We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

5,200
Open access books available

129,000
International authors and editors

150M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Reducing Routing Loops Under Link-State Routing in Wireless Mesh Networks

Takuya Yoshihiro and Masanori Kobayashi

Additional information is available at the end of the chapter

1. Introduction

In Mobile Ad Hoc Networks (MANETs), one important issue is how to provide stable communication between nodes against transition of network state such as node mobility or quality transition of communication links. In MANETs, due to the fragile nature of wireless links, routing protocols are designed to be more robust and resilient against failure, while restraining the network load of control messages even if nodes are distributed densely. Four routing protocols, i.e., AODV[1], DSR[2], OLSR[3], TBRPF[4] have been standardized so far. Although each of which has its own mechanism that is convenient for MANET, they do not still displayed a sufficient performance to be applied in practice.

One of the drawbacks in these routing protocols is that they do not consider the transition of link quality in their process of computing forwarding paths; they use hop counts as their basic criteria to compute forwarding paths of packets. However, the transition of link quality including link failure is not avoidable in MANET because its basic requirements include node mobility and wireless links. To take this important characteristic into account, many routing metrics have been proposed that quantifies link quality from various points of view [6]-[11]. For MANETs in which mobility is included in general, the event that affects the most on communication performance is link failure due to mobility. Thus, most of the proposed link metrics tries to quantify the probability of link failure by way of measuring node speed, RSSI (Received Signal Strength Indication), and so on [6]-[11]. For Wireless Mesh Networks (WMN) [5] in which nodes are stationary, because the risk of link failure is far smaller than MANET, the link quality that should be quantified is the quality of communications such as communication speed, delay, and stability of links [12]-[16]. For example, ETX (Expected Transmission Count) [12], which is one of the most commonly used link metrics, quantifies the average transmission count in 802.11 MAC computed from success ratio

of MAC transmission, and ETT (Expected Transmission Time) [13] extends ETX to quantify the average transmission time of a MAC frame in the link.

As far as proactive link-state routing such as OLSR is concerned, it is well understood that introducing dynamic link metrics make networks far robust and resilient, and consequently improve performance of networks in practical situations. However, simultaneously, such dynamic metrics cause communication paths to be changed frequently. Note that the paths flapping behavior is not always bad, because it is the result of continuous effort of routing protocols to find better quality paths. Nevertheless, it certainly increases the risk of several inconvenient phenomena such as packet looping.

Packet looping is one of the very harmful problems because looping packets travel along the same link repeatedly and consume significant capacity of the network. In general, larger number of looping packets appears when the network topology changes including link metrics more frequently. In other words, dynamic metrics by nature involves the risk of this kind of instability in exchange for the flexibility against wireless instability. Therefore, it is one of the goals for us to reduce the harmful influence of packet loops, while simultaneously holding the flexibility brought from dynamic metrics.

Note that, in wireless multi-hop routing, there are several causes of reducing communication performance other than packet looping, and they are deeply related with one another. Not only packet loops, but also congestions due to interference, and further link failures due to wireless instability or mobility are also regarded as the essential elements that should be considered in MANET routing schemes. Especially, interference would be the most focused element in the current state of the art. However, in wireless networks, packet looping and interference are deeply related with each other so that improving performance from the viewpoint of looping would also be an important part of the contribution.

In this paper, we present a new loop reduction method for Wireless Mesh Networks, which follows the description of the literature on loop-free techniques for wired and wireless networks. This article is organized as follows. In Section 2, we first describe a concise description of proactive link-state routing protocols and related techniques in WMNs, including dynamic metrics. In Section 3, we review the literature of loop prevention methods for wired networks. In Section 4 we describe the loop prevention methods proposed for WMNs. Then, in Section 5, we present a new loop prevention method and its evaluation results. Finally in Section 6 we conclude the article.

2. Packet looping problem and its harmful influences

Packet looping is a harmful phenomenon in which packets are forwarded among the same nodes. Looping packets significantly consume resources of networks, and consequently cause severe congestion. Packet looping traditionally has been discussed in wired networks, where looping occurs typically when a link fails. Link failure triggers the process of paths re-computation in routing protocols. Then, in the transient state to converge to the new

shortest paths state, routing tables among some nodes possibly create loops temporarily. This kind of looping is traditionally called as "routing-table loops" or just "routing loops."

See Figure 1 for an example of a routing loop as a result of link failure. Figure 1(a) shows an initial link metrics on the circular network with three nodes. The shortest paths from *A* (resp. *B*) to *C* computed by *A* itself (resp. *B* itself) is also indicated here. Suppose that link *(A, C)* fails. With this topology change, the shortest paths are finally changed to the state shown in Figure 1(c). However, in the transient state where only *A* knows the link failure and *B* does not know it (Figure 1(b)), *A* forwards packets destined to *C* using the next hop *B*, while *B* forwards those using the next hop *A*, then they loops between *A* and *B*. This loop continues until *B* updates its next hop nodes and consumes significant network resources of *A* and *B*.
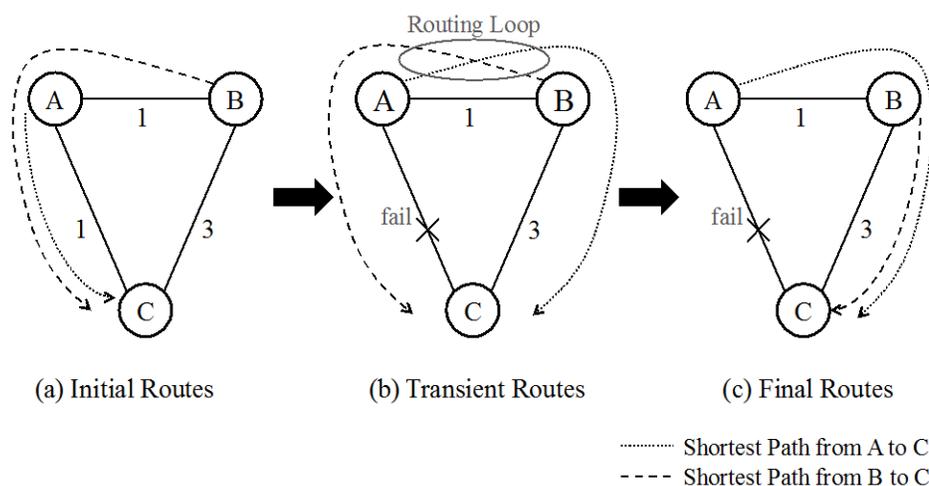

Figure 1. A Routing Loop in Case of Link Failure

Routing loops caused of failure occurs in both wired and wireless networks. However, wireless network has another type of risk for routing loops. In wireless multi-hop networks with proactive link-state routing schemes such as OLSR, it is general to deploy a dynamic metric to improve the performance of networks over instable wireless links. With dynamic metrics, frequent changes of link metrics arise to be a major cause of routing loops.

The typical example of loops caused from metric change is shown in Figure 2. In Figure 2(a), the initial metrics are shown with the same topology as in Figure 1. The shortest paths from *A* and *B* to *C* is the direct path to *C*. Suppose that metrics of links *(A, C)* and *(B, C)* are simultaneously raised to 3. In the final state shown in Figure 2(c), the shortest paths are the same as the initial state. However, in the transient state shown in Figure 2(b) where *A* knows the metric change of only *(A, C)* and *B* knows the metric change of only *(B, C)*, the shortest path for *C* computed in *A* is via *B* while the one computed in *B* is via *A*, then the packets loop between *A* and *B*. You may imagine that the situation shown in this example may be rare in practice. However, similar situation frequently occurs under dynamic metrics. Imagine the case heavy traffic suddenly appears in a network. Dynamic metrics sensitively respond to the traffic load and then many metric values would be raised simultaneously.

Note that the routing loops cause severe degradation of network performance. Speakman et al. [17] measured the impact of packet looping through their proposed method that detects looping packets at each node and drops them immediately. From the evaluation with mobility scenarios, they found the throughput of the network is improved by at most 10%, which shows the significant occupancy of resources with looping packets.

In wireless multi-hop routing, there are several causes of reducing performance other than packet looping, and they are deeply related with one another. Generally, the cause that has the largest impact on the performance would be the interference among radios. Because in wireless networks radio spreads for all directions, packet transmission on a link can be disturbed by other link's transmissions. Especially, in case of multi-hop networks over 802.11 MAC, hidden terminals significantly degrade the network performance by generating severe congestion. If such congestion is extreme, naturally wireless links are to fail. Link failure in turn invokes paths flapping, and the path flapping creates loops. Then, the loops again grow up congestions. In this way, the circulation of those harmful influences is formed. To reduce these harmful influences, it is essential to take measures for each of the causes. As one of the measures to deal with this situation, the techniques to reduce routing loops would be an important part to improve the performance of wireless multi-hop networks.
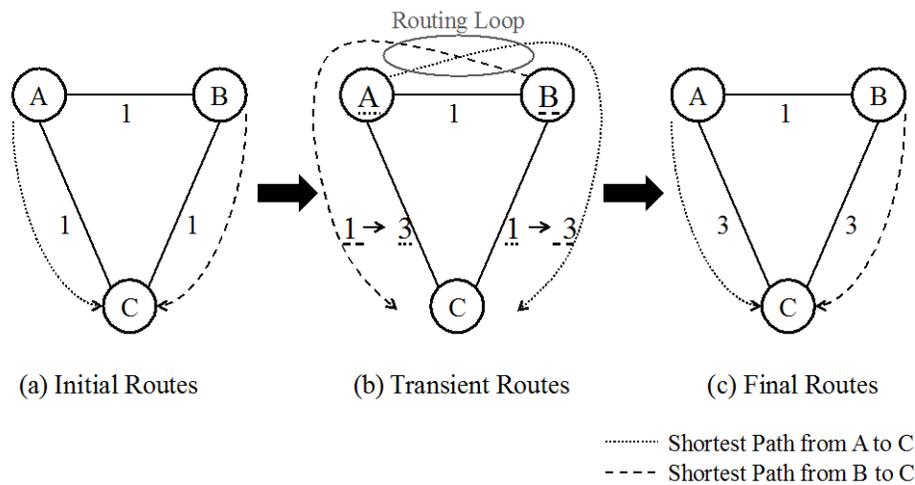

Figure 2. A Routing Loop in Case of Metric Changes

3. Literature on loop-prevention techniques for wired networks

3.1. Loop-prevention in distance-vector schemes

We start with the literature of loop prevention techniques for wired networks. To construct loop-free paths for every destination of a given network is a primary routing problem in the Internet. The most general approach for this problem is to use shortest-paths as the forward-

ing path for each destination. There are two major routing strategies, e.g., Distance-Vector schemes and Link-State schemes, which have been deployed in the Internet for a long time.

Distance-Vector routing schemes are deployed from the early stage of the Internet, and one of them is standardized as the representative routing protocol called RIP (Routing Information Protocol) [19]. RIP is based on the distributed Bellman-Ford algorithm [18]. Distance-vector routing scheme is driven with a simple mechanism: each node maintains a distance-table in which the distance for each destination is held, and advertises the distance-table to every neighbor periodically. Each node has only to choose the neighbor that has the shortest distance for each destination to construct its routing table. Unfortunately, this simple scheme has a serious problem so called the count-to-infinity problem [18]. The count-to-infinity problem occurs in case of topology change such as link failure, where the distance for a destination increases repeatedly among involved nodes until reaching the maximum distance defined in the protocol. In this period of time, packets are forwarded among those involved nodes and loop among them.

Several solutions are presented to reduce the harmful influence of the count-to-infinity problem. As a simple solution deployed in the early days, techniques so called split-horizon and poison-reverse [18] are well known. Although they can effectively reduce the affect of count-to-infinity phenomenon, the influence of the problem is not still negligible. One of the early-days solutions for this problem is to exchange full paths information in the routing scheme. This approach is currently known as Path-Vector routing, which is still used in inter-AS routing protocol BGP (Border Gateway Protocol) [21]. (Note that BGP is currently deployed not only for preventing loops, but also for several other functionalities such as controlling policies.) However, maintaining full paths information for every destination requires so much cost. To reduce the cost of control messages, Cheng et al. proposed a method to limit the information of forwarding paths to destinations that is transmitted into the network, with which they still prevent count-to-infinity problems [22]. Unfortunately, although this method mostly prevents routing loops, it cannot still eliminate temporary routing loops completely.

The first solution that prevents count-to-infinity problem without using path information was proposed by Garcia-luna-aceves, which was called as DUAL [23]. In [23], he presented locally computable sufficient conditions to be loop-free when a node changes its successor (next-hop) nodes to forward packets. Namely, when a node wants to change its successor node for a destination, it firstly checks the condition of loop-freedom. If the condition is met, it safely changes the successor node. Otherwise, it invokes a diffusing computation by sending a query to its neighbors to find a feasible successor. A neighbor node that received the query again send a query to its neighbors if it does not have a feasible successor, and when it received all the responses from its neighbors, it surely finds a feasible successor and returns the response with the successor information to the sender of the query. When the sender received the responses from all neighbors, it determines the new feasible successor. This coordination of nodes guarantees the selection of a "safe" successor to be loop-freedom at any instant. The mechanism of DUAL is implemented in the routing protocol EIGRP (Enhanced Interior Gateway Protocol) [24].

DUAL is improved in [25], in which only one-hop query processing is required, by means of using the predecessor information (i.e., information of forwarding paths to reach destinations) in the similar way as BGP. Later, Schmid et al. proposed to prevent the count-to-infinity problem without changing the message format of RIP routing protocols, if all link costs in a network is uniform [26].

3.2. Loop-prevention in link-state schemes

Link-state routing is another representative routing strategy that is also deployed from the early stage of the Internet. In this category of routing family, IS-IS (Intermediate-Systems, Intermediate-Systems) [27] and OSPF (Open Shortest Path Fast) [28] are the representative standardized routing protocols. In the link-state routing schemes, every node advertises the neighbor information (i.e., link information) to have all nodes in the network share the same image of whole network topology, and then every node computes the shortest paths on the shared network topology. Because all neighbor information is advertised through the network, link-state routing schemes require larger cost of control messages than distance-vector routing schemes.

Link-state routing schemes exchange more information among nodes, and it significantly reduces the time for path converging. Nevertheless, the risk of routing loops still remains in the face of topology changes. DUAL [23], which we described as a loop-free technique for distance-vector algorithms, again can be applied for link-state routing schemes so that link state routing works without routing loops under the diffusing computation. However, because this mechanism is based on distance-vector schemes and suitable to use on it, implementing it over link-state scheme is a little complicated. So, as a simple and feasible method to perform loop-free convergence to the new state of routing tables against single link/node/SRLG (Shared Risk Link Group) failure, Francois et al. proposed a method to control the order of updating routing tables not to create loops without any additional messages [29]. In their method, when a router computed a new successor to forward packets, it waits for a while until neighbor routers no longer select it as their successors, before updating its successor. To shrink the waiting time, simple messages to guarantee the order of updating successors can be used. Francois et al. further proposed a method to perform a planed link failure without loops by increasing a link cost gradually until no flow uses the link [30]. The mathematical analysis of the same problem is seen in [31]. In this method, they present algorithms to compute the sequence of intermediate values of link costs with which we can change a link cost to an arbitrary value without loops at any instant. With this sequence of link costs, it is possible to control the amount of traffic on the link, i.e., we can decrease or increase the traffic on a link gradually without rapid change of the link load.

4. Literature on loop-prevention techniques for wireless networks

4.1. Loop-prevention in reactive routing schemes

For wireless networks, generally two routing strategies are deployed, i.e., proactive and reactive routing. Proactive strategy takes a similar mechanism as wired networks; it always

maintains a routing table that includes next-hop nodes for all destinations. In Contrast, reactive strategy is a new approach for MANET in which network load of control messages are reduced by searching forwarding paths to create its routing table entries on demand.

Reactive routing schemes such as AODV searches a path when a request for data delivery occurs, and by that they save the network load of control messages for unnecessary destinations. Thus, the family of reactive routing schemes is suitable for the case where communication amount or destinations to deliver data are limited. Reactive routing schemes are by nature loop-free because, once the path is determined, it is basically used unless a link or a node on the path fails. However, they require a path repairing process when the path is broken, which inevitably degrades the performance. To overcome this degradation, the method called ROAM [32] is proposed that allows nodes to change their successors without any message exchange. By means of using loop-free conditions given in [23], ROAM also guarantees loop-freedom with the more flexible paths control functionality. Also, [33] presents a method to improve the efficiency of path repairing in reactive routing schemes using the loop-free condition given in [23].

On the other hand, proactive routing schemes such as OLSR always maintain a path for every destination using routing tables. Note that the majority of the proactive routing schemes proposed so far are based on link-state strategy. In the link-state routing, every node in a network advertises its neighbor information (i.e., link information). As a result, all nodes in the network share the topology of the network, from which nodes compute their routing tables. Proactive routing schemes are able to begin communication without delay of searching paths, whereas they requires constant load of control messages. Because the load of control messages in dense networks is significant, OLSR deploys an effective load reduction technique called MPR (Multi-Point Relay), which limits the relay nodes in the flooding procedure that advertises messages throughout a network.

4.2. Dynamic metrics for proactive networks

In proactive link-state routing schemes, routing tables have to be maintained so that better paths are always available over the transition of wireless link quality. However, the initially standardized routing protocols including OLSR do not take it into account because they compute the shortest paths with respect to hop-count. To achieve more flexible routing over unstable wireless links, dynamic metrics were introduced into the shortest-path routing.

For wireless mesh networks, De Couto, et al. first proposed a routing metric called ETX [12], which quantifies the average transmission count of an 802.11 link required to have packet received by the other node. Because 802.11 link requires an acknowledgement to complete a transmission, ETX of a link is computed as $ETX = \frac{1}{d_f \times d_r}$, where d_f is the success transmission ratio to the neighbor via the link and d_r is that of the reverse direction. Value d_f are computed via periodically transmitted probe packets with the formula $d_f = \frac{c\tau}{w}$, where τ is the time interval of probe packets, w is the measuring time range, and c is the number of received probe packets within time w. Value d_r is computed in the same way for the reverse

direction at the neighbor node, and is carried by probe packets to the neighbor to compute the ETX value of the link.

ETX is extended by introducing communication speed of links as ETT (Expected Transmission Time), which quantifies the average transmission time to have an 802.11 frame received via a link [13]. WCETT (Weighted Cumulative ETT) is also proposed in the same paper [13], which takes bottleneck channel affection into account to compute path metrics under multi-channel environments. Note that WCETT is not link metrics but path metrics ("link metric" here means the additive metric where a path metric is the sum of link metrics included in the path), which is expressed as $WCETT = (1 - \beta)\sum_{i=1}^{n} ETT_i + \beta Max_{1 \leq j \leq k} X_j$, where n is the number of hops of the forwarding path, k is the number of available channels, and $X_i = \sum_{Hop\ i\ on\ channel\ j} ETT_j$. By including the level of bottleneck channel affection $\beta Max_{1 \leq j \leq k} X_j$, WCETT quantifies the quality of a whole path.

Unfortunately, this path metric approach may include routing loops even in a static metric situation. For this problem, Sobrinho introduced a necessary and sufficient condition for path metrics to be loop-free in static metric situation, which is called isotonicity [14]. As a path metric that hold isotonicity, Yang et al. proposed a path metric called MIC (Metric of Interference and Channel-switching) [15], which metric values can be decomposed to the isotonic metrics in a virtual network. This characteristic enables MIC to be computed efficiently using the general shortest-path computation algorithms such as Dijkstra's algorithm.

4.3. Loop-prevention for proactive networks

It is now well understood that dynamic metrics significantly improves the network performance and the robustness against instability of wireless links. However, dynamic metrics by nature increase the frequency of changing forwarding paths, which in contrast causes instability of communications in networks. One of the significant effects introduced by dynamic metrics is the routing loop problem described in Section 2. As described there, the routing loop problem is one of the major causes of instability in wireless multi-hop networks.

One naive idea to eliminate routing loops in wireless networks is to apply the loop-free techniques proposed for wired networks. However, with dynamic metrics, it is hard to apply these loop-free techniques such as DUAL because metric change is too frequent. In such cases, diffusing computations such as DUAL requires too many requests for changing successors. Consequently, for wireless multi-hop networks, we need another lower-cost approach against routing loop problems.

As a loop aware routing scheme for MANETs, LLD (Loop-free Link Duration) is proposed [35]. LLD extends proactive routing schemes such as OLSR [3] by introducing dynamic metrics. In LLD, based on the assumption that longer lasting links are more stable in probability, each link metric is decreased constantly as time passes from an initial value. Namely, the longer a link stays stable, the smaller its metric becomes. Every link metric $\delta^t(l)$ of link l at time t is managed by one of the end node of l with the formula $\delta^t(l) = ab^t + c$ as long as the link is judged as "stable," where t is the time passed by since the link was born, and

b ($0<b<1$) is a ratio of metric decreased per unit time. When a link is judged as "unstable," the metric is reset with the initial value $\delta^0(l)=a+c$. This judgment should be done carefully because frequent reset of metrics leads instability of forwarding paths.

In LLD routing protocol, link metrics are updated periodically and the timing of them is roughly synchronized in the network. With the synchronization, we can decrease all link metric values in the same ratio, which guarantees loop-freedom as long as no link becomes "unstable." However, not only it cannot ensure loop-freedom when unstable link appears, but also it requires additional messages for synchronization.

As a method to reduce routing loops in wireless mesh networks, LMR (Loop-free Metric Range) is presented in [36]. The idea of LMR is to prevent rapid changes of metrics by applying the changeable range of metrics per unit time. So, LMR can be applied to any dynamic link metric proposed ever. The changeable range in LMR is expressed as the formula $m_{l,t-t'}\cdot r^{-t'}\leq m_{l,t}\leq m_{l,t-t'}\cdot r^{t'}$, where $m_{l,t}$ is the metric of link l at time t, and r is the coefficient that we call *metric stretch*. Namely, a metric can be changed in a unit time by the value multiplied by metric stretch r.

Note that, reference [36] theoretically proved that we could achieve loop-freedom at any instant with sufficiently small value of r as long as no link fails. Unfortunately, such loop-free value of r is too small to use in practice. Although the loop-free value of metric stretch r depends on several parameters such as network diameter, the value of r to be loop-free is less than 0.5% per unit time for a practical case, where we usually suppose the time interval of control messages in the deployed routing protocol as the unit time. However, reference [36] showed that, even with values r larger than the loop-free threshold, we could reduce routing loops in wireless mesh networks. Here, note that a trade-off is observed that loop packets are reduced for smaller r, but simultaneously the flexibility of paths selection is also reduced so that more congestions and further link failure occur.

5. A new loop-reduction technique for wireless mesh networks

5.1. Idea for loop reduction

In this section, we propose another method to reduce looping in Wireless Mesh Networks that can work in combination with LMR. LMR reduces loop packets by limiting the range of metric changes to prevent rapid transition of metrics. Namely, packet loops are reduced by means of suppressing paths flapping. Here, note that the metric stretch r of LMR that guarantees loop-freedom depends on the network diameter. Specifically, the larger the network diameter in hop count is, the smaller the metric stretch to guarantee loop-freedom is. This means that it is difficult to guarantee loop-freedom for farther destination because larger number of links is included in the path. In other words, packets destined to nearer nodes would less likely to create a loop.

To lower the metric stretch of LMR, it would be reasonable to limit the length in hop count of each single forwarding path. In our method, this is done by splitting a forwarding path

between two nodes into the sequence of partial forwarding paths, such that the length of each partial path in hop count is smaller than a certain value k. This is performed using loose source routing, in which a packet should visit several intermediate nodes before reaching its final destination.

By limiting the path length within k, the threshold of the metric stretch of LMR to be loop-free becomes smaller, because we can use the threshold value in which network diameter is assumed to be k. For example, if $k=3$, and the maximum and the minimum link metric are $M_{max}=5$ and $M_{min}=1$, respectively, the metric stretch should hold $r\leq1.0667$ to guarantee loop-freedom according to [36]. For $k=4$ and $k=5$, the conditions are $r\leq1.05$ and $r\leq1.04$, respectively.

5.2. A new loop-reduction method

We propose a new method to reduce routing loops for wireless mesh networks. As described in the previous section, we split a forwarding path between two nodes into a sequence of partial paths to limit the length of each partial path. To achieve this, we introduce a technique of loose source routing, where the list of intermediate nodes to visit is held in each packet header, and the packet is repeatedly forwarded to the next intermediate node to finally reach its destination.

In our method, these intermediate nodes are set along the shortest paths to the destination. Specifically, every time a packet reaches its intermediate node, the node set the next intermediate node with the node k-hop ahead in the shortest path to the destination. Note that, with this method, we require only one additional field in the packet header for an intermediate node.

We explain our method with an example shown in Figure 3. There is a network with 10 nodes and assume that the proposed method is working with $k=2$. Suppose that a packet is sent from A destined to F, and the computed shortest path from A to F is expressed by the sequence of nodes A, B, C, D, E and F. In A, because the intermediate-node field is empty, A sets the field with C, which is the 2-hop ahead in the forwarding path to F, and forwards the packet to the next-hop node to reach the intermediate node C. Node B forwards a packet to C, which is the next-hop node for the intermediate node C. At node C, because C is the intermediate node of the packet, node C update the intermediate-node field with E, which is the node 2-hop ahead on the path to the destination F, and C forwards packets to D. Node D simply forwards packet to E. At node E, because the destination F lies within k hop distance, it forwards the packet to F with its intermediate-node field left empty. The proposed method forwards packets to their destination in this way.

To implement this scheme in practice, nodes are required not only to prepare extra field for an intermediate node in the packet header, but also to maintain an extra table that manages an intermediate node for each destination. Because this scheme extends link-state routing schemes, computing intermediate nodes can be easily done in the process of shortest-paths computation. Consequently, the intermediate-node table, in which the node k-hop ahead on

the shortest path for each destination is held, is computed every time a node computes the shortest paths to create its routing table.
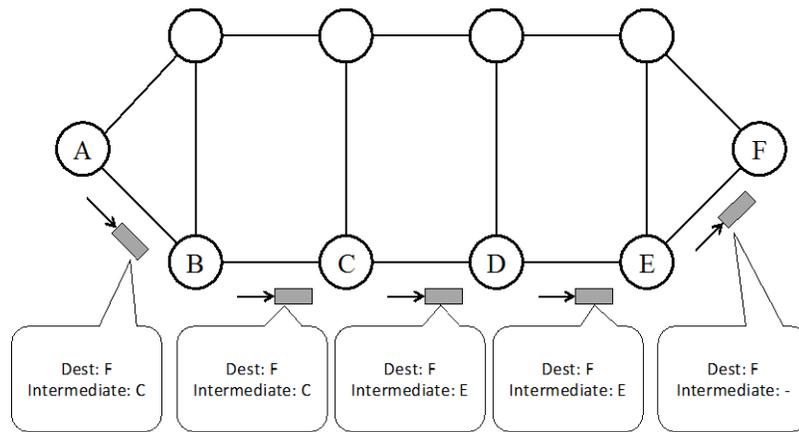

Figure 3. Proposed Method, Case of *k*=2

5.3. Simulation setup

We evaluated the proposed method through computer simulation with network simulator Qualnet [38]. We implemented ETX link metric and the proposed method by modifying the OLSRv2-Niigata module [39].

As a network topology, we used a 7 x 7 grid as shown in Figure 4, where nodes are placed with interval of 300 meters. Because we set the transmit power with 85dB, only the neighbors in vertical and horizontal directions are connected by wireless links. Each node has a single 802.11 interface with an omni-antenna, and the communication speed is fixed to 2Mbps.

We generated four CBR (Constant Bit Rate) flows in diagonal directions, i.e., from *A* to *D*, *D* to *A*, *B* to *C*, and *C* to *B*, where the packet size is 512 bytes and the communication speeds tried are 40, 60, and 80kbps. We ran the simulation for 15 minutes: during 2 minutes from the beginning, we just wake up all routers and wait for the convergence of routes, and after that we generate the four CBR flows for 10 minutes.

As for OLSR parameters, we use the default values for HELLO_INTERVAL and TC_IN-TERVAL, i.e., 2 and 5 seconds, respectively. Because we deployed ETX routing metric, we set the parameter NEIGHBOR_HOLD_TIME with 20 seconds, i.e., 10 sequential loss of HELLO messages cause links to fail. We set TC_REDUNDANCY=2 to disable the mechanism of MPR (Multi-Point Relay), which reduce the load of control messages by limiting relay nodes and advertised links, to exclude the affect of advertised link selection. With this parameter settings, all nodes relay control messages and all links are advertised over the network.

We compared the cases of (i) the conventional method only with ETX metric, (ii) the proposed method with ETX metric, and the proposed method with ETX and LMR. As for LMR, we tried several parameters of metric stretch r. We performed 30 trials for each parameter, and compared the performance using the average of them.
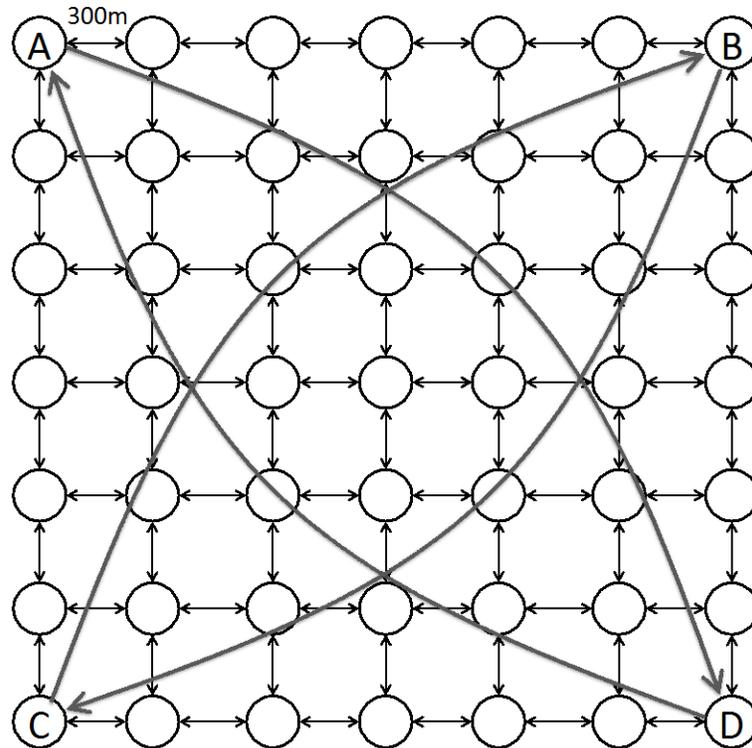

Figure 4. Simulation Settings

5.4. Simulation results

Figures 5-10 show the results of the case where the transmission rates of the four flows are all 40, 60, and 80kbps. In these figure, the results of every combinations of the values k=2, 3, …, 6 of the proposed method and the metric stretch values r=0.01, 0.03, 0.05, 0.1, 0.2 of LMR.

In Figure 5, the number of loop packets in the 40kbps case is shown. The number of loop packets is relatively low in total, but we see that the ETX cases take especially high value, which surely indicates the effects of the proposed method on reducing loop packets. It is wondering that loop packets increase when the metric stretch takes smaller values. The reason is that; when the metric stretch is small, forwarding paths tend to be persistent and

paths selection does not react against rapid transition of link quality, resulting in link failure. Link failure causes paths re-computation and consequently leads packet loops.

On the other hand, in this result, packet loops do not occur frequently when the metric stretch is high or LMR is not applied. This is why the network is not so congested that cause rapid metric changes and then routing loops. For the evidence of this point, see the result of packet delivery ratio shown in Figure 6, where packet delivery ratio is very high as much as 90%, indicating that the congestion level is low.

Figure 6 also shows the interesting tendency that the packet delivery ratio is the best when the metric stretch r takes 1.03. We point out that the main reason of this phenomenon is interference among nodes; when r takes large values, forwarding paths change frequently and it becomes the situation where many nodes have packets to forward. The throughput degrades in CSMA/CA when the number of nodes in contention increases. Namely, when r takes small values, loop packets due to link failure degrades the performance, whereas when r takes large values, heavy interference degrades the performance. The result of Figure 6 shows that the balance point of the metric stretch in this network is around r=1.03.

In Figures 7-10, we show the results in the cases of more loaded scenarios where we generate 60kbps and 80kbps flows. Although the whole tendency is the same as the 40kbps scenario, the number of loop packets increases and the packet delivery ratio decreases in total. Note that in these cases packet loss with the reason "queue over" significantly increases, which indicates the heavy load of the network. Note that, in Figure 9, the cases of k=3, 4, 5 especially take better performance of loop packets reduction. Note that in the cases where r is less than 1.03, routing loops due to link failure dominates, and the cases where both LMR and the proposed method are applied takes better performance.
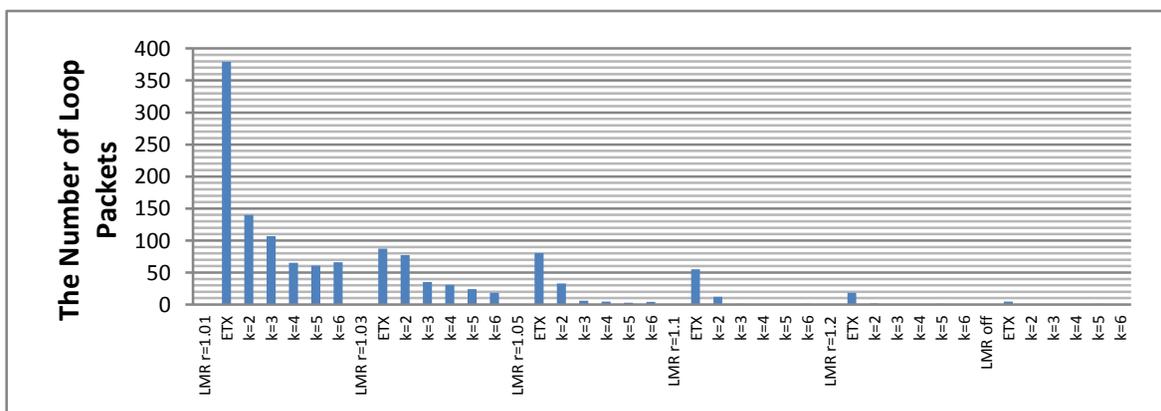

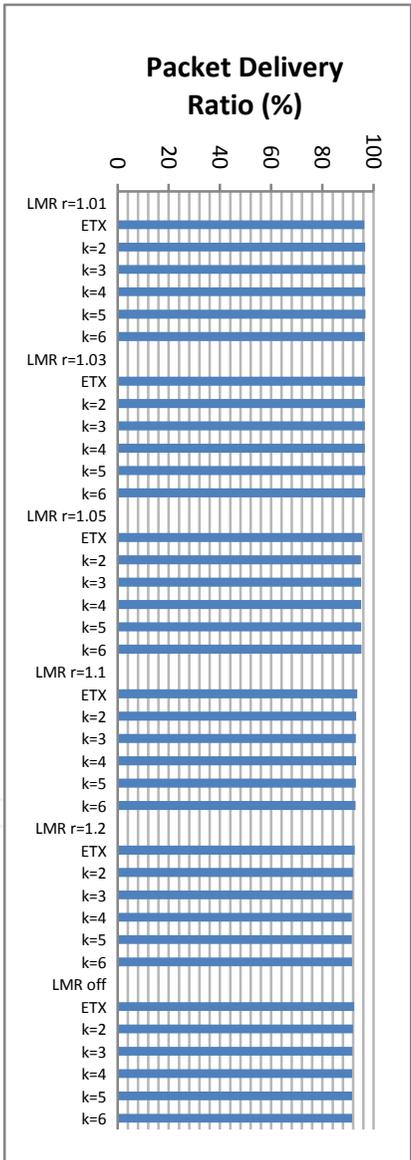**Figure 5.** Number of Loop Packets (Case of 40kbps flows)

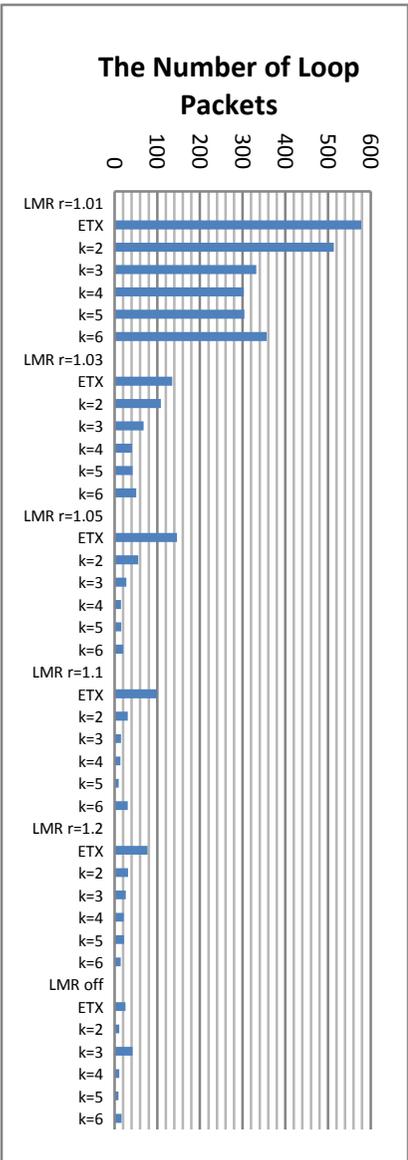**Figure 6.** Packet Delivery Ratio (Case of 40kbps flows)

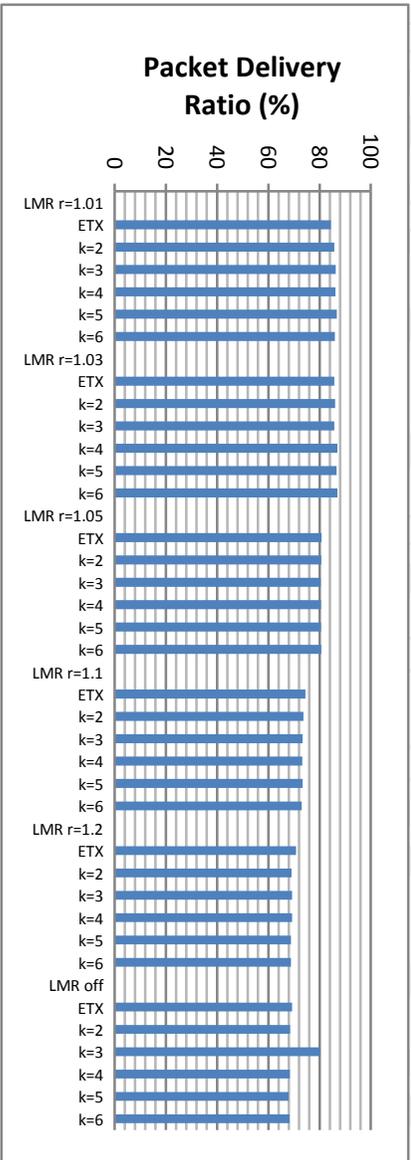**Figure 7.** Number of Loop Packets (Case of 60kbps flows)

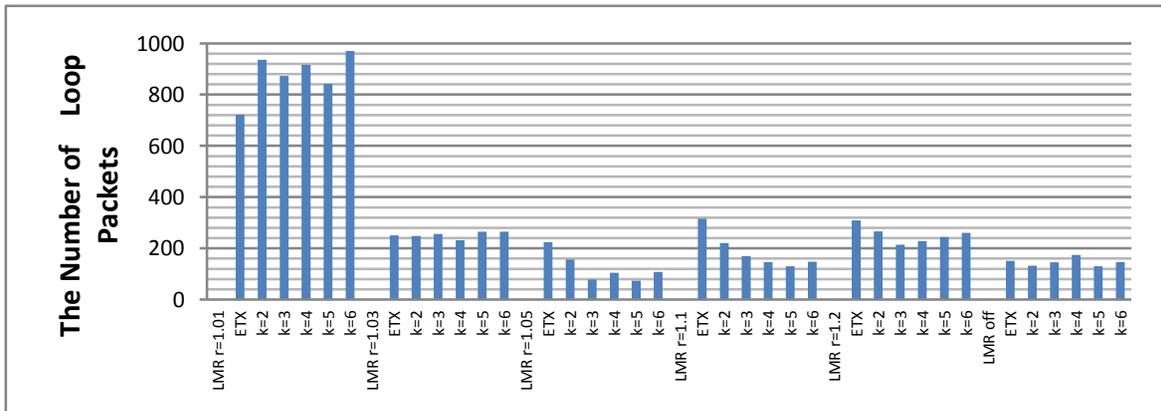**Figure 8.** Packet Delivery Ratio (Case of 60kbps flows)

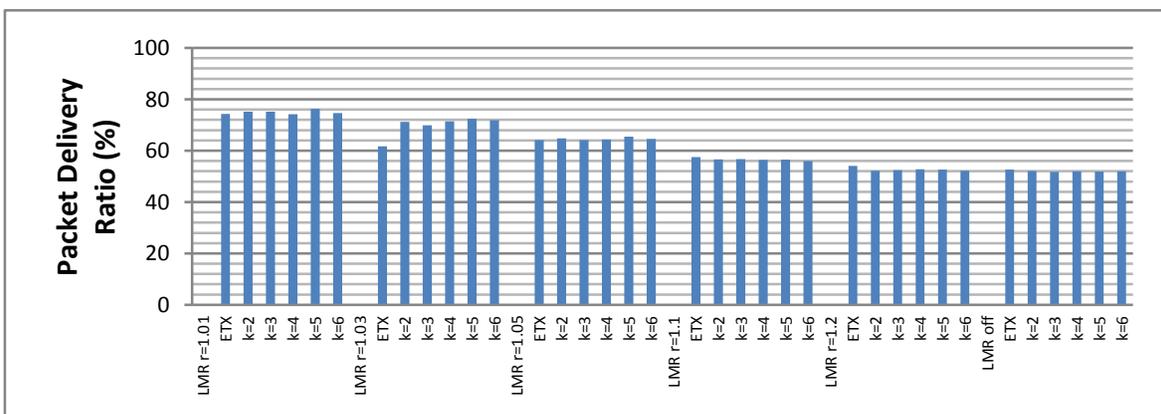**Figure 9.** Number of Loop Packets (Case of 80kbps)

Figure 10. Packet Delivery Ratio (Case of 80kbps)

5.5. Discussion

We evaluated the performance of the proposed method in combination with the other loop reduction method LMR. The simulation results showed that the proposed method solely did not work well and it effectively worked when LMR is applied together. In other words, the combination of the proposed method and LMR works well to reduce routing loops in wireless mesh networks.

Routing loop problem in wired networks has been traditionally regarded as a harmful problem to be avoided, and it is the same in wireless mesh networks. The harmful influence of the looping packets has not been frequently focused on in wireless mesh networks, only because the problem of interference has currently far larger impact on the performance. Although the harmful affection is not always reflected on throughput or packet delivery ratio, looping packets causes large variation of jitter and throughput on time series, which surely degrades the quality and the stability of the communications.

To improve the quality and stability of communications, we have proposed two methods, both of which attain loop-reduction against metric changes. One of the important problems here is that the two methods do not assume link failure, whereas practically congestions easily cause link failure and the consequent packet loops in wireless mesh networks. To make the most of the applied methods and achieve stable communications in wireless mesh networks, it is desirable to develop a congestion control method that prevents link failure even in case of congestion. In combination with such congestion control methods, the harmful influence of looping packets in wireless networks would be significantly improved.

6. Conclusion

In this article, we reviewed the literature of loop-free routing in wired and wireless networks, and further we proposed a new loop-reduction technique for wireless mesh networks. We see that, in proactive routing schemes, it is promising to apply dynamic metrics to afford flexibility against wireless instability, and for this reason it is difficult to apply a family of loop-free techniques developed for wired networks.

We have several loop reduction techniques for wireless multi-hop networks, but it is not still sufficient in performance to provide stable communications in wireless multi-hop networks because loops are not still eliminated. We see that the main cause of looping under the two loop reduction methods, i.e., LMR and the proposed method in this paper, is link failure due to congestion. To provide stable communications over wireless mesh networks, a method is required to prevent link cuts even in case of congestion. To develop such congestion control methods, which works in combination with the two loop-reduction methods, is one of the important tasks to realize wireless mesh networks that can provide stable and reliable communications without routing loops.

Author details

Takuya Yoshihiro and Masanori Kobayashi

Wakayama University, Japan

References

[1] Perkins, C., Belding-Royer, E. & Das, S. (2003), Ad Hoc On-demand Distance Vector (AODV) Routing, IETF Request For Comments (RFC) 3561, IETF.

[2] Johnson, D., Hu, Y. & Malts, D. (2007). The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4, IETF Request For Comments (RFC) 4728, IETF.

[3] Clausen, T. & Jacquet, P. (2003). Optimized Link State Routing Protocol (OLSR), IETF Request For Comments (RFC 3626), IETF.

[4] Ogier, R., Templin, F. & Lewis, M., (2004). Topology Dissemination Based on Reverse-Path Forwarding (TBRPF), IETF Request For Comments (RFC 3684), IETF.

[5] Akyildiz, I.F. & Wang, X. (2009), Wireless Mesh Networks, John Wiley & Sons Ltd Publication.

[6] Cheng, Z. & Heinzelman, W.B. (2004). Exploring Long Lifetime Routing (LLR) in ad hoc Networks, Proceedings of 7th ACM International Symposium on Modeling, Analysis and imulation of Wireless and Mobile Systems, pp.203–210.

[7] Zhao, M. & Wang, W. (2007). The impacts of radio channels and node mobility on link statistics in mobile ad hoc networks, Proceedings of IEEE Global Telecommunications Conference (Globecom2007), No.1, pp.1205-1209.

[8] Tickoo, O., Raghunath, S. & Kalyanaraman, S. (2003). Route Fragility: A Novel Metric for Route Selection in Mobile Ad Hoc Networks, Proceedings of IEEE ICON03, pp. 537–542.

[9] Trivi ˜no-Cabrera, A., Nieves-P´erez, I., Casilari, E. & Gonz´alez-Ca˜nete, F.J. (2006). Ad Hoc Routing Based on the Stability of Routes, Proceedings of the 4th ACMInternationalWorkshop on MobilityManagement and Wireless Access, pp.100–103.

[10] Yawut, C., Paillassa, B. & Dhaou, R. (2007). On Metrics for Mobility Oriented Self Adaptive Protocols, Proceedings of in Wireless and Mobile Communications 2007 (ICWMC2007).

[11] Qin, L. & Kunz, T. (2006). Mobility Metrics to Enable Adaptive Routing in MANET, Proceedings of IEEE Wireless and Mobile Computing, Networking and Communications 2006 (WiMobapos2006), pp.1–8.

[12] De Couto, D., Aguayo, D., Bicket, J. & Morris, R. (2003). A High-Throughput Path Metric for Multi-Hop Wireless Routing, Proceeding of ACM Annual International Conference on Mobile Computing and Networks (MOBICOM2003), pp.134–146.

[13] Draves, R., Padhye, J. & Zill, B. (2004). Routing in Multi-Radio, Multi-HopWirelessMesh Networks, Proceedings of ACM Annual International Conference on Mobile Computing and Networks (MOBICOM2004), pp.114–128.

[14] Sobrinho, J. L. (2003). Network Routing with Path Vector Protocols: Theory and Applications, Proceedings of the ACM 2003 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM2003), pp.49–60.

[15] Yang, Y., Wang. J. & Kravets, R. (2005). Interference-aware Load Balancing for Multi-hop Wireless Networks, In Technical Report UIUCDCS-R-2005-2526, Department of Computer Science, University of Illinois.

[16] Jin, S., & Mase K., A Hidden-Exposed Terminal Interference Aware Routing Metric for Multi-Radio and Multi-Rate Wireless Mesh Networks, IEICE Transactions on Communications, Vol.92-B(No.4), pp.709–716.

[17] Speakman L., Owada Y., Mase K., Looping in OLSRv2 in Mobile Ad-Hoc Networks, Loop Suppression and Loop Correction, IEICE Transactions on Communications, Vol.E92-B(No.4), pp.1210–1221.

[18] A.S. Tanenbaum, Computer Networks Fourth Edition, Publisher: Prentice Hall PTR, ISBN:9780130661029, 2003.

[19] G. Malkin, "RIP Version 2," IETF RFC2453, November 1998.

[20] J.M. Jaffe and F.M. Moss, "A Responsive Routing Algorithm for Computer Networks," in IEEE Transactions on Communications, Vol. COM-30, No. 7, July 1982, pp. 1758-1762.

[21] Y. Rekhter, T. Li, and S. Hares, "A Border Gateway Protocol 4 (BGP-4)," IETF RFC 4271, January 2006.

[22] C. Cheng, R. Riley, S. P. R. Kumar, J. J. Garcia-Luna-Aceves, A loop-free extended Bellman-Ford routing protocol without bouncing effect, SIGCOMM Comput. Commun. Rev., Vol. 19, No. 4. (September 1989), pp. 224-236.

[23] Garcia-Luna-Aceves, J. J. (1993), Loop-free Routing using Diffusing Computations, IEEE/ACM Transactions on Networking, Vol.1 (No.1), 130–141.

[24] R. Albrightson, J.J. Garcia-Luna-Aceves, and J. Boyle, "EIGRP – A Fast Routing Protocol Based on Distance Vectors," in Proceedings of Network/Interop, Las Vegas, NV, May 1994.

[25] J.J. Garcia-Luna-Aceves and Shree Murthy, A Path-Finding Algorithm for Loop-Free Routing, IEEE/ACM Trans. Networking, Vol.5, pp.148—160, 1997.

[26] Andreas Schmid and Christoph Steigner, Avoiding Counting to Infinity in Distance Vector Routing, TELECOMMUNICATION SYSTEMS, Volume 19, Numbers 3-4 (2002), 497-514, DOI: 10.1023/A:1013858909535

[27] IS-IS (Intermediate Systems – Intermediate Systems), ISO/IEC 10589, 2002.

[28] Moy, "OSPF version 2," IETF RFC2328, April 1998.

[29] Francois, P. & Bonaventure, O. (2007), Avoiding Transient Loops During the Convergence of Link-state Routing Protocols, IEEE/ACM Transactions on Networking, Vol. 15 (No. 6), 1280–1932.

[30] Francois, P., Shand, M. & Bonaventure, O. (2007). Disruption-free Topology Reconfiguration in OSPF Networks, Proceedings of IEEE INFOCOM2007.

[31] Ito, H., Iwama, K., Okabe, Y. & Yoshihiro, T. (2003). Avoiding Routing Loops on the Internet, Theory of Computing Systems, Vol.36, 597–609.

[32] J. Raju and J. J. Garcia-Luna-Aceves, "A New Approach to On-demand Loop-free Multipath Routing," In IEEE IC3N'99, pp.522-7, 1999.

[33] J. J. Garcia-Luna-Aceves, "A New Approach to On-demand Loop-free Routing in Ad Hoc Networks," In Proceedings of the twenty-second annual symposium on Principles of distributed computing (PODC'03), pp. 53 – 62, 2003.

[34] Yoshihiro, T. (2009). Loop-free Link Stability Metrics for Proactive Routing in Wireless Ad Hoc Networks, Proceedings of IEEE ICC2009, pp.1–5.

[35] Takuya Yoshihiro (2011). LLD: Loop-free Link Stability Metrics for Proactive Link-State Routing in Wireless Ad Hoc Networks, Mobile Ad-Hoc Networks: Protocol Design, Xin Wang (Ed.), ISBN: 978-953-307-402-3, InTech, Available from: http://www.intechopen.com/books/mobile-ad-hoc-networks-protocol-design/lld-loop-free-link-stability-metrics-for-proactive-link-state-routing-in-wireless-ad-hoc-networks

[36] T. Yoshihiro, "Reducing Routing Loops under Dynamic Metrics in Wireless Mesh Networks," IEEE Global Communication Conference, Exhibition, and Industory Forum (Globecom2010), pp.1-6, 2010.

[37] Qualnet, http://www.scalable-networks.com/

[38] OLSRv2-Niigata, http://www2.net.ie.niigata-u.ac.jp/nOLSRv2/olsrv2/Welcome.html