GPAA 109486—22/3/2005—KALYAN—141842

The International Journal of Parallel, Emergent and Distributed Systems,
Vol. 00, No. 0, 2005, 1–15

A highly reliable multi-path routing scheme for *ad hoc* wireless networks

FEI DAI and JIE WU

Department of Computer Science and Engineering, Florida Atlantic University, Boca Raton, FL 33431

This paper presents a new routing scheme for *ad hoc* wireless networks that provides fresh routing information along active routes with affordable cost. The proposed routing mechanism, called *proactive route maintenance* (PRM), is used to replace the naive route mechanism in existing reactive (on-demand) routing protocols to enhance route reliability and reduce the frequency of expensive route discoveries. The assumption behind PRM is the communication locality in *ad hoc* wireless networks. That is, most data packets are transported along a few active routes. Data packets are forwarded via multiple optimal paths to meet certain QoS requirements, such as fault tolerance and load balance. Routing information is disseminated along active routes and advertised only by active nodes that forward data packets. Alternative paths are dynamically discovered and maintained by active nodes and their 1-hop neighbors (called passive nodes). The routing overhead in passive nodes is light. PRM maintains reliable end-to-end connections even in dynamic networks with relatively low overhead, and has the desirable properties including high delivery ratio, low latency, fair load distribution, self-healing and self-optimization.

Keywords: Ad hoc networks; Multipath routing; On-demand routing; Quality-of-service (QoS); Route maintenance; Simulation

1. Introduction

Wireless network architecture can be divided into two categories [18]: The infrastructure-aided single hop model and the peer-to-peer multihop model. The former and centralized model is still dominant in wireless LANs and cellular networks. But the latter, called *wireless mesh networks* [3], are emerging to provide extended coverage, higher reliability and ease of setup. An *ad hoc* wireless network (or simply *ad hoc network*) [7] is a special type of wireless mesh networks, in which a collection of mobile hosts with wireless network interfaces form a temporary network, without the aid of any established infrastructure (i.e. base stations) or centralized administration (i.e. mobile switching centers).

Quality-of-Service (QoS) routing has been motivated by multimedia applications, such as voice channels, live videos and document transfer. QoS routing selects paths based on QoS metrics to satisfy specific requirements, such as end-to-end delay, delay jitter, bandwidth and packet loss probability. In *ad hoc* networks using a contention-based MAC layer such as IEEE 802.11, enforcing hard QoS guarantees is difficult because of the lack of a resource

---

*Corresponding author. Email: jie@cse.fau.edu

The International Journal of Parallel, Emergent and Distributed Systems
ISSN 1744-5760 print/ISSN 1744-5779 online
http://www.tandf.co.uk/journals
DOI: 10.1080/17445760500095035

allocation mechanism. Nevertheless, *soft* QoS routing is still possible, where the long-term statistic value of theses QoS metrics are enhanced for smooth communication.

Routing in *ad hoc* networks is difficult for two reasons: Highly dynamic network topology and limited scarce bandwidth and energy resources. A perfect routing protocol would be providing accurate routing information when needed, while wasting no network resource in maintaining inactive routes. Ad hoc routing protocols are either *proactive* or *reactive*. Proactive protocols like DSDV [16] periodically disseminate routing information over the entire network regardless of neediness and suffer from high overhead. Reactive (i.e. *on-demand*) protocols such as DSR [9] and AODV [17] do not update routing information unless a new path is requested (route discovery) or an old path is broken (route recovery). Route discovery and recovery are usually conducted via network-wide flooding of route query (RREQ) packets, which causes route setup delay and high cost. *Hybrid* protocols like ZRP [6] use proactive approaches in small regions called zones and reactive approaches outside the local zones. However, the size of the zone is either too small to provide fresh information for an active route, or too large to be cost-effective.

Multipath routing has been used in wired networks to improve throughput, fault tolerance and load balance. Due to the inter-route interference in wireless transmissions, multipath routing is less effective in achieving high throughput. Nevertheless, its abilities to provide multiple alternative paths and improve the distribution of communication loads are desirable for *ad hoc* networks, where wireless links fail frequently and mobile hosts suffer from limited power. In most existing multipath schemes for *ad hoc* networks [11–13,15,21], multiple paths are divided into primary (i.e. active) paths that actually forward data packets, and backup paths that are activated only after active paths fail. Usually, a shortest path serves as the primary path, and others become backup paths. However, without maintenance, the backup paths may fail without alert, and cause longer delay and more packet losses. Simultaneously forwarding data packets via all paths can detect path failures promptly and achieve better load distribution. But this also causes out-of-order delivery and data transmission along non-optimal paths.

This paper proposes a hybrid routing protocol that maintains robust multipath routes with relatively low overhead. This protocol uses the same route discovery mechanism as in reactive protocols, but the maintenance of the active routes is proactive, which adapts well to the highly dynamic networks, and reduces the frequency of route recoveries. The proposed *proactive route maintenance* (PRM) mechanism provides fresh routing information at where it is needed with affordable cost. PRM maintains a mesh containing multiple overlapping optimal and sub-optimal paths from the source to destination. All optimal paths are active. A data packet can travel to the destination via a randomly selected path. Sub-optimal paths serve as off-line backups, which are activated after all optimal paths have failed. As this mesh-structure is self-healing and self-optimizing, most link failures can be tolerated without causing route failure or non-optimal routing. PRM is a distributed routing scheme. A freshness-based mechanism similar to those in DSDV and AODV is used to ensure loop freedom.

PRM can be used for soft QoS routing, where a multipath route is automatically adjusted to meet certain QoS requirements. In the case of a single path failure, the data traffic can switch to alternative paths to avoid packet losses. The automatic repair mechanism reduces the frequency of route recoveries, which also reduces the overall delay and delay-jitter. The traffic load of a connection is distributed to multiple paths, which lower the chance of the early power depletion of the heavy-loaded nodes. The multipath transmission can also avoid hot spots and achieve higher throughput.

The assumption behind PRM is the *communication locality* in *ad hoc* networks. That is, most traffic is caused in a few data flows. If we measure the size of an *ad hoc* network with its node number N, there are $\Omega(N^2)$ possible source/destination pairs, but the network capacity is $O(\sqrt{N})$. Therefore, only a small number ($c\sqrt{N}$ out of N^2) of simultaneous data flows can be supported by an *ad hoc* network, unless the channel bandwidth could be increased dramatically, which is not likely in a short term vision, or the per connection traffic would be decreased as the network size increases, which is also impractical. Actually, communication locality is implicitly assumed by all reactive protocols. If the data traffic is distributed in many short-lived connections, the reactive protocols will be more expansive than proactive protocols.

The rest of this paper is organized as follows: Section 2 discusses related work in multipath routing, localized route maintenance and loop-free routing. Section 3 presents the PRM extension to reactive routing protocol and compares it with several existing schemes. Section 4 gives our simulation result. Section 5 concludes this paper.

2. Related work

Multipath routing has been used in wired networks to achieve high throughput, load balance and fault tolerance. Among routing protocols for *ad hoc* networks, TORA [14] explicitly supports multipath routing but lacks accurate distance metrics for optimal routing. Both ROAM [19] and MDVA [20] are designed to provide multipath routing. But their proactive manner makes them more suitable for static or low mobility networks. More recently, pure reactive protocols, such as AODV [17] and DSR [9], have been extended to support multipath routing [11–13,15,21]. Nasipuri, Castañeda and Das [13] suggested preserving two link-disjoint paths to the destination, at the source and at each intermediate node, one as the primary path and the other as the backup. Analysis and simulation show that providing intermediate nodes with backup paths increases the life span of active route. Pearlman, Haas, Sholander and Tabrizi [15] proposed a diversity injection scheme for DSR to find node-disjoint paths. The route reply process is modified so that intermediate nodes may redirect RREPs along multiple paths back to source. Lee and Gerla [11] proposed another scheme to find maximally disjoint paths. In their split multipath routing (SMR) extension to DSR, intermediate nodes may forward, not drop, a duplicate RREQ, if this RREQ takes a route different from the previous received RREQ. Wu and Harms [21] discussed and compared both schemes. Marina and Das [12] proposed on-demand multipath distance vector routing (AOMDV), an extension to AODV. AOMDV also enable intermediate nodes to forward multiple RREPs along link-disjoint paths. An extra *first hop* field is added to RREQs to distinguish disjoint paths.

Multipath routing in *ad hoc* networks has a different set of objectives from that in wired networks. It is shown in [15] that, due to the signal interference between multiple paths (the *coupling problem*), the throughput benefit of multipath routing is trivial, even for wireless networks using multiple channel schemes. Most on-demand protocol extensions [11–13] focus on fault tolerance. That is, by activating backup paths after the primary path fails, frequency of route discovery can be reduced, which means less routing overhead and smaller average end-to-end delay. It is also shown in [13] that keeping multiple paths in each intermediate node can further decrease the frequency of route discovery. However, backup paths may fail before the primary path fails, and there is no mechanism to find new alternate paths before the next route discovery. Load balance is another concern. It is shown in [21] simultaneously forwarding data packets with multiple paths can improve the distribution of network loads, and avoid the situation that a few critical nodes quickly deplete their power.

Some localized route maintenance schemes [4,10,22] have been designed to control the route recovery cost in reactive protocols. Castenada, Das and Marina [4] suggested to exploit the path locality and node locality in mobile wireless networks. When recovering a broken path, the source will issue a limited flooding within a few hops around the old path. Lee and Gerla proposed AODV-BR [10], where a 1-hop local repair scheme is proposed. In this scheme, nodes along the primary path overhear passing-by routing reply (RREP) packets to construct more backup paths. Wu, Ni, Tseng and Sheu [22] proposed a similar scheme for local route recovery and optimization. All nodes along the primary path overhears both RREPs and data packets. If one node detected a better path than the current one, it will send a RREP to the upstream node, asking it to switch next hop. However, local maintenance schemes in [10,22] use routing information collected in the last route discovery, which becomes stale quickly in highly dynamic networks. Boppana and Konduru proposed ADV [2], a DSDV-like protocol with some on-demand features. In ADV, only the routing information about active receivers (i.e. destinations of some data packets) is disseminated in the network, and the information propagation speed, depends, on the data traffic volume. ADV demands explicit initialization and termination of connections, where the status change (i.e. active or inactive) of receives is broadcast to the entire network. None of these schemes uses multipath routing.

Loop-free routing is not a trivial issue in *ad hoc* networks. DSR [9] uses source routing to avoid loop, with the penalty of longer packet headers. TORA [14] uses distributed routing, is loop free even in partitioned networks, but is less effective in finding optimal paths. Distance vector protocols [12,16,17,19,20] maintain shortest paths. But loops may appear when a node increases its distance mark, and before the distance mark of its upstream nodes converges, it may select an upstream node as its next hop. ROAM [19] and MDVA [20] use diffusion computation to avoid loop. A node increasing its distance mark cannot switch next hop until all upstream nodes have updated their distance marks. DSDV [16], AODV [17] and AOMDV [12] use destination-issued sequence number to compare the freshness of two distance marks. A node can only use a next hop with a fresher distance mark. Diffusion computation relies on the reliable hop-to-hop coordination, which is costly in *ad hoc* networks. The liveness (i.e. the recovering speed from a link failure) of freshness-based approaches depends, on the frequency that new sequence numbers are issued. DSDV uses constant frequency and suffers efficiency penalty as it floods sequence numbers all over the network. AODV does not issue new sequence numbers except during a route discovery. Therefore, it is hard for a node in a broken route to switch its next hop. In AOMDV, a node with multiple next hops computes its distance as based on the maximum distance of its next hops. It can tolerate more link failures before triggering a route repair or recovery process. However, it comes with an expense of routing via non-optimal paths.

3. Proactive route maintenance

We assume *ad hoc* networks with fixed transmitter range and bidirectional links. There is no neighbor discovery mechanism. A node is invisible to its neighbors unless it advertises its existence. But the MAC layer can detect a link failure during unicast transmission.

3.1 Protocol overview

PRM is the combination of reactive route discovery and proactive route maintenance. The route discovery part (called the *base protocol*) can be any reactive routing protocols such as DSR and AODV. Currently a *naive route maintenance* mechanism is used in most reactive routing protocols. Data packets travel along a path constructed during the last route discovery. If a link failure is detected by an intermediate node, it drops the packet and sends a route error (RERR) packet back to the source. If the source has a backup path to the destination, it will switch to the backup path; otherwise, a flooding-based route recovery process is triggered. As a replacement of the naive route maintenance, PRM has several desirable properties:

Freshness. All nodes near an active route have the up-to-date routing information. Invalid paths will be eliminated, new paths recognized, and non-shortest paths replaced by shorter paths.

Robustness. An active node that is forwarding data packets usually maintains several fresh alternative paths. After one path fails, the data packet can still be forwarded along another path instead of being dropped. The multipath route can be self-healed by recognizing new alternative paths. The extra delay and overhead caused by the frequent route discovery operations is avoided. On the other hand, PRM does not guarantee a route to the destination, and will resort to a route discovery operation if there is no alternative path available.

Light-weighted maintenance. Unlike in existing proactive routing protocols, the route maintenance is confined to those small areas surrounding active routes, where control packets make only a small portion of data transmission. As the lifetime of a route is lengthened, the overhead of the proactive route maintenance can be compensated by the less frequent route discovery operations.

Figure 1 is a snapshot of an active route forwarding data packets. For the sake of clarity, only one pair of source s and a destination d is considered in the following discussion. Maintenance of multiple active routes can be conducted independently, or with bundled

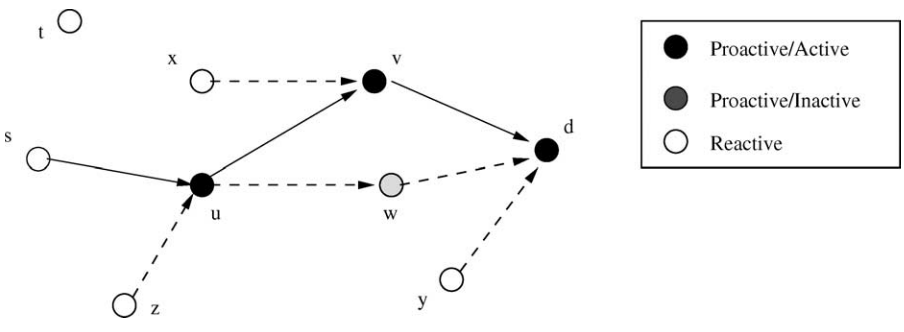


Figure 1. An active route maintained by PRM.

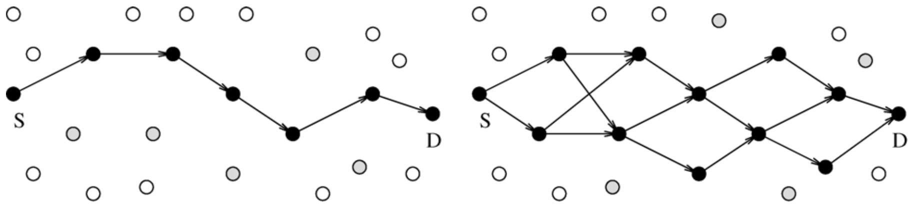
control packets to minimize the overhead. For each specific destination, a node is either in *reactive* or *proactive* mode. At a reactive (white) node, data packets are either forwarded to its next hop discovered by the base protocol, or to a neighboring proactive node, whichever applies. Each proactive node has a *watermark*. Data packets can only travel from high watermark nodes to low watermark nodes. Proactive nodes are either *active* or *inactive*. An active (black) node that is actively forwarding data packets advertises its watermark periodically. An inactive (gray) node, which is not forwarding/receiving data packets, does not advertise its watermark unless it forms a shortcut between two active nodes. In figure 1, data packets are forwarded along path $s \rightarrow u \rightarrow v \rightarrow d$. The destination d and two intermediate nodes u and v are active nodes. The source node s is a reactive node since it has not advertised its watermark yet. Node w has detected an optimal path $u \rightarrow w \rightarrow d$ and advertised its watermark. It is still inactive because it has not forwarded data packets yet.

Watermarks of proactive nodes form a gradient field that attracts data packets to the destination. Usually, the destination node has the lowest watermark. The watermark of a non-destination node is computed based on neighbors' watermarks. A freshness-based mechanism similar to that of DSDV and AODV is used to ensure loop freedom in PRM. In a valid path, a previous hop always has a higher watermark than the watermark of a next hop. A node will never raise its watermark. Therefore, a loop is impossible with monotonously decreasing watermarks.

After a route is constructed by the base routing protocol and used to forward data packets, proactive nodes emerge in the corridor area connecting the source and destination. These proactive nodes form a mesh, where each node has several alternative next hops. At each step, a random next hop is selected to forward a data packet. Nodes can move in and move out of the corridor without compromising the connectivity of the mesh. The overhead of advertising watermarks of proactive nodes is bounded by the size of the corridor. The corridor width, depends, on the traffic volume. As shown in figure 2, with low traffic load, there is only one active (black) node at each step. The width of the corridor area, including both active nodes and inactive (gray) nodes, is at most three. Under heavy traffic load, previously inactive nodes will be activated to forward data packets, which in turn will solicit more inactive nodes. In both scenarios, the scalability of PRM is ensured, as the number of control packets is always proportional to the number of data packets.

3.2 Routing algorithm

The watermark of a proactive node is a 4-tuple (*seqno*, *hops*, *type*, *id*), where *seqno* is a destination-issued sequence number, *hops* is its distance to destination in hops, *type* is either



(a) Under light data traffic load (b) Under heavy data traffic load

Figure 2. The width of the active varies under different data traffic loads.

active or *inactive* and *id* is its unique id. A reactive node has a watermark $(seqno, \infty, -, -)$. A proactive node will not knowingly select a reactive node as its next hop. But a reactive node may be the next hop of another reactive node. For two watermarks $wm_i = (seqno_i, hops_i, type_i, i)$ of node i and $wm_j = (seqno_j, hops_j, type_j, j)$ of node j, we say wm_i is higher than wm_j (i.e. $w_i > w_j$) if

$$(-seqno_i, hops_i) > (-seqno_j, hops_j)$$

Each node i maintains a list *WM* of its neighbors' watermarks, where $WM(j)$ represents the latest watermark of a neighbor j. The water mark of the current node i, $WM(i)$ is initialized to $(0, \infty, -, -)$ and evolves as follows:

```
WATERMARK(WM)
if $i$ is the destination then
    return $(WM(i)seqno + 1, 0, -, -)$
else
    $wm \leftarrow min(WM - WM(i))$
    $wm.hops \leftarrow wm.hops + 1$
    $wm.id \leftarrow i$
    if $wm > WM(i)$ then
        return $(WM(i).seqno + 1, \infty, -, -)$
else
    return $wm$
```

Each time a new watermark is generated, the destination issues a new sequence number to maintain high liveness. For each non-destination node, its watermark is computed based on the lowest watermark of its neighbors. It is ensured that the new watermark will be lower than the last one. Watermark advertisement is different for active and inactive nodes. A node is considered active if it has sent or received at least ACTPKTNUMTHRESHOLD data packets during last ACTTIMEWINDOW seconds; otherwise, it is inactive. Each active node i periodically broadcasted its watermark as follows:

```
ACTIVETIMEOUT(WM)
if $i$ is an active node then
    $wm \leftarrow$ WaterMark($WM$)
    $wm.type \leftarrow active$
    if $wm.hops \neq \infty$ then
        $WM(i) \leftarrow wm$
    Broadcast $WM(i)$ to 1-hop neighbors
```

Procedure ACTIVETIMEOUT is executed for every ACTTIMEWINDOW seconds. If the current node is active and the new watermark has a valid distance value, the new watermark is advertised to neighbors. An inactive node will advertise its watermark only when it can provide an optimal path to an active neighbor, as shown in the following procedure.

```
RECEIVENONERROR(wm′,WM)
$j \leftarrow wm'.id$
$WM(j) \leftarrow min(WM(j), wm')$
```

```
if $i$ is inactive and $j$ is active then
    $wm \leftarrow$ WaterMark($WM$)
    if $WM(j).seqno \leq wm.seqno \land WM(j).hops > wm.hops$ then
    $wm.type \leftarrow inactive$
    $WM(i) \leftarrow wm$
    Send $WM(i)$ to $j$
```

When a node j receives a data packet, and it has at least one neighbor with a lower watermark, node j will randomly select a next hop to forward the packet. If no such next hop is available, it drops the packet and broadcasts an error message $wm = (WM(j).seqno + 1, \infty, inactive, j)$ to its neighbors. The following procedure is triggered when a node i receives an error message from j, or when a link failure (i, j) is detected.

```
ReceiveError($j$, $WM$)
$WM \leftarrow WM - WM(j)$
$wm \leftarrow$ WaterMark($WM$)
if $wm.hops = \infty \land WM(i).hops \neq \infty$ then
    $wm.type \leftarrow inactive$
    $WM(i) \leftarrow wm$
    Broadcast $WM(i)$ to 1-hop neighbors
```

If there are alternative next hops remaining at i, nothing need be done; otherwise, if i's previous watermark is non-infinity, an error message is broadcast to i's neighbors. The same procedure is repeated in i's upstream neighbors until an alternative path is found or the source node is reached. If the source node has an infinity watermark, the base protocol is invoked to construct a new route.

3.3 An example

We use an example to illustrate PRM operations. Figure 3 shows an *ad hoc* network with 4 mobile nodes. The source node s and the destination node d are stationary, and the other nodes u and v move from the left to the right. Initially, all nodes have the watermark $(0, \infty, -, -)$ and data packets are forwarded along path $s \rightarrow v \rightarrow d$, which is discovered by the base protocol (figure 3(a)).

After ActTimeWindow seconds, the destination node d begins to advertise its watermark $(1, 0, -, -)$, which is received by nodes u and v. Both u and v can use d as the next hop, since their watermarks are higher than that of d. The dashed line from u to d indicates the potential next-hop relationship. Currently no data packets have been forwarded through this link (figure 3(b)).

After other ActTimeWindow seconds, d issues a new sequence number. Because node v is actively forwarding data packets, it computes a new watermark $wm = (1, 1, active, v)$ based on the previously received watermark $(1, 0, -, -)$ of d. wm is lower than the last $WM(v) = (0, \infty, -, -)$ and is advertised to neighbors. Destination d simply ignores this message. Node u updates its local copy of $WM(v)$ and, since it cannot provide an optimal alternative path to v, does not advertise an inactive watermark. Instead, v is considered as a potential next hop

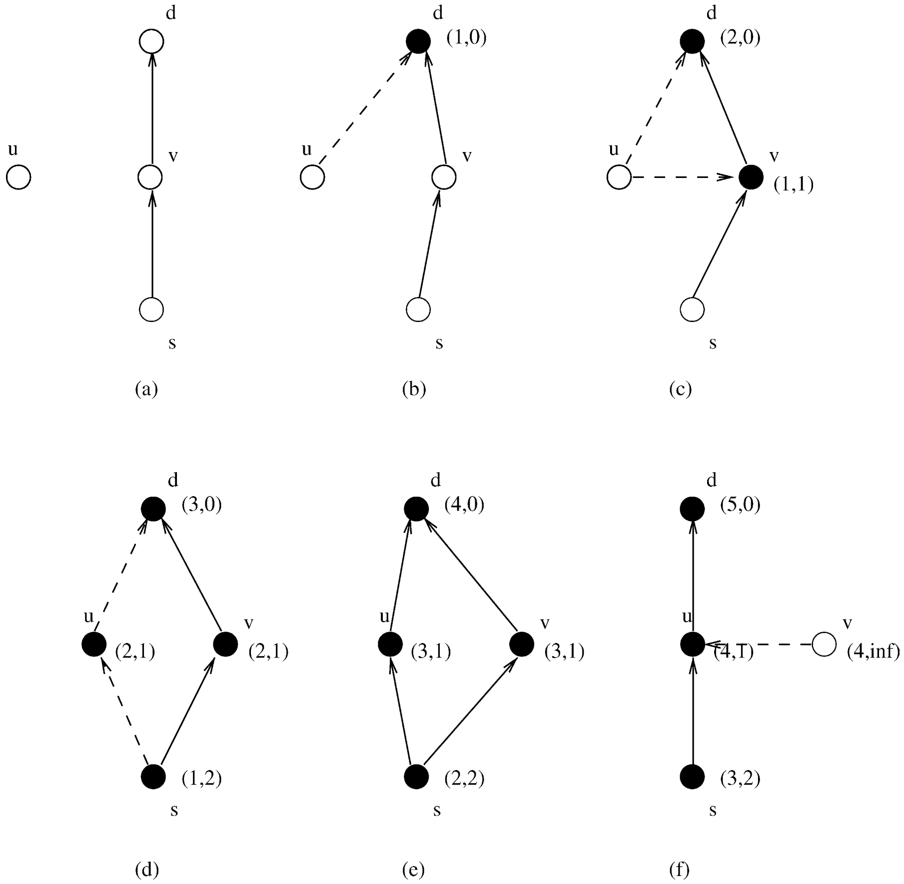


Figure 3. A scenario study of PRM. Watermarks are denoted by a pair (*seqno*, *hops*) with *inf* representing ∞. The type of each node is represented by white (reactive), gray (inactive) and black (active) colors. Solid lines denote data traffic and dashed lines denote next hop relationship.

by u. Node s also updates its local copy of $WM(v)$ and, since it is not a proactive node yet, does not respond to wm (figure 3(c)).

In the next step, active nodes d, v, and s advertise their watermarks. After hearing the watermark $(1, 2, \mathit{active}, s)$, node u determines that it can provide an optimal alternative path to node s. Node u updates its watermark to $WM(u) = (2, 1, \mathit{inactive}, u)$ and unicasts the new watermark to s. Since the new $WM(u)$ is no higher than $WM(v)$, node u can no longer use node v as a next hop. On the other hand, on receiving $WM(u)$, node s can view node u as an alternative next hop. But at this time, no data packets have been forwarded to u yet (figure 3(d)).

Next, node s uses nodes u and v alternatively as its next hops. Therefore, both nodes u and v are activated and advertise watermarks. Node u cannot use v as a next hop, since its advertised *hops* is not larger than that of v. For the same reason, node v cannot use u as a next hop. Both nodes u and v forward data packets directly to d (figure 3(e)).

After node v moves out of the transmitter ranges of nodes s and d, it detects failures of two links (u, d) and (u, s) and recomputes the new watermark $wm = (3, 2, \mathit{active}, v)$ based on $WM(u) = (3, 1, \mathit{active}, u)$. Since $WM(v)$ is higher than the last advertised $WM(v) = (3, 1, \mathit{active}, v)$, the buffered data packets are dropped, the new $WM(v)$ is set to

$(4, \infty, -, -)$, and node v becomes a reactive node. Nodes d and u are not affected by this event, since they are not using node v as a next hop. Node s, on detecting the failure of link (s, u), removes its local copy of $WM(v)$ and forwards the following data packets to node u only (figure 3(f)).

3.4 Correctness

PRM guarantees loop freedom. Given a graph $G = (V, E)$, a directed graph $G' = (V, E')$ can be induced from the next hop relationship for a specific destination, where a directed link (u, v) exists in E' if and only if node u can use node v as a next hop. We say a routing protocol is loop free, if the induced graph is a directed acyclic graph for every destination in every instant. It has been proved in [16] that a routing protocol is loop free, if it uses monotonously decreasing watermarks and each node selects only low watermark nodes as next hops. Therefore, PRM is loop free in the subnetwork consisting of proactive nodes. In the subnetwork consisting of reactive nodes, loop freedom is ensured by the base protocol. However, a loop may occur in a network with both reactive and proactive nodes, without an appropriate mechanism that coordinates PRM and the base protocol.

Figure 4 shows a loop involving reactive nodes and previously proactive nodes: (a) Originally, s is active, and its watermark $(k, 1, active, s)$ is overheard by a reactive node v. (b) After link (s, d) breaks, s advertises its new watermark $(k + 1, \infty, -, -)$ and becomes a reactive node. However, the advertisement is lost and not heard by node v. (c) A new path $s \rightarrow v \rightarrow u \rightarrow d$ is discovered by the base protocol. However, node u still remembers s's old watermark, and a loop is formed between nodes s and v.

We use a simple mechanism to prevent such a loop: (1) a proactive node will not use the next hop provided by the base protocol, and (2) when a new path is discovered by the base protocol, the watermark of all involved nodes are set to $(k + 1, \infty, -, -)$, where k is the maximum *seqno* in all their previous watermarks. We define this watermark as the *path low bound*. As shown in figure 4(d), since the new watermark of node v is smaller than the last heard watermark of s, the next hop relationship (v, s) no longer exists. The following theorem guarantees loop freedom under such a mechanism.

THEOREM 1. *If the base routing protocol is loop free, and nodes in each path discovered by the base protocol are assigned a watermark of the path low bound, then PRM guarantees loop freedom.*

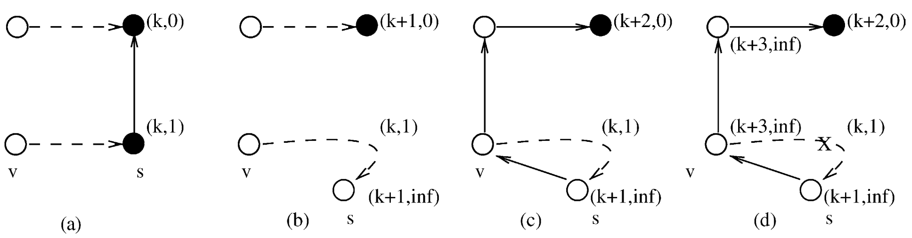


Figure 4. Without appropriate mechanism, routing loop may emerge among reactive nodes and previously proactive nodes.

Proof. Suppose at one moment, a loop $v_1 \rightarrow v_2 \rightarrow \cdots \rightarrow v_l \rightarrow v_1$ exists in the induced graph G'. Because the base routing protocol is loop free, the loop contains at least one link that is directed from a high watermark node to a low watermark node. Without loss of generality, let (v_l, v_1) be that link. That is, $WM(v_k) > WM'(v_1)$, where $WM'(v_1)$ is the latest watermark of v_1 heard by v_l. We will show that $MM(v_l) \leq WM(v_1) \leq WM'(v_1)$, which is a contradiction.

In the above loop, v_1 can select v_2 as its next hop, it is either because $WM(v_1) > WM'(v_2)$, where $WM'(v_2)$ is the latest watermark of v_2 that is heard by v_1, or because both nodes v_1 and v_2 are reactive nodes belonging to the same path discovered by the base protocol and, hence, $WM(v_1) = WM(v_2)$ is the low bound of this path. In either case, we have $WM(v_2) \leq WM(v_1)$. Similarly, we have $WM(v_3) \leq WM(v_2) \leq \ldots \leq WM(v_{k+1}) \leq WM(v_k) \leq \ldots \leq WM(v_l) \leq WM(v_{l-1})$. That is, $WM(v_l) \leq WM(v_1)$.

3.5 Extensions

Here we discuss two extensions of PRM. The first one can further expand the route life span and the second one can reduce the watermark advertisement cost. As shown in figure 3(f), if node v ($WM(v) = (3, 1, active, v)$) loses its next hop d, it cannot use node w ($WM(u) = (3, 1, active, u)$) as its next hop, because $WM(v) > WM(u)$ is not satisfied. All buffered data packets are dropped. Consider another scenario as shown in figure 5(a) and (b), where nodes u and v are stationary and nodes d and s move to the left and right, respectively. After node v detects the failure of link (v, d), it sends watermark $(4, \infty, -, -)$ and becomes a reactive node. After node s detects the failure of link (s, u) and receives v's watermark, it becomes a reactive node too. Although a path $s \rightarrow v \rightarrow u \rightarrow t$ is available, node s cannot use it.

The first extension solves this problem by redefining the "higher than" relationship to enable the usage of non-optimal backup paths. According to the new definition, we say a watermark $wm_i = (seqno_i, hops_i, type_i, i)$ is higher than another watermark $wm_j = (seqno_j, hops_j, type_j, j)$ if

$$(-seqno_i, hops_i, type_i, i) > (-seqno_j, hops_j, type_j, j)$$

For type comparison we assume *inactive* > *active*. If two watermarks have the same sequence number and distance value, the one with an *active* type is higher than the one with an *inactive* type. Given all other fields equal, the watermark with the larger node id is higher. When forwarding data packets, optimal paths (i.e., nodes with smaller *hops* in their watermarks) are considered first. If no optimal path is available, sub-optimal paths (i.e., nodes with the same *hops* as the current node) can be used. Procedure ReceiveError in Section 3.2 is changed so that a proactive node will not become reactive unless it has lost all next hops, optimal or sub-optimal. In line 3 of procedure ReceiveError, even if the condition $wm.hops = \infty$ is true, it is possible that a sub-optimal backup path is available. Procedure ReceiveNonError needs no change, as inactive sub-optimal paths will be reported to neighboring active nodes as well.

The extended PRM is illustrated in figures 5 (c)–(f). When node v loses the optimal path through d, it switches to a sub-optimal path through u. It is possible because its watermark is higher than that of node u based on the new definition. Node v cannot adjust its *hops*, as that will raise its watermark. Node v does not report an error to s, as it

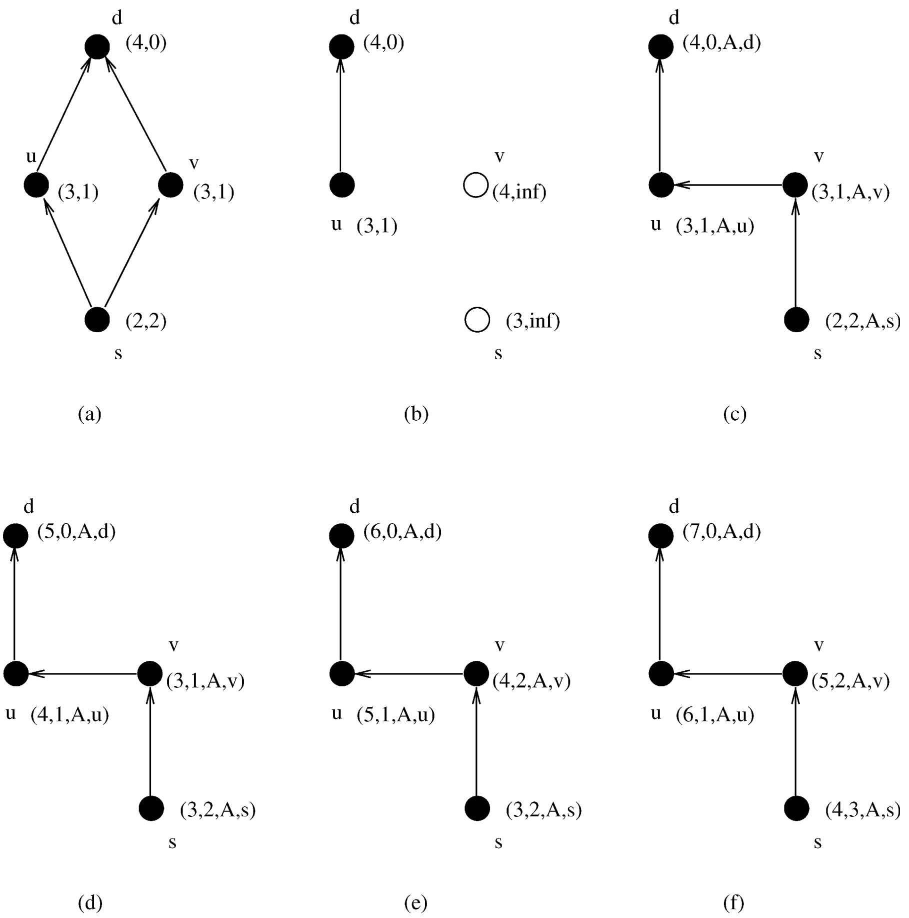


Figure 5. The extended PRM activates a backup path after the optimal path fails.

still has a valid next hop. Node *s* continues forwarding data packets to v. Eventually, node v receives a new watermark from *u* with a larger *seqno*, and with this new *seqno*, it can safely raise *hops* in its watermark. Finally, node *s* receives the correct *hops* from v and adjust its *hops* as well.

The second extension uses two flags *sol_opt* and *sol_bak* in each active watermark to control the number of inactive watermark advertisements. Those two flags are set to *true* by default, and two system parameters are used to measure the saturation level of available paths (i.e. next hops). If the number of optimal paths in a node is larger than or equal to OptPathThreshold, its *sol_opt* flag is set to *false*. If the number of optimal and backup paths in a node is larger than or equal to BakPathThreshold, its *sol_bak* flag is set to *false*. An inactive node sends optimal paths to nodes with their *sol_opt* flags set to *true* only and sends sub-optimal paths to nodes with their *sol_opt* flags set to *true* only.

In addition, watermarks for different destinations in a node are bundled into one packet to reduce the number of control packets. In order to pack the inactive watermark advertisements in the same packet, the respond operation in the procedure ReceiveNonError is deferred to the time when active watermarks are generated and advertised.

3.6 Comparison with existing protocols

To facilitate the further understanding, we compare PRM with existing proactive routing protocols, reactive routing protocols using multiple static path, and reactive routing protocols with enhanced route maintenance mechanisms. Proactive routing protocols like DSDV [16] are much more expensive than PRM, as the routing information of all destinations is distributed over the entire network. In PRM, only the information about the destinations of active routes are distributed in their corresponding stripe areas ("digital valleys"). In another proactive protocol ADV [2], the overhead is reduced by controlling the data volume and propagating speed of the routing information. However, this protocol is based on a much more complicated model than PRM, as it tries to use one mechanism to handle two tasks with fundamental differences, route discovery and route maintenance, at the same time.

On-demand multipath routing protocols [11,12,13,15,21] are used to lengthen the life span of a route without causing significant overhead. Since multiple paths are based on the routing information gathered in the last route discovery, the accuracies deteriorates at about the same pace. Therefore, the life span of the multipath route in these protocols is much shorter than that in PRM. Of course, which scheme is better, lower maintenance cost or longer route life span, needs to be verified in a simulation study.

Several enhanced route maintenance schemes, like AODV-BR [10] and localized route discovery [4], are based on the same rationale behind the on-demand multipath routing protocols; that is, a little longer route life span for low or no extra overhead. Wu *et al.*'s scheme [22] is very similar to PRM. The operation of eavesdropping passing by data packets is equivalent overhearing active watermarks, and advertising shorter paths to the source is similar to advertising an inactive watermark. An obvious difference is that PRM uses a mesh to forward data packets, which is more robust in a network with relatively high mobility. More importantly, PRM is based on a model that is independent of the underlying route discovery mechanism and is more generic. For example, by giving higher priority to nodes with higher energy levels in the next hop selection operation, PRM can support energy-aware routing; by applying swarm intelligence [1] to control the frequency of watermark advertisement, the overhead of PRM may be further reduced, etc.

4. Simulation

We implemented PRM on Network Simulation *ns*2 [5], where the CMU implementation of AODV [8] is used as the base protocol. The performance of PRM-enhanced AODV has been compared with the original AODV protocol. The complete simulation results are yet to come. But our preliminary results obtained from a basic version of PRM without any enhancement suggest that it is a promising scheme.

The simulated network has 50 nodes randomly deployed in a $670 \times 670\,\mathrm{m}^2$ area, a uniform transmission range of 250 m, and a bandwidth of 2 Mb. The MAC layer protocol is IEEE 802.11. We use 10 CBR flows with 4 packet per second and 512 bytes per packet. Node movement follows the random waypoint model with a maximal speed of 10 m/s. Among the protocol parameters, ActPktNumThreshold is 1, and ActTimeWindow is 1 s. The simulation lasts 200 s. Table 1 shows the preliminary results obtained by a single run under such configuration. Compared with the original AODV protocol, the PRM-enhanced protocol has slightly higher delivery ratio and significantly lower average end-to-end delay. PRM has also

Table 1. Preliminary simulation result.

Protocol	*Delivery Ratio (%)*	*Average Delay (s)*	*Discovery Overhead (Pkt)*	*Maintenance Overhead (Pkt)*
AODV	89.00	0.299	2236	0
PRM	91.29	0.153	1688	5033

lower route discovery overhead in number of control packets. However, PRM has a higher route maintenance overhead.

We expect better results from the ongoing extensive simulation, where the two extensions of PRM are also implemented. Specifically, the first extension will improve the delivery ratio and reduce the route discovery cost; the second one will decrease the number of control packets. We also expect the performance of AODV to be worse in larger networks under a higher mobility level.

5. Conclusion

This paper proposes a hybrid routing scheme called PRM for *ad hoc* networks. It combines reactive route discovery with proactive route maintenance to achieve several desirable properties. PRM is not a stand-alone protocol but rather a replacement of the naive route maintenance in most existing reactive routing protocols. Compared with existing local route repair and optimization schemes, PRM uses much fresher routing information and, therefore, is much more effective in highly dynamic networks. By forwarding data on several optimal paths, PRM achieves better load distribution and higher route reliability. In reactive routing protocols, this high reliability implies less route discovery cost and route setup delay.

PRM has relatively low overhead. Only nodes in active routes and their neighbors exchange routing information. The route advertising frequency is proportional to the passing by traffic volume. This overhead can be further reduced by embedding route advertisement in data packets or MAC layer control packets. By exploiting communication locality and applying proactive maintenance to a confined area, we provide a new paradigm for high reliability, low overhead, multipath routing in *ad hoc* networks. Our future work includes a comprehensive simulation study of PRM, and variations of PRM based on more sophistical techniques.

Acknowledgements

This work was supported in part by NSF grants CCR 0329741, and EIA 0130806. Contact address: E-mail: jie@cse.fau.edu

References

[1] Amin, K.A., Mayes, J.T. and Mikler, A.R., 2001, Agent-based distance vector routing, *Proceedings of the 3rd International Workshop on Mobile Agents for Telecommunication Applications (MATA 2001), Lecture Notes in Computer Science (LNCS)*, **2164**, 41–50.

[2] Boppana, R.V. and Konduru, S.P., 2001, An adaptive distance vector routing algorithm for mobile, *ad hoc* networks, *IEEE Infocom*, April.

[3] Bulk, F., 2004, Understanding wireless mesh networks, *Mobile Pipeline*. http://www.mobilepipeline.com/howto/21600011.

[4] Castenada, R., Das, S.R. and Marina, M.K., 2002, Query localization techniques for on-demand routing protocols for mobile *ad hoc* networks, *ACM/Kluwer Wireless Networks (WINET) Journal*, **8**(2), 137–151.

[5] Fall, K. and Varadhan, K., 2002, *The ns Manual (formerly ns Notes and Documentation)*. The VINT Project, A collaboratoin between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC, Available from http://www.isi.edu/nsnam/ns/doc/

[6] Haas, Z.J. and Perlman, M.R., 1997, The zone routing protocol (ZRP) for *ad hoc* networks. IETF Internet Draft. http://www.ietf.org/internet-drafts/draft-ietf-manet-zone-zrp-00.txt

[7] Imielinski, T. and Korth, H., 1996, *Mobile Computing* (Kluwer Academic Publishers).

[8] Johnson, D.B., Broch, J., Hu, Y.-C., Jetcheva, J. and Maltz, D.A., 1999, *The CMU Monarch Project's Wireless and Mobility Extensions to ns. Technical Report* (Carnegie Mellon University).

[9] Johnson, D.B. and Maltz, D.A., 1996, In: T. Imielinski and H. Korth (Eds.) *Dynamic Source Routing in Ad hoc Wireless Networks*, chapter 5, 153–181 (Kluwer Academic Publishers).

[10] Lee, S.J. and Gerla, M., 2000, AODV-BR: Backup routing in *ad hoc* networks, *ACM/Kluwer Mobile Networks and Applications (MONET) Journal*, 1311–1316.

[11] Lee, S.J. and Gerla, M., 2001, SMR: Split multipath routing with maximally disjoint paths in *ad hoc* networks. *Proceedings of ICC 2001*, 3201–3205 (Finland: Helsinki).

[12] Marina, M.K. and Das, S.R., 2001, On-demand multipath distance vector routing for *ad hoc* networks. *Proceedings of the International Conference for Network Procotols (ICNP)*.

[13] Nasipuri, A., Castañeda, R. and Das, S.R., 2001, Performance of multipath routing for on-demand protocols in *ad hoc* networks, *ACM/Kluwer Mobile Networks and Applications (MONET) Journal*, **6**(4), 339–349.

[14] Park, V.D. and Corson, M.S., 1997, A highly adaptive distributed routing algorithm for mobile wireless networks, *Proceedings of the IEEE INFOCOM*, 1405–1413.

[15] Pearlman, M.R., Haas, Z.J., Sholander, P. and Tabrizi, S.S., 2000, On the impact of alternate path routing for load balancing in mobile *ad hoc* networks, *Proceedings of the ACM MobiHoc*, 3–10.

[16] Perkins, C.E. and Bhagwat, P., 1994, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, *Proceedings of the ACM SIGCOMM*, 234–244.

[17] Perkins, C.E. and Royer, E.M., 1999, *Ad hoc* on-demand distance vector routing, *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, 90–100.

[18] Ganz, A. and Xue, Q., 2002, Qos routing in mesh-based wireless networks, *International Journal of Wireless Information Networks*, **9**(3), 179–190.

[19] Raju, J. and Garcia-Luna-Aceves, J.J., 1999, A new approach to on-demand loop-free multipath routing, *Proceedings of the International Conference on Computer Communications and Networks (IC3N)*, 522–527.

[20] Vutukury, S. and Garcia-Luna-Aceves, J.J., 2001, MDVA: A distance-vector multipath routing protocol, *Proceedings of the IEEE INFOCOM*, 557–564.

[21] Wu, K. and Harms, J., 2002, Multipath routing for mobile *ad hoc* networks, *Journal of Communications and Networks*, **4**(1).

[22] Wu, S.-L., Ni, S.-Y., Tseng, Y.-C. and Sheu, J.-P., 2000, Route maintenance in a wireless mobile *ad hoc* network. *Proceedings of 33rd Hawaii International Conference on System Sciences*.

Author Queries

JOB NUMBER: MS 109486—
JOURNAL: GPAA

Q1 Kindly check the running title.