A Redundant Traffic Load Routing Policy in Complex Network

Ling Xing¹,², Qiang Ma¹,², Li Wei¹

1. School of Information Engineering, Southwest University of Science and Technology, Mianyang, China
2. Robot Technology Used for Special Environment Key Laboratory of Sichuan Province, Mianyang, China

E-mail: xingling_my@163.com, maqiang_my@163.com, weili_my@163.com

Abstract: To tackle the network congestion issue caused by redundant traffic, a redundant traffic load routing policy is proposed. During the network dynamics, nodes can be aware of the traffic redundancy and take it into consideration to decide paths choosing. Nodes in the network transfer packets by trading off the shortest path length and the edges’ dynamic information flow in order to maximize network communication capacity. The processing ability of a node is proportional to its weighted value, which is defined as its nodes’ degree. Theoretical analysis shows that network phase transition point is the hybrid effects of network edges’ bandwidth and nodes’ processing capability. Simulation is carried out in BBV weighted network and results show that this novel routing policy can avoid choosing nodes which are more vulnerable to congestion. It can alleviate nodes congestion and improve the network overall throughput.

Keywords: routing policy; redundant traffic; network congestion; complex network.

1. Introduction

With the rapid growth of network information and fast expansion of network scale, network traffic congestion has become a common issue. It becomes the main obstacle for network development. Many reasons simultaneously cause network congestion, e.g., network nodes’ ability to process packets, link bandwidth and network traffic redundancy [1-6]. The network redundant traffic is made up of repeated requested traffic flow in specific time duration [7-12]. Redundant traffic occupies large network bandwidth and decreases communication inefficacy, especially in complex networks where nodes and network topology evolve simultaneously. Complex networks which suitably models real networks such as Internet, social networks and communication networks, have attracted attentions from many researches.

Many of current routing researches are from the perspective of optimizing network communication efficiency and the routing policy is studied on the scale-free complex networks. The classic shortest path routing algorithm optimizes packets’ transmission path but neglects network congestion brought by nodes which have large traffic volume to handle, thus it is not able to deal with redundant traffic properly. Echenique in [13] proposed a shortest path and dynamic load combined routing algorithm for complex networks but treating that all the nodes have the same capability to handle packets. Then researches in [14-16] improved the routing policies by assuming nodes’ ability to handle packets is proportional to nodes’ degree or betweenness. Xiang [17] focused on the effect of link bandwidth distribution on transmission capacity and network overall performance and proposed network link bandwidth based routing method, which is obtained by multiplying network average node degree by network scale.

Both the dynamic algorithms and combined routing policies improve the efficiency of network transmission in some degree. However, they are based on the scale-free network and not the weighted network. The weighted network is more akin to real-life network. Chen [18] defined a kind of shortest path in weighted network and proposed packets routing method based on the weighted network. Wang [19] improved the network traffic throughput by building a novel weighted network with tunable clustering coefficients in which nodes with stronger degree have privilege to transfer packets. However, those works do not take a look at redundant traffic and performance of the routing in the network can be improved.

We consider the network as a weighted network with scale-free characteristics and focus on packets traffic routing policy which is capable to sense information flow. Also we systematically research on the congestion problem by studying the hybrid effects of nodes’ ability and link bandwidth in complex network. Currently, most works on routing policy assume the initial network state is load free, which means the network is in idle state and no packets transmission happens. However, we do not consider traffic load as zero during the idle
state but incorporate redundant traffic into the network. Thus even in initial state the network has some traffic and we define that traffic as initial redundant traffic load. In real life network, redundant packets flow count for most of the network bandwidth. When data are transmitted over the network happens, exist of redundant traffic worsens network congestion problem. In this paper we propose a redundant load traffic routing policy in complex network to alleviate the congestion by traffic redundancy.

The paper is constructed as follows. In the next section the routing policy is explained in detail and in section 3 we conduct theoretical analysis on the network phase change for the proposed routing policy. In section 4 we present out simulation result and we conclude our work in section 5.

2. Evolution process of routing policy

Nodes in our network model represent routers or hosts in communication network. All nodes have the ability to generate, transmit, accept and handle data packets. In reality nodes differ in their abilities to handle information traffic and usually nodes with higher accessing times have better ability. The ability of node $i$ to transmit packets in unit time is denoted by $c_i$, which means the node can transmit packets as many as $c_i$ to the next routing node. In our routing policy the value of $c_i$ relates to the weighted value of node $i$, which is represented by $s_i$. The relation between node’s ability and its weighted value is $c_i = \max(s_i, 1)$ and it means that nodes with higher weighted value can process packets more easily.

The dynamic evolution process of our routing policy is as follows:

Step 1) Network initiation. Assign edges in the weighted network values as redundant traffic, i.e., the redundant load. The network starts with load traffic and the redundant traffic counts for the initial network bandwidth in some degree.

Step 2) New data packets generation. In every time step, new $R$ data packets are generated in the network. Source nodes and destination nodes for the newly generated packets are randomly chosen from previous network nodes. The new packets are appended to the rear of source packets queue to be transmitted. Also data packets transmitted from other nodes are sequentially queued in the rear of the new destination nodes. We assume the length of queue as unlimited. Thus every queue of the nodes consists of new generated packets and the packets transmitted from other nodes.

Step 3) Routing decision. When source node transmits data packets, it firstly searches its neighbor nodes to check if destination node exists. If it exists then source node transfer the data packets to the destination node and those packets disappear in the network as they have been handled. If it does not exist, then it transfers the data packets to neighbor node with the least trade-off value $\eta$. Thus when data packets arrive in the new neighbor nodes, they search for new destination nodes of the currently arrived nodes until they find their destination nodes.

The trade-off value $\eta$ is a hybrid of two parts, namely the length of shortest path and the node’s ability of processing packets. It is obtained as in Eq. (1), where $\eta_i$ is the value for node $i$ and the parameter $\mu$ is an adjusting parameter which falls into the range $[0, 1]$.

$$\eta_i = \mu d_i + (1 - \mu) \frac{q_i}{s_i}$$

where $d_i$ is length of the shortest path from node $i$ to other nodes in the network, and $q_i$ is the length of the queue of waiting data packets of node $i$. $s_i$ is the sum of weight value of node $i$’s all neighbor nodes and is shown in Eq. (2).

$$s_i = \sum_{j \in \tau(i)} w_{ij}$$

where $\tau(i)$ is a set consisting of all neighbor nodes of node $i$ and $w_{ij}$ is the weight value for the edge from node $i$ to node $j$. When $\eta$ equals one, our policy turns out to be the shortest path policy. Thus the shortest path routing policy can be seen as a specific example of our routing method.

Step 4) Data packets transmission. Redundant traffic in the network is defined as data packets with the same source and same destination. For simplicity we treat packets as similar packets only if both their source and sink address are the same. When the redundant packets transfer over the edges, weighted value of the underlying edges increases. Data packets are eliminated from the network once they arrive at the destination nodes.

The proposed routing policy considers the hybrid effects of length of shortest path and the traffic flow, as is seen in Eq.(1). At the routing initial step, trade-off values for all nodes are calculated. When data packets are to be decided to route, the neighbor node with the least value $\eta_i$ is chosen for next transmitting node.
networks, the degree value of a node is the sum of all edges of its neighbor nodes and it reflects the degree of importance of the node. The greater the value is, the more importance it poses, e.g., the hub nodes in real-life networks. We refer to those nodes with greater degree value as central nodes. Central nodes receive more packets to process. Consequently, the length of queue of waiting packets increases and it may exceed the limit of packets processing of central nodes. So we make the value of $\eta_i$ increase when $q_i$ increases, which means nodes with increasing length of queue has relatively less probability to be chosen as a candidate to route. Meanwhile, nodes with smaller value $\eta_i$ may be considered as routing nodes, which have greater value $s_i$ and smaller length value $q_i$. In this way our policy can sense the information flow of edges and improve the routing performance.

3. Theoretical analysis

Network performance can be measured by the process ability of generation, transmission and processing of all data packets in the network. New data packets are generated at every time step and let $R$ denote the packet generation rate. When $R$ increases gradually, the network transits from free phase into congestion phase once value $R$ reaches at a critical value. The data packet generation rate at the critical phase is represented by $R_c$. Also $R_c$ states the network data packets throughput or maximum load capacity. Then network is in congestion or not whether $R$ exceeds $R_c$.

The congestion status of network is described by parameter $H$, which is calculated as follows:

$$H(R) = \lim_{t \to \infty} \frac{w(t)}{Rt}$$  \hspace{1cm} (3)

where $w(t)$ is total number of data packets in the network at time $t$. When $R \leq R_c$, the number of arrived packets nearly equals the newly generated packets and network is in a stable state with $H \approx 0$. However, when $R > R_c$, the new packets exceeds the newly arrived packets and network congestion occurs. The value of $H$ increases with the generation of new packets. Congestion status is more obvious when $H$ is greater.

3.1 Network throughput based on nodes congestion

Betweenness is used as an effective mean to describe centrality of network [20]. For the network with constant processing ability of all nodes, the parameter $R_c$ is estimated by betweenness. On the contrary for the network with variable processing ability, $R_c$ is estimated by effective betweenness. In our complex network, we consider the processing ability as different values and choose effective betweenness for $R_c$ calculation. Node effective betweenness is defined in Eq.(4), where $b_{v}^{\text{eff}}$ is the effective betweenness for node $v$, $\sigma_{sd}(v)$ is the number of paths from node $s$ to node $d$ and $\sigma_{sd}(v)$ is the number of paths going through node $v$ from node $s$ to node $d$. At every time step, the number of packets arriving at node $i$ is $Rb_i / N(N-1)$ and $b_i / N(N-1)$ is the probability of a packet arriving at the node with betweenness $b_i$. $N$ is the total number of nodes in the network. Let $C_i$ denote the processing ability of node $i$. If $Rb_i / N(N-1) > C_i$, remaining packets which can not be processed accumulates at node $i$. Then it causes network congestion.

$$b_{v}^{\text{eff}} = \sum_{s,d} \frac{\sigma_{sd}(v)}{\sigma_{sd}}$$ \hspace{1cm} (4)

In order to avoid congestion, all nodes in the network should satisfy the condition that $Rb_i / N(N-1) \leq C_i$, that is

$$R \leq \frac{C_{\text{min}}N(N-1)}{b_{v}^{\text{eff-max}}}$$ \hspace{1cm} (5)

where $C_{\text{min}}$ is the minimum of all nodes’ processing ability and $b_{v}^{\text{eff-max}}$ is the maximum of all nodes’ effective betweenness. The network throughput based on nodes congestion denoted by $R_{c\text{-node}}$ is calculated in Eq. (6).

$$R_{c\text{-node}} = \frac{C_{\text{min}}N(N-1)}{b_{v}^{\text{eff-max}}}$$ \hspace{1cm} (6)

3.2 Network throughput based on edges congestion

Network traffic flow status on the edges can be described by edges betweenness. Similar to node effective betweenness, edge effective betweenness is defined as in Eq. (7).
where $b^e_{ij}$ denotes the edge effective betweenness for the edge $l_{ij}$ which is starting from node $i$ to node $j$, $\sigma_{sd}$ is the number of paths from node $s$ to node $d$ and $\sigma_{sd}(l_{ij})$ is the total number of path going through edge $l_{ij}$ from node $s$ to node $d$.

Let $G_{ij}$ denote the usable bandwidth on the edges $l_{ij}$. When the traffic on the edge is less than the edge’s usable bandwidth, no congestion happens. Otherwise, not all packets on the edge can be transferred and congestion occurs. The probability of a packet going through edge $l_{ij}$ is $b^e_{ij} / N(N-1)$. As the deduction of Eq. $(7)$, the network throughput based on edges congestion is calculated in Eq. $(8)$.

$$R_{e\text{-edge}} = \frac{G_{\text{max}} N(N-1)}{b^e_{\text{max}}}$$

where $b^e_{\text{max}}$ is the maximum of all edges’ effective betweenness and $G_{\text{max}}$ is the minimum of all edges’ usable bandwidth.

3.3 Network throughput

We combine both throughputs on nodes and edges, and then the network throughput is defined in Eq. $(9)$.

$$R_e = \min (R_{e\text{-node}}, R_{e\text{-edge}})$$

Thus value of $R_e$ is the minimum of $R_{e\text{-node}}$ and $R_{e\text{-edge}}$, which means our data packets routing policy considers the hybrid effects of both nodes’ processing ability and the edges’ bandwidth.

4. Simulation and results

We perform our simulation over the BBV weighted scale-free network [21]. First, we experimentally obtain the optimum parameter value for trade-off $\mu$. Then values for $R_e$ under different scenario is calculated. The weighted values of edges in the network are assumed as the initial redundant traffic load. Meanwhile every data packet when transmitted needs one bandwidth unit. The parameter $H$ is calculated every 50 time steps and corresponding parameter $R$ runs 10000 time steps.

4.1 Choose of optimum parameter $\mu$

Figure 1 draws the network phase transition points $R_e$ for different values of $\mu$. When $\mu$ equals zero or one, the throughput of network traffic declines to the lowest. While $\mu$ is in the range $(0, 1)$, network throughput increases gradually. It nearly converges to a stable value along the increase of value $\mu$. In the network with redundant traffic, $R_e$ reaches its highest value when $\mu$ falls in the range $[0.5, 0.9]$. Meanwhile for network without redundancy, $R_e$ gets its highest value when $\mu$ is in the range $[0.7, 0.9]$. Therefore value of $\mu$ in $[0.7, 0.9]$ can guarantee the maximum throughput of network whether with redundancy or not. We choose $\mu$ to be 0.8 for the following simulation.

4.2 Network throughputs for two kinds of congestions

We calculate the theoretical value for $R_e$ under two statuses and compare them with simulation results. In our BBV network, there are 1000 nodes and we set minimum of nodes’ processing ability to 5 and the $b^e_{\text{max}}$ obtained is 8056. By Eq. $(6)$ the $R_{e\text{-node}}$ is 620. Similarly, we set minimum of all edges’ usable bandwidth to 2 and the network bandwidth to 10. Network bandwidth, denoted by $B$, is the available bandwidth for every edge

$$\min(0, 0) = 0$$

Figure 1. Relation between $R_e$ and $\mu$
and it is the sum of usable bandwidth and used bandwidth. The $b^{\text{eff}}_{\text{max}}$ from the BBV network is 6054. Then by Eq. (8) the $R_\text{c-edge}$ is 330. Therefore from Eq. (9) $R_c$ is 330.

Figure 2 draws the network congestion status for two routing policies, i.e., the shortest path routing (SPR) and our proposed redundant load routing (RLR). Value B is set to 10 and it means the effect of network traffic congestion caused by edges is much more obvious than that by nodes. The theoretical value for $R_c$ is 330 while the simulation value for $R_c$ is about 350. Also value of $R_c$ for SPR is much less than the corresponding value for RLR. This is due to that our policy considers the real-life redundant traffic and it improves network throughput.

![Figure 2. Congestion Status for SPR and RLR with bandwidth 10](image)

We then conduct our simulation for a larger bandwidth. B is set to 100 and the result for $R_\text{c-edge}$ is 710. Then $R_c$ is 620. The congestion status for parameter H is shown in Figure 3. The simulation value for $R_c$ is around 650. Contrary to the above setting, the restricting factor for the network throughput is the nodes’ processing ability. For these two scenarios, the theoretical calculation for $R_c$ are nearly equal to the simulation results.

![Figure 3. Congestion Status for SPR and RLR with bandwidth 100](image)

### 4.3 Network throughputs with or without redundant traffic load

We show the network congestion status for routing methods with and without redundancy. Figure 4 and Figure 5 draw the congestion results for network bandwidth of 10 and 100, respectively. It can be seen from these that the routing algorithm with redundant traffic load consideration can shift right the network phase transition point and thus improve the network traffic throughput. Also the blue line which denotes the routing policy with no redundancy always lies above the red line which denotes the one with redundancy after network phase change. It means that exist of redundant traffic strengthens the congestion of network traffic and the elimination of redundant traffic improves network performance.

![Figure 4. Congestion Status for network with and without redundant load when B=10](image)
4.4 Network throughput for different bandwidths

The network congestion status with packet generation rate of proposed routing policy is shown in Figure 6, where bandwidth of 10, 100 and 500 are drawn. It can be clearly seen that with the increase of bandwidth the point of $R_c$ is shifted right. It justifies that to increase bandwidth can alleviate the network congestion problem.

When bandwidth is small, the increase of bandwidth has obviously positive effects on the network throughput improvement. Since the restriction by network bandwidth is much stronger than the network nodes’ processing ability. The small value of bandwidth cannot guarantee the successful transmission of a large number of network packets and those packets accumulate on certain nodes, which in turn decreases the processing ability of those nodes and causes nodes’ packets congestion. While bandwidth increases to a certain range, the bandwidth has relatively smaller effects on network performance than nodes’ processing ability. It can be proved in Figure 6 that $R_c$ shifts right at about only 50 units between lines of bandwidth 10 and bandwidth 500, while $R_c$ shifts right at 300 units between lines of bandwidth 10 and 100. Then it states that to increase network bandwidth is an effective mean to improve network communication efficiency.

5. Conclusions

Complex networks have been the ideal model of many real-life networks. Motivated by the congestion problem of network traffic flow, we have proposed redundant traffic load routing policy. Two factors, i.e., network edges’ bandwidth and nodes’ processing capability, are combined into our method. At the network initiation, redundant traffic is previously loaded to simulate the real-network. In the following steps, packets with same source and destination addresses are regarded as redundant traffic. The routing process is conducted on the basis of a trade-off between nodes’ ability and shortest path parameters.

The most advantages of our method are that our policy can avoid choosing of nodes with higher congestion status and routing process can be aware of network bandwidth and network nodes’ capability. We have simulation on the BBV scale-free network and have found out that the optimal trade-off value is 0.8. By varying network bandwidth parameters we have also shown that network phase transition is affected by both bandwidth and nodes’ processing ability. By comparing network with redundant and without redundant consideration, routing policy with redundant traffic load can ease traffic congestion problem and improves network performance.

Acknowledgment

This research is partially supported by National Nature Science Foundation of China (Grant No. 61171109), Applied Basic Research Programs of Sichuan Science and Technology Department (Grant No. 2014JY025, No.
References


