Available online at www.sciencedirect.com
SCIENCE DIRECT®

Computer Communications 29 (2006) 1316–1329

computer communications

www.elsevier.com/locate/comcom

Dynamic bandwidth allocation for QoS routing on TDMA-based mobile ad hoc networks

Kuei-Ping Shih *, Chih-Yung Chang, Yen-Da Chen, Tsung-Han Chuang

Department of Computer Science and Information Engineering, Tamkang University, Tarnshui 251, Taipei, Taiwan, ROC

Abstract

The paper proposes several dynamic bandwidth allocation strategies for QoS routing on TDMA-based mobile ad hoc networks. Comprehensively, these strategies are called a distributed slots reservation protocol (DSRP). In DSRP, QoS routing only depends on one-hop neighboring information of each mobile host (MH). In addition, slot inhibited policies (SIPs) and slot decision policies (SDPs) are proposed to determine which slots are valid to use and which slots in the valid slots can be used actually, respectively. In SDPs, three heuristic policies, 3BDP, LCFP, and MRFP, are proposed to increase the success rate of a QoS route and alleviate the slot shortage problems. Moreover, a slot adjustment protocol (SAP) is proposed for a conflicting MH to coordinate the slot usage of its neighbors during the route reservation phase in order to accommodate more routes in the network. The slot adjustment algorithm (SAA) invoked in SAP is a branch-and-bound algorithm, which is an optimum algorithm in terms of the number of slots to be adjusted, on the premise that not to break down any existing route. QoS route maintenance and improvement are also provided. By the simulation results, the proposed protocol cannot only increase the success rate in search of a route with bandwidth requirement guaranteed but also raise the throughput and efficiency of the network.

Keywords: Mobile ad hoc networks (MANETs); Routing protocol; Quality of service (QoS); Time division multiple access (TDMA); Wireless networks

1. Introduction

A mobile ad hoc network (MANET) is an infrastructure-less network consisting of numbers of mobile hosts (MHs) communicating with one another via relaying messages among MHs through multihop wireless links. The major challenges of a MANET are its fast changing of topology and the lack of global information. An MH has only its neighboring information due to its transmission capability, such as transmission range, battery power, and so on. These result in the difficulty of developing an efficient routing protocol for MANETs. A routing protocol for MANETs should be capable of not only finding a route for the communicating MHs but also tolerating highly dynamic changing of topology. A large number of researches on routing for MANETs could be found in the literature such as AODV, DSR, TORA, and so on Refs. [1–7]. Most of them try to find a shortest route between the source and the destination.

Recently, the need to support real-time applications, such as audio or video transmissions, is getting more and more. QoS-support routing protocol is thus, becoming important as well. In addition to finding a route from a source to a destination, a QoS routing for MANETs should also guarantee the end-to-end QoS requirement of the route, such as bandwidth needs, delay constraints, and so on. Recently, QoS routing protocols on MANETs have been studied extensively Refs. [8–23]. However, in Refs. [8–14], QoS routing protocols are designed for layer 3 (network layer). The QoS requirements in these protocols are assumed to be known in advance. Nevertheless, the QoS requirement much depends on the real situation from lower layers. Therefore, it would be more practical if a cross-layer protocol can be devised for QoS routing on MANETs. Thus, some researches take both the MAC and network layers into consideration to design QoS routing protocols for MANETs Refs. [15–23].

QoS routing protocols on CDMA over TDMA channel model are proposed in Refs. [15–20]. The multi-path schemes, Refs. [15–17], are adopted to discover QoS routes in a bandwidth-limited network. A ticket-based algorithm is proposed in Ref. [15] to search for a satisfactory multi-path route. The global link-state information from source to destination is needed in Ref. [16] to search a QoS multi-path route. In Ref. [17], a spiral-multi-path QoS routing protocol is

---

* Corresponding author. Tel.: +886 2 2621 5656x2748; fax: +886 2 2620 9749.

E-mail addresses: kpshih@mail.tku.edu.tw (K.-P. Shih), cychang@mail.tku.edu.tw (C.-Y. Chang).

doi:10.1016/j.comcom.2005.10.009

proposed not only to search QoS routes but also to enhance route-robustness and route-stability. Note that if the network bandwidth is sufficient, the protocols, Refs. [15–17], will just search a single route to satisfy the QoS bandwidth requirements. In Refs. [18–20], the protocols construct a single QoS routing path depending on one-hop neighbors information. The hidden terminal problem is covered by assigning different codes for the concurrent transmissions within two-hop distance. In Ref. [20], the QoS routing protocol adopts the algorithm in Ref. [18] to resolve slot-scheduling problem. The protocol in Ref. [20] takes not only the bandwidth but also battery life into consideration to search a QoS route.

The protocols, Refs. [15–20], are designed for CDMA over TDMA channel model. The cost of the code assignment should be counted into the protocol as well. However, the code assignment is not discussed in Refs. [15–20]. Due to the high cost of using CDMA model, some researches investigate to find the QoS route only on the TDMA channel model Refs. [21,22].

A TDMA-based bandwidth reservation protocol for QoS routing in MANETs is proposed in Ref. [21]. This protocol can solve the hidden-terminal and exposed-terminal problems by taking two-hop neighbors' slots usages into consideration. Each MH maintains the two-hop neighbors' slots usages to select the appropriate slots for each link of the route. If there are many slots to select for a link, random policy is adopted. However, it has a high possibility that the protocol cannot find any route satisfying the QoS requirements, even there exists a QoS route. On the other hand, the QoS routing protocol proposed in Ref. [22] is also based on the TDMA channel model. The protocol maintains one-hop neighboring information for the selection of slots in order to find a QoS route. A route maintenance mechanism for reconstructing a route when the route is broken is described as well. Different from the above channel models, Ref. [23] considers the QoS routing in a contention-based MANETs.

In this paper, a distributed slots reservation protocol (DSRP) for QoS routing on TDMA-based mobile ad hoc networks is proposed. The QoS requirement considered in the paper is the bandwidth requirement, in terms of data slots. The hidden-terminal and exposed-terminal problems are taken into consideration. Slots reuse is the main concept of the protocol. The main differences of the DSRP from the Ref. [22] are slots reuse and slots adjustment mechanism. The slots with least conflict to other MHs or have been used by other MHs are used with high priority if they are valid to use. Thus, the successful rate to find a QoS route can be increased correspondingly. A slot adjustment mechanism is to adjust the slots usages of an MH to tolerate more routes passed by when the slots in an MH have been reserved or used by another route. Doing so can increase the successful rate of discovering a QoS route. Route maintenance is to maintain the connectivity of the QoS route and improve the efficiency of the route. Extensive experiments are performed to verify the superiority of the proposed protocol. The proposed protocol does outperform than the existing methods in call success rate and average delay time.

In general, a QoS service means a constant traffic flow lasts for a period of time and no acknowledgement is required. Thus, the paper assumes that the QoS requirement does not change during route discovery and transmission. On the other hand, since the traffic of a QoS service is constant, the slots reserved in a time frame will be automatically reserved for the following time frames until the end of the service. In addition, no route from the destination to the source for ACK transmission is considered as well.

The rest of this paper is organized as follows. Section 2 presents the system model and describes the challenges in designing the QoS routing protocol on TDMA-based mobile ad hoc networks. Some heuristic policies used for slots selection are proposed in Section 3. Section 4 describes the distributed slots reservation protocol (DSRP), including QoS route discovery and reservation, and QoS route maintenance and improvement. Simulation results are presented and analyzed in Section 5. Section 6 concludes the paper.

2. Preliminaries

2.1. The system model and terminology

The paper proposes a distributed slots reservation protocol (DSRP) for QoS routing on TDMA-based mobile ad hoc networks. TDMA-based channel model is shown in Fig. 1, where time is divided into TDMA frames. A TDMA frame is composed of a *control* subframe and a *data* subframe. Each subframe contains a number of slots. Each slot can be used for one packet transmission. It is worth mentioning that control packets are scheduled to be transmitted in control subframes and data packets are in data subframes.

Some technical terms used throughout the paper are defined as follows. Let $\mathcal{N}$ be the set of MHs, NB_A denotes the neighbors of A, $A \in N$. That is $\mathrm{NB}_A = \{B \in \mathcal{N} | B$ is within the transmission range of $A\}$. If $B \in \mathrm{NB}_A$, there is a link between A and B, and vice versa. The link is denoted $\overline{AB}$. Here, the link refers to the bidirectional link. Unidirectional link is not considered in the paper. Let $\mathcal{L}$ be the set of links. $\mathcal{L} = \cup_{A \in \mathcal{N}} \{\overline{AB} | B \in \mathrm{NB}_A\}$. Therefore, a mobile ad hoc network can be viewed as a graph $\mathcal{G} = (\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ and $\mathcal{L}$ are defined as above.

Let n be the number of slots in a data subframe and $\mathcal{T}$ the set of slots in a data subframe. TS_A is defined as a set of data slots which A is using to send currently, RS_A a set of data slots which

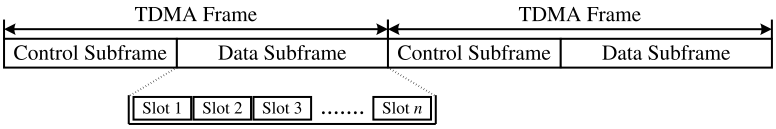

Fig. 1. TDMA frame structure.

A is using to receive, VS_A a set of valid slots for A that A can schedule to use to send, $\mathrm{VS}_{\overline{AB}}$ a set of valid slots for the link $\overline{AB}$ that A can schedule to send to B and B can schedule to receive from A simultaneously, and $\mathrm{SS}_{\overline{AB}}$ a set of data slots that are selected for the link $\overline{AB}$. The relationship among VS_A, $\mathrm{VS}_{\overline{AB}}$, and $\mathrm{SS}_{\overline{AB}}$ is $\mathrm{SS}_{\overline{AB}} \subseteq \mathrm{VS}_{\overline{AB}} \subseteq \mathrm{VS}_A$. Note that the time slots discussed in the paper all mean the data slots, if no otherwise are notified.

2.2. Challenges of QoS routing on TDMA-based MANETs

It is difficult to design a QoS routing on TDMA-based mobile ad hoc networks. There are many challenges to be conquered, such as the hidden terminal problem, exposed terminal problem, and the slots shortage problems. The challenges of QoS routing on TDMA-Based MANETs are briefly described as follows.

2.2.1. Hidden and exposed terminal problems

Suppose A, B, and $C \in \mathcal{N}$. If $B \in \mathrm{NB}_A$ and $B \in \mathrm{NB}_C$, but $A \notin \mathrm{NB}_C$ and $C \notin \mathrm{NB}_A$, A and C are hidden terminals to each other. Under the circumstance, it will cause collisions if A and C send to B simultaneously due to the invisibility of hidden terminals. The hidden terminal problem is known to be a serious problem in multihop mobile ad hoc networks since, it will cause collisions and further degrade the system throughput.

On the other hand, the exposed terminal problem is explained as follows. Suppose A, B, C, and $D \in \mathcal{N}$. Assume that B wants to send to A ($A \in \mathrm{NB}_B$) and C wants to send to D ($D \in \mathrm{NB}_C$). If $C \in \mathrm{NB}_B$, $A \notin \mathrm{NB}_C$, and $D \notin \mathrm{NB}_B$, the two communications (B sends to A and C sends to D) do not interfere with each other and can occur simultaneously. However, according to carrier sense multiple access (CSMA), one, either B or C, is inhibited from transmitting to its destination due to its exposure to another. The exposed terminal problem is viewed as a source of an inefficiency of the network.

Take Fig. 2 as an example. Suppose S schedules to send to B in slots $\{1, 2\}$. B receives from S in slots $\{1, 2\}$ and forwards to E in slots $\{3, 4\}$.

If E forwards to G in slots $\{1, 2\}$, the collision will occur at B. This is the hidden terminal problem. Therefore, when an MH wants to transmit, it should take the hidden terminal problem into consideration in order not to collide with others. On the other hand, follow the above example that S sends to B in slots $\{1, 2\}$ and B forwards to E in slots $\{3, 4\}$. If C wants to send to A, C cannot reserve slots $\{3, 4\}$ to send, since B has used slots $\{3, 4\}$ to send. This is the exposed terminal problem. When an MH wants to transmit, if it can consider the exposed terminal problem and use the slots that its neighbors have used, the others MHs can have more free slots to use. Thus, solving the exposed terminal problem cannot only increase the network efficiency but also increase the successful rate of QoS routes findings.

2.2.2. Slot shortage problems

In addition to the hidden and exposed terminal problems, slot shortage problems are also great challenges in designing a QoS routing on TDMA-based MANETs. Slot shortage problems mean the problems that there should exist at least one route from the source to the destination, which satisfies the QoS requirement. However, for the sake of inappropriate slots selection, the route that originally exists cannot be discovered now. The slot shortage problem caused by inappropriate slots selection will affect on not only the discovery of self-route but also the discoveries of the neighboring routes, which are termed the SSSR and SSNR problems, respectively, and are explained as follows.

2.2.3. Slot shortage for self-route (SSSR)

As shown in Fig. 2, suppose S wants to construct a QoS route to D and the QoS requirement is two slots. The slots $\{6, 7, 8\}$ in G have been used. Take the route P: $S \to B \to E \to G \to D$ as an example. Assume S uses slots $\{1, 2\}$ to transmit to B and B uses slots $\{3, 4\}$ to forward to E. Slots $\{3, 4\}$ in E should be used to receive from B. Furthermore, E cannot use slots $\{1, 2\}$ to forward to G as well due to the hidden terminal problem. As a result, E has no sufficient slots to forward to G. Under such a circumstance, the route P cannot satisfy the QoS requirement. On the contrary, for the same route P, if S uses slots $\{5, 6\}$ to transmit to B and B uses slots $\{7, 8\}$ to forward to E, E can use slots $\{1, 2\}$ to forward to G. Thus, G can use slots $\{3, 4\}$ to forward to D. Therefore, the route P can satisfy the QoS requirement.

2.2.4. Slot shortage for neighboring routes (SSNR)

Slot shortage problem is also possible to affect the neighboring routes. Consider the same example. Suppose the slots in black color are used. For the same route P, according to

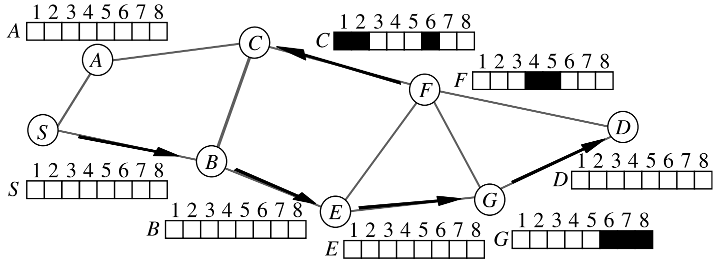

Fig. 2. An illustrative example for the hidden terminal, exposed terminal, SSSR, and SSNR problems.

the previous result, S, B, E, and G use slots {5, 6}, {7, 8}, {1, 2}, and {3, 4} to transmit to B, E, G, and D, respectively. If there is a neighboring route P' from F to C, the QoS requirement is also two slots. One can easily observe that there are no sufficient slots for F to transmit to C. However, if S, B, E, and G, respectively, use slots {7, 8}, {5, 6}, {3, 4}, and {1, 2} to transmit to B, E, G, and D instead, F can have enough slots ({7, 8}) to transmit to C. Thus, not only P but also P' can satisfy their QoS requirements.

Taking slot shortage problems into consideration can increase not only the successful rate of route discovery but also the network throughput. So slot shortage problems are important issues in designing a QoS routing protocol on TDMA-based MANETs.

The above problems are often encountered while designing a QoS routing protocol for TDMA-based MANETs. Thus, the paper takes these problems into consideration and proposes some heuristic slot selection policies to resolve or alleviate these problems. The slot selection policies are described in the following section.

3. Slot selection policies

3.1. Slot inhibited policies (SIPs)

For a link $\overline{AB}$, slot inhibited policies (SIPs) are the polices used for A and B to decide the free slots that are valid to use. That is $\mathrm{VS}_{\overline{AB}}$. The similar ideas can be found in Refs. [21,22]. For completeness, the ideas of SIPs are explained as follows. SIPs take the hidden terminal and exposed terminal problems into consideration. Let $A, B \in \mathcal{N}$. If a slot $t, t \in \mathcal{T}$, is valid to use for A to send to B, t should satisfy the following three policies. The first policy, which is termed sip1, requires that slot t should be free to use for the two MHs. Namely, both A and B have not used slot t neither to send nor to receive. The second policy, which is termed sip2, requires that no neighbor of A uses slot t to receive. The third policy, which is termed sip3, requires that no neighbor of B use slot t to send. On the contrary, if slot t can satisfy the above three policies, slot t is a valid slot for A to use to send to B. Lemma 1 concludes the results.

Lemma 1. Let $A, B \in \mathcal{N}$, and $\mathrm{B} \in \mathrm{NB_A}$. Slot $t \in \mathcal{T}$ is valid for A to send to B if t satisfies the following three policies, and vice versa.

sip1: $t \notin \mathrm{TS}_A$, $t \notin \mathrm{RS}_A$, $t \notin \mathrm{TS}_B$, *and* $t \notin \mathrm{RS}_B$.
sip2: $t \notin \mathrm{RS}_C$, $\forall C \in \mathrm{NB}_A$.
sip3: $t \notin \mathrm{TS}_D$, $\forall D \in \mathrm{NB}_B$.

Lemma 1 has taken the hidden terminal and exposed terminal problems into consideration. Thus, whether a slot is valid to use or not can be determined by Lemma 1. As a result, the set of valid slots for a link, say $\overline{AB}$, $\mathrm{VS}_{\overline{AB}}$ can be determined by the following theorem.

Theorem 1. Let $A, B \in \mathcal{N}$, and $\mathrm{B} \in \mathrm{NB_A}$. Suppose A wants to send data to B. SIP1, SIP2, and SIP3 are the sets of slots that satisfy the policies of sip1, sip2, and sip3, respectively.

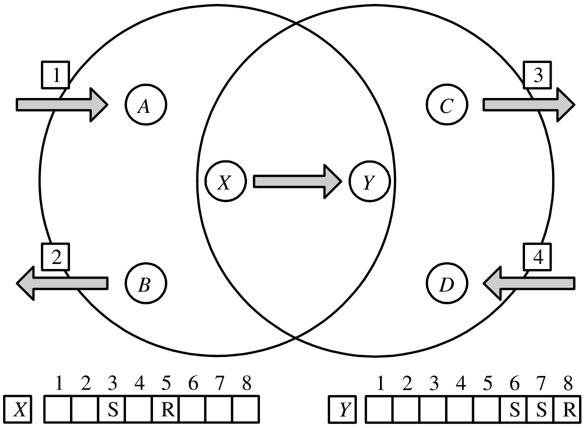

Fig. 3. Demonstration of slot inhibited policies (SIPs).

$\mathrm{SIP1} = \{t \in \mathcal{T} \mid t \notin \mathrm{TS}_A, t \notin \mathrm{RS}_A, t \notin \mathrm{TS}_B, \text{and } t \notin \mathrm{RS}_B\}$.
$\mathrm{SIP2} = \{t \in \mathcal{T} \mid t \notin \mathrm{RS}_C, \forall C \in \mathrm{NB}_A\}$.
$\mathrm{SIP3} = \{t \in \mathcal{T} \mid t \notin \mathrm{TS}_C, \forall C \in \mathrm{NB}_B\}$.

The set of valid slots for A to use to send to B, $\mathrm{VS}_{\overline{AB}}$, can be obtained as follows.

$$\mathrm{VS}_{\overline{AB}} = \{t \in \mathcal{T} \mid t \in \mathrm{SIP1} \cap \mathrm{SIP2} \cap \mathrm{SIP3}\}$$

Take Fig. 3 as an example A, B, C, D, X, and $Y \in \mathcal{N}$. $A, B \in \mathrm{NB}_X$ and C, $D \in \mathrm{NB}_Y$. Suppose there are eight slots in a data subframe, and X intends to send data to Y. In DSRP, only one-hop neighbors information is required. Therefore, by Theorem 1, X and Y have to cooperate to obtain VS_X. To begin with, X collects its one-hop neighboring slot information, $3 \in \mathrm{TS}_X$, $5 \in \mathrm{RS}_X$, $1 \in \mathrm{RS}_A$, $2 \in \mathrm{TS}_B$, $8 \in \mathrm{RS}_Y$, and $\{6, 7\} \in \mathrm{TS}_Y$, and initiates SIP1 and SIP2. The set of valid slots in X, VS_X, which has been defined in Section 2.1, is the intersection of SIP1 and SIP2. Consequently, $\mathrm{VS}_X = \{1, 2, 4\} \cap \{2, 3, 4, 5, 6, 7, 8\} = \{2, 4\}$. This information will be enclosed in the RREQ packet and be sent to Y. As a result, Y can obtain $\mathrm{VS}_{\overline{XY}}$ by intersecting VS_X with SIP3. Thus, $\mathrm{VS}_{\overline{XY}} = \mathrm{VS}_X \cap \{1, 2, 4, 5, 6, 7, 8\} = \{2, 4\}$.

3.2. Slot decision policies (SDPs)

In the previous subsection, Theorem 1 is used to determine which slots are valid to be used for a link. In the section, Slot Decision Policies (SDPs) are proposed to decide which slots are to be reserved for a link. SDPs include three heuristic policies: three-hop backward decision policy (3BDP), least conflict first policy (LCFP), and most reuse first policy (MRFP) to alleviate the slots shortage problems. They are described as follows.

3.2.1. Three-Hop backward decision policy (3BDP)

Three-Hop backward decision policy (3BDP) is a heuristic policy for an MH backward to determine the slots for the previous third link in order not only to satisfy the QoS requirement but also to reduce the slot shortage problems. It is obvious that if the slot selection is only consider its one-hop neighbors slots information, it is quite possible to have no

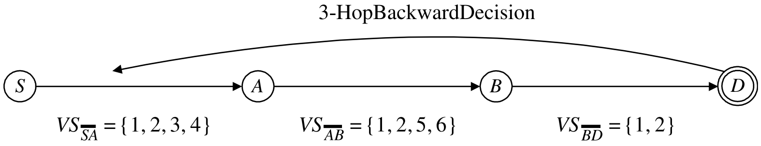

Fig. 4. Demonstration of 3-Hop backward decision policy (3BDP).

sufficient slots for the remaining links. The slot shortage problems will occur frequently.

Take Fig. 4 as an example. Suppose S wants to send to D. The QoS requirement is two slots. The sets of valid slots for the links $\overline{SA}, \overline{AB}$ and $\overline{BD}$ are {1, 2, 3, 4}, {1, 2, 5, 6}, and {1, 2}, respectively. In this example, neither slot 1 nor slot 2 can be selected for the link $\overline{SA}$. Otherwise, the following two links will have no sufficient slots to reserve in order to satisfy the QoS requirement. However, both S and A are not aware of this situation since only one-hop neighbor information is obtained. Therefore, it is very possible to select inappropriate slots such that there is no route from S to D, which can satisfy the QoS requirement. On the other hand, if a slot is selected for a link, the slot cannot be selected again for the following two links due to the hidden terminal problem. That is, a slot can be used just once for three continuous links. Thus, if taking three hops' valid slots into consideration, the possibility of SSSR problem can be greatly reduced. That is the reason why 3BDP is designed.

The effect of the 3BDP is quite obvious. However, it is still possible to have SSSR problem even if 3BDP is adopted, but the possibility is quite low. Actually, the best way to find an optimum slot selection for all MHs in a route to satisfy the QoS requirement is to collect all valid slots information of all MHs in the route to the destination and the destination decides the slots scheduling. However, it is time-consuming, which has been shown to be an NP-hard problem Ref. [22]. On the other hand, to do so will make the RREQ packets grow explosively and are prone to be collided. Therefore, there is a tradeoff between the control overhead and the precision of slots information. Thus, an experiment is made to investigate the relationship between the call success rate and the number of hops of backward decision. Fig. 5 shows the simulation results, where DH is the number of backward decision hops. Obviously, when DH = 1 or 2, the success rates are far lower than those for DH = 3 or higher DH values. However, the success rates for DH = 4 or 5 are very close to that for DH = 3. As a result, collecting three hops information to decide which slots to be used is much cost-efficient.

3.2.2. Least conflict first policy (LCFP)

The principle of least conflict first policy (LCFP) is that the slots with the least conflict with the next two links are preferable to be selected first. It is because a slot can be used just once for three continuous links. Thus, if the slots, which are also available in the next two continuous links, are selected, the available slots in the next two links may be insufficient to satisfy the QoS requirement. Thus, slots shortage problem may happen in the next two links. Therefore, the slots with the least conflict with the next two links are preferable to be selected first. Consequently, the probability of the SSSR problem will be reduced accordingly.

Recall the example shown in Fig. 4 to show the LCFP. Assume that, by 3BDP, D is going to determine which slots to be selected for the link $\overline{SA}$. In this example, $VS_{\overline{SA}} = \{1,2,3,4\}$. The slots {1, 2} are also appeared in $VS_{\overline{AB}}$ and $VS_{\overline{BD}}$. On the contrary, the slots {3, 4} are only appeared in. Thus, by LCFP, D will select {3, 4} in advance, instead of the slots {1, 2}. That is, $SS_{\overline{SA}} = \{3,4\}$. Obviously, the slots {1, 2} should not be selected for the link $\overline{SA}$ in order to avoid the SSSR problem.

3.2.3. Most reuse first policy (MRFP)

The most reuse first policy (MRFP) is used when the valid slots after filtered by the LCFP are still more than the QoS requirement. The concept of MRFP comes from slot reuse. MRFP takes the neighboring slots utilization rate into consideration. Since each MH has one-hop neighboring slot information, it is not difficult to obtain the slot utilization rate. The slots with higher utilization rate have higher priority to be selected. By this way, the slot utilization rate can be increased. Thus, it is possible that more routes can exist at the same time in the network. Therefore, the SSSR and SSNR problems can be mitigated as well. On the other hand, MRFP can also be viewed as a solution to alleviate the exposed terminal problem.

Fig. 6 is an example to show how the MRFP can ease the SSSR problem. Suppose S intends to send data to D and the QoS requirement is two slots. The valid slots information for each links are $VS_{\overline{SA}} = \{1,2,3,4,5\}, VS_{\overline{AB}} = \{1,2,3,4\}$ and $VS_{\overline{BD}} = \{5,6,7,8\}$. The slot utilization rates, denoted as UR, corresponding to $VS_{\overline{SA}}$, $VS_{\overline{AB}}$, and $VS_{\overline{BD}}$ are assumed to be {1, 1, 0, 0, 0}, {0, 0, 1, 1}, and {0, 0, 1, 1}, respectively. Assume that D is going to select slots for the link $\overline{AB}$ The slots in $VS_{\overline{SA}}$ are appeared once either in $VS_{\overline{AB}}$ or in $VS_{\overline{BD}}$. Therefore, according to LCFP, the least conflict slots are also {1, 2, 3, 4, 5}. However, the QoS requirement is two slots. Instead of randomly selecting two slots, MRFP will select slots

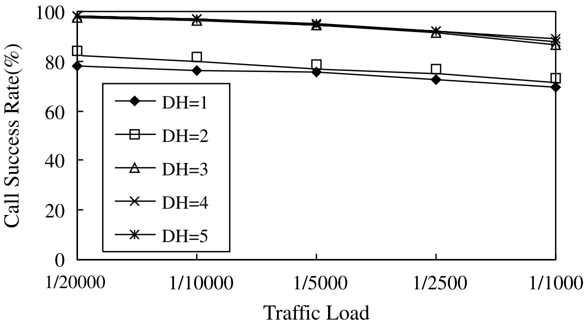

Fig. 5. The number of hops of backward decision vs. the call success rate.

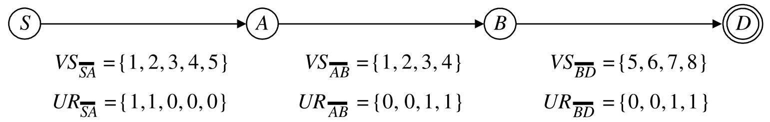

Fig. 6. Illustration of the MRFP to alleviate the SSSR problem, where *UR* denotes the corresponding slot utilization rate.

{1, 2} since the utilization rates of these two slots are the highest. Thus, $\mathrm{SS}_{\overline{SA}} = \{1,2\}$.

After $\mathrm{SS}_{\overline{SA}}$ is determined, the remaining valid slots in $\mathrm{VS}_{\overline{AB}}$ and $\mathrm{VS}_{\overline{BD}}$ are slots $\{3,4\}$ and slots $\{5,6,7,8\}$. Applying LCFP and MRFP repeatedly, the slots selected for the links $\overline{AB}$ and $\overline{BD}$ are slots {3, 4} and slots {7, 8}, respectively. In short, $\mathrm{SS}_{\overline{AB}} = \{3,4\}$ and $\mathrm{SS}_{\overline{BD}} = \{7,8\}$. Therefore, the QoS route from *S* through *A* and *B* to *D* can be successfully constructed and the route can satisfy the QoS requirement.

MRFP can assist not only in alleviating the SSSR problem but also in mitigating the SSNR problem. Fig. 7 is an example to show how the MRFP can mitigate the SSNR problem. Suppose the QoS requirement is one slot. *A*, *C* and *X* are neighbors to each other. *A* is using slot {1} to transmit to *B*. *X* and *D* want to send data to *Y* and *C*, respectively. $\mathrm{VS}_{\overline{XY}} = \{1,2\}$ and $\mathrm{VS}_{\overline{DC}} = \{2\}$. If *X* randomly selects slot {2} to send, there is no valid slot for $\overline{DC}$ Therefore, there is no QoS route for $\overline{DC}$. On the contrary, if taking the slot utilization rate into consideration, slot {1} will be selected for $\overline{XY}$ since *A* is using slot {1} to send to *B*. Thus, *D* can use slot {2} to send to *C* as well. All the three QoS routes can be constructed successfully. As a result, the SSNR problem can be mitigated under this situation. Note that, to some extent, MRFP preferring *X* using slot {1} to send to *Y* because this slot is also used by *A* is a kind of solution to the exposed terminal problem.

The slot inhibited policies (SIPs) and the slot decision polices (SDPs), including three-hop backward decision policy (3BDP), least conflict first policy (LCFP), and most reuse first policy (MRFP), are all included in the distributed slot reservation protocol (DSRP). The following section describes the DSRP in details.

4. The distributed slots reservation protocol (DSRP)

The distributed slots reservation protocol (DSRP) is an on-demand slots reservation protocol for QoS routing on TDMA-based mobile ad hoc networks. When a source *S* wants to communicate with a destination *D*, the *QoS* route discovery phase will be initiated by *S* for broadcasting a route request (RREQ) packet to search a QoS route to *D*. During the *QoS* route discovery phase, slots inhibited policies (SIPs) decide which slots are valid to use for a link, and slot decision policy (SDP) decides which slots to be used for a link. *QoS* route reply and reservation phase will be initiated to reserve slots for the route by replying a route reply (RREP) packet destined to *S* while *D* receives the RREQ packet. However, the slots that have been decided to reserve in *QoS* route discovery phase may be reserved by other requests during the reply and reservation phase. Hence, instead of rediscovering a new route, a slot adjustment protocol (SAP) will be initiated to coordinate the slots scheduling of the conflicting MH with its one-hop neighbors' scheduling. On the other hand, if the QoS route is broken or changed while it is in use, the *QoS* route maintenance and improvement protocol will be triggered to fill up the breakage or improve the change of the route. This is the main concept of the DSRP. The details of the DSRP will be described as follows.

4.1. Qos route discovery phase

The QoS route discovery phase is initiated whenever an MH wants to communicate with another MH. Let *S* be the source and *D* the destination. At the beginning, *S* broadcasts a route request (RREQ) packet to search a QoS route to *D*. The routing information carried in the RREQ packet is described as follows.

⟨S_id, D_id, T_id, RID, Path, BR, VS, UR⟩

- S_id: source MH ID
- D_id: destination MH ID
- T_id: transmitter MH ID
- RID: route ID
- BR: QoS bandwidth requirement
- Path: the partial path that has been discovered so far
- VS: valid slots information
- UR: valid slots utilization rates

Note that the RID is unique for each route to avoid endless looping. VS records the valid slots information, including the valid slots sets for the previous two continuous links and the $\mathrm{VS}_{\mathrm{T\_id}}$ for the receiver to calculate the valid set of the current link. UR is the slots utilization rates corresponding to the valid slots sets recorded in VS. *Path* will record the visited MHs so far and the corresponding selected slots (SS sets) for the previous links.

Initially, *S* collects and calculates the valid slots of VS_S by sip1 and sip2, and accounts the utilization rates of the

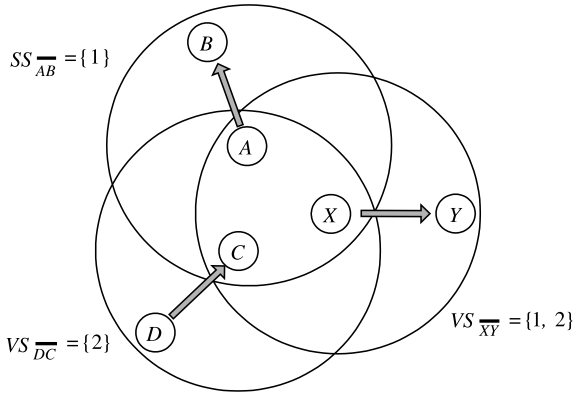

Fig. 7. Illustration of the MRFP to alleviate the SSNR problem.

corresponding slots, UR_S. S records the information in the corresponding fields in the RREQ packet and broadcasts it to its neighbors.

Without loss of generality, assume that Y broadcasts an RREQ packet to its neighbors and Z receives the RREQ packet, $Z \in \mathrm{NB}_Y$. The previous three MHs recorded in the *Path* are W, X, and Y, in sequence. That is, $P{:}\cdots \rightarrow W \rightarrow X \rightarrow Y$. When receiving an RREQ packet from Y, Z will check whether this RREQ packet has been received before or not. If this RREQ packet has been received before, this RREQ packet will be discarded.

Otherwise, Z will calculate $\mathrm{VS}_{\overline{YZ}}$ by intersecting VS_Y (recorded in the RREQ packet) and itself SIP3. If $\mathrm{VS}_{\overline{YZ}} \geq \mathrm{BR}$, according to 3BDP, Z will determine which slots are to be used for the link $\overline{WX}$ by using LCFP and MRFP. Therefore, $\mathrm{SS}_{\overline{WX}}$ is determined. If either $\mathrm{VS}_{\overline{YZ}}$ or $\mathrm{SS}_{\overline{WX}}$ is smaller than BR, Z will stop forwarding the RREQ packet. Otherwise, Z will calculate the $\mathrm{SIP1} \cap \mathrm{SIP2}$, pack all required information to a new RREQ packet, and forward the RREQ packet to its neighbors.

If Z is the destination, Z should respond for the determinations of the slots selections for the previous three links. Then *QoS* route reply and reservation phase will be initiated by D to send RREP to S and reserve the slots along the path. The detailed procedure is shown in Algorithm 1.

```
Algorithm 1 Partial route discovery algorithm
/* Suppose Z receives a broadcast packet RREQ from Y, and there is a partial path $P = \{W \rightarrow X \rightarrow Y$ recorded in RREQ packet*/
if this RREQ has been received before then
    Discard the RREQ packet
    Exit this procedure
else
    $\mathrm{VS}_{\overline{YZ}} = \mathrm{VS}_Y \cap \mathrm{SIP3}$
    if $\mathrm{VS}_{YZ} \geq \mathrm{BR}$ then
        Append Z to Path.
        if more than three MHs recorded in Path then
            $\mathrm{SS}_{\overline{WX}} = \mathrm{SDP}(\overline{WX})$
        end if
        $VS_Z = \mathrm{SIP1} \cap \mathrm{SIP2}$
        if $Z = D$ then
            $\mathrm{SS}_{\overline{XY}} = \mathrm{SDP}(\overline{XY})$
            $\mathrm{SS}_{\overline{YZ}} = \mathrm{SDP}(\overline{YZ})$
            Route_Reservation()
            Exit this procedure
        end if
        Broadcast RREQ
    end if
end if
```

It is worth mentioning that, in route discovery phase, the slots selected to be used are not actually reserved. The reservations of these slots are left to the route reservation phase. It is because in route discovery phase, all the routes starting from the source are tentative. If these slots are reserved in route discovery phase, it is required to release these slots if the route is not selected by the destination. Moreover, if these slots are reserved in route discovery phase, it is very possible that there are no sufficient slots for the other routes. The slot shortage problems, both SSSR and SSNR, will occur much frequently.

4.2. Qos route reply and reservation phase

Let $M \in n$. When M receives the RREP, it checks that if it is listed in *Path* of RREQ or not. If M is in *Path*, it reserves the relative slots and sends the RREP packet to the source along the *Path*.

However, it is very possible that during the reply and reservation phase, the slots which an MH is going to reserve have already been reserved by other requests. To avoid rediscovering another route from the source to the destination, the paper investigates a novel protocol, named slot adjustment protocol (SAP), to coordinate the MH and its one-hop neighbors slots usages. Under the conditions that do not break the existing routes, SAP tries to accommodate more QoS routes in order to increase network throughput, reduce the control overhead, and diminish the route discovery latency. The details of the SAP are described as below.

4.2.1. Slot adjustment protocol (SAP)

Let M be a conflicting MH. M will collect slot information of its neighbors by broadcasting an SA_REQ packet. The neighbors with which M is communicating, either to be a transmitter or to be a receiver, need to reply their slot usages while receiving the SA_REQ packet. These neighbors are called *SA_members*. A branch-and-bound algorithm, named Slot Adjustment Algorithm (SAA), is invoked to coordinate the slot usages of M and its *SA_members*. If the slots need to be adjusted, M will send an SA_INFO packet to acquire the agreements of the *SA_members* the adjustment of slots. If the adjustment is valid, the *SA_members* will acknowledge M with an SA_ACK packet. If all *SA_members* agree with the adjustment, an SA_EXE packet will be sent by M to confirm the adjustment.

While an *SA_member*, say N, receives the SA_REQ packet, according to its current slot usage status, N will calculate the current used slots which are in use to communicate with M and the valid slots that can be selected for their communication. In other words, if N is a receiver, N will calculate the $\mathrm{SS}_{\overline{MN}}$ and the $\mathrm{VS}_{\overline{MN}}$. Otherwise, if N is a transmitter, N will figure out the $\mathrm{SS}_{\overline{NM}}$ and the $\mathrm{VS}_{\overline{NM}}$. It is worth mentioning that the calculations of the SS and VS sets should follow the SIPs policies and take its neighbors information into consideration.

The operations of the slot adjustment protocol are described in Algorithm 2. The details of the slot adjustment algorithm (SAA) are explained as follows.

```
Algorithm 2 Slot adjustment protocol (SAP)
/* my_id = N */
if Slot(s) to be reserved is(are) in use then
    Broadcasts SA_REQ to SA_members
```

```
end if
if Receive a packet from $M$ then
  switch Packet Type do
    case {Type==SA_REQ}
      if I am an SA_member then
        if I am a transmitter then
          $VS_N = SIP1 \cap SIP2$
        else
          $VS_N = SIP1 \cap SIP3$
        end if
        Add $VS_{\overline{N}}$ in SA_REP and Reply SA_REP
      end if
      break
    case {Type==SA_REP}
      if I am the receiver of the SA_REP then
        if I am a receiver then
          $SS_{\overline{MN}} = VS_M \cap SIP3$
        else
          $SS_{\overline{NM}} = VS_M \cap SIP2$
        end if
        repeat
          Wait for SA_REP
        until (Timeout) or (All SA_REPs are received from all SA_members)
        if I have received SA_REPs from all SA_members then
          Call Slot Adjustment Algorithm
          Broadcast SA_INFO to SA_members with the slot adjust info.
        end if
      end if
      break
    case {Type==SA_INFO}
      if I agree with the slot adjustment denoted in the SA_INFO then
        Reply SA_ACK
      end if
      break
    case {Type==SA_ACK}
      if I have received all SA_ACKs from all SA_members then
        Broadcast SA_EXE
      end if
      break
    case {Type==SA_EXE}
      Adjust slots according to the slot adjustment recorded in SA_EXE
      break
  end switch
end if
```

4.2.2. Slot adjustment algorithm (SAA)

SAA is a branch-and-bound technique. SAA has the following features. First, SAA will not break down the existing routes. Second, the number of MHs to be adjusted is the least. Third, the number of slots to be adjusted is also the least. Actually, SAA is an optimum algorithm in terms of the numbers of slots and MHs to be adjusted.

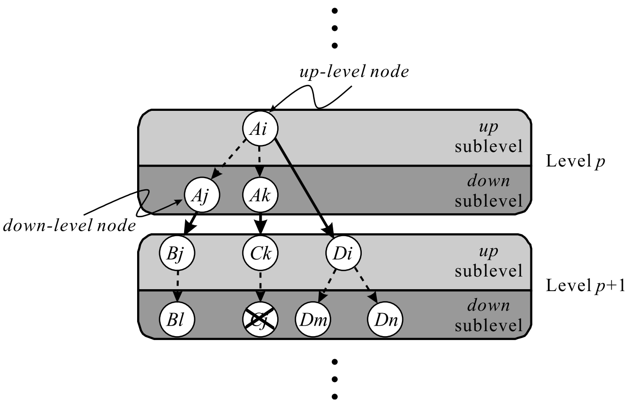

Fig. 8. The architecture of a slot adjustment tree.

SAA will construct a slot adjustment tree for each conflicting slot according to the slot information indicated in the SA_REPs from all *SA_members*, where a conflicting slot means the slot which is to be reserved but has been used. The architecture of a slot adjustment tree is illustrated in Fig. 8. A slot adjustment tree is layered and each layer is numbered by a level number. The root is at level 0. A level can be further divided into two sublevels: *up* and *down* sublevels, as shown in Fig. 8. In a slot adjustment tree, a tree node is represented by an *SA_member* in conjunction with a slot. A tree node in the up sublevel is called an *up-level node* while in the down sublevel a *down-level node*. Basically, there are two kinds of links, *intra-level links* and *inter-level links*, which are shown in Fig. 8 as the dotted lines and the solid lines, respectively. An intra-level link connects an up-level node with a down-level node, both within the same level, while an inter-level link connects two nodes within two neighboring levels. It is worth mentioning that the two end-nodes of an intra-level link must have the same *SA_member* but associate with different slot numbers. On the contrary, the two end-nodes of an inter-level link must have the same slot number but with different *SA_members*. The reasons of the phenomena come from the specific meanings of the types of the links and are explained as follows.

For some intra-level link, assume that Ai and Aj are the two end-nodes of the intra-level link, where A is an *SA_member* and i, j are the associated slot numbers. Ai and Aj are within the same level. Without loss of generality, let Ai be an up-level node and Aj a down-level node. The intra-level link means that A has to adjust the slot i and A can find a valid slot j from its VS set. In other words, if the slot adjustment can be done by an *SA_member* and no need to adjust the slots in the other *SA_members*, an intra-level link is involved. That is why the two end-nodes of an intra-level link have the same *SA_member* but with different slot numbers.

On the other hand, for some inter-level link, assume that Ai and Di are the two end-nodes of the inter-level link, where A and D are two *SA_members* and i is the associated slot number. Without loss of generality, Ai is in level p, for some p and Di is in level $p+1$. The inter-level link means that A has to adjust the slot i but A intends to use slot i. Thus, if A uses slot i, it will conflict with D who is using slot i currently since $i \in \mathrm{SS}_D$. Therefore, D has to adjust slot i in order not to conflict with A. In other words, if the slot adjustment cannot be done by an *SA_member* and needs the adjustment in the other *SA_member*, an inter-level link between the two *SA_members* is involved. That is also why the two end-nodes of an inter-level link have the same slot number but with different *SA_members*.

A tree node can be pruned if there exists another tree node whose slot is the same as the tree node but its level is lower than that of the tree node. For the example shown in Fig. 8, the tree node *Cj* at level $p+1$ can be pruned since there is another tree node *Aj* at level p. The reason is that if *Cj* is not pruned, there must have an inter-level link connecting *Cj* to another *SA_member* such that *j* belongs to the SS set of the *SA_member*. The situation will be completely the same with that of *Aj*. As a result, there is no need to expand *Cj*. In the slot adjustment tree, each path from the root to an unpruned leaf is a valid adjustment. If there exist more than one valid adjustment, SAA will choose the path with the least number of levels. If the numbers of levels of the valid adjustments are the same, the least number of tree nodes will be selected. Otherwise, randomly selection is adopted.

For example, let the conflicting MH be *X*. The conflicting slot is occurred between the communication pair of *X* and *Y*. Currently, *A*, *B*, *C*, and *D* are communicating with *X*. Thus, the *SA_members* include *A*, *B*, *C*, *D*, and *Y*. The slot usage status of *SA_members* is shown in Table 1. In the example, the total number of data slots is 10. It is worth mentioning that the sets SS and VS shown in Table 1 are the slots that *SA_members* use to communicate with *X*. In other words, these sets stands for the slot usage for the link between *X* and the corresponding *SA_member*, not for either node. Moreover, these slots are obtained according to the SIPs policies and have taken the neighbors information into consideration.

Suppose the conflicting slots are the 1st and the 7th slots. The slot adjustment trees constructed by SAA corresponding to slot 1 and slot 7 are illustrated in Fig. 9. Due to the space limitation, the construction of the slot adjustment tree for slot 1 by SAA is explained here. The construction of the slot adjustment tree for slot 7 can be done likewise.

The explanation is depth-first. At the beginning, the root of the slot adjustment tree is *Y*1. Since VS_Y contains slots 3 and 5 (slot 7 cannot be counted since it is another conflicting slot), it means that slot 1 is possible to be adjusted to slot 3 or slot 5. Therefore, two tree nodes, *Y*3 and *Y*5, are generated at the down sublevel of the same level with that of *Y*1. Two intra-level links, *Y*1–*Y*3 and *Y*1–*Y*5, are generated as well. For node *Y*3, SAA will check that whether slot 3 is in the SS set of some other *SA_member*. If no, the node *Y*3 is a promising node and the path from *Y*1 to *Y*3 is a valid adjustment. However, in the example, slot 3 is in *B*'s SS set. Hence, an inter-level link from *Y*3 to *B*3 is constructed. Similarly, SAA will check whether there is any other slot in VS_B which can be used to replace slot 3. Obviously, slots 9 and 10 are valid to use. Consequently, two nodes, *B*9 and *B*10, are generated and the corresponding two intra-level links are constructed as well. Since both slot 9 and slot 10 are not appeared in the SS set of some other *SA_member*, *B*9 and *B*10 are all promising nodes. Therefore, the two pathes, $Y_1 \rightarrow Y_3 \rightarrow B_3 \rightarrow B_9$ and $Y_1 \rightarrow Y_3 \rightarrow B_3 \rightarrow B_{10}$, are all valid adjustments.

Table 1
An example to illustrate SAA

SA_member	SS	VS
A	1, 2	1, 2, 3, 5, 6
B	3, 4	3, 4, 9, 10
C	5, 6	3, 4, 5, 6
D	7, 8	5, 7, 8, 10
Y	1, 7	1, 3, 5, 7

The table is the slot usage status indicated in SA_REPs from all *SA_members*.

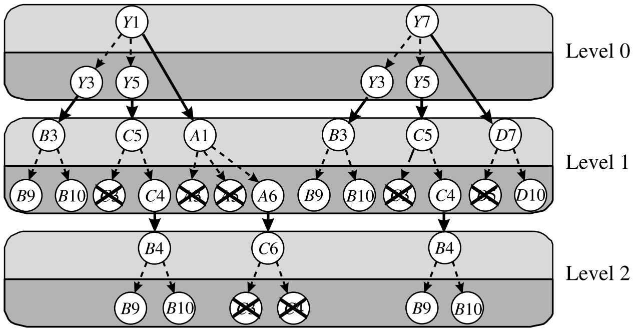

Fig. 9. The slot adjustment trees corresponding to slot 1 and slot 7, respectively.

Likewise, the subtree rooted at *Y*5 can be constructed as well. On the other hand, for the *Y*1 node, if *Y* intends to use slot 1, SAA will check whether slot 1 is in the SS set of some other *SA_member*. In the example, slot 1 is in SS_A. Therefore, a new node, *A*1, is constructed at the one more level of *Y*1 and a corresponding inter-level link is constructed as well. Similarly, slots 3, 5 and 6 are valid to replace slot 1, since slots 3, 5, and 6 are in VS_A. Therefore, nodes *A*3, *A*5, *A*6 are generated and three intra-level links are constructed. However, *A*3 and *A*5 are pruned since there are *Y*3 and *Y*5 at level 0, respectively. The generation of the subtree rooted at *A*6 can also be constructed as well. The total valid adjustments corresponding to slot 1 and slot 7 are listed in Table 2. The results of SAA are also pointed out.

Finally, the results of the slot adjustment algorithm are as follows. For the conflicting slot 1, *Y* will use slot 3, instead of slot 1, to communicate with *X* and *B* has to adjust slot 3 to slot 9

Table 2
The valid paths for slot 1 and slot 7

Slot	Valid path	# Levels	# Tree nodes	Result of SAA
1	*Y*1–*Y*3–*B*3–*B*9	2	4	✓
	*Y*1–*Y*3–*B*3–*B*10	2	4	
	*Y*1–*Y*5–*C*5–*C*4–*B*4–*B*9	3	6	
	*Y*1–*Y*5–*C*5–*C*4–*B*4–*B*10	3	6	
7	*Y*7–*Y*3–*B*3–*B*9	2	4	
	*Y*7–*Y*3–*B*3–*B*10	2	4	
	*Y*7–*Y*5–*C*5–*C*4–*B*4–*B*9	3	6	
	*Y*7–*Y*5–*C*5–*C*4–*B*4–*B*10	3	6	
	*Y*7–*D*7–*D*10	2	3	✓

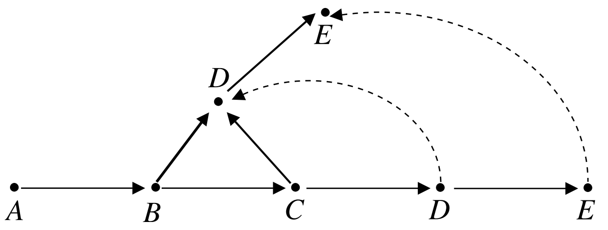

Fig. 10. An example of route improvement.

to communicate with X. On the other hand, for the conflicting slot 7, Y can use slot 7 to communicate with X. However, D has to adjust slot 7 to slot 10 to communicate with X.

4.3. Qos route maintenance and improvement

The dynamic changing of network topology is one characteristic of MANETs. Thus, even though a route path has been constructed successfully, it is also possible that the route path is broken or changed while it is in use. Thus, if an MH X detects the breakage or change of the route, the route maintenance and improvement protocol will be initiated to fill up the breakage or improve the change of the route. For completeness, the *QoS* route maintenance and improvement is briefly described as follows.

When X discovers a breakage of a route, X broadcasts an MT_REQ packet to search a new route to the destination. If the destination receives the MT_REQ packet, it will reply an MT_REP packet to establish a partial route to X. Otherwise, if X cannot find a route to the destination, X will send an MT_ERR packet to the source in order to re-initiate the DSRP to discover another new route. If the slot of an MH is not used for a period, the MH will automatically release the slot.

Since network topology is changing with the time, it is possible that the original route can be shortened in order to obtain better performance, such as the example shown in Fig. 10. In the example, the original path is $P_1=\{A\rightarrow B\rightarrow C\rightarrow D\rightarrow E\}$. Suppose D moves to the transmission range of B but still in the transmission range of C and E moves to a new location but still can connect with D. When B detects that D moves to its transmission range and the route to E can be shortened, B will send an IMP_REQ packet to D to initiate a Route Improvement operation. If $VS_{\overline{BD}}$ satisfies the QoS bandwidth requirement, D will reply an IMP_REP packet to B and reserve the corresponding slots to establish a new route. Thus, the route P_1 from A to E can be shortened to $P_2=\{A\rightarrow B\rightarrow D\rightarrow E\}$. Route improvement can also increase the number of routes in the network and raise the network throughput.

5. Simulation results

The performance of the DSRP in comparison with the related work Refs. [21,22] is evaluated via simulation study from several aspects, such as call success rate, network throughput, control overhead, average delay time, and the storage to perform the protocol. In the simulation, there are 30 MHs in 1000 m × 1000 m area. The transmission range is 300 m, the transmission rate is 11 M bit/s, and the QoS bandwidth requirements are 2 slots and 4 slots. Each frame has 16 data slots and each time slot is 5 ms. The source-destination pairs are randomly generated and the total simulation time is 1000 s.

5.1. Call success rate

The call success rates of each protocol (DSRP, DSRP w/o SAP, Zhu and Corson's, and Liao etc.'s) are compared under various traffic loads in Fig. 11. DSRP w/o SAP scheme is DSRP without the slot adjustment protocol. The mobility is 10 m/s. When the traffic is light, most of slots in each MH are idle, and there will be many valid slots in each link. Thus, each protocol can easily to search a QoS route in the network and has much better performance in call success rate. Observe that the best call success rate with 2 slots and 4 slots bandwidth requirement of DSRP are almost upward 98%.

However, when the traffic becomes heavy, the slots in each MH are used frequently, and then the number of valid slot in a link may be insufficient. Each link of a route does not easily to satisfy the QoS bandwidth requirement anymore. Thus, the call success rate of each protocol decreases. In consideration of slot conflict and slot reuse, DSRP can easily get more valid slot in a

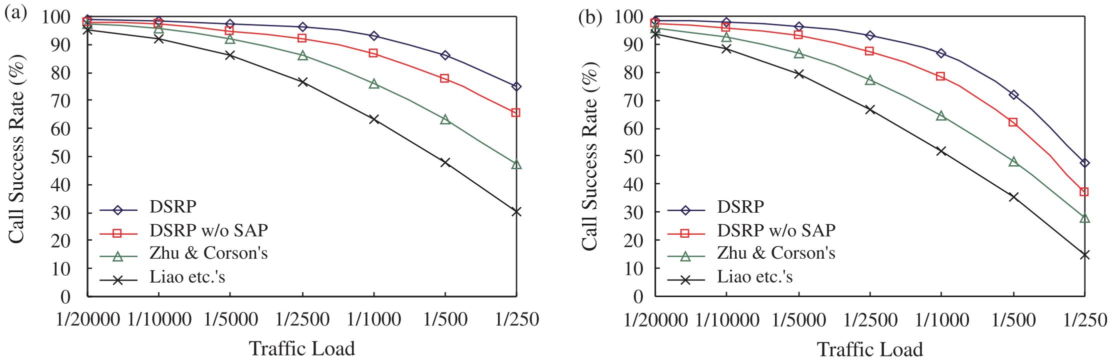

Fig. 11. The comparisons of DSRP with Zhu and Corson's and Liao etc.'s in terms of call success rate for variant traffic load:(a) QoS bandwidth requirement is 2 slots; (b) QoS bandwidth requirement is 4 slots.

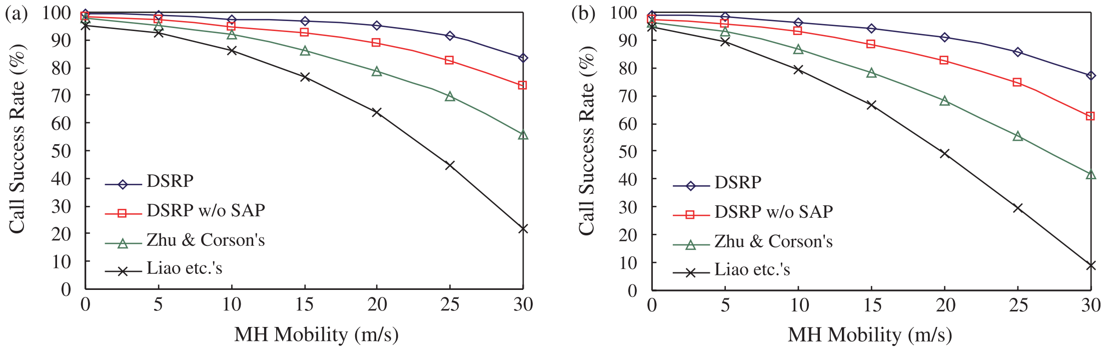

Fig. 12. The comparisons of DSRP with Zhu and Corson's and Liao etc.'s in terms of call success rate for variant mobility: (a) QoS bandwidth requirement is 2 slots; (b) QoS bandwidth requirement is 4 slots.

link by SIPs and select more suitable slot to send in a route by SDPs. Therefore, DSRP performs better in call success rate than other protocols. Note that the more slots QoS bandwidth requires, the more the call success rate decreases. The reason for this phenomenon is that the number of data slot in a TDMA frame is limited. If the more slots are needed in a link, then it is difficult to exist more QoS route in a link at the same time. Thus, the call success rate will decrease when the slot requirement in a link increases.

The Fig. 12 is shown the call success rates of different protocols in various mobility. The traffic load is 1/5000 (ms/frame). Under slow mobility, the calls of each protocol are almost successfully if there exist the routes. As the mobility increases, the routes may break easily because network topology changes rapidly. So the performances of all protocols decrease. However, DSRP, DSRP w/o SAP, and Zhu and Corson's are with route maintenance or route adjustment mechanisms. Thus, the route will be reconstructed quickly. When the route breaks, Liao etc.'s needs to reconstruct a QoS route from source to destination. Therefore, the performance of the calls in Liao etc's. is the worst. But the route maintenance of DSRP and DSRP w/o SAP reconstruct route more efficient than Zhu and Corson's. So the calls of Zhu and Corson's is lower than DSRP and DSRP w/o SAP. The SAP adjusts the slots when the fault occurred in slots reservation phase. So the DSRP performs better than DSRP w/o in call success rates.

5.2. Network throughput

In calculating network throughput, we only count packets successfully arriving at their destinations. According to the results shown in Fig. 13, when the traffic is slight, the throughput is low because of fewer data in transmission. As traffic becomes heavy, more data is in transmission, and the network throughput will increase. But network throughput is also related to call success rate. Higher call success rate means that there is more data transmission in the networks. So the performance of each protocol in network throughput is similar to the performance of each protocol in call success rate.

5.3. Control overhead

The control overhead is defined the number of control packets are needed for transmitting a unit of data successfully. The simulation results are shown in Fig. 14. As the traffic increases, the control overhead of all the protocols almost do not grow up apparently except DSRP. That means no matter the traffic in the networks is slight or heavy, the number of control packets for constructing a route is stable.

Liao etc.'s needs to maintain two-hop slots information, but others just maintain slots information of one-hop neighbors. Therefore, the control overhead of Liao etc.'s is the highest. The route adjustment scheme executed need some control packets. So the control overhead of DSRP is more than Zhu and Corson's and grows up with traffic increasing.

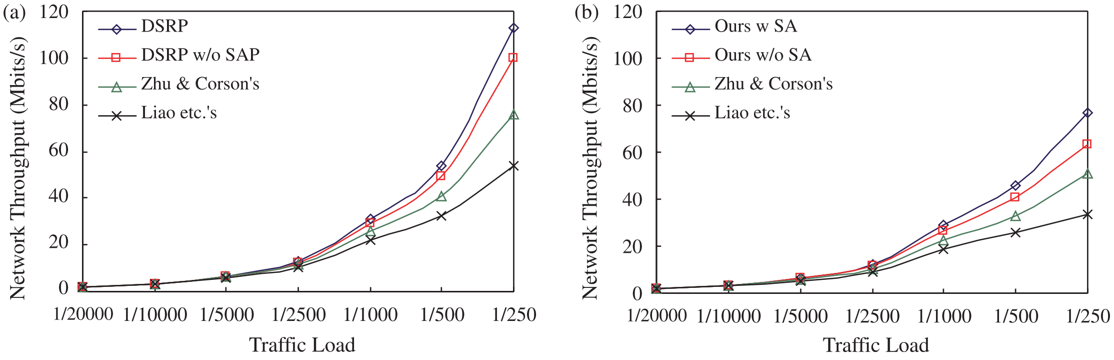

Fig. 13. The comparisons of DSRP with Zhu and Corson's and Liao etc.'s in terms of network throughput for variant traffic load: (a) QoS bandwidth requirement is 2 slots; (b) QoS bandwidth requirement is 4 slots.

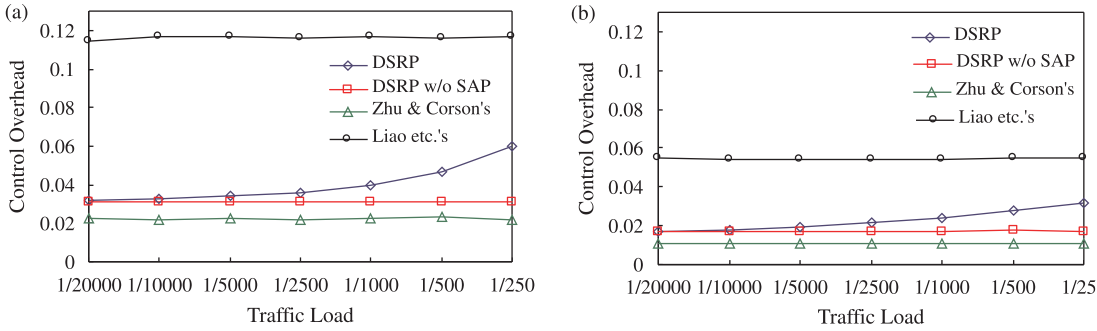

Fig. 14. The comparisons of DSRP with Zhu and Corson’s and Liao etc.’s in terms of control overhead for variant traffic load: (a) QoS bandwidth requirement is 2 stats; (b) QoS bandwidth requirement is 4 stats.

Note that the control overhead of each protocol with 4 slots required is lower than the control overhead of each protocol with 2 slots required. The reason for the phenomenon is that the control overhead of establishing a QoS route is the same no matter how many QoS slots are needed. Consequently, the more QoS slots for a link needs, the less control overhead for a packet is.

5.4. Average delay time

The simulation result of average delay time is shown in Fig. 15. Delay time is the time which constructing a successful route needs. In Fig. 15, the average delay time of all protocols is stable except DSRP. The Liao etc.’s could discover a route by two-hops neighboring information, but DSRP and Zhu and Corson’s are only by one-hop neighbors information. So more calculation and decision are needed. Therefore, the delay time of Liao etc.’s is the fewest. DSRP considers more factors in slots decision than Zhu and Corson’s. Then the calculation and decision time of DSRP is more than Zhu and Corson’s. However, the difference is similar. Note that the average delay time of DSRP grows up with traffic increasing because that slot adjustment protocol works in DSRP. Thus, a QoS route constructed needs much time than other protocols.

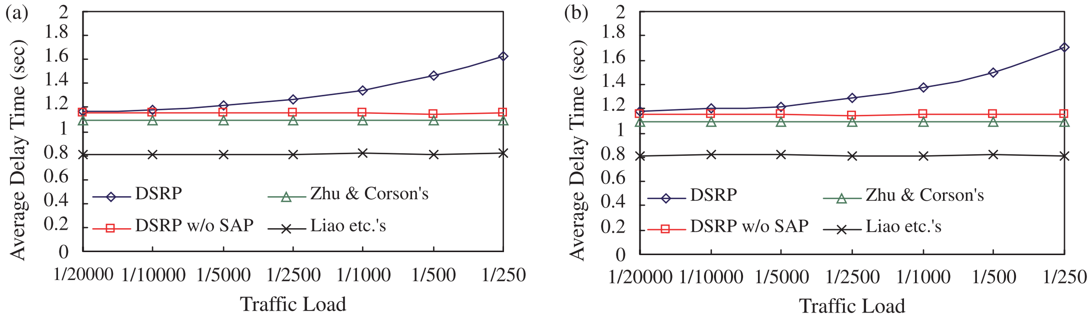

Fig. 15. The comparisons of DSRP with Zhu and Corson’s and Liao etc.’s in terms of average delay time for variant traffic load: (a) QoS bandwidth requirement is 2 slots; (b) QoS bandwidth requirement is 4 slots.

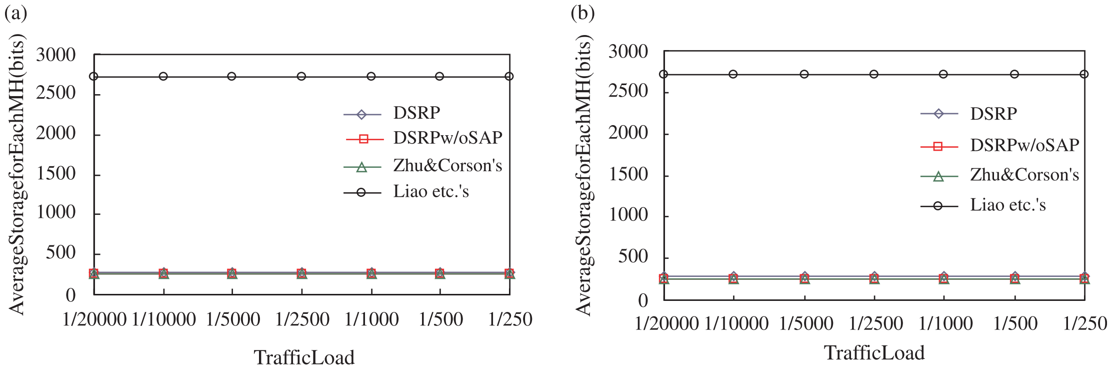

Fig. 16. The comparisons of DSRP with Zhu and Corson’s and Liao etc.’s in terms of average storage for each MH with variant traffic load: (a) QoS bandwidth requirement is 2 slots; (b) QoS bandwidth requirement is 4 slots.

5.5. Average storage for each node

Average storage for each MH means that when operating routing protocol, how much memory space needs. The average storage for each MH in different protocols are shown in Fig. 16. The memory space are needed approximately 2700 bits for Liao etc.'s protocol because of maintaining the two-hops neighboring information. However, DSRP and Zhu and Corson's protocol just only need to record the one-hop neighboring information. Thus, the memory space are needed approximately 300 bits. So the average storage for each MH in Liao etc.'s protocol is higher than others.

6. Conclusions

As well-known, QoS service is much demanding with the increasing of the real-time multimedia applications nowadays. Multimedia transmission over wireless networks is becoming important as well. Thus, QoS routing on MANETs is deemed a significant research issue.

This paper proposed several dynamical bandwidth allocation strategies for QoS routing on TDMA-based MANETs. Comprehensively speaking, a distributed slots reservation protocol (DSRP) is proposed, which can construct QoS routes only depending on one-hop neighbor information of each MH. In DSRP, the slot inhibited policies (SIPs) are used to determine valid slots for a link. From the valid slots, which slots are actually selected to use is determined by slot decision policies (SDPs). In SDPs, three heuristic policies, 3BDP, LCFP, and MRFP, are proposed. The main concept behind LCFP and MRFP is the slot reuse. The more the slots are reused, the more the QoS routes exist. Thus, the slot shortage problems, including SSSR and SSNR problems, can be alleviated accordingly. On the other hand, a slot adjustment protocol (SAP) is proposed to coordinate the slot usage of a conflicting MH with its neighboring MHs during the route reply and reservation phase in order to accommodate more routes in the network. The slot adjustment algorithm (SAA) invoked in SAP is a branch-and-bound algorithm, which is an optimum algorithm in terms of the number of slots to be adjusted. Besides, QoS route maintenance and improvement is also provided. The simulation results show that DSRP cannot only increase the success rate in search of a QoS route with bandwidth requirement guaranteed but also raise the network throughput and efficiency of the network. DSRP indeed works well for QoS routing on TDMA-based mobile ad hoc networks.

Currently, wireless sensor networks (WSNs) are getting important both in academic and in industrial fields. QoS routing on WSNs is also an important research issue, especially for event-driven WSNs. WSNs have the characteristics that sensor nodes are low-power, short-range, and small-size devices. Sensors are stationary, randomly deployed and prone to failure. Therefore, investigating a scalable and dynamic QoS routing on TDMA-based WSNs is our possible future work.

Acknowledgements

This work was supported by the National Science Council of the Republic of China under Grants NSC 94-2213-E-032-014 and NSC 94-2524-S-032-003.

References

[1] J. Broch, D.A. Maltz, D.B. Johnson, Y.C. Hu, J. Jetcheva, A performance comparison of multi-hop wireless ad hoc network routing protocols, in: Proceedings of the ACM International Conference on Mobile Computing and Networking (MOBICOM), 1998, pp. 85–97.

[2] J. Broch, D.B. Jphnson, D.A. Maltz, The dynamic source routing protocol for mobile ad hoc networks, IETF Internet Draft (draft-ietf-manet-dsr-03.txt), October 1999.

[3] D.B. Johnson, D.A. Maltz, Dynamic source routing in ad hoc wireless networks, in: T. Imielinski, H. Korth (Eds.), Mobile Computing, vol. 353, Kluwer Academic Publishers, Norwell, Mass., 1996, pp. 153–181.

[4] C.E. Perkins, E.M. Royer, Ad hoc on-demand distance vector routing, in: Proceedings of the Second IEEE Workshop on Mobile Computing Systems and Applications, 1999, pp. 90–100.

[5] C.E. Perkins, E.M. Royer, S.R. Das, Ad hoc on-demand distance vector (AODV) routing, IETF Internet Draft (draft-ietf-manet-aodv-06.txt), July 2000.

[6] C.E. Perkins (Ed.), Ad Hoc Networking, Addison–Wesley, Reading, MA, 2001.

[7] V. Park, M.S. Corson, A highly adaptive distributed routing algorithm for mobile wireless networks, in: Proceedings of the IEEE INFOCOM, the Annual Joint Conference of the IEEE Computer and Communications Societies, 1997, pp. 1405–1413.

[8] T.W. Chen, P. Krzyzanowski, M.R. Lyu, C. Sreenan, J. Trotter, A VC-based API for renegotiable QoS in wireless ATM networks, in: Proceedings of International Conference on Universal Personal Communication (ICUPC), 1997.

[9] T.W. Chen, M. Gerla, T.C. Tsai, QoS routing performance in a multi-hop, wireless networks, in: Proceedings of International Conference on Universal Personal Communication (ICUPC), 1997.

[10] S. Chen, K. Nahrstedt, Distributed quality-of-service routing in ad hoc networks, IEEE Journal on Selected Areas in Communications 17 (1999).

[11] S.B. Lee, A.T. Campbell, QoS issues in mobile ad hoc networks, in: Seminar on Mobile Multimedia Communication—Systems and Networks, Invited Talk Dagstuhl, Germany, 1999.

[12] S.H. Shah, K. Nahrstedt, Predictive location-based QoS routing in mobile ad hoc networks, in: Proceedings of the IEEE International Conference on Communications (ICC), vol. 2, 2002, pp. 1022–1027.

[13] C.R. Lin, On-demand QoS routing in multihop mobile networks, in: Proceedings of the IEEE INFOCOM, the Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 3, 2001, pp. 1735–1744.

[14] W. Su, M. Gerla, Bandwidth allocation strategies for wireless ATM networks using predictive reservation, in: Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), vol. 4, 1998, pp. 2245–2250.

[15] W.-H. Liao, Y.-C. Tseng, S.-L. Wang, J.-P. Sheu, A multi-path QoS routing protocol in a wireless mobile ad hoc network, Telecommunication Systems (special issue of best papers from ICN2001) 19 (2002) 329–347.

[16] Y.-S. Chen, Y.-C. Tseng, J.-P. Sheu, P.-H. Kuo, An on-demand, link-state, multi-path QoS routing in a wireless mobile ad-hoc network, Computer Communications 27 (2004) 27–40.

[17] Y.-S. Chen, Y.-T. Yu, Spiral-multi-path QoS routing in a wireless mobile ad hoc network, IEICE Transactions on Communications E87-B (2004) 104–116.

[18] C.R. Lin, J. Liu, QoS routing in ad hoc wireless networks, IEEE Journal on Selected Areas in Communications 17 (1999) 1426–1438.

[19] C.R. Lin, An on-demand QoS routing protocol for mobile ad hoc network, in: Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), vol. 3, 2000, pp. 1783–1787.
[20] W. Zhang, J. Li, X. Wang, Cost-efficient QoS routing protocol for mobile ad hoc networks, in: Proceedings of the International Conference on Advanced Information Networking and Applications, vol. 2, 2005, pp. 13–16.
[21] W.-H. Liao, Y.-C. Tseng, K.-P. Shih, A TDMA-based bandwidth reservation protocol for QoS routing in a wireless mobile ad hoc network, in: Proceedings of the IEEE International Conference on Communications (ICC), vol. 5, 2002, pp. 3186–3190.
[22] C. Zhu, M.S. Corson, QoS routing for mobile ad hoc networks, in: Proceedings of the IEEE INFOCOM, the Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 2, 2002, pp. 958–967.
[23] S.L. Chen, W.B. Heinzelman, QoS-aware routing based on bandwidth estimation for mobile ad hoc networks, IEEE Journal on Selected Areas in Communications 23 (2005) 561–572.