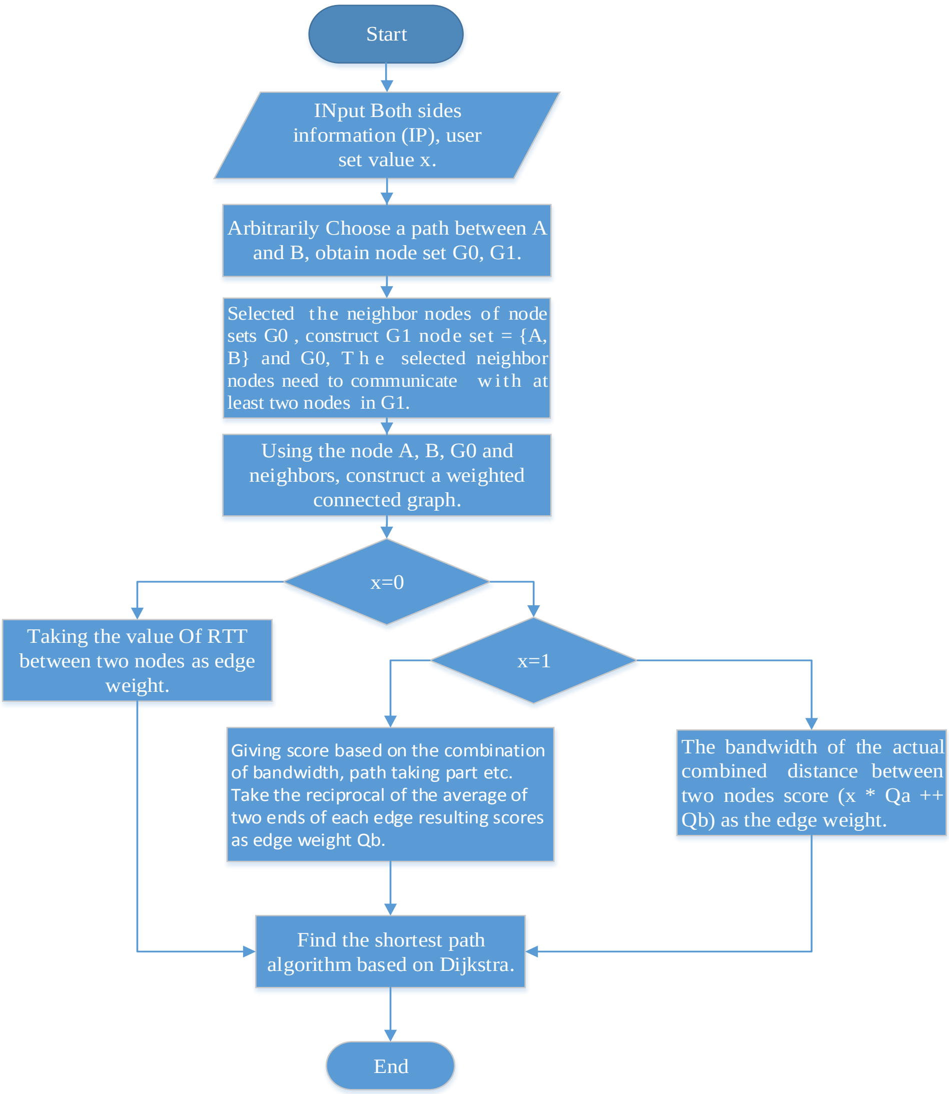
Start
INput Both sides information (IP), user set value x.
Arbitrarily Choose a path between A and B, obtain node set G0, G1.
Selected the neighbor nodes of node sets G0 , construct G1 node set = {A, B} and G0, The selected neighbor nodes need to communicate with at least two nodes in G1.
Using the node A, B, G0 and neighbors, construct a weighted connected graph.
x=0
Taking the value Of RTT between two nodes as edge weight.
x=1
Giving score based on the combination of bandwidth, path taking part etc. Take the reciprocal of the average of two ends of each edge resulting scores as edge weight Qb.
The bandwidth of the actual combined distance between two nodes score (x * Qa ++ Qb) as the edge weight.
Find the shortest path algorithm based on Dijkstra.
End

Figure 7. The NSDA Algorithm Flow Chart

(4)Change weight. To make a flexible combination of anonymity and network overhead, set a parameter to satisfy the needs of different kinds of application.

Instant messages user may choose the low-latency anonymous communication, which also choose edge-weighted connected graph.

File transfer user can select high bandwidth anonymous communication, namely alter above values, give a comprehensive score of the bandwidth of each node, the number of paths each node take part in, the online time, the trust degree, etc. Take the reciprocal of the average of two ends of each edge resulting scores as edge weight, and then construct a weighted connected graph.

If the user wants to set their own anonymity, he can take the comprehensive score as Qa, and the round-trip delay between nodes resulting as Qb, combining those two weights to a new edge-weight. Make a combination of bandwidth, online time, delay, path taking part etc. Setting is (x*Qa+Qb), x is a random number in (0, 1) set by user, to construct the shortest path.

$$f(x)=\begin{cases}\text{Qa}, & \text{x}=0\\ \text{Qb}, & \text{x}=1\\ \text{x*Qa+Qb}, & \text{x}\in(0,1)\end{cases}$$

(5)Using Dijkstra to get shortest path, namely select a path from node A to node B, but since there may be a weighted connected graph edges have the same weight, it may be more than one shortest path, if there are two or more, according to the system described in the previous mark of autonomy to select the number of autonomous systems containing more paths to ensure the path complexity, reduce probability of being compromised, improve anonymity. A flowchart showing the algorithm (Figure 7).

4. NSDA Algorithm Analysis

4.1. NSDA Algorithm Characteristics

4.1.1. Optimal Random Path

Given after the communicating parties has determined, at first, randomly select a path to guarantee the anonymity of anonymous path, a random path. Performance of this path is unknown, although it may be able to guarantee certain anonymity. Performance might be so poor, that the user cannot bear. Thus NSDA algorithm selects neighbor nodes based on a random path, and then construct topology, namely local topology C '(as shown below), so that a new path within the scope of the topology can be selected based on Dijkstra algorithm. This new path is not only a simple path, but also is a local topology optimal solution, that is, the local topology best suited to the needs of the anonymous user. Performance of the new path must be no worse than random paths, if a new path with the same original random path, then prove that the random path is the best path within the scope of the topology. Although not all of the algorithm obtained are optimal topology of C, for the global topology is very time-consuming, and it is usually difficult to obtain a global network topology in P2P networks. Adding online time factor to the choice of standard algorithms added online time, to some extent, ensuring the stability of the chosen path, so that within a certain period of time, the path is stable.

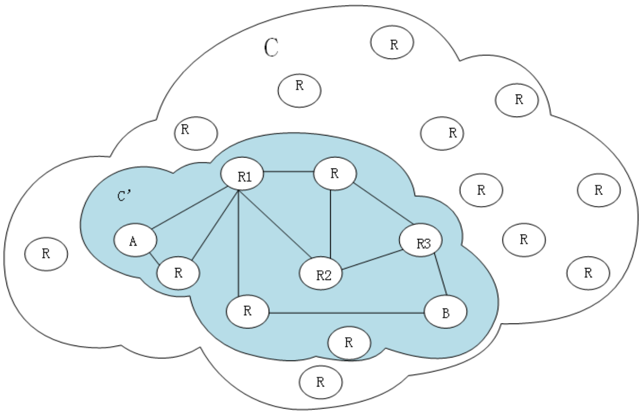

Figure 8. Comparison between Local Topology and Global Topology

4.1.2. Flexible Node Balancing

The algorithm balanced node and link attributes, allowing users freely to set the two properties based on different preferences and applications, such as instant messaging users may favor low latency communication, so choose link attributes (delay) weighted connected graph, and file transfer users may tend to good communication, care less about file transfer time, so choose a weighted node attributes connected graph. In short, users can freely change settings to adjust the performance of anonymous communication systems. This not only meets needs of users, but also reduces the traffic and communication delays in anonymous network

4.1.3. The Number of Paths Nodes Taking Part

In NSDA algorithm, the number paths nodes taking part added to the attribute parameter, this parameter can improve effective use of the network node resources. Giving link congestion may occur in network, result in delays and other problems. From the perspective of anonymity, selected node has not participated in the anonymous path yet, firstly can take full advantage of network resources, increasing network utilization, secondly unlikely to cause attention of attacker, thus attacker is not easy to pinpoint their targets. Thirdly, it effectively avoid network "hot spots", and reduce the possibility of significant increasing load nodes, even overload conditions, and load imbalances.

In addition, through the definition of nodes trust degree, and the reflection shows in node weights, recording the performance of nodes, and thus influence nodes selection. Node trust degree will avoid malicious nodes selection, but also avoid only capturing resources from the network, rather than contributing own resources.

4.2. Simulation Tool

To evaluate the performance of the NSDA algorithm, this paper, based on the simulation environment NS3, design a mechanism for the complete implementation of NSDA algorithm, this section mainly describe relative implementation method and strategies. The mechanism contains some predefined parameters, different data structures and components. By running this mechanism and collect operating results, we can have objective judgement about design and performance NSDA algorithms. This will help to guide future work, such as how to improve the algorithm performance, or add other attributes. In addition, the simulation algorithm runs NSDA may expose a few issues, which helps developers find design flaws and make modifications.

NS3 (Network Simulator 3) is an open source discrete event simulation tool, mainly for network research, which implements the abstract network elements can simulate TCP, IP, UDP, and other network protocol. NS3 and NS2 basically the same, but in NS2 need to use Tcl language topology, and NS3 topology can be fully realized by C ++, this system achieve more consistent, easy to customize modified. Besides, the NS3 architecture is clearer than NS2, more comprehensive and accurate simulation of the underlying network, so this article choose NS3 as a simulation tool.

In NS3, there are the animation or text two available ways to describe the simulation results. For these two describing ways, NAM (Network Animator) and XGraph are default tool used in conjunction with NS3. XGraph is interactive analysis which is based on the simulation results of text, NAM is a graphical analysis which is based on the simulation results of animation. Although the NAM is also interactive, but it can read the results by the end of the simulation, and simulate the real-time simulation of effects, such as the working status of nodes, the process of packet delivery on the link, and traffic change on the link, or even packet loss. Compared with the existing popular mapping tools, XGraph interface is aging and function is relatively simple. Therefore, we choose another drawing software Origin, Origin is currently world recognized scientific standard data analysis and mapping software. In this paper, using Origin software to analyze experimental data generated under different network topologies.

4.3. The NSDA Simulation Framework

The simulation experiments was under Ubuntu system using simulation software NS3. Given the common of random path, this experiment was running under a random path between a sender and receiver. After the simulation needs had been determined, according to this requirement, we designed the following simulation architecture:

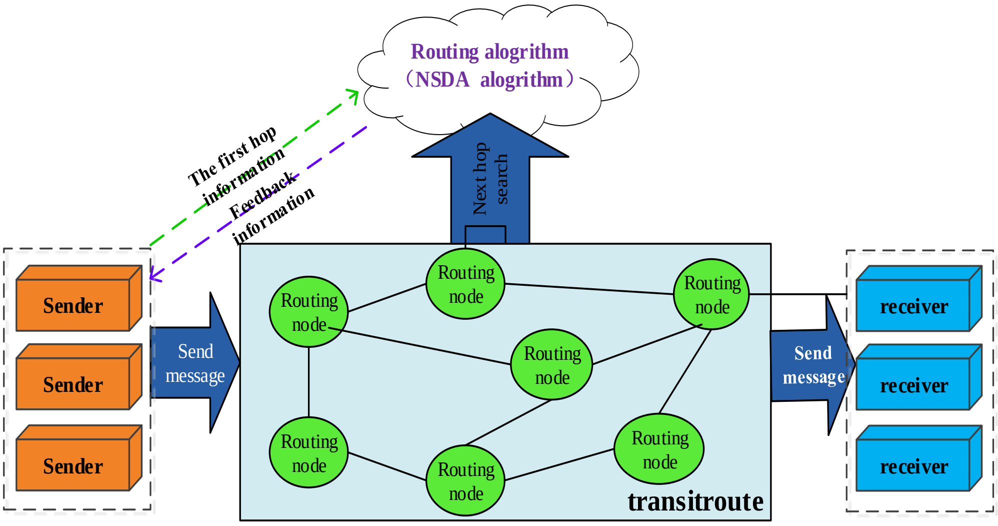

Figure 9. The NSDA Simulation Framework

First, based on the random path between the sender and receiver The NSDA simulation framework, we can determine the local topology of the network from the neighboring node matches NSDA algorithm configured (shown in Figure 9). According NSDA algorithm then generates a new path, afterwards send bank the new generation of first-hop path R1 to the sender.

Then, the sender sends a message to the first hop R1, after R1 have received the message, the NSDA PATH search algorithm generated to find the next hop R2, sequentially transmitted until a message is sent to the recipient. After the architectural design accomplished, we classified the entity architecture design based on anonymous communication as sender module, receiver module, intermediate node modules and message modules.

(1)Sender module

Sender module that is designed for the sender, message is sent by sending a timer at regular intervals to support the next trigger. In this experiment, given a set time interval has nothing to do with the measurement of the experiment, so we set send messages in every50 milliseconds, and send 1000 packets.

```
   Sender::Sender()
   sockets(),
   totalToSend(1000),
   alreadySend(0){}
   void Sender::Send (void){
      MHeader head;
      head.SetSequence (alreadySend);
      head.SetsTime(Simulator::Now().GetMicroSeconds());
      Ptr<Packet> p = Create<Packet> (50);
      p->AddHeader (head);
      int nextjumpId = Algorithm::GetInstance()->GetNext(id);
   if ((sockets[nextjumpId]->Send (p)) >= 0){
         ++alreadySend;
         NS_LOG_INFO (Simulator::Now ().GetSeconds ()<<": -[" <<id<<"] send
packet["
         <<alreadySend-1<<"] to ["<<nextjumpId<<"]");}
   if (alreadySend < totalToSend) {
         Simulator::Schedule (MilliSeconds(50), &Sender::Send, this);}}
```

(2) Receiver module

Receiver module is designed mainly for the receiving node, and it contains relatively simple function, mainly for receiving data and recording time stamp data to calculate the link delay.

```
   void Receiver::Rcv(Ptr<Socket> socket){
       Ptr<Packet> packet;
       Address from;
      while ((packet = socket->RecvFrom (from))){
         if (packet->GetSize () > 0){
             MHeader head;
             packet->PeekHeader(head);
             uint32_t Sequence = head.GetSequence();
             uint64_t stime = head.GetsTime();
             uint64_t rtime = Simulator::Now().GetMicroSeconds();
            NS_LOG_INFO (Simulator::Now ().GetSeconds ()<<": +["
 <<id<<"] rcv packet["<<Sequence<<"]");
   info.insert(std::make_pair(Sequence,std::make_pair(stime,rtime)));}
```

(3)Intermediate node

Intermediate node in anonymous path mainly implemented forwarding function, after receiving data from previous hop, to find the next hop based on NSDA algorithm. Then the data can be sent to the next hop. In case of the nodes may be malicious, if it is malicious node, the system anonymity of anonymous system will be decreased. Therefore, in experiment, we set some nodes as malicious nodes and add malicious attributes to define nodes in previous. If the value of malicious attribute is true, it means that the node is marked as malicious nodes, otherwise normal node. When message was forwarded in network, if it went through a malicious node, the malicious node count in message header will plus one. In this way, the number of malicious nodes involved in an anonymous path can be recorded.

(4)Message module

To obtain the data for simulation experiment, you need to set a specific type of message. In this design experiment, the data is carried in the form of message header, wherein data is divided into separate two kinds: time stamp and malicious nodes. In order to obtain accurate data under running system, header design is shown as below.

Message sequence number	Sending time	Receiving time	Malicious nodes number

Figure 10. The Form of Message Header

Among them, timestamp is the time of sender module sending message and the receiver module receiving time.In message header, adding malicious attribute is to measure the probability of malicious node taking part in anonymous path.

4.4. Results of Simulation and Analysis

After designed the above modules framework, we need to build a network topology to realize NSDA algorithm. Giving that the NSDA algorithm is based on a P2P network, so we don't need a directory server, etc. In the network all nodes are equal, in this paper, we set up a network topology which contained 50 nodes, and the topology is disordered. First, we set the random path length n, construct a path of length n. In this experiment, we set the path length is 10. Then, set the neighbor node number is k, i.e. the degree of each node is k. To measure the connectivity influences on algorithm, we respectively set k as 2, 5 and 10.

Finally, set the time delay for each link, in view of the diversity of nodes in the P2P network, in this experiment, we set the delay of each link randomly, the range is limited to 0.5ms to 15ms.

Having accomplished the above-described structure, for the simulation of different neighbor nodes influencing on algorithm, we pick three types network topology, i.e. k as 2, 10 and 50. Obtained test data by sending 1000 packets respectively. Using Origin to analyze data and, we have the following results:

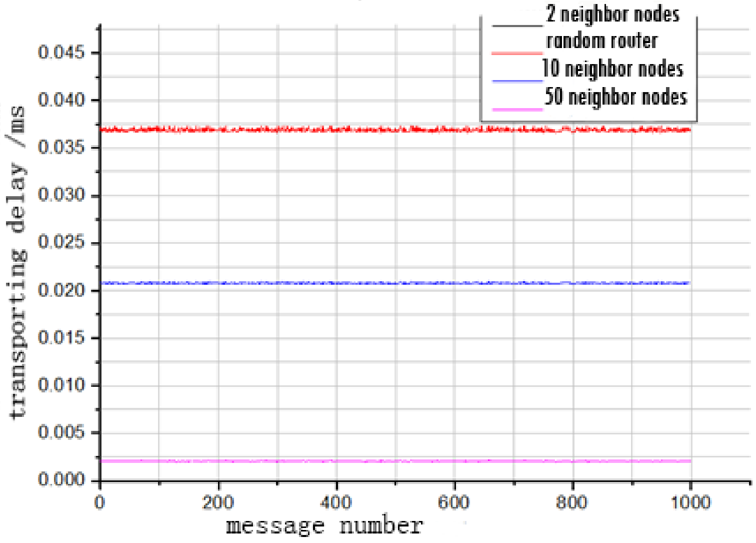

Figure 11. Different Nodes Compared with Neighbor under 100 Nodes

5. Conclusion

The limited resources and routing efficiency both are the core issues in the anonymous P2P network. Since node resource question is the core issue of anonymous P2P networks, and routing algorithm is the core algorithm anonymous affect system performance, we mainly focuses on the discovery of particular node(node selection) resources, then found more suitable node resources to build anonymous path. Moreover, we proposed and implement a novel node-select algorithm based on Dijkstra algorithm, named NSDA algorithm. The algorithm can combine the node properties with link properties to select node, which can adjust the system performance and anonymity. In order to evaluate the NSDA algorithm, we have done experiments in Network Simulator 3, and we also explain the reason why we choose Network Simulator 3 as the simulate tool. In the end, we analyze the characteristics of NSDA algorithm according to the results of experiments.

(1) When there were only two neighbor nodes, the NSDA algorithm cannot applied. In the experiment, if it cannot choose a path through NSDA algorithm, then choose random nodes on the default path. As shown, the selected path is same as initial default path, obviously, NSDA algorithm is more appropriate for complex network.

(2) Through Figure.11 contrast, we can find that the larger the node degree, the lower path delay under NSDA algorithm, and significantly lower than the random path delay. Apparently, NSDA algorithm performed better in complex P2P network.

Acknowledgements

This work is supported by the following programs: the National Natural Science Foundation of China under Grant No.61170273; 2010 Information Security Program of China National Development and Reform Commission with the title “Testing Usability and Security of Network Service Software”.

References

[1] D. Chaum, “Untraceable electronic mail, return addresses, and digital pseudonyms”, Communications of the ACM, vol. 24, no. 2, **(1981)**.

[2] P. Mittal, M. Wright, N. Borisov and G. Danezis, “Anonymous Communication Using Social Networks”, In the Proceedings of the 20th Annual Network and Distributed System Security Symposium (NDSS2013), San Diego, California, USA, **(2013)**.

[3] R. Küsters, T. Truderung and Andreas, “Vogt. Formal Analysis of Chaumian Mix Nets with Randomized Partial Checking”, Proceedings of the 2014 IEEE Symposium on Security and Privacy, **(2014)**.

[4] C. Gülcü and G. Tsudik, “Mixing E-mail with BABEL”, Network and Distributed Security Symposium (NDSS’96), **(1996)**, pp. 2-16.

[5] G. Danezis, R. Dingledine and N. Mathewson, “Mixminion: Design of a Type III Anonymous Remailer Protocol”, Proceedings of the 2003 IEEE Symposium on Security and Privacy, **(2003)**.

[6] U. Möller, L. Cottrell, P. Palfrader and L. Sassaman, “Mixmaster Protocol - Version 2”, Draft, **(2003)**.

[7] T. Metrics, https://metrics.torproject.org/bubbles.html#country

[8] C. Diaz, S. J. Murdoch and C. Troncoso, “Impact of Network Topology on Anonymity and Overhead in Low-Latency Anonymity Networks”, Proceedings of the 10th Privacy Enhancing Technologies Symposium (PETS 2010), Berlin, Germany, **(2010)**.

[9] G. Danezis, “Mix-networks with Restricted Routes”, Proceedings of Privacy Enhancing Technologies workshop (PET 2003), **(2003)**, pp. 1-17.

[10] M. J. Freedman and R. Morris, “Tarzan: A Peer-to-Peer Anonymizing Network Layer”, Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS 2002), Washington, DC, **(2002)**.

[11] M. Rennhard and B. Plattner, “Introducing MorphMix: Peer-to-Peer based Anonymous Internet Usage with Collusion Detection”, Proceedings of the Workshop on Privacy in the Electronic Society (WPES 2002), Washington, DC, USA, **(2002)**.

[12] R. Dingledine, N. Mathewson and P. Syverson, “Tor: The Second-Generation Onion Router”, Proceedings of the 13th USENIX Security Symposium, **(2004)**.

[13] C. Grothoff, “An Excess-Based Economic Model for Resource Allocation in Peer-to-Peer Networks”, Wirtschaftsinformatik, **(2003)**.

[14] M. Reiter and A. R. Crowds, “Anonymity for Web Transactions”, ACM Transactions on Information and System Security vol. 1, no. 1, **(1998)**.

[15] M. Herrmann and C. Grothoff, “Privacy Implications of Performance-Based Peer Selection by Onion Routers: A Real-World Case Study using I2P”, Proceedings of the 11th Privacy Enhancing Technologies Symposium (PETS 2011), Waterloo, Canada, **(2011)**.

[16] M. Dorigo, V. Maniezzo and A. Colorni, “Ant system: optimization by a colony of cooperating agents”, Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions, vol. 26, no. 1, **(1996)**, pp. 29-41.

[17] D. J. Watts and S. H. Strogatz, “Collective dynamics of ‘small-world’networks”, nature, vol. 393, no. 6684, **(1998)**, pp. 440-442.

Authors

Tian-Bo Lu, he was born in Guizhou Province, China, 1977. He is an Associate professor in School of Software Engineering, Beijing University of Posts and Telecommunications, China. His technical interests include information security and computer network.

Jiao Zhang, she was born in Shandong Province, China, 1991. She is a graduate student in School of Software Engineering, Beijing University of Posts and Telecommunications, China. His technical interests include information and network security, anonymous communication.

Ling-Ling Zhao, she is a graduate student in School of Software Engineering, Beijing University of Posts and Telecommunications, China. Her technical interests include Cyber-Physical System and P2P network.

Yang Li, he was born in Hunan Province, China, 1978. He is a PhD and his technical interests include information security, distributed computing and P2P network.

Xiao-Yan Zhang, she was born in Shandong Province, China, 1973. She is an Associate professor in School of Software Engineering, Beijing University of Posts and Telecommunications, Beijing, China. Her technical interests include software cost estimation and software process improvement.