PAPER

Spiral-Multi-Path QoS Routing in a Wireless Mobile Ad Hoc Network*

Yuh-Shyan CHEN[†a)], ***Member and*** **Yu-Ting YU**[†], ***Nonmember***

SUMMARY A mobile ad hoc network (MANET) is characterized by multi-hop wireless links in the absence of any cellular infrastructure as well as by frequent host mobility. This paper proposes a SMPQ: Spiral-Multi-Path QoS routing protocol in a MANET, while the MAC sub-layer adopts the CDMA-over-TDMA channel model. This work investigates the bandwidth reservation problem of on-demand QoS routing in a wireless mobile ad-hoc network. The proposed approach increases the ability of a route to identify a robust path, namely a *spiral-multi-path*, from source host to destination host, in a MANET to satisfy certain bandwidth requirements. Two important contributions of the proposed spiral-multi-path are: (1) the spiral-multi-path strengthens route-robustness and route-stability properties and (2) the spiral-multi-path increases the success rate of finding the QoS route. Performance analysis results demonstrate that our SMPQ protocol outperforms other protocols.
key words: *CDMA-over-TDMA, mobile ad hoc network (MANET), mobile computing, quality-of-service (QoS) routing, multi-path, wireless network*

1. Introduction

Mobile ad hoc networks (MANETs) [8] consist of wireless hosts that communicate with each other in the absence of a fixed infrastructure. Due to considerations such as radio power limitations, power consumption, and channel utilization, a mobile host might not be able to communicate directly with other hosts in a single-hop fashion. A multi-hop scenario occurs where packets sent by a source host are retransmitted by several intermediate hosts before reaching the destination host. In a MANET, host mobility can cause frequent unpredictable topology changes, thus designs of MANET QoS routing protocols are more complicated than those of traditional networks because they require strong fault-tolerant capabilities. Since a MANET is characterized by its fast-changing topology, extensive research efforts have been devoted to the design of routing protocols for MANETs [4], [5]. These works have all addressed some of the following issues: how to discover a route from a source node to a destination; how to maintain a route while it is being used; and how to deliver data packets to the intended destination host. However, these protocols, when searching for a route to the destination, are only concerned with shortest-path routing and the availability of a multitude of routes in the MANET's dynamically changing environment. That is, only best-effort data traffic is provided. Connections with quality-of-service (QoS) guarantees in a MANET should be intensively investigated to support the fundamental operation for multimedia transmissions.

Some researchers have recently begun to study the QoS issue in MANETs [2], [3], [6], [7], [9]–[14]. A ticket-based QoS routing protocol was initially proposed in [2] to find a route satisfying certain bandwidth and delay constraints. The basic idea uses *ticket*-based scheme to limit the number of route-searching packets to avoid blind flooding operations. A multi-path QoS routing protocol was recently proposed by Liao et al. [9]. This multi-path approach uses multiple paths to satisfy the original bandwidth constraints. A bandwidth requirement is split into sub-bandwidth requirements, where each one is responsible for one of the multiple paths. This scheme obtains a better success rate of finding a QoS route, especially if the MANET is in a low-bandwidth environment. Observe that Liao's scheme [9] does not consider the radio interference problem. To further consider the radio interference problem, a CDMA-over-TDMA channel model is firstly assumed by Lin et al. in [10], [11], and a QoS routing protocol is developed in [10], [11]. Using the model, the use of a time slot on a link is only dependent on the status of its one-hop neighboring links. This QoS routing protocol calculates the end-to-end path bandwidth in a MANET, where MAC-sublayer is the CDMA-over-TDMA channel model. To support the mobility-tolerant capability, Chen et al. [5] recently presented a robust spiral-path routing protocol in [5] and robust spiral-path multicast protocol in [4]. However, these results are not the QoS-awareness routing protocols.

To support the QoS guarantees, we propose a scheme in this study to combine the spiral-path and multi-path routing result to develop a new QoS routing protocol, which inherits the robust path capability from the spiral-path schemes [4], [5] and the better success rate from the multi-path scheme [9]. Efforts will

Manuscript received March 4, 2002.
Manuscript revised May 6, 2003.
[†]The authors are with the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan, Republic of China.
a) E-mail: yschen@cs.ccu.edu.tw
[*]A preiminary version of this paper was appeared in Proceedings of IEEE ICOIN-16, Feb. 2002, Korea. This work is supported by the National Science Council of the Republic of China under Grant #NSC90-2213-E-305-001 and #NSC91-2213-E-194-042.

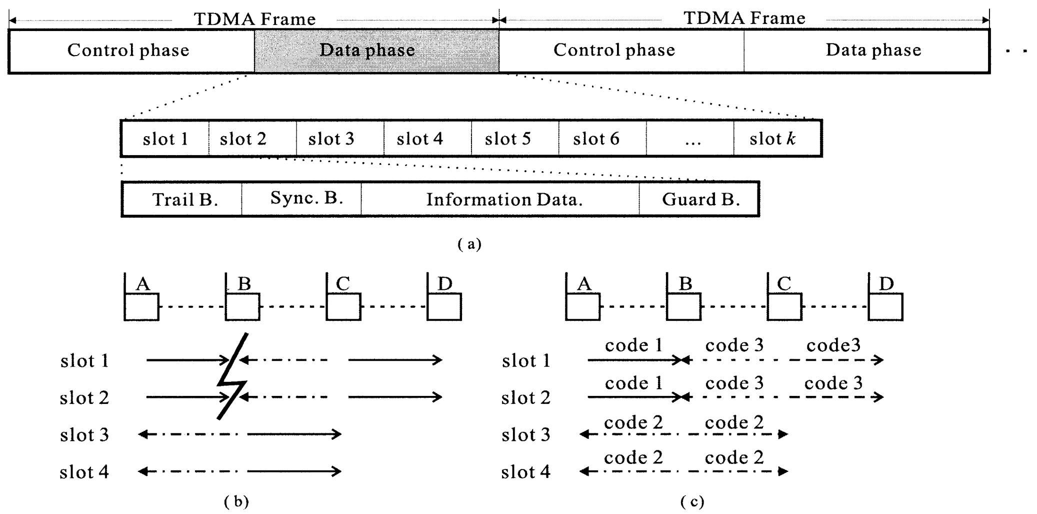

Fig. 1 (a) The TDMA frame structure, (b) the hidden-terminal problem, and (c) the CDMA-over-TDMA scheme.

be made in this paper to develop a QoS route with a high success rate of finding a QoS route with mobility-tolerant capability. In this paper, we propose a SMPQ: spiral-multi-path QoS routing protocol to identify a robust QoS path structure, namely a *spiral-multi-path*, from the source host to the destination host to satisfy certain bandwidth requirements in a MANET. The SMPQ protocol calculates the end-to-end path bandwidth, where the MAC sub-layer is the CDMA-over-TDMA channel model. Two important contributions of the proposed SMPQ protocol are that: (1) a spiral-multi-path strengthens the route-robustness and route-stability properties, and (2) a spiral-multi-path increases the success rate of finding QoS routes. Finally, performance analysis results demonstrate that our SMPQ protocol outperforms other protocols.

The rest of the paper is organized as follows. Section 2 presents basic ideas and motivations. Our protocol is developed in Sect. 3, and experimental results are discussed in Sect. 4. Section 5 concludes this paper.

2. Basic Ideas and Challenges

The ad hoc wireless network [8] is modeled as an undirected graph $G = (V, E)$, where V is set of $|V|$ nodes and E is a set of $|E|$ undirected links connecting nodes in V. Each node has a unique identifier and represents a mobile host having a wireless communication device with transmission range R, and infinite storage space.

The SMPQ (spiral-multi-path QoS) routing protocol is a fully distributed scheme, which aims to dynamically identify the *spiral-multi-path*, from the source host to the destination host, in a MANET to satisfy bandwidth requirements. Before formally defining the spiral-multi-path, assumptions of the network model are described as follows. The MAC sub-layer in our model is implemented using the CDMA-over-TDMA channel model. The CDMA-over-TDMA channel model is split into the different codes of CDMA-over-TDMA model and common code of CDMA-over-TDMA model. This paper considers the common code of CDMA-over-TDMA model, so the adjacent nodes can't simultaneously send data.

Each frame is divided into a control phase and a data phase as illustrated in Fig. 1(a). The CDMA-over-TDMA channel model is assumed to follow the same model as defined in [10], [11]. The CDMA (code division multiple access) is overlaid on top of the TDMA infrastructure. Multiple sessions can share a common TDMA slot via CDMA. Figure 1(b) shows a hidden-terminal problem. To overcome this problem, an orthogonal code used by a host should differ from that used by any of its two-hop neighbors as illustrated in Fig. 1(c). A code assignment protocol should be supported (this can be regarded as an independent problem, which can be found in [10], [11]). The bandwidth requirement is realized by reserving time slots on links. Under such a model, the use of a time slot on a link only depends on the status of its one-hop neighboring links. This model may be emulated by wireless LAN cards which follow the IEEE 802.11 standard [1]. Each data phase of a frame is assumed to be partitioned into κ time slots. A free time-slot list is defined as $F\{\alpha_1, \alpha_2, \cdots, \alpha_\kappa\}$ or denoted as $\{\alpha_1, \alpha_2, \cdots, \alpha_\kappa\}$. Let $(N_1, N_2, \cdots, N_k)$ denote a path from node $N_1, \cdots,$ to node N_k. For instance, $F\{1, 3, 4, 5, 6\}$ represents the free time slots $1, 3, 4, 5,$ and 6.

Observe that our spiral-multi-path scheme is constructed by combining the spiral-path and multi-path schemes [5], [9]. We initially reviewed the spiral-path and multi-path routing protocols in [5], [9]. First, a spiral-path routing was proposed by Chen et al. in [5]. The spiral-path is a robust routing structure, denoted as spiral-path P_k, as defined below.

Definition 1: Spiral-path [5]**:** Given a path P, every node of path P connects to the next k-hop node in path P by some extra, disjointed links. Therefore, path P and all extra links form a spiral-path P_k, where the length of the extra links is equal to k.

For instance, a spiral-path P_k is illustrated in Fig. 2(b). The route-robustness capability of the spiral-path is achieved by maintaining backup links. The greater the number of backup links there is, the higher the route robustness will be. If any link fails, the failed link is immediately replaced by a backup link. If we consider the spiral-path P_2, then the length of all extra links is equal to 2 as illustrated in Fig. 2(a). Observe that this paper considers the spiral-path P_2 using our spiral-multi-path routing.

Second, a multi-path scheme was proposed by Tseng et al. [9]. The multi-path scheme splits the bandwidth requirement into sub-bandwidth requirements to improve the success rate of identifying the QoS route. Figure 3(a) gives an example if the bandwidth requirement equals B units. Figure 3(b) shows that there is no uni-path which can satisfy the bandwidth requirement from nodes S to D. Figure 3(c) illustrates the use of multi-path routing, while each of the multi-paths is responsible for a small bandwidth requirement. The multi-path approach offers a higher success rate for finding a satisfactory QoS route than do those protocols which try to find a uni-path. Unfortunately, multi-path routing does not provide mobility-tolerant capability.

The spiral-path [5] is a uni-path routing scheme, but our proposed spiral-multi-path scheme is a multi-path routing scheme. Our scheme provides mobility-tolerant capability. The definition of a spiral-multi-path, denoted SMP, is very similar to the definition of the spiral-path P_k. A branch node is formally defined herein before defining the spiral-multi-path. A node is said to be a branch node if at least two disjointed paths exist from the same node, then such nodes are branch nodes. For instance as shown in Fig. 4, node G is a branch node due to the four disjointed paths from node A. Observe that, for spiral-paths [5], only uni-paths, between two adjacent branch nodes, act as primary paths for transmitting data. The spiral-multi-path is formally defined below.

Definition 2: Spiral-multi-path: A spiral-path is said to be a spiral-multi-path if the spiral-path uses multiple paths as the primary path to transmit data

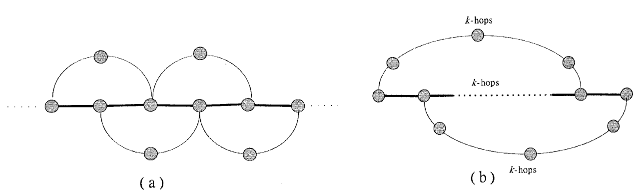

Fig. 2 A spiral-path approach.

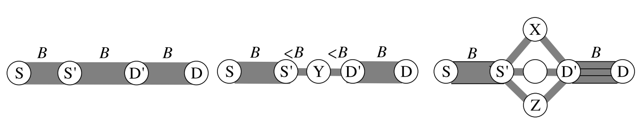

Fig. 3 A multi-path approach.

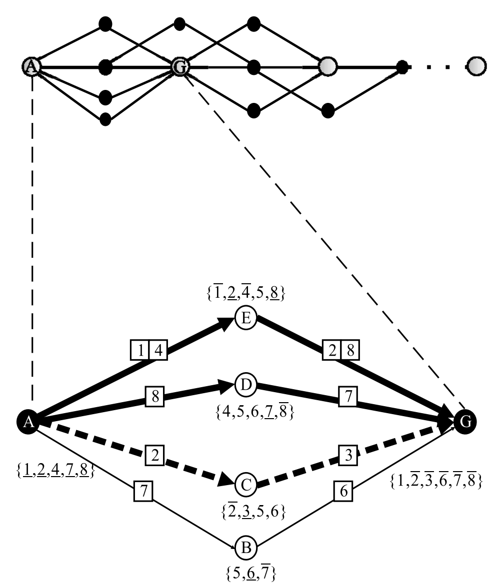

Fig. 4 Example of a spiral-multi-path.

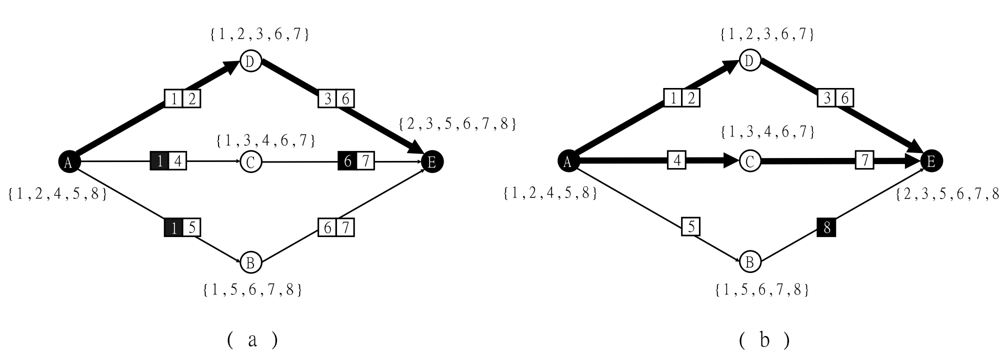

Fig. 5 Time-slot reservation scheme for a (a) uni-path and a (b) multi-path.

between two adjacent branch nodes.

The spiral-path is a special case of a spiral-multi-path if the primary path from A to G is a uni-path. However, the spiral-multi-path is seen as a spiral-path, except when using a multi-path as the primary path. As illustrated in Fig. 4, two disjoint paths (A, E, G) and (A, D, G) are used for transmitting data. The design challenge is to build in differences of time slot reservations for the uni-path and the multi-path between each pair of adjacent branch nodes. If we denote $\overline{XY}$ as the link between nodes X and Y, and given two adjacent branch nodes A and E as shown in Fig. 5(a), time slots reserved for $\overline{AD}$ and $\overline{AC}$ are $\{1, 2\}$ and $\{1, 4\}$, respectively. Slot 1 is shared by $\overline{AD}$ and $\overline{AC}$, since only one link is used in the uni-path routing. Similarly, time slots reserved for $\overline{DE}$ and $\overline{CE}$ are $\{3, 6\}$ and $\{6, 7\}$, while slot 6 is shared by $\overline{DE}$ and $\overline{CE}$. Observe that multi-path routing does not allow the same time slots to be reserved for different links between a pair of adjacent branch nodes. Figure 5(b) shows that the reserved time slots for $\overline{AD}$ and $\overline{AC}$ are $\{1, 2\}$ and $\{4\}$. Efforts are made to reserve time slots for spiral-multi-path routing.

3. Our Spiral-Multi-Path QoS Routing Protocol

This section introduces the SMPQ protocol by constructing a spiral-multi-path. Our on-demand QoS routing protocol is achieved by development of *retaining local link-state information, route-discovery*, *route-reply*, and *route-maintenance* phases. Initially, some local link-state information is collected for each node in a MANET. In the route-discovery phase, the QoS request packet floods the MANET to determine all available spiral-multi-paths from source to destination. A QoS route-reply packet, in the route-reply phase, is sent back from the destination to the source to confirm the reserved time slots. A route-maintenance phase is performed in order to maintain the spiral-multi-path. The robust-path capability is strengthened by using the spiral-multi-path.

3.1 Phase 1: Retaining Local Link-State Information

This phase aims not only to identify *branch nodes* and *branch supernodes,* but also to retain local link-state information. The identification of *branch nodes* and *branch supernodes* allows the possibility of constructing a spiral-multi-path. Moreover, local link-state information is used in the route discovery phase to reserve the path bandwidth for the spiral-multi-path.

Some notations are defined herein. If the free time slot lists of two neighboring nodes A and B are $\{\alpha_1, \alpha_2, \cdots, \alpha_{\kappa_1}\}$ and $\{\beta_1, \beta_2, \cdots, \beta_{\kappa_2}\}$, $\kappa_1 \neq \kappa_2$, we define an intersection function $\cap(\{\alpha_1, \alpha_2, \cdots, \alpha_{\kappa_1}\}, \{\beta_1, \beta_2, \cdots, \beta_{\kappa_2}\}) = (\gamma_1, \gamma_2, \cdots, r_{\kappa_3})$, where $(\gamma_1, \gamma_2, \cdots, r_{\kappa_3}) \in \{\alpha_1, \alpha_2, \cdots, \alpha_{\kappa_1}\}$, $(\gamma_1, \gamma_2, \cdots, r_{\kappa_3}) \in \{\beta_1, \beta_2, \cdots, \beta_{\kappa_2}\}$, and $\kappa_3 \leqq \min\{\kappa_1, \kappa_2\}$. Let $(\gamma_1, \gamma_2, \cdots, r_{\kappa_3})$ represent shared free time slots between nodes A and B. This indicates that time slots for communicating between A and B should be selected from $(\gamma_1, \gamma_2, \cdots, r_{\kappa_3})$. For example as shown in Fig. 6(a), the free time-slot lists of S and A are $\{1, 2, 4, 7, 8\}$ and $\{1, 3, 4, 5, 6\}$, respectively, so $\cap(\{1, 2, 4, 7, 8\}, \{1, 3, 4, 5, 6\}) = (1, 4)$ as illustrated in Fig. 6(b). Node S communicates with node A in time slots $(1, 4)$.

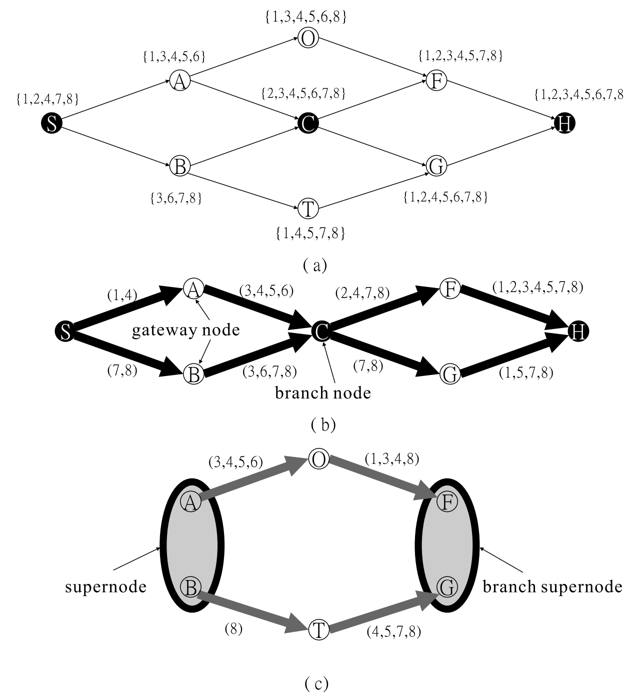

Fig. 6 Identifying branch nodes/supernodes and their link bandwidths.

Recalling the definition of a branch node, Fig. 6(a) indicates that nodes S and C are branch nodes and two disjointed paths (S, A, C) and (S, B, C) exist between S and C. In the following, we present how to identify the branch nodes in a MANET. The identification is accomplished by periodically flooding a Beacon packet within two hops. The beacon format is denoted as **Beacon**(*hopnumber*, *path_record, free_slots*$_{path\_record[i]}$), where *hopnumber* denotes the packet lifetime by limiting the number of hops, *path_record* is a path using an array to store it and records the history-path from the beacon-initiation node to the current node, and *free_slots*$_{path\_record[i]}$ records the free time-slot list for each node *path_record*$[i]$. The time interval between two Beacon packets is a variable parameter, where the time interval is depended on the mobility status. This indicates that the time interval is long if the mobility is low, and the time interval is short if the mobility is high. We formally describe the identifying operation of the branch node below.

A1) Each node N initiates and floods a **Beacon** ($hopnumber--$, $path\_record[1]=[N]$, $free\_slots_N$), where $hopnumber$= 3 (or 4) records the free time-slot list into $free\_slots_N$, and re-forwards the **Beacon** packet to all neighboring nodes. The flooding process is repeatedly performed until $hopnumber$= 0, while $path\_record$ records the history-path from node N to the current node and $free\_slots_{path\_record[i]}$ records the free time-slot list in node $path\_record[i]$.
- For instance as shown in Fig. 6(a), if a **Beacon**$_1$ packet travels along path (S, A, C), node C eventually receives the **Beacon**$_1$($path\_record = [S, A, C]$, $free\_slots_i$), where $free\_slots_S = \{1, 2, 4, 7, 8\}$, $free\_slots_A = \{1, 3, 4, 5, 6\}$, and $free\_slots_C = \{2, 3, 4, 5\ , 6, 7, 8\}$.

A2) The shared time-slot list of a link bandwidth in path $path\_record$ is obtained by calculating the intersection function $\cap(free\_slots_{path\_record[i]}, free\_slots_{path\_record[]i+1})$, where $i \geqq 1$.
- For instance as shown in Fig. 6(b), the shared time-slot list of link $\overline{SA}$ is $\cap(free\_slots_{path\_record[1]}, free\_slots_{path\_record[2]}) = \cap\,(free\_slots_{\ path\_record_S}, free\_slots_{path\_record_A}) = \cap(\{1, 2, 4, 7, 8\}, \{1, 3, 4, 5, 6\}) = (1, 4)$.

A3) Let a node receive two different **Beacon** packets, denoted as **Beacon**$_1$ and **Beacon**$_2$. If $path\_record[1]$ of **Beacon**$_1$ is equal to $path\_record[1]$ of **Beacon**$_2$, then the node is a branch node.
- For instance as shown in Fig. 6(b), node C also receives a second **Beacon**$_2$($path\_record = [S, B, C]$, $free\_slots_i$) from node S, where $free\_slots_S = \{1, 2, 4, 7, 8\}$, $free\_slots_B = \{3, 6, 7, 8\}$, and $free\_slots_C = \{2, 3, 4, 5\ , 6, 7, 8\}$. Node C is a branch node since $path\_record[1]$ of **Beacon**$_1$ = $path\_record[1]$ of **Beacon**$_2$ = S and $path\_record[3]$ of **Beacon**$_1$ = $path\_record[3]$ of **Beacon**$_2$ = C.

Given a pair of branch nodes, a node is called a *gateway node* if the node can communicate with both branch nodes. In the following, we formally define the supernode and the branch supernode.

Definition 3: Supernode: All of the gateway nodes between a pair of branch nodes are logically said to be a supernode. Let $<\alpha_1, \alpha_2, \cdots, \alpha_{\kappa_1}>$ denote a supernode containing nodes $\alpha_1, \alpha_2, \cdots, \alpha_{\kappa_1}$.

For instance in Fig. 6(b), nodes A and B are gateway nodes, and Fig. 6(c) illustrates that gateway nodes A and B form supernode $S = <A, B>$, and supernode $S' = <F, G>$.

Definition 4: Branch supernode: A supernode is said to be a branch supernode if at least two disjointed paths exist from the same supernode.

For instance as shown in Fig. 6(c), supernode $S' = <F, G>$ is a branch supernode since there are two different paths from supernode $S = <A, B>$. Similarly, supernode S is also a branch supernode. We now describe how to identify a branch supernode and its corresponding link-state information.

B1) If branch node E receives **Beacon** packets from two neighboring branch nodes E' and E'', two sets of nodes are formed. One supernode, denoted α, is a gateway node between nodes E and E'. Another supernode, denoted β, consists of all gateway nodes between nodes E and E''. Each node which is a supernode must know all other nodes in the same supernode.
- For instance as illustrated in Fig. 6(b), node C is a branch node, and nodes S and H are neighboring branch nodes of node C. Two supernodes are $<A, B>$ and $<F, G>$, respectively.

B2) Each node A in a supernode periodically floods a **Beacon**($hopnumber--$, $path\_record[1] = [A]$, $free\_slots_N$) packet to an other neighboring supernode β, where $A \in$ supernode α.
- For instance as shown in Fig. 6(c), node $A, \in$ supernode $<A, B>$, sends a **Beacon** packet through node O to node F which $\in$ supernode $<F, G>$. Note that node F receives **Beacon**($path\_record = [A, O, F]$, $free\_slots_i$) from node S, where $free\_slots_A = \{1, 3, 4, 5, 6\}$, $free\_slots_O = \{1, 3, 4, 5, 6, 8\}$, and $free\_slots_F = \{1, 2, 3, 4, 5, 7, 8\}$.

B3) This step is the same as step **A2**.
- For instance as shown in Fig. 6(c), the free time-slot list of link $\overline{AO}$ is $\cap(free\_slots_{path\_record[1]}, free\_slots_{path\_record[2]}) = \cap\ (\ free\_slots_{path\_record_A}, free\_slots_{path\_record_O}) = \cap(\{1, 3, 4, 5, 6\}, \{1, 3, 4, 5,\ 6,\ 8\}) = (3, 4, 5, 6)$.

B4) If a supernode receives at least two **Beacon** packets sent from the same supernode, then such a supernode is a branch supernode.
- For instance as shown in Fig. 6(c), two **Beacon** packets travel along path (A, O, F) and (B, T, G), since nodes A and $B \in$ supernode $<A, B>$ and nodes F and $G \in$ supernode $<F, G>$, therefore both supernodes $<A, B>$ and $<F, G>$ are branch supernodes.

3.2 Phase 2: Route-Discovery Phase

Herein, we discuss the discovery phase of the SMPQ routing protocol. The discovery phase is built by a subpath bandwidth reservation operation. This operation is repeatedly applied to each pair of branch nodes and then applied to each pair of branch supernodes. Our purpose is to pre-reserve time slots during the construction of the spiral-multi-path. The time-slot allocation problem is a NP-complete problem [9]. For this phase, we present a heuristic algorithm to efficiently overcome

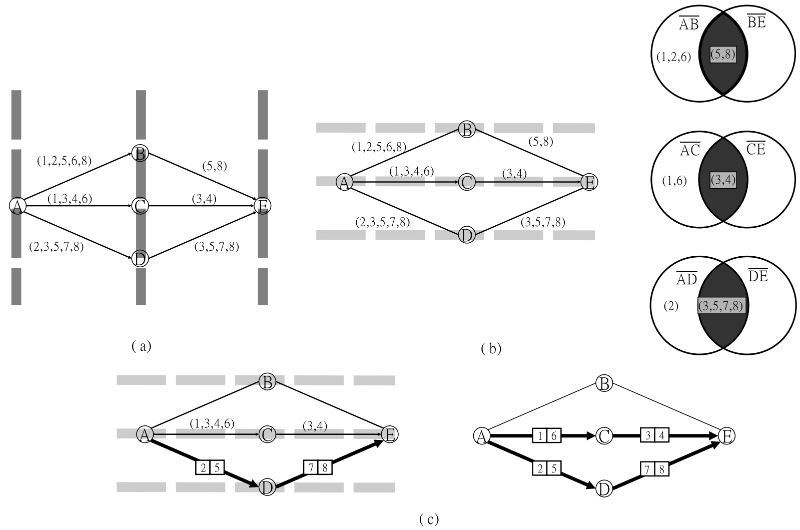

Fig. 7 A basic sub-path bandwidth reservation operation on a pair of branch nodes.

the problem. A sub-path bandwidth reservation operation is given herein.

A Sub-Path Bandwidth Reservation Operation is performed between two adjacent branch nodes B and B'. Assume that there are κ disjointed paths between B and B', where each sub-path length is equal to 2. If the bandwidth requirement is γ, we reserve time slots on α disjointed paths, where $\alpha \leqq \kappa$, such that the total path bandwidth is equal to γ. A priority value for each path is determined, so that we reserve time slots on these α disjointed paths, $\alpha \leqq \kappa$, in order of the priority value. The step **C1** may produce many disjointed paths with the same maximum sub-path bandwidth. Then, step **C2** aims to provide a priority rule among these disjointed paths with same maximum sub-path bandwidth to determine how to allocate the time slots.

C1) Each path calculates its own maximum sub-path bandwidth between nodes B and B'. The maximum sub-path bandwidth is obtained by calculating the maximum sub-path bandwidth without considering radio interference with other paths. Observe that the time slot allocation of adjacent two links must be different, where the MAC sublayer is adopted by the CDMA-over-TDMA channel model. A path has the highest priority value if that path has the highest maximum sub-path bandwidth.

- For example as shown in Fig. 7(a), there are three paths $(A, B, E), (A, C, E)$, and (A, D, E) between branch nodes A and E. Time slots $\{1, 2\}$ can be allocated on link $\overline{AB}$, and time slots $\{5, 8\}$ is allocated into link $\overline{BE}$. Similarly, slots $\{1, 6\}$ and $\{3, 4\}$ can be allocated into links $\overline{AC}$ and $\overline{CE}$, respectively. Moreover, time slots $\{3, 5\}$ and $\{7, 8\}$ can be allocated into links $\overline{AD}$ and $\overline{DE}$, or time slots $\{7, 8\}$ and $\{3, 5\}$ can be allocated into links $\overline{AD}$ and $\overline{DE}$. Therefore, the maximum sub-path bandwidths of paths $(A, B, E), (A, C, E)$, and (A, D, E) are 2, 2, and 2, respectively, so the three paths have the same priority value.

C2) After the step **C1**, if there are $\chi \leqq \kappa$ disjointed paths with the same maximum sub-path bandwidth, we still need to calculate the priority value for $\chi \leqq \kappa$ disjointed paths. These priority values are used to determine a priority sequence, while the priority sequence is the order of allocating time slots among χ disjointed paths with the same maximum sub-path bandwidth. We adopt a set-intersection operation for the link bandwidth of χ disjointed paths to determine the priority values. If (X, Y, Z) denotes a path of $\chi \leqq \kappa$ disjointed paths, then $\cap(\overline{XY}, \overline{YZ})$ is calculated. Each set-intersection operation produces intersection elements. The greater the number of intersection elements there is, the more free time-slots there will be. A path has a higher priority value if this path has a greater number of intersection elements.

- For instance as shown in Fig. 7(b), there are three disjointed paths (A, B, E), (A, C, E), and (A, D, E). Let $\cap(\overline{AB}, \overline{BE})$ denote the set-intersection operation on the link bandwidth of links $\overline{AB}$ and $\overline{BE}$. Therefore, $\cap(\overline{AB}, \overline{BE}) = \cap((1, 2, 5, 6, 8), (5, 8)) = (5, 8)$, $\cap(\overline{AC}, \overline{CE}) = \cap((1, 3, 4, 6), (3, 4)) = (3, 4)$, and $\cap(\overline{AD}, \overline{DE}) = \cap((2, 3, 5, 7, 8), (3, 5, 7, 8)) = (3, 5, 7, 8)$. The priority value of multi-path (A, D, E) is larger than those of (A, C, E) and (A, B, E), since $|(3, 5, 7, 8)| > |(3, 4)|$ and $|(5, 8)|$. Finally, a possible priority sequence is determined to be $(A, D, E) > (A, C, E) = (A, B, E)$.

C3) After a priority sequence is determined by steps **C1** and **C2**, we apply the bandwidth reservation operation for paths between branch nodes accord-

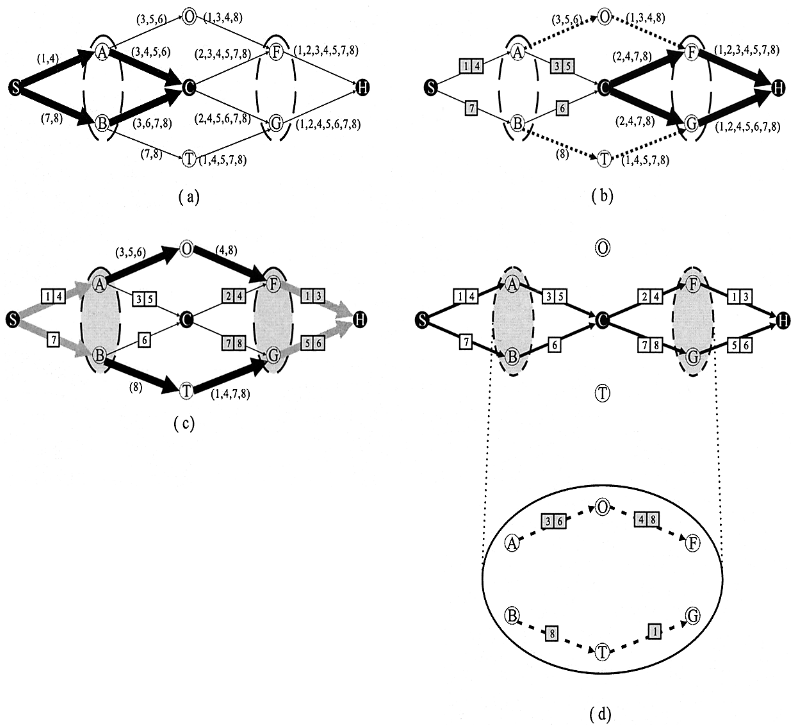

Fig. 8 Route-discovery operation.

ing to the priority sequence. We repeatedly reserve time slots until total path bandwidths are equal to or larger than the original bandwidth requirement γ.

- For instance as shown in Fig. 7(c), if the bandwidth requirement is 4 slots, we first reserve $(2, 5)$ and $(7, 8)$ to path (A, D, E). We then reserve $(1, 6)$ and $(3, 4)$ to path (A, C, E), such that the total path bandwidths is 4 slots.

We now present the route-discovery operation by repeatedly performing the basic time-slot reservation operation. One important notation is defined herein. Let $[\overline{\alpha_1},\ \underline{\beta_1}, \overline{\alpha_2},\ \underline{\beta_2}, \cdots \overline{\alpha_\kappa},\ \underline{\beta_\kappa}]$ denote a feasible path with its path bandwidth from source to destination, where $\overline{\alpha_i}$ denotes a link bandwidth between two branch nodes and $\underline{\beta_i}$ represents a link bandwidth between two branch supernodes. For instance, a path bandwidth $[\overline{3}, \underline{3}, \overline{4}]$ from nodes S to H is given in Fig. 8. The destination eventually acquires many paths with corresponding path bandwidths; a final path with a suitable path bandwidth is determined based on the bandwidth requirement and mobility-tolerant capability, which are described in the QoS route-reply phase. We now formally describe the construction of a possible spiral-multi-path and its path bandwidth.

D1) The source node initiates and floods the MANET with a QoS route-discovery packet, and then performs a basic time-slot reservation operation between the source node and all neighboring branch nodes, where each path records its link bandwidth $[\overline{\alpha_1}]$ in its route-discovery packet.

- For instance as shown in Fig. 8(a), $[\overline{3}]$ is obtained from nodes S to C through multi-paths (S, A, C) and (S, B, C), where (S, A, C) has two time-slots and (S, B, C) has one time-slot.

D2) A branch node, which receives a route-discovery packet with record $[\overline{\alpha_1}]$, continues to perform a basic time-slot reservation operation between the next possible neighboring branch node, and the path bandwidth becomes $[\overline{\alpha_1}, \overline{\alpha_2}]$. It should be noted that some free time-slots must be modified before performing the basic time-slot reservation operation. This is because these time-slots have already been allocated in step **D1**.

- For instance as shown in Fig. 8(b), consider multi-paths (C, F, H) and (C, G, H). Note that the link bandwidth of $\overline{CF}$ changes from $(2, 3, 4, 5, 7, 8)$ to $(2, 4, 7, 8)$ since $(3, 5)$ are already allocated to link $\overline{AC}$; and the link bandwidth of $\overline{CG}$ changes from $(2, 4, 5, 6, 7, 8)$ to $(2, 4, 7, 8)$ since (5) and (6) are already allocated to multi-paths (S, A, C) and (S, B, C). Observe that the slot reservation of $\overline{CF}$ must both consider the free time slots on links $\overline{BC}$ and $\overline{AC}$. Time slot (6) is already allocated into $\overline{BC}$, and not affect the result for $\overline{CF}$ since the free time slots of $\overline{CF}$ does not have the time slots (6). Therefore, $[\overline{3}, \overline{4}]$ is obtained.

D3) Two supernodes S and S' based on steps **D1** and **D2** are formed, and a basic time-slot reservation operation is similarly performed between these two

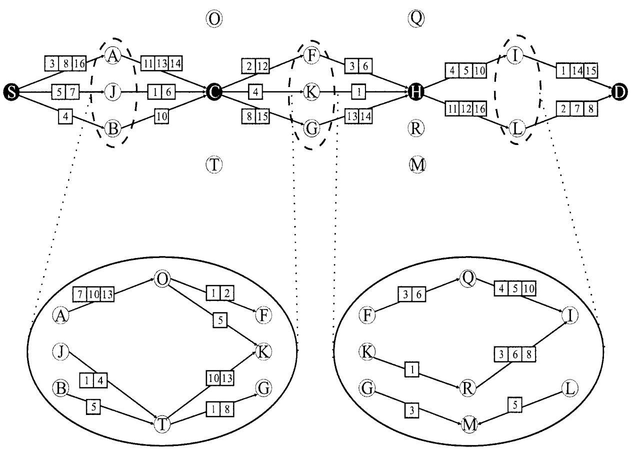

Fig. 9 A spiral-multi-path with path bandwidth $[\overline{6}, \underline{6}, \overline{5}, \underline{5}, \overline{6}]$.

supernodes. Surely, some free time-slots must be modified before performing the basic time-slot reservation operation. This is because these time-slots have already been allocated in steps **D1** and **D2**. Therefore, we let the path bandwidth become $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}]$. It should be noted that each node e in supernode S must possibly find multi-paths (or a uni-path) to any node in supernode S' such that the total sub-path bandwidths of the multi-paths are equal to the sub-bandwidth requirement of node e.

- For instance as shown in Fig. 8(c), a path bandwidth $[\overline{3}, \underline{3}, \overline{4}]$ is constructed. Another instance is given in Fig. 9, where node A is in supernode $<A, J, B>$ and the other supernode is $<F, K, G>$. The sub-bandwidth requirement of node A is 3 since time slots $(3, 8, 16)$ are reserved for link $\overline{SA}$. Therefore, node A finds multi-paths (A, O, F) and (A, O, K), and the total sub-path bandwidth is equal to 3. Note that since the bandwidth of links $\overline{FH}$ and $\overline{KH}$ are 2, the bandwidths of both links $\overline{OF}$ and $\overline{OK}$ are required to equal to 2. This is because the final time-slot reservation must be exactly confirmed in the route-reply phase.

D4) Repeatedly executing the steps **D2** and **D3** until arriving at the destination, a path with path bandwidth $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots \overline{\alpha_\kappa}, \underline{\beta_\kappa}]$ is eventually obtained at the destination.

Intuitively, many paths, each containing spiral-multi-paths with path bandwidths of $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots, \underline{\beta_{\kappa-1}}, \overline{\alpha_\kappa}]_i$, $i \geqq 1$, can be obtained at the destination node. For another example, a path from S to D with a path bandwidth of $[\overline{6}, \underline{6}, \overline{5}, \underline{5}, \overline{6}]$ is illustrated in Fig. 9. A simple selection criterion is given in the QoS route-reply phase to choose a suitable spiral-multi-path as the final one.

3.3 Phase 3: Route-Reply Phase

Assume that there are many paths, each one with a path bandwidth $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots, \underline{\beta_{\kappa-1}}, \overline{\alpha_\kappa}]_i, i \geqq 1$, that are obtained at the destination node. Before sending the packet back to the source, the destination must select a final spiral-multi-path and reserve time-slots for the final one while releasing time-slots for the unused spiral-multi-paths.

Consider that $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots, \underline{\beta_{\kappa-1}}, \overline{\alpha_\kappa}]_i, i \geqq 1$ represents a spiral-multi-path. Given that the bandwidth requirement from the source is γ, the path-selection criteria of the final spiral-multi-path are given herein.

E1) Mobility-tolerant capability for gateway nodes: A spiral-multi-path satisfies the condition

$$[\overline{\alpha_j}]_i \geqq \gamma, \text{for all } 1 \leqq j \leqq \kappa.$$

E2) Mobility-tolerant capability for branch nodes: A spiral-multi-path satisfies the condition

$$[\underline{\beta_j}]_i \geqq \gamma, \text{for all } 1 \leqq j \leqq \kappa - 1.$$

Our spiral-multi-path has to at least satisfy condition **E1** such that a QoS route is found which possibly contains the on-line recovery capability for the gateway node. There is the on-line recovery capability for any branch node if the spiral-multi-path only satisfies condition **E1**. Notably, all backup paths in $[\underline{\beta_j}]_i$ are used to recover the failed node, which is a branch node. It is possible to find a path without such on-line recovery capability for a branch node. For instance, given $\gamma = 3$, a path with $[\overline{6}, \underline{0}, \overline{5}, \underline{0}, \overline{6}]$ satisfies condition **E1**, but a path with $[\overline{6}, \underline{6}, \overline{2}, \underline{5}, \overline{6}]$ does not. Further, if we hope our spiral-multi-path has the on-line recovery capability for gateway/branch nodes, the spiral-multi-path must satisfy both conditions **E1** and **E2**. For instance, given $\gamma = 3$, a path with $[\overline{6}, \underline{6}, \overline{5}, \underline{5}, \overline{6}]$ satisfies the E1 and E2 conditions. Given a spiral-multi-path with path bandwidth $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots, \underline{\beta_{\kappa-1}}, \overline{\alpha_\kappa}]_i, i \geqq 1$, an average extra bandwidth or EB_{av} of the spiral-multi-path is calculated by

$$\begin{aligned} EB_{av} &= \overline{EB}_{av} + \underline{EB}_{av} \\ &= \frac{\sum_{i=1}^{\kappa} |\overline{\alpha_i} - \gamma|}{\kappa} + \frac{\sum_{i=1}^{\kappa-1} |\underline{\beta_i} - \gamma|}{\kappa - 1}, \end{aligned}$$

where $\overline{EB}_{av}$ denotes the average extra bandwidth for every pair of branch nodes, and $\underline{EB}_{av}$ represents the average extra bandwidth for every pair of branch supernodes. Recalling the above example, a path with $[\overline{6}, \underline{6}, \overline{6}, \underline{5}, \overline{6}]$ satisfies both conditions **E1** and **E2**, and its $EB_{av} = \overline{EB}_{av} + \underline{EB}_{av} = \frac{|6-3|+|5-3|+|6-3|}{3} + \frac{|6-3|+|5-3|}{2} = 5.16$. Notice that the EB_{av} is used to make a path-selection decision if there are many spiral-multi-paths which satisfy both conditions **E1** and **E2**. We observe that the larger the value of EB_{av} is, the higher the QoS route stability will be. To maintain the QoS route stability, a spiral-multi-path with high EB_{av} is selected as our final spiral-multi-path. In particular, a spiral-multi-path has no mobility-tolerant capability if $\overline{EB}_{av} = \underline{EB}_{av} = 0$. We further discuss the impact of performance under different ratios of $\overline{EB}_{av}$ and $\underline{EB}_{av}$ in our simulation.

A simple route reply phase dispatches a route-reply packet from destination to source, in order to reserve all time-slots in the selected spiral-multi-path with path bandwidth $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots, \underline{\beta_{\kappa-1}}, \overline{\alpha_\kappa}]_j$,where $1 \leqq j \leqq i$. Additionally, the destination also dispatches packets to all other spiral-multi-paths releasing their bandwidth resource. Each spiral-multi-path with path bandwidth $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots, \underline{\beta_{\kappa-1}}, \overline{\alpha_\kappa}]_j$,where $1 \leqq j \leqq i$, merely needs to reply a packet to reserve the path bandwidth $[\overline{\gamma}, \underline{\gamma}, \overline{\gamma}, \cdots, \underline{\gamma}, \overline{\gamma}]$ if the bandwidth requirement is γ. However, an extra link bandwidth $|\overline{\alpha}_x - \gamma|$ or $|\underline{\beta_x} - \gamma|$ is reserved to provide mobility-tolerant capability. After a final spiral-multi-path with path bandwidth $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots \overline{\alpha_{\kappa-1}}, \underline{\beta_{\kappa-1}}, \overline{\alpha_\kappa}]$ is determined, the reply packet is dispatched according to the following rules.

F1) The destination initiates a reply packet to a pre-branch node, if there is a uni-path with a pre-reserved time-slot number at least larger than γ, then the uni-path is chosen as the primary path by reserving γ time slots. All other possible multi-paths are backup paths. If there is no uni-path whose pre-reserved time-slot number is smaller than γ, then multi-paths are used as the primary paths. A greater number of pre-reserved time-slot path are reserved first, which aims to use a fewer number of multi-paths. Observe that a path with a fewer number of pre-reserved time slots will be chosen as the last of our multi-primary-paths. This is because we can retain other paths with more time-slots to serve as backup paths.

- For instance as shown in Fig. 10 if a path bandwidth $[\overline{6}, \underline{6}, \overline{5}, \underline{5}, \overline{6}]$ is determined, destination node D sends a reply packet to a pre-branch node H, which is a uni-path (D, I, H) with three time-slots. Another path (D, L, H) with three time-slots is a backup path.

F2) The branch node continually re-forwards the reply packet to the next pre-branch node, and then performs a similar operation as described in step **F1**.

- For instance as shown in Fig. 10, branch node H sends the reply packet to branch node C for which there is no path with three time-slots. Therefore, multi-paths (H, F, C) and (H, K, C) are used in order to reserve three time-slots. However, path (H, G, C) with two time slots is also reserved as a backup path. Observe that we do choose path (H, K, C) as one of primary paths, and let (H, G, C) be a backup path. This can achieve a better mobility-tolerant capability than if we use path (H, G, C) as the primary path.

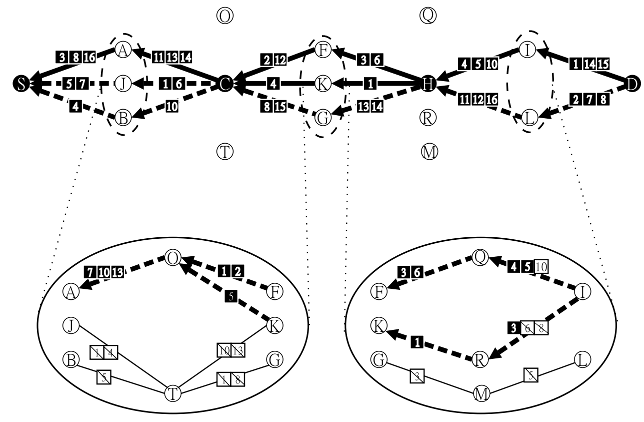

Fig. 10 QoS route-reply operation.

F3) Two supernodes S and S' based on steps **F1** and **F2** are formed. Every node in the primary paths of supernode S must try to connect to nodes in the primary paths of supernode S' under the bandwidth requirement γ. Note that all multi-paths in step **F3** belong to backup paths.

- For instance as shown in Fig. 10, node $I \in <I, L>$ is in a primary path connecting nodes F and K, which are in a primary path of supernode $<F, K, G>$. Multi-paths (I, Q, F) and (I, R, K) with two and one time slots, respectively, are possibly reserved as backup paths.

F4) Repeatedly execute the steps **F2** and **F3** until reaching the source, such that a path with path bandwidth $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots \overline{\alpha_\kappa}, \underline{\beta_\kappa}]$ is constructed.

Eventually, a spiral-multi-path is constructed with reserved a path bandwidth of $[\overline{\alpha_1}, \underline{\beta_1}, \overline{\alpha_2}, \underline{\beta_2}, \cdots \overline{\alpha_\kappa}, \underline{\beta_\kappa}]$ under the bandwidth requirement γ. A full example is given in Fig. 10 for reserving a spiral-multi-path with path bandwidth $[\overline{6}, \underline{6}, \overline{5}, \underline{5}, \overline{6}]$.

3.4 Phase 4: Route-Maintenance Phase

The main contribution of our proposed protocol is to provide on-line route-recovery capability. A node is said to be a failed node if it is moving out of the original transmission radius or if it has already failed. We show how to achieve the on-line recovery capability such that the QoS route can be continually maintained. The on-line recovery capability of our proposed protocol is achieved using a multi-path and spiral-path replacement strategy. The proposed replacement strategy is given according to the different roles of a failed node.

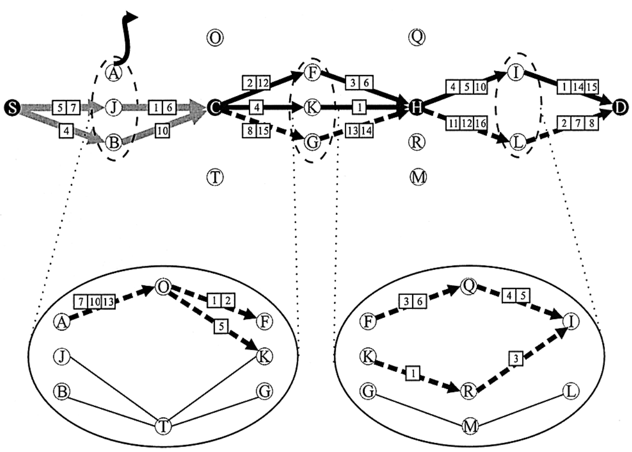

Fig. 11 Tolerating a failed gateway node.

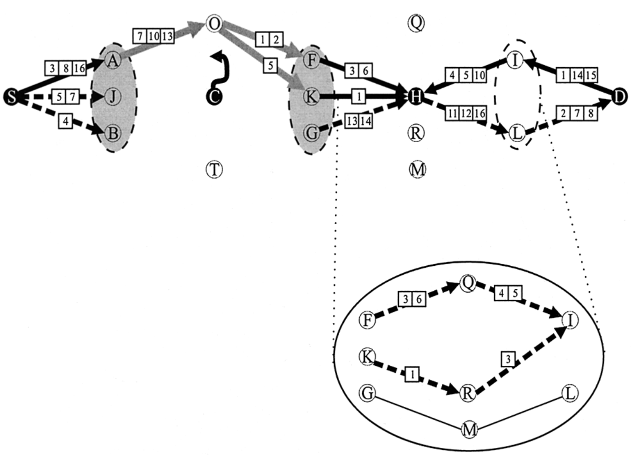

Fig. 12 Tolerating a failed branch node.

G1) If the failed node is a gateway node, all backup multi-paths between branch nodes are used to replace the failed path. Here we use an ACK. packet between branch nodes to make sure that the data message has been successfully received by the downstream branch nodes. That is, if an upstream branch node waits for a period of time and still does not receive an ACK. packet from the downstream branch node, then it detects a failed path, and it will initiate the backup multi-paths to replace the failed path.

- For instance as shown in Fig. 11, if node A is a failed node and path (S, A, C) with sub-path bandwidth $= 3$ is broken, then the backup paths (S, J, C) and (S, B, C) with a total sub-path bandwidth $= 3$ are used to replace the failed path.

G2) If the failed node is a branch node, all backup spiral-multi-paths between branch supernodes are used to replace the failed paths.

- For instance, as shown in Fig. 12, two supernodes are $< A, J, B >$ and $< F, K, G >$. If branch node C is a failed node and the multi-paths (A, C, F) and (A, C, K), whose path bandwidths $= 3$, are broken, the backup spiral-multi-paths (A, O, F) and (A, O, K), whose path bandwidths $= 3$, are immediately used to replace the failed paths and to maintain the bandwidth requirement.

4. Experimental Results

Our scheme mainly combines the spiral-path-based protocols [4], [5] and multi-path QoS routing protocol in [9]. This simulation mainly compares our scheme with Lin's scheme [10]. This is because that the spiral-path-based schemes [4], [5] are the routing protocols without providing the QoS guarantees. In addition, the multi-path QoS routing protocol in [9] does not further consider the radio interference problem. That is, the Liao's scheme does not adopt the CDMA-over-TDMA channel model in the MAC-sublayer. Therefore, we compare our result with Lin's scheme [10] to demonstrate the efficiency.

To illustrate the efficiency of our proposed scheme and compare it to Lin's scheme [10], a simulator is developed. The simulator is simulated in a 1000×1000-m^2 area. The radio transmission range is 400 m. The data rate is 4 Mbit/s. The duration of each time slot of a time frame, as shown in Fig. 1(a), is assumed to be 5 ms, and the duration of a control slot is 0.1 ms. The source and destination are selected randomly. Once a QoS request is successful, a time slot is reserved for all the subsequent packets. The reservation is released when either the data transmission process is finished or the link is broken. The total simulation time is 10^6 ms. The simulation parameters in the simulator are given below.

- The *number of mobile hosts* ranges from 20 to 40.
- The *maximum number of time slots* ranges from 8, 12, to 16.
- Three *different bandwidth requirements* are 1, 2, and 4 slots, which are denoted Lin-x in Lin's scheme [10], and SMPQ-x in our proposed scheme, where $x = 1, 2$, or 4.
- The *network bandwidths* are denoted as low, medium, and high, which assume 25%, 50%, and 75% of the free time slots, respectively.

The mean values of inter-arrival times for the SMPQ-x and Lin-x schemes, where $x = 1$, 2, and 4, are 100, 50, and 25 ms, respectively. A packet is dropped if the packet staying in a node exceeds the maximal queuing delay time, which is set to four frame lengths (328 ms). Additionally, performance metrics of our simulation are defined below.

- *Success_Rate* (SR): the number of successful QoS route requests divided by the total number of QoS route requests from source to destination.

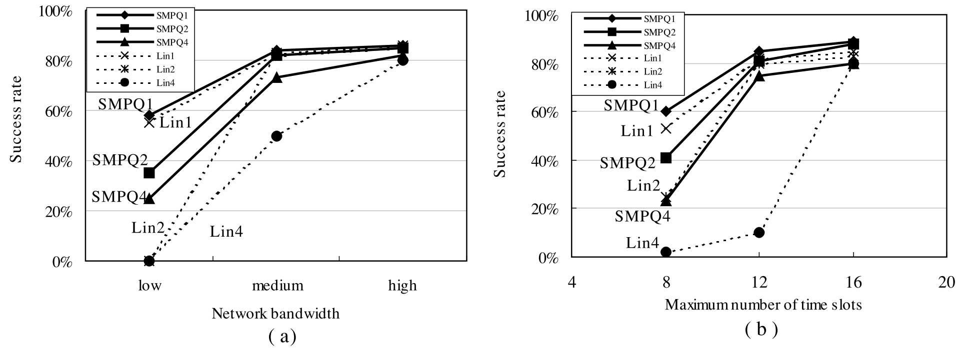

Fig. 13 Performance of success rate vs. effect of (a) network bandwidth and (b) maximum number of time slots.

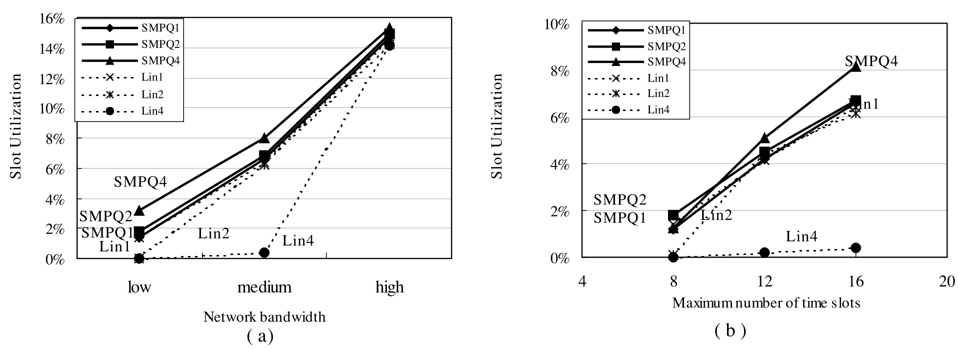

Fig. 14 Performance of slot utilization vs. effect of (a) network bandwidth and (b) maximum number of time slots.

- *Slot_Utilization* (SU): the average slot utilization of every link in all QoS routes.
- *OverHead* (OH): the number of packets used for constructing and maintaining the QoS route from source to destination.

It is worth mentioning that an efficient QoS routing protocol is achieved with a high success rate (SR), high slot utilization (SU), and low overhead (OH). In the following, we illustrate our simulation results of SR, SU, and OH from several prospects.

4.1 Performance of *Success Rate* (SR)

The observed results of our SMPQ and Lin's schemes are shown in Fig. 13 to reflect the performance of success rate vs. network bandwidth and number of time slots. Two types of effects are discussed.

1a) *Effects of Network Bandwidth*: Each value in Fig. 13(a) is obtained by assuming that the number of time slots is 16 and the number of mobile hosts is 30. From Fig. 13(a), we observe that for low network bandwidth, our SMPQ obtains a higher SR than does Lin's. For example, for low network bandwidth, SR values of SMPQ-1, SMPQ-2, and SMPQ-4 are 58%, 35%, and 25%, and SR values of Lin-1, Lin-2, and Lin-4 are 56%, 0%, and 0%, respectively. This indicates that our spiral-multi-path scheme more easily finds a QoS route than does Lin's scheme for a MANET with low network bandwidth. Basically, a higher network bandwidth can acquire a higher value of SR. Observe that our scheme and Lin's acquire the same values of SR in a MANET with high network bandwidth.

1b) *Effects of the Maximum Number of Time Slots*: Every value in Fig. 13(b) is obtained by assuming that the number of mobile hosts is 30 and the network bandwidth is medium. Figure 13(b) shows that our scheme has higher SRs than does Lin's under various maximum numbers of time slots. It can be observed that SR values of SMPQ-x are always larger than those of Lin-x, where x={1,2,4}. For instance, if the maximum number of time slots is 12, then the SR values of SMPQ-1, SMPQ-2, SMPQ-4, Lin-1, Lin-2, and Lin-4 are 84%, 81%, 72%, 80%, 79%, and 15%, respectively.

4.2 Performance of *Slot Utilization* (SU)

The observed results of performance of slot utilization vs. network bandwidth and number of time slots are given in Fig. 14. Two types of effects are discussed below.

2a) *Effects of Network Bandwidth*: The simulation as-

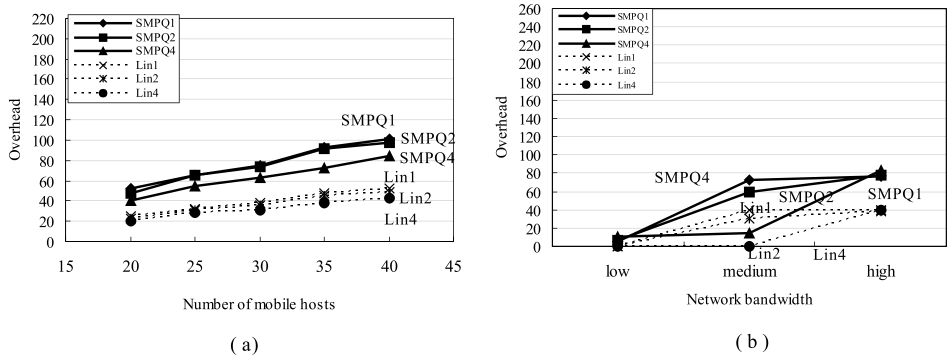

Fig. 15 Performance of overhead vs. effect of (a) number of mobile hosts and (b) network bandwidth.

sumption is the same as for case 1a). A higher SU implies a better scheme. Figure 14(a) shows that our scheme has better SU values than does Lin's for various network bandwidths under different bandwidth requirements. In a MANET with low network bandwidth, when the bandwidth requirement increases, Lin's SU values decrease and our SU values increase, since our scheme can acquire better SR values for low network bandwidth. For instance, with a low network bandwidth, SU values of SMPQ-1, SMPQ-2, SMPQ-4, Lin-1, Lin-2, Lin-4 are $6.8\%, 7\%, 8\%$, $6.4\%, 6.2\%$, and 0.6%, respectively.

2b) *Effects of the Maximum Number of Time Slots*: The simulation assumption is the same as that for case 1b). Figure 14(b) shows that our scheme obtains higher slot utilization than does Lin's under various maximum numbers of time slots. Observe that the SU is very low for Lin-4 as shown in Fig. 14(b), because it is not easy for Lin's scheme to find a QoS route if the bandwidth requirement is 4. For instance, if the maximum number of time slots is 12, then the SU values of SMPQ-1, SMPQ-2, SMPQ-4, Lin-1, Lin-2, and Lin-4 are $4.1\%, 4.3\%, 4.7\%, 4.23\%, 4.2\%$, and 0.12%, respectively.

Overall, our approach is a multi-path routing result, the multi-path routing scheme can naturally increase the success rate of constructing a QoS route. The best on slot utilization results in high success rate of a QoS route, especially for the case with the high bandwidth requirement. Therefore, the SMPQ4 has the best result on the slot utilization. Furthermore, Lin's scheme is the uni-path routing result, the success rate of searching a QoS route is lower than our approach. Therefore, the worst result on the slot utilization will be obtained if the bandwidth requirement is high, i.e., Lin4. Although our scheme has better results in success rate and slot utilization, our scheme has a heavier overhead of packets in a MANET than does Lin's scheme for the purpose of maintaining the spiral-multi-path routing structure as stated below.

4.3 Performance of $OverHead$ (OH)

The observed results of the performance of overhead vs. the number of mobile hosts and network bandwidth are illustrated in Fig. 15. Two kinds of effects are shown.

3a) *Effects of the Number of Mobile Hosts*: Each value in Fig. 15(a) is obtained by assuming that the number of time slots is 16 and the network bandwidth is medium. Since our scheme must collect time slot information for two- and three-hop neighbors, extra control packets are needed. Therefore, our scheme uses more packets than Lin's scheme. Although a lower OH implies a better scheme, our scheme has a higher OH value than does Lin's. Figure 15(a) shows that our scheme's overhead is about two times than that of Lin's scheme. However, this is the cost of our scheme. For instance, if the number of mobile hosts is 30, then OH values of SMPQ-1, SMPQ-2, SMPQ-4, Lin-1, Lin-2, and Lin-4 are $75, 74, 63$, $38, 36$, and 31, respectively.

3b) *Effects of Network Bandwidth*: The simulation assumption is the same that for case 1a). Figure 15(b) illustrates that the OH value of our protocol is also dependent on the conditions of network bandwidth. Figure 15(b) shows that our scheme has higher values for OH than does Lin's scheme for various network bandwidths. Observe that the OH values are the same for high network bandwidth as illustrated in Fig. 15(b). For instance, if both schemes are in high network bandwidth, then OH values of SMPQ-x and Lin-x are about 80 and 40, where $x = 1, 2$, and 4.

In summary, although our protocol has high overhead, our scheme achieves better success rate and slot utilization. This confirms that our proposed spiral-multi-path routing protocol outperforms the existing protocol.

5. Conclusions

In this paper, we present an efficient on-demand QoS routing protocol for a wireless mobile ad hoc network. This paper proposes SMPQ: Spiral-Multi-Path QoS routing protocol in the MANET. Our protocol aims to dynamically identify the *spiral-multi-path*, from source host to destination host, in a MANET to satisfy certain bandwidth requirements. Two important contributions of the spiral-multi-path are that (1) the spiral-multi-path enhances the QoS route robustness and route stability, and (2) the spiral-multi-path promotes the success rate of finding the QoS route. Performance analysis results demonstrate that the proposed SMPQ protocol outperforms other protocols.

References

[1] IEEE Std 802.11-1997, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Institute of Electrical and Electronics Engineers, Inc. USA.

[2] S. Chen and K. Nahrstedt, "Distributed quality-of-service routing in ad hoc networks," IEEE J. Sel. Areas Commun., vol.17, no.8, pp.1488–1505, Aug. 1999.

[3] S. Chen, K. Nahrstedt, and Y. Shavitt, "A QoS-aware multicast routing protocol," Proc. IEEE INFOCOM, pp.1594–1603, 2000.

[4] Y.-S. Chen, T.-S. Chen, and C.-J. Huang, "SOM: Spiral-fat-tree-based on-demand multicast protocol in a wireless ad-hoc network," Comput. Commun., vol.25, no.17, pp.1684–1695, Nov. 2002.

[5] Y.-S. Chen and K.-C. Lai, "MESH: Multi-eye spiral-hopping protocol in a wireless ad hoc network," IEICE Trans. Commun., vol.E84-B, no.8, pp.2237–2248, Aug. 2001.

[6] Y.K. Ho and R.S. Liu, "On-demand QoS-based routing protocol for ad hoc mobile wireless networks," Proc. Fifth IEEE Symposium on Computers and Communications (ISCC), pp.560–565, 2000.

[7] Y.C. Hu and D.B. Johnson, "Caching strategies in on-demand routing protocols for wireless ad hoc networks," Proc. MOBICOM, pp.231–242, 2000.

[8] D.B. Johnson and D.A. Maltz, "Dynamic source routing in ad hoc wireless networks," in Mobile Computing, ed. T. Imielinski and H. Korth, Chapter 5, pp.81–153, 1996.

[9] W.H. Liao, Y.C. Tseng, J.P. Sheu, and S.L. Wang, "A multi-path QoS routing protocol in a wireless mobile ad hoc network," Telecommun. Syst., vol.19, no.3-4, pp.329–347, 2002.

[10] C.R. Lin, "On-demand QoS routing in multihop mobile networks," Proc. IEEE INFOCOM 2001, pp.1735–1744, April 2001.

[11] C.R. Lin and J.S. Liu, "QoS routing in ad hoc wireless networks," IEEE J. Sel. Areas Commun., vol.17, no.8, pp.1426–1438, Aug. 1999.

[12] H.C. Lin and P.C. Fung, "Finding available bandwidth in multihop mobile wireless networks," Proc. IEEE Vehicular Technology Conference, pp.15–18, May 2000.

[13] A.K. Talukdar and B.R. Badrinath, "Rate adaptation schemes in networks with mobile hosts," Proc. MOBICOM, pp.169–180, 1998.

[14] Y.C. Tseng, J.P. Sheu, and C.Y. Lin, "A new mutli-channel MAC protocol with on-demand channel assignment for mobile ad hoc networks," Int'l Symp. on Parallel Architectures, Algorithms and Networks (I-SPAN), pp.232–237, 2000.

Yuh-Shyan Chen received the B.S. degree in computer science from Tamkang University, Taiwan, Republic of China, in June 1988 and the M.S. and Ph.D. degrees in Computer Science and Information Engineering from the National Central University, Taiwan, Republic of China, in June 1991 and January 1996, respectively. He joined the faculty of Department of Computer Science and Information Engineering at Chung-Hua University, Taiwan, Republic of China, as an associate professor in February 1996. He joined the Department of Statistic, National Taipei University in August 2000, and then joined the Department of Computer Science and Information Engineering, National Chung Cheng University in August 2002. Dr. Chen served as IASTED Technical Committee on Telecommunications for 2002–2005, Program Committee Member of IEEE ICPADS 2001, IASTED CCN 2002, MSEAT 2003, IEEE ICCCN 2001–2003, ICPP 2003, IEEE ICDCS 2004, and IASTED CSA 2004. He was a Workshop Co-Chair of the 2001 Mobile Computing Workshop, and Guest Editor of Journal of Internet technology, special issue on "Wireless Internet Applications and Systems" (2002) and Telecommunication Systems, special issue on "Wireless Sensor Networks" (2004). His paper wins the 2001 IEEE 15th ICOIN-15 Best Paper Award. His recent research include wireless LAN, mobile computing, wireless mobile ad-hoc network, wireless sensor network, and m-learning system. Dr. Chen is a member of the IEEE Computer Society, and Phi Tau Phi Society.

Yu-Ting Yu received the B.S. degree in statistics from National Taipei University, Taiwan, Republic of China, in June 2001. Her research includes wireless network analysis and mobile ad hoc networking.