Synergizing Wireless Communication Technologies to Improve Internet Downloading Experiences

Ting-Yu Lin, *Member*, *IEEE*, Tai-Yi Huang, and Chia-Fu Hsu

Abstract—Considering downloading traffic from the internet, in this paper, we propose a synergized framework (SF), consisting of heterogeneous wireless communication technologies, multi-mode (multi-interface) mobile, and fixed wireless hosts capable of operating over multiple orthogonal (non-overlapping) radio channels, to realize better downloading experiences for users via cooperation between different wireless technologies. An SyNerGized (SNG) routing protocol is devised to enable the proposed framework. Given perceived network information, SNG performs computations based on linear formulations and obtains an optimized route for packet delivery. To adapt to network dynamics, a reactive version of SNG, entitled Reactive SyNerGized (RSNG) routing protocol, is proposed to alleviate the network from constantly keeping track of link capacities within a certain scope of neighborhood. Since the downloading throughput may be bounded by either the internet gateway capacity λ_{gw} or ad hoc throughput λ_{ah}, RSNG judiciously propagates Route REQuest (RREQ) until the downloading throughput is bounded by λ_{ah} over the ad hoc domain, effectively eliminating unnecessary RREQ flooding. Our main objective is to improve achieved user downloading throughput via the cooperative (synergized) communication model and its corresponding routing mechanisms. Simulation results demonstrate the benefits brought by the unified architecture and corroborate the efficacy of the proposed routing techniques.

Index Terms—Synergized framework (SF), wireless internet access, multi-hop ad hoc network, routing protocol, linear optimization

1 Background

In the past decade, we have witnessed a multitude of communication technologies evolved into mature wireless Internet access options. Wireless wide-area networks (WWANs), wireless metropolitan-area networks (WMANs), and wireless local-area networks (WLANs) possess complementary characteristics in terms of transmission range and attainable data rate. Table 1 summarizes respective features of those state-of-the-art wireless communication systems (statistics excerpted partially from the empirical data documented in [3], [11], [15], [24], and [26]). Among those technologies, IEEE 802.11 family standards are traditionally classified as WLAN systems, while WCDMA/HSDPA, LTE, and WiMAX are usually recognized as WWAN or WMAN cellular communication platforms. Furthermore, according to the attainable data rates, cellular standards are further categorized into 3G (WCDMA), 3.5G (HSDPA), and 4G (LTE and WiMAX) systems. Generally speaking, WLAN systems have limited transmission distances (normally in tens of meters), but achieve higher data communication rates (in tens of, or even hundreds of Mbps). Infrastructure access to the IP network in WLANs is via access points (APs). Due to its limited communication range, the IEEE 802.11-based WLAN systems have developed a multi-hop relaying mode (termed as ad hoc multi-hop mode) to extend effective AP coverage. On the other hand, 3G WWAN systems, such as WCDMA, are capable of transmitting data for long distances (several kilometers), but have relatively low data rates (in hundreds of Kbps to at most 2 Mbps). Access to the IP core network in WWAN/WMAN systems is via base stations (BSs). Several researchers have observed the complementary characteristics between WLANs and 3G WWANs, along with the usefulness offered by the ad hoc multi-hop relaying mode, therefore proposed to integrate these heterogeneous systems in order to provide an integrated wireless environment capable of serving ubiquitous connections with high data rates. Such integration issues and the benefits of enabling cooperation (interworking) between 3G, WLAN, and ad hoc multi-hop communication models can be found in [9], [12], [13], [17], [18], [20], [22], and [23].

The integration of various wireless technologies is a nontrivial task, involving designs at all layers of the protocol stack. We observe that designing at the network layer is perhaps the most critical yet challenging in the sense that different capabilities and functionalities of heterogeneous wireless access platforms need be considered in a unified routing process. Most past integration solutions tap into the aforementioned complementary characteristics (long/short transmission distance accompanied by low/high data rate) possessed by respective wireless systems when designing their routing algorithms. However, with the emergence of high-speed 4G technologies (LTE and WiMAX), the

• *T.-Y. Lin is with the Department of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu 300, Taiwan. E-mail: ting@cm.nctu.edu.tw.*
• *T.-Y. Huang and C.-F. Hsu are with the Institute of Communications Engineering, National Chiao Tung University, Hsinchu 300, Taiwan. E-mail: airwalx@gmail.com, tsukinowabr.cm99g@nctu.edu.tw.*

Manuscript received 06 Oct. 2012; revised 08 July 2013; accepted 10 July 2013. Date of publication 22 July 2013; date of current version 14 Oct. 2014. Recommended for acceptance by P. Bellavista.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2013.147

TABLE 1
Comparison of Wireless Technologies

Technology / Attribute	IEEE 802.11a, b, g	IEEE 802.11n	WCDMA / HSDPA	LTE	WiMAX
Frequency band	2.4 ~ 2.497 GHz (b, g, n) 5.15 ~ 5.35 GHz (a, n) 5.725 ~ 5.285 GHz (a, n)		700 ~ 2100 MHz		2.3 ~ 2.4 GHz 2.496 ~ 2.69 GHz 3.4 ~ 3.6 GHz 3.4 ~ 3.8 GHz
Standard specified data rate	54 Mbps (a, g) 11 Mbps (b)	600 Mbps	2 Mbps (WCDMA) 14 Mbps (HSDPA)	300 Mbps (2 x 2 MIMO)	75 Mbps (2 x 2 MIMO)
Commercial data rate	~ 27 Mbps (a) ~ 22 Mbps (g) ~ 5 Mbps (b)	50 ~ 144 Mbps	384 Kbps ~ 2 Mbps (WCDMA) 1~10 Mbps (HSDPA)	10 ~ 100 Mbps	2 ~ 10 Mbps
Transmission range	30 ~ 45 m (a) 30 ~ 100 m (b) 30 ~ 100 m (g)	50 ~ 300 m	3 ~ 12 km	5 ~ 100 km	2 ~ 10 km

complementarity becomes no longer salient, making the integration task even more challenging. In this paper, we try to address this new challenge by proposing a synergized framework (SF), considering various state-of-the-art wireless access technologies, and design an optimized routing process for such a generic hybrid networking environment.

Fig. 1 illustrates the concept of a synergized framework (SF), where clients E, F, G, and H have direct Internet connections via BS/AP and can act as potential proxies (gateways) for other clients. In this framework, participating clients may have multiple interfaces operating over orthogonal (non-overlapping) radio channels. For instance, there are 3 non-overlapping channels in IEEE 802.11 b/g, while 8 (up to 12) non-overlapping channels are available in IEEE 802.11a. With the prevalence of inexpensive wireless hardware components, it becomes affordable to equip multiple radio interfaces on a communication host (client). As shown in Fig. 1, clients A, D, F, and G are capable of transmitting simultaneously over orthogonal channels, Ch1 and Ch2, with different link qualities with their own neighboring clients. Imagine a routing protocol designed to synergize the cooperation between participating clients. Take client A in Fig. 1 for example, what attribute(s) should act as the metric(s) in determining the best route to access the Internet? Obviously, among the four proxy (gateway) options, theoretically LTE BS can provide the highest downlink data rate for client A via direct link with proxy H (1 relaying hop). However, the radio link quality over Ch1 between clients A and H is weak, unfortunately invalidating the high downlink bandwidth provided by LTE BS. On the other hand, since radio link qualities between A-D (over Ch2) and D-G (over Ch1 and Ch2) are both good, client A may instead consider selecting WiMAX BS as the downlink provider via proxy G, despite the fact of having to travel 2 relaying hops. In addition, by leveraging the multiple interfaces available on clients A, D, and G, the routing protocol can possibly enable multi-path flow by simultaneously taking route G-D-A over high-quality Ch2 and route G-D-B-A over high-quality Ch1 to further increase the effective downloading rate for client A. Consequently, the best route in this case does not necessarily involve proxy with the highest downlink capacity, neither proxy with the shortest relaying path. In other words, *either proxy (gateway) capacity or relaying hop distance alone no longer serves as the single metric for selecting the best route* in such a synergized (integrated) framework. Rather, those factors, together with ad hoc multi-hop link qualities, should all be taken into consideration in a holistic manner. Motivated by the interacting tradeoffs, we design routing protocols that try to optimize the foregoing factors when selecting the best routes. Another observed benefit shown in Fig. 1 is that client B, while temporarily out of direct

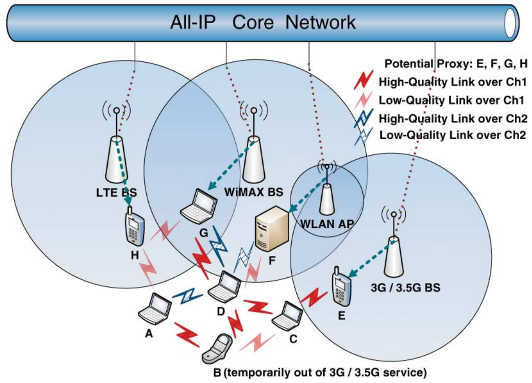


Fig. 1. Illustration of the proposed synergized framework (SF) to enable seamless internetworking between (mobile) wireless devices.

connection with its BS, can select another route via multi-hop relaying to reach the core network and still enjoy the Internet services (with possibly better quality than its original downlink rate) under this cooperative model. Such hybrid network combining centralized Internet access model and distributed ad hoc multi-hop communication behavior is generally beneficial in terms of reduced deployment costs for infrastructure providers and increased connectivity opportunities for end users (clients).

The remainder of this paper is organized as follows. In Section 2, we review related earlier research efforts in the area of integrating heterogeneous wireless technologies, and point out our unique contributions. Section 3 presents the SNG protocol, elaborating on network neighborhood discovery process, route optimization details, and packet forwarding procedures. In Section 4, a reactive routing approach, entitled RSNG, is proposed for the purpose of adapting to dynamic environments. We conduct extensive simulations to validate the routing performance of our SNG and RSNG protocols, while exhibiting the benefits brought by the synergized framework in Section 5. Finally, Section 6 draws our conclusion.

2 Related Work and Our Contributions

Due to the availability of dual-mode terminals and popularity of ad hoc networking technologies, several pioneering works have been proposed to explore the multi-hop relaying possibility in a cellular system [5], [25], [28]-[32]. In [5], an ad hoc Global System for Mobile Communications (A-GSM) architecture is presented to provide connectivity for users in dead spots. The proposed architecture intends to increase cellular capacity and link robustness. With the similar idea in mind, the authors in [25] devise an opportunity-driven multiple access (ODMA) mechanism to support multi-hop connections. Specifically, ODMA breaks down a single CDMA transmission into a number of smaller radio hops to relay the packets. Due to reduced transmit power and co-channel interference, the cellular coverage-capacity tradeoffs can be optimized. However, ODMA does not support communications for users outside the cellular coverage. Also exploiting dual-mode terminals capabilities, the iCAR authors in [28] introduce the ad hoc relay stations (ARSs) to be deployed by the network operator at cell boundaries with limited mobility under the control of cellular mobile switching center

(MSC). The deployed ARSs have two interfaces capable of communicating with the cellular base station and, simultaneously, communicating with other ARSs in ad hoc mode via WLAN interface. ARSs in iCAR can divert excess traffic from a congested cell to other lightly-loaded neighboring cells. With the traffic adaptations performed by ARSs, iCAR aims to balance traffic load between cells. Furthermore, iCAR also increases cellular coverage by enabling users out of cellular coverage to access the system through the assistance of deployed relay stations (ARSs). Since mobile hosts (MHs) in iCAR have only one air interface for communicating with the cellular system, in [31], the authors extend iCAR by including another ad hoc network interface (A-interface) into MHs (so that MHs can participate in the relaying procedure as well). An adaptive routing protocol, entitled as ARFA, is introduced to facilitate flexible access (FA) in the extended iCAR-FA architecture. Another proposal to achieve cellular load balancing can be found in [29] and [30], where a mobile-assisted data forwarding (MADF) mechanism is introduced to forward part of the traffic in a crowded (hot) cell to some free (cold) cells. Different from the usage of stationary ARSs adopted by iCAR, MADF utilizes mobile stations (MSs) that are located between hot and cold cells as relaying nodes. By implementing MADF in Aloha and TDMA networks, the authors show that the throughput in a hot cell, which is surrounded by several cold cells, can be significantly improved. In [32], the self-organizing packet radio ad hoc network with overlay (SOPRANO) project is another effort to incorporate the ad hoc relaying capability in a cellular system. Several aspects, including bandwidth allocation, access control, routing, traffic control, and profile management, have been investigated in the proposed SOPRANO architecture. Focusing on connection establishment and self-organization issues, the SOPRANO authors investigate the optimal transmission strategy in the multi-hop network with the objective of enhancing cellular capacity. All of the aforementioned research works belong to earlier efforts in the area of integrating cellular and ad hoc multi-hop networks. However, those previous works do not consider integration with WLAN APs, let alone those state-of-the-art broadband cellular systems emerging in recent years.

Motivated by the idea of incorporating ad hoc multi-hop relaying mode into infrastructure-based single-hop cellular communication, a number of follow-up research works have been proposed to integrate 2.5G/3G/3.5G WWAN with IEEE 802.11-based WLAN [6], [21], [27]. In [21], a unified cellular and ad hoc network (UCAN) architecture is proposed to enhance cellular throughput by providing low data-rate users with better downlink channel quality through proxy clients (acting as Internet gateways). A mobile client in UCAN has dual interfaces connecting to both 3G infrastructure and IEEE 802.11 ad hoc multi-hop network. The basic rationale behind UCAN is to find proxy clients with higher downlink data rates for users experiencing poor cellular channel qualities, where selected proxy client should perform the downloading on behalf of requesting user and then forward received packets in ad hoc multi-hop relaying mode via IEEE 802.11 interface to the intended destination (requesting user). Two proxy discovery mechanisms are introduced: greedy and on-demand protocols. The greedy protocol is proactive and tries to locate possible proxy client with better downlink quality, starting the search from immediate neighbors, 2-hop neighbors, 3-hop neighbors, and so on, in a greedy manner until no better neighbor can be found. This approach is simple but comes with one attendant drawback: such a greedy path may not always locate the proxy with the best cellular channel rate, due to possible local minimum occurring in the neighborhood of the requesting client. In order to address this problem, another on-demand protocol is proposed to perform proxy discovery in UCAN. Instead of greedily reaching out from the requesting client, on-demand protocol searches for the best proxy by propagating requesting message through the ad hoc network (with a limited number of hops controlled by a time-to-live, TTL, field). The UCAN authors evaluate the performance in a simulated HDR cell, with IEEE 802.11b as the simulated ad hoc interface. Also utilizing dual-mode terminals as relay proxies, the authors in [27] include WLAN APs as possible Internet gateways to provide seamless roaming between WWANs and WLANs. The proposed integrated WWAN/WLAN two-hop-relay architecture intends to enhance cellular system capacity and extend WLAN coverage for up to two hops. More recently, a cross-layer study over integrated 3G and WLAN systems has been presented to enable interoperability between heterogeneous communication environments [6]. The suggested cross-layer algorithm jointly performs 3G resource allocation and ad hoc routing in order to increase 3G system performance. With a slightly different design metric from the previous attempts, the authors in [6] also try to select relaying route without disturbing existing WLAN background traffic. We observe that the underlying goal of foregoing research works mainly focuses on improving (downlink) capacity in cellular systems. Moreover, no specific proxy (gateway) load balancing strategy is available to judiciously divert user traffic to another possible candidate when the downlink bandwidth of selected proxy (gateway) is partially occupied by existing users. In this paper, we consider a generic networking paradigm that contains heterogeneous wireless Internet access technologies in a unified architecture, referred to as a synergized framework (SF), to enable interoperability between those communication platforms. We aim to improve effective user downloading throughput, and perform gateway load balancing among requesting users in order not to exhaust some high-capacity proxies (gateways), which are commonly favored by proxy (gateway) selection algorithms. In addition, as the development of upcoming broadband 4G systems, such as WiMAX and LTE standards, progresses aggressively, we observe that *Internet gateways may no longer be the communication bottleneck*. On the contrary, communication bottleneck may exist in the ad hoc relaying domain. However, none of the above works considers the ad hoc bottleneck, where wireless link qualities vary and medium contentions dominate the effective throughput. Furthermore, the channel diversity provided by multiple orthogonal (non-overlapping) radio channels has not been leveraged to further increase the ad hoc network capacity either.

2.1 Our Contributions

While most traffic in the proposed SF is destined for Internet access, such communication behavior shares the similar traffic pattern manifested by a wireless mesh network (WMN). Internet data transmissions over WMNs are realized by *anycast* routing,[1] which is usually adapted from *unicast* routing techniques over the ad hoc networks. Plenty of routing protocols targeted on ad hoc and wireless mesh networks have been developed [7], [8], [16]. However, there are structural differences between our SF and WMN. In essence, participating devices in the SF usually roam around in the environment where surrounding wireless link qualities vary over time. Furthermore, Internet gateways in the SF have *heterogeneous* downlink rates through different cellular access technologies (as illustrated in Fig. 1), whereas a WMN generally deals with multiple homogeneous gateways [7], [16]. As a result, anycast WMN routing mechanisms are not directly applicable to effectively find the best gateway in our SF. Specifically, *a good routing over the SF should consider heterogeneous gateway capabilities, varying surrounding wireless link qualities, and also possible channel diversity provided by multi-interface devices, in order to achieve optimized network-wide throughput performance*. For this purpose, we propose the proactive SNG (Section 3) and reactive RSNG (Section 4) routing protocols, particularly designed for the SF environment. Different from existing mesh routing, our protocols compute the best routes based on a more *diverse communication model*.

In addition, thanks to the near-saturation of iPhones and Android-based handsets, it is likely that many roaming clients in the SF are dual-mode (or multi-mode) smartphones having both 3G (or possibly 4G in the near future) and WiFi connections. We observe that with two Internet access choices, those smartphones simply choose WiFi over 3G or allow users set manually [1], [4], not really synergizing heterogeneous communication standards in a mixed mode. However, *a good selection of Internet access method involves more than the instantaneous downlink rate of a single communication technology and can possibly be enhanced by performing multi-hop relaying to utilize another cellular access system* (as illustrated in Fig. 1). In light of this, we make a unique contribution by proposing SNG and RSNG protocols aiming to really take advantage of various contemporary wireless Internet access technologies within a user's neighborhood. With our protocols running on clients, a user may give up its own Internet connection and turn to other's after performing our evaluation algorithms (unlike the scenario in a WMN where gateway nodes always use their own Internet connections). We expect to realize *moderate synergy* of different communication technologies, so that users can benefit from this cooperative model.

3 SNG Routing Protocol

In this section, we present our SNG routing protocol, which is customized to operate over the proposed synergized framework (SF). As explained and illustrated in Fig. 1, the SF is an integrated network containing heterogeneous wireless Internet access technologies (with distinct downlink capacities), a variety of communication devices (with different numbers of interfaces), and various ad hoc links (with varying channel qualities). In order to obtain the best route for a downloading request made by a participating client (user) in such a hybrid network, the SNG protocol takes a holistic approach by considering available proxies (gateways), remaining downlink capacities, and ad hoc link connection qualities, when making an optimized routing decision. Below we describe the necessary components that are included in the SNG mechanism. In Section 3.1, a network construction procedure based on periodical table exchange is introduced. The obtained network information is then used to compute the best (optimized) route for the requested downloading flow based on linear programming methodology. We provide the detailed optimization formulations in Section 3.2. Once the best route and corresponding link flows are determined, the data downloading can be performed via the selected proxy (gateway). Section 3.3 describes the signaling process and packet forwarding strategy. Finally, we summarize the SNG routing protocol in Section 3.4.

3.1 Network Information Construction

Since the SNG routing protocol tries to take both the infrastructure downlink capabilities and ad hoc connectivity status into account, a moderate amount of network information is necessary in determining a good route. Fortunately, since most devices in SF have their own Internet connections (though may not be good enough), gateway searches can be limited within a reasonable range of neighborhood without being propagated throughout the whole network. These searches are performed in hopes of finding better routes (with higher downloading qualities) from the Internet. In the ad hoc multi-hop network domain, SNG requires every participating client (node) to estimate and periodically update the average data rates for wireless links originating from its immediate neighbors. The estimation can be achieved using certain packet-pair probing mechanism [10]. By counting the number of successfully received broadcast advertisements, a node can obtain the packet delivery ratio from a neighbor to approximate the effective link rate. More details for improving the estimation accuracy can be found in [10]. The estimated data rates are kept by a node in the **C-Table** (capacity table) to reflect the ad hoc wireless link capacities over certain channels. In addition to the link capacities from neighbors, a node should also include the information regarding proxy (gateway) availability and attainable downlink bandwidth (if capable of acting as a gateway) in the C-Table. Specifically, the **Gateway Bit** (set to *true* or *false*) and **Gateway Capacity** fields should be provided in the C-Table as well.

Whenever receiving a C-Table with new entries from others, a node learns and records those new entries in its own C-Table. In this way, the network knowledge perceived by a node can be effectively expanded. To ensure the freshness of received C-Table, a node is required to attach and increase the **Sequence Number** whenever advertising a new C-Table. In addition, we incorporate a **TTL** (time-to-live) field to limit how far (how many hops) the C-Table can travel. This limitation intends to impose a scoped neighborhood discovery on the table exchange process. Although a more complete network knowledge can lead to a better optimized routing decision (as one may soon observe from the computation models presented in Section 3.2), too many C-Table exchanges pose significant communication overhead, possibly trading off the optimized benefit that can be achieved. We investigate

1. Anycast routing is designed for networks where some client nodes require a route to any member from a certain group of service nodes. In the context of WMNs, the mesh nodes are the clients, and the gateways are the service nodes.

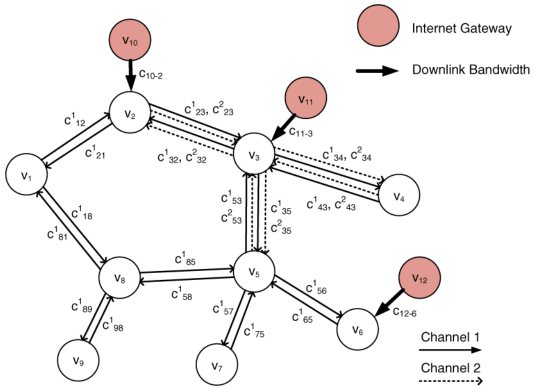


Fig. 2. Constructed network topology, along with corresponding link capacities, potential gateways, and available Internet downlink bandwidths within 3-hop neighborhood of client node v_1.

this design issue in Section 5.1. In fact, the simulation results show that the proposed SNG protocol only needs a moderate amount of discovered neighborhood information to outperform other routing approaches.

Fig. 2 illustrates an example 3-hop neighborhood discovered by client (node) v_1. Potential proxy clients for v_1 downloading request include v_2, v_3, and v_6, connecting to Internet gateways v_{10}, v_{11}, and v_{12}, respectively. Note that nodes v_{10}, v_{11}, and v_{12} represent BSs/APs capable of providing downlink bandwidths from the IP infrastructure. In our routing modeling (presented in Section 3.2), Internet gateways are special clients that do not generate downloading requests. Two interfaces (operating on non-overlapping Channel 1 and Channel 2) are available at nodes v_2, v_3, v_4, and v_5, making possible simultaneous communications over both channels. Furthermore, wireless link asymmetry is considered in our route computation to reflect realistic wireless radio channel conditions. As shown in Fig. 2, link capacity from node v_3 to v_5 over Channel 1 (c_{35}^1) is not necessarily identical to link capacity in the reverse direction (c_{53}^1). Similarly, c_{35}^2 is not always equal to c_{53}^2 when Channel 2 is used.

3.2 Downlink Flow Routing Computation

Given the neighborhood information obtained by limited C-Table exchanges, a requesting client (node) computes the best route and corresponding traffic flow distributions based on linear optimization. Define a directed graph $G = (V, E)$, where set V contains all nodes (clients), and set E includes all edges (wireless links). Below we introduce the notations to be used in our optimization formulations.

- As explained in Fig. 2, there are two types of clients: clients that request for downloading services (mobile clients), and clients that do not generate downloading requests (Internet gateways). We use V^m to denote the set of mobile clients, and V^g to indicate the set of Internet gateways. Consequently, V^m and V^g are disjoint sets and their union equals to set V ($V = V^m \cup V^g$).
- For wireless link originating from node v_i to node v_j over channel k, we denote the edge as e_{ij}^k, while e_{ji}^k is used to represent the edge in the reverse direction over the same channel. Assume there are totally K non-overlapping channels in the network. In case of interface k $(1 \le k \le K)$ not being equipped on either node v_i or v_j or both nodes, the edges e_{ij}^k and e_{ji}^k simply do not exist.

TABLE 2
Summary of Notations Used in Our Optimization Model

Notation	Description
V^g	Set of potential gateways (proxies)
V^m	Set of mobile clients
N_i	Neighbor set of node v_i
I_{ij}^k	Set of interfering links for communication between node v_i and v_j over channel k
c_{ij}^k	Estimated link capacity from node v_i to v_j over interface k
f_{ij}^k	Downlink traffic flow from node v_i to v_j over interface k (to be computed)

- Set N_i contains all immediate neighbors of node v_i.
- To model the medium contention behavior in the IEEE 802.11-based ad hoc network domain, we define I_{ij}^k as the set of interfering links that interfere with communication from node v_i to node v_j over channel k. One may define this set differently based on various interference models. In our modeling, we adopt to include all wireless links within two hops of nodes v_i and v_j in set I_{ij}^k, considering RTS/CTS four-way handshaking is used in the IEEE 802.11 DCF access mode.
- Define c_{ij}^k as the estimated link capacity (rate) from node v_i to node v_j over channel (interface) k. In case of interface $k\,(1 \le k \le K)$ not being equipped on either node v_i or v_j or both nodes, we simply let $c_{ij}^k = c_{ji}^k = 0$. Furthermore, wireless link asymmetry is considered, so that c_{ij}^k is not necessarily equal to c_{ji}^k. However, if either c_{ij}^k or c_{ji}^k is zero, while the other is not ($\forall v_i, v_j \in V^m$), we should avoid such links by letting $c_{ij}^k = c_{ji}^k = 0$, in order for the IEEE 802.11 acknowledgement mechanism to work correctly (requiring bi-directional links).
- Finally, f_{ij}^k denotes the optimized traffic flow distribution from node v_i to v_j over channel (interface) k that we intend to obtain through our computations.

A brief summary of notations is provided in Table 2.

For each potential gateway candidate $v_g \in V^g$, a requesting node $v_r \in V^m$ evaluates the maximum downlink throughput attainable by current gateway capacity and routing flow distributions. The evaluation is based on linear optimization by setting the objective function as

$$Maximize \quad \lambda = \sum_{k=1}^{K} \sum_{v_j \in N_g} f_{gj}^k, \tag{1}$$

where λ represents the effective downlink throughput injected into the ad hoc multi-hop network, while satisfying the following constraints.

Flow Conservation Constraint:

For requesting node $v_r \in V^m$

$$\sum_{k=1}^{K} \sum_{v_j \in N_r} f_{jr}^k = \sum_{k=1}^{K} \sum_{v_j \in N_g} f_{gj}^k, \tag{2}$$

while for other nodes $v_i \in V^m$, $v_i \neq v_r$

$$\sum_{k=1}^{K} \sum_{v_j \in N_i} f_{ij}^k = \sum_{k=1}^{K} \sum_{v_j \in N_i} f_{ji}^k, \tag{3}$$

ensuring equal incoming and outgoing flows.

Capacity Constraint:

For reasonable non-negative flow computations, the estimated link capacities represent the upper bounds for feasible f_{ij}^k values, thus we have

$$0 \leq f_{ij}^k \leq c_{ij}^k, \quad \forall e_{ij}^k \in E. \tag{4}$$

MAC Contention Constraint:

In IEEE 802.11-based ad hoc multi-hop networks, wireless medium is shared and contended by all links within the interference range of each other. Such contention behavior can be modeled as

$$\sum_{e_{pq}^k \in I_{ij}^k} \frac{f_{pq}^k}{c_{pq}^k} \leq 1, \forall v_i, v_j \in V^m. \tag{5}$$

After evaluating each gateway candidate based on the above optimization procedures, requesting node v_r selects the gateway with the maximum λ value computed by the linear programming model. The client connected to the selected gateway should then be notified to serve as downloading proxy for v_r, and calculated flow distributions should be provided to corresponding relaying nodes (clients).

3.3 Packet Forwarding Strategy

Once the proxy (gateway) client is determined, a requesting client sends out the **Gateway Request (GREQ)** packet to notify the proxy. The GREQ packet should contain computed λ value, so that the gateway downlink capacity can be refreshed to reflect the occupied bandwidth by the current requesting client. In addition, all corresponding relaying nodes are expected to receive the **Forward Request (FREQ)** packets for initiating the relaying process. Each sent FREQ packet should include computed flow distributions to facilitate routing performed by a relaying node. Like the refreshed gateway downlink capacity, all involved link capacities need be updated to reflect the current bandwidth occupancy. Both the GREQ and FREQ are unicast packets (acknowledgement mechanism used) to provide transmission reliability.

Based on the optimized flow computation results, multi-path packet delivery routes are possibly obtained to leverage channel diversity brought by multiple interfaces. The benefits and feasibility of exercising multi-path packet forwarding have been discussed in previous works [14], [19]. In this paper, we also implement the multi-path packet forwarding by distributing traffic flows according to the calculations obtained from our linear formulations.

3.4 SNG Routing Protocol Summary

Algorithm 1 provides the pseudo-code for our proposed SNG routing algorithm. In a nutshell, whenever there is a downloading request ($\mathrm{DLRequest} == true$) issued, the SNG routing daemon computes an optimized route based on discovered neighborhood information, which includes network configurations, available gateways, attainable downlink capacities from the IP network, and estimated ad hoc link rates. The ultimate goal is to maximize the perceived downloading data rate for a requesting client that participates in the synergized framework (SF).

Algorithm 1 SNG Routing Algorithm

```
while (!exit) do
    Periodically estimate neighboring link capacities;
    Periodically exchange and maintain SNG C-Table;
    If DLRequest == true then
        Construct the network graph G = (V, E);
        For each gateway candidate v_g ∈ V^g do
            Solve LP (input: c_ij^k, I_ij^k, K, V^g, V^m, E);
            output: f_ij^k, λ;
        end for
        Send GREQ to selected gateway node with the
        max λ;
        Send FREQ to all relaying nodes;
        Refresh all link & gateway capacities;
        Perform downloading;
    end if
end while
```

4 RSNG Routing Protocol

Due to the proactive routing nature, two salient drawbacks can be observed in the previously proposed SNG protocol, when it comes to dynamic networking environments. Firstly, the entries kept in a node's C-Table may easily become outdated due to mobility or unstable wireless link conditions. Unless the C-Tables are exchanged frequently enough (at the cost of increased communication overhead), data packets are likely to travel over bad routes as a result of stale C-Table entries. If this happens a lot, the optimized performance expected by SNG will be compromised. Secondly, C-Table exchanges are performed constantly between nodes, even if there are no traffic needs. This means that the network has a constant background overhead, which occupies a certain amount of available system bandwidth but may contribute little to actual throughput improvement. In light of the above observations, we propose a reactive version of SNG, entitled RSNG (Reactive SyNerGized) routing protocol, in this section.

The RSNG is a reactive routing protocol, which acquires network information in an on-demand manner. Like SNG, nodes also maintain **C-Tables**, but those tables are *kept locally without being exchanged*. On packet arrivals, RREQs (Route REQuests) are generated by the requesting node (denoted as v_r) to search for good Internet downloading routes. A node constructs its own C-Table from received RREQs. In other words, an RREQ packet starts from the requesting node v_r and passes on wireless link information (capacities) along the route it travels. Consequently, a node that receives the RREQ learns relative route information, which *gradually grows into a partial network graph when several RREQs from different routes are gathered and compiled into the local C-Table*.

We elaborate the RSNG route discovery process in Sections 4.1 and 4.2. Detailed control packet formats and

RSNG protocol summary are provided in Sections 4.3 and 4.4, respectively.

4.1 Route Discovery

The C-Table used here is similar to that of the SNG protocol, but differs in that the gateway information (**Gateway Bit** and **Gateway Capacity**) is no longer needed by the RSNG C-Table. Specifically, an RSNG C-Table maintained by a node contains wireless ad hoc link capacities collected from received RREQs. For requesting node v_r, a node, say v_i, receiving the RREQ calculates the best route and corresponding traffic flow distributions based on the optimization technique (presented in Section 3.2) with local C-Table as the input assuming v_i as the only gateway having unlimited downlink rate. In this manner, v_i obtains the *achievable downloading throughput over the ad hoc multi-hop network domain*, termed $\lambda_{ah}(v_r)$, when itself acts as the proxy (gateway) for requesting node v_r. Define $\lambda_{gw}(v_i)$ as the actual Internet downlink rate supported by node v_i (gateway throughput of v_i). Consequently, the attainable (expected) downloading throughput for v_r through proxy v_i can be obtained by $\min\{\lambda_{ah}(v_r), \lambda_{gw}(v_i)\}$. In the RREQ packet generated by v_r, the potential proxy field (denoted as v_p) is initiated as v_r with expected downloading throughput λ_{exp} set to $\lambda_{gw}(v_r)$. If node v_i discovers that a higher expected downloading throughput can be achieved through itself, the v_p and λ_{exp} fields will be updated before rebroadcasting the RREQ packet. As RREQ propagates, better routes are potentially to be discovered by the requesting node v_r.

Below we describe the handling schemes for a received new RREQ and duplicate RREQ, separately.

4.1.1 New RREQ

Whenever a node receives a new RREQ, it evaluates the possibility of providing a better Internet connection for the requesting node. During this RREQ propagation process, an essential question is *how far an RREQ should travel, so that the best proxy node can be discovered with reasonable amount of controlling overhead*. Since no destination address can be specified in the RREQ packet, we certainly do not want RREQ to be flooded throughout the entire ad hoc network domain unnecessarily. By investigating the problem of when it will become unnecessary for the RREQ to continue propagating, we exercise a trick here, which leverages the following property: *In a multi-hop ad hoc network, the end-to-end throughput for a route over possibly multiple channels decreases monotonically or stays the same, as one more wireless hop is introduced.* In other words, the end-to-end throughput improvement in the ad hoc network domain will be non-positive (zero or negative) as one more hop is added into the existing route. Now that the downloading throughput may be bounded by either the downlink rate λ_{gw} or ad hoc throughput λ_{ah}, we adopt the handling principle to *propagate RREQ in search of better* λ_{gw} *until the downloading throughput is bounded by* λ_{ah} *over the ad hoc domain*.

Two cases that terminate RREQ propagation should be discussed respectively.

- Termination Condition I: When the obtained $\lambda_{ah}(v_r)$ is no greater than the currently expected downloading throughput λ_{exp} brought by the RREQ packet, this indicates that v_p is already the best proxy over this route. In this case, node v_i sends an RREP to notify the requesting node v_r of relative route information.
- Termination Condition II: When Termination Condition I does not hold, and the calculated $\lambda_{ah}(v_r)$ is no greater than the actual supported downlink rate $\lambda_{gw}(v_i)$, this implies that the downloading throughput is now bounded by the ad hoc throughput. In this case, node v_i realizes it is the best proxy over this route, and sends an RREP to v_r with modified route information (indicating itself as the proxy node).

When neither Termination Condition I nor II hold (that is, $\lambda_{ah}(v_r) > \lambda_{exp}$ and $\lambda_{ah}(v_r) > \lambda_{gw}(v_i)$), node v_i checks if itself can act as a better proxy, does necessary modifications to the RREQ packet and then rebroadcasts it. Following the above principles, we judiciously restrict the RREQ propagation within a reasonable neighborhood region. However, in some occasions, an RREQ packet may stop propagating too early due to an incomplete C-Table kept by the receiving node. To solve this, we introduce a hop count parameter h_{min} to enforce the RREQ be propagated for *at least* h_{min} hops. In this way, nodes are able to collect reasonably sufficient links information into their C-Tables. Consequently, new RREQs may be generated on behalf of the requesting node if better routes are discovered from better-informed C-Tables (described in the next paragraph). Note that h_{min} indicates the least number of hops an RREQ needs to travel, thus it is also possible for an RREQ to go more than h_{min} hops if RSNG determines it is still beneficial to propagate this RREQ (that is, neither Termination Condition I nor II hold).

4.1.2 Duplicate RREQ

When a node receives an RREQ that has been processed before (by checking the requesting node address and associated sequence number), it updates the local C-Table and assesses whether a better route can be obtained, instead of directly discarding this RREQ packet. In case a better route exists, node v_i generates a new RREQ (with increased sequence number) containing relative route information, and then issues the new RREQ on behalf of requesting node v_r.

4.2 RSNG Route Discovery Example

Fig. 3(a) illustrates a route discovery example, where node A is the requesting node that issues an RREQ packet with h_{min} set to 1. In this example, since nodes B and D have no Internet connection, the potential proxy v_p remains node A itself with expected downloading throughput $\lambda_{exp} = 0$ and the RREQ continues propagating because neither termination conditions occur. When node G receives the RREQ from node D, it calculates $\lambda_{ah}(A)$ discovering that $\lambda_{ah}(A) > \lambda_{exp}$ and $\lambda_{ah}(A) < \lambda_{gw}(G)$ (Termination Condition II). Therefore node G updates the potential proxy v_p to itself with $\lambda_{exp} = \lambda_{ah}(A) = 0.5$, computes corresponding flow distributions over the route G $\rightarrow$ D $\rightarrow$ A, and then sends an RREP to notify the requesting node A. Likewise, when node C receives the RREQ from node B, it discovers itself as a better potential proxy. However, unlike the case of node G, neither termination conditions are satisfied at node C. This implies that downloading throughput is actually bounded by the Internet downlink rate supported by node C, thus it is still beneficial to pass on the RREQ. Consequently, node C rebroadcasts the RREQ with v_p set to itself and $\lambda_{exp} = \lambda_{gw}(C) = 1.0$ (expected downloading throughput via node C), which is received by

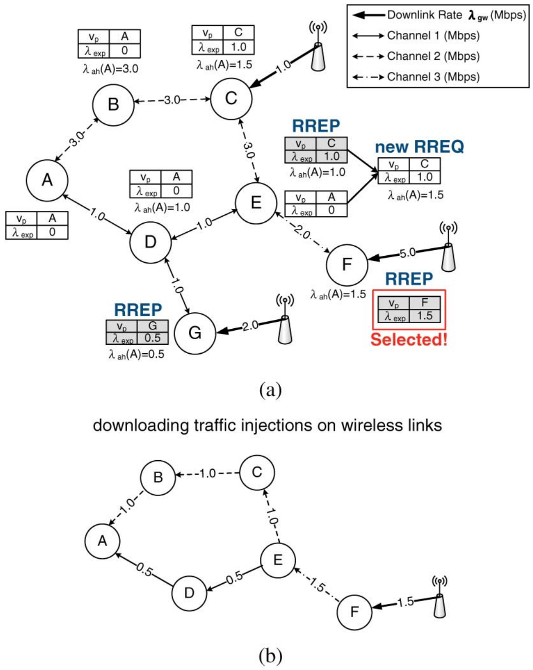


Fig. 3. A working example of the proposed RSNG routing protocol, illustrating the processes of (a) route discovery and (b) performing data downloading (with expected downloading speed at 1.5 Mbps).

node E. Since node E has no Internet connection, the potential proxy remains node C. Meanwhile, node E discovers that the calculated $\lambda_{ah}(A)$ is no greater than λ_{exp} (Termination Condition I), hence computes corresponding flow distributions over the route $C \rightarrow B \rightarrow A$ and sends an RREP back to node A. Later node E also receives a duplicate RREQ from node D, which updates its local C-Table with expanded network information. In this case, node E recomputes $\lambda_{ah}(A)$ based on the updated C-Table, and discovers a better ad hoc route, which turns out to be multi-path, combining routes $E \rightarrow C \rightarrow B \rightarrow A$ (over Channel 2) and $E \rightarrow D \rightarrow A$ (over Channel 1). Therefore node E generates a new RREQ (with increased sequence number) that contains relative link capacities along the discovered multi-path route. In the new RREQ, the potential proxy remains node C with $\lambda_{exp} = 1.0$ because node E has no Internet connection ($\lambda_{gw}(E) = 0$). Finally, node F receives the new RREQ, which need not be further rebroadcast because $\lambda_{ah}(A) > \lambda_{exp}$ and $\lambda_{ah}(A) < \lambda_{gw}(F)$ (Termination Condition II) is satisfied. An RREP indicating v_p set to node F and $\lambda_{exp} = \lambda_{ah}(A) = 1.5$ is then sent back to node A. Collectively, the requesting node A receives three RREPs replied by nodes G, E, and F, from which node A determines a best route with the highest λ_{exp}. As shown in Fig. 2, we successfully obtain a multi-path downloading route (discovered by node F), containing $F \rightarrow E$ (over Channel 3), $E \rightarrow C \rightarrow B \rightarrow A$ (over Channel 2), and $E \rightarrow D \rightarrow A$ (over Channel 1).

We wrap up the RSNG route discovery process by providing the RREQ forwarding pseudo-code in Algorithm 3. In the algorithm, $\lambda_{exp}^{best}(v_r)$, $v_p^{best}(v_r)$, and $\lambda_{ah}^{best}(v_r)$ are locally kept parameters utilized to trace the best λ_{exp}, v_p, and λ_{ah} for some requesting node v_r that a node has obtained so far. Parameters $v_p^{best}(v_r)$ and $\lambda_{exp}^{best}(v_r)$ record the best potential proxy with corresponding best expected downloading throughput among all processed RREQs. The parameter $\lambda_{ah}^{best}(v_r)$ is used by a node to locally keep track of the best λ_{ah} it has ever obtained, such that a node can decide whether to generate a new RREQ or not (refer to lines 41-49 of Algorithm 2). The above three parameters are refreshed mainly due to the expanded network knowledge perceived by the local C-Table kept by a node.

Algorithm 2 RSNG RREQ Forwarding Mechanism

1: Initially $\lambda_{exp}^{best}(v_i) = \lambda_{ah}^{best}(v_i) = \lambda_{ah}(v_i) = 0, \forall v_i \in V^m$;
2: Set λ_{gw} as the actual Internet downlink capacity supported by myself;
3: On receiving an RREQ originated from some requesting node v_r;
4: **if** ($\lambda_{exp} > \lambda_{exp}^{best}(v_r)$) **then**
5:     Set $\lambda_{exp}^{best}(v_r) = \lambda_{exp}$ and $v_p^{best}(v_r) = v_p$;
6: **end if**
7: Locally update RSNG C-Table;
8: **if** (C-Table Chgs == $true$) **then**
9:     Solve LP (input: C-Table with myself as the only gateway having unlimited downlink capacity);
10:     output: $\lambda_{ah}(v_r)$ and relative link flow distributions;
11: **end if**
12: // by checking v_r address and sequence number
13: **if** (new RREQ) **then**
14:     $h = h - 1$; // initially $h = h_{min}$, a parameter determined by the requesting node v_r
15:     **if** ($h > 0$) **then**
16:         **if** ($\lambda_{exp} < \min\{\lambda_{ah}(v_r), \lambda_{gw}\}$) **then**
17:             Set $\lambda_{exp} = \lambda_{exp}^{best}(v_r) = \min\{\lambda_{ah}(v_r), \lambda_{gw}\}$;
18:             Set potential proxy v_p and $v_p^{best}(v_r)$ to myself;
19:         **end if**
20:         Rebroadcast RREQ;
21:     **else**
22:         **if** ($\lambda_{exp} \geq \lambda_{ah}(v_r)$) **then**
23:             Send RREP to v_r with current v_p as the best proxy;
24:         **else if** ($\lambda_{gw} \geq \lambda_{ah}(v_r)$) **then**
25:             Set $\lambda_{exp} = \lambda_{exp}^{best}(v_r) = \lambda_{ah}(v_r)$;
26:             Set potential proxy v_p and $v_p^{best}(v_r)$ to myself;
27:             Send RREP to v_r with myself as the best proxy;
28:         **else**
29:             **if** ($\lambda_{exp} < \lambda_{gw}$) **then**

Set $\lambda_{exp} = \lambda_{exp}^{best}(v_r) = \lambda_{gw}$;
Set potential proxy v_p and $v_p^{best}(v_r)$ to myself;
end if
Rebroadcast RREQ;
end if
end if
if $(\lambda_{ah}(v_r) > \lambda_{ah}^{best}(v_r))$ **then**
Set $\lambda_{ah}^{best}(v_r) = \lambda_{ah}(v_r)$;
end if
else
// generate a new RREQ on behalf of v_r if better route exists
if $(\lambda_{ah}(v_r) > \lambda_{ah}^{best}(v_r))$ **then**
Set $\lambda_{ah}^{best}(v_r) = \lambda_{ah}(v_r)$;
if $(\lambda_{exp}^{best}(v_r) < \min\{\lambda_{ah}(v_r), \lambda_{gw}\})$ **then**
Set $\lambda_{exp}^{best}(v_r) = \min\{\lambda_{ah}(v_r), \lambda_{gw}\}$ and $v_p^{best}(v_r)$ to myself;
end if
Generate and issue a new RREQ with increased sequence number, h set to h_{min}, $\lambda_{exp} = \lambda_{exp}^{best}(v_r)$, v_p set to $v_p^{best}(v_r)$, and related route information;
else
Simply drop the RREQ packet;
end if
end if

4.3 Routing Control Packets

In this section, we describe the control packet formats used by our SNG and RSNG routing protocols. These control packets are implemented in our routing protocol simulations. In Fig. 4, a reserved value (binary 11) of the Type field in IEEE 802.11 MAC Header (Frame Control) is utilized as a handler for our routing daemon. The Subtype field further indicates the kind of routing control packets. In the current design, we define six subtypes of routing control packets, as shown in Fig. 4. The first three subtypes, namely Probing Packet, GREQ, and FREQ, are used by both the SNG and RSNG protocols. The fourth subtype of routing control is the C-Table Exchange specifically designed for our SNG protocol, while the last two subtypes (RREQ and RREP) are used by the RSNG protocol for route discovery.

Fig. 5 shows the detailed formats of RSNG RREQ and RREP packets. For some requesting node v_r, an RREQ packet is generated that initiates the Remaining Hops $h = h_{min}$, sets the Potential Proxy v_p to itself with Expected Downloading Throughput $\lambda_{exp} = \lambda_{gw}(v_r)$ (the actual Internet downlink rate supported by itself). As RREQ propagates, relative nodes and links information need be carried on. In the beginning, the node list only contains the requesting node v_r and there is no link information. Along the route as RREQ travels, receiving nodes append themselves and add link capacities they have learned into the RREQ packet. For example, in Fig. 3(a), when node B receives RREQ from node A over Channel 2, node B appends its address and adds the estimated capacities of links A $\rightarrow$ B and B $\rightarrow$ A into the RREQ. Similarly, node C provides its address along with estimated capacities of links B $\rightarrow$ C and C $\rightarrow$ B, and then passes on the RREQ. Since the number of nodes and the number of links contained in an RREQ packet are not necessarily equal, node list and link list are maintained separately, as shown in Fig. 5(a). Each link is represented by a transmitting node (Tx) and a receiving node (Rx) over certain channel. In order to save space, instead of using the 32-bit address for Tx and Rx, we use a 4-bit address indexed in the node list. This also explains why the Potential Proxy v_p is a 4-bit address. Such representation system can be easily extended to include multi-path route information. For instance, in Fig. 3(a), the new RREQ that node E generates actually contains a node list of {A, B, C, D, E} and all link capacities along the multi-path route. Based on the links information brought by the new RREQ, node F is able to compute the best route and corresponding flow distributions using the optimization technique modeled in Section 3.2. Consequently, an RREP packet is generated, with format

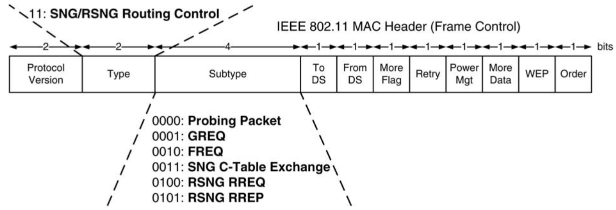
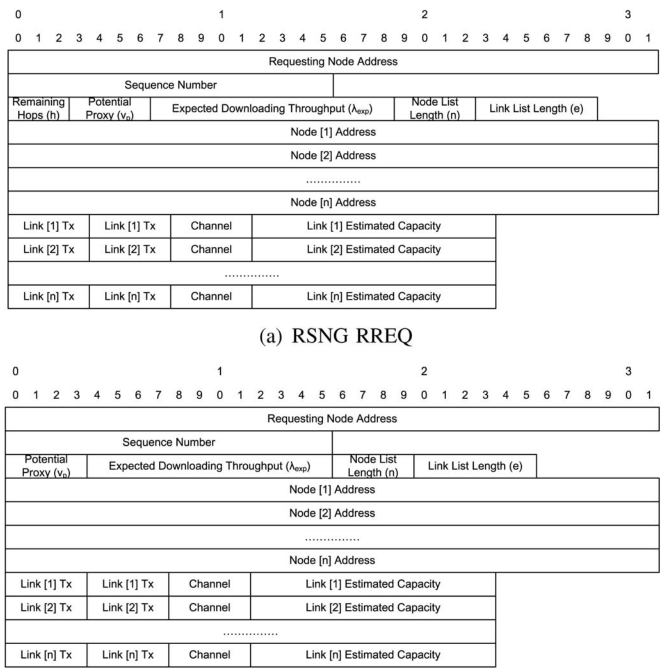


Fig. 4. Summary of routing control packets used by our SNG and RSNG protocols.



(a) RSNG RREQ

(b) RSNG RREP

Fig. 5. The detailed formats of (a) RREQ and (b) RREP packets used by the RSNG routing protocol.

shown in Fig. 5(b), containing all relaying nodes and flow distributions over corresponding links ($\mathrm{F} \rightarrow \mathrm{E}$, $\mathrm{E} \rightarrow \mathrm{C}$, $\mathrm{E} \rightarrow \mathrm{D}$, $\mathrm{C} \rightarrow \mathrm{B}$, $\mathrm{B} \rightarrow \mathrm{A}$, $\mathrm{D} \rightarrow \mathrm{A}$).

Note that, since RREQ does not carry Internet downlink rates (λ_{gw}), a node that prepares the RREP simply assumes λ_{exp} as the downlink rate supported by the potential proxy v_p. After all, λ_{exp} indicates a lower bound for $\lambda_{gw}(v_p)$, and using a value greater than λ_{exp} for $\lambda_{gw}(v_p)$ will not affect the results of our linear optimization. Specifically, a node that generates RREP solves the linear calculation using its local C-Table with the only gateway node v_p having downlink rate $\lambda_{gw}(v_p) = \lambda_{exp}$ as the input. Take Fig. 3(a) for example, when node E, unaware of $\lambda_{gw}(C)$, prepares the RREP, it sets $\lambda_{gw}(C) = \lambda_{exp} = 1.0$ (which happens to be the exact downlink rate that node C supports) and obtains a downloading route $\mathrm{C} \rightarrow \mathrm{B} \rightarrow \mathrm{A}$ with corresponding flow distributions.

4.4 RSNG Routing Protocol Summary

Different from the previously proposed SNG protocol, the RSNG protocol does not perform C-Table exchanges. Instead, the RSNG C-Tables are maintained locally, and expanded gradually from received RREQ packets that travel through various routes. Whenever there is a downloading request, a node issues an RREQ packet and collects multiple RREPs, from which a best route with the maximum λ_{exp} value is selected for realizing the downloading. We provide the RSNG routing pseudo-code in Algorithm 3.

Algorithm 3 RSNG Routing Algorithm

```
while (!exit) do
    Periodically estimate neighboring link capacities;
    Locally maintain RSNG C-Table from received
    RREQ packets;
    if DLRequest == true then
        Issue an RREQ to discover multiple routes;
        Select the best route with maximum λ_exp from
        received RREP packets;
        Send GREQ to notify the proxy node;
        Send FREQ to all relaying nodes;
        Perform downloading;
    end if
end while
```

5 Performance Evaluation

To validate the performance of proposed routing mechanisms, we implement our SNG and RSNG protocols in the ns-2 simulator.[2] In the ad hoc domain, IEEE 802.11 MAC protocol with RTS/CTS four-way handshaking is used, and 3 IEEE 802.11b non-overlapping channels are simulated. Default transmit power and Two-Ray Ground propagation model are adopted, leading to 250 m transmission distance and 550 m interference range. For each client running the routing optimization, the *lpsolve* tool is utilized for the computation [2]. As presented in Section 3.1, SNG does not intend to collect a global network information. Instead, a TTL field is used to limit how far the C-Table can propagate. In the simulations, we use the parameter h for SNG to indicate how many hops the C-Table can travel. For RSNG, nodes construct the C-Table from received RREQ messages. The RSNG C-Table is kept locally without being exchanged over the air. On packet arrival, RSNG initiates route discovery by propagating RREQ packets. We use the parameter h_{min} in the simulations to make sure a certain scope of neighborhood (for at least h_{min} hops) is searched through by the RREQ packets. RSNG at the requesting node collects three RREP packets or await a 2-second timeout to expire before deciding on the best downloading route. Two other routing mechanisms, Greedy (introduced in [21]) and DSR+ (adapted from [8]), are also implemented for comparison purpose. The Greedy protocol searches for proxy client with better downlink quality, starting from the immediate neighbors, in a greedy manner until no better neighbor can be found. DSR+ is an extended version of DSR protocol, which inherits the shortest path metric of DSR routing logic. It basically selects the proxy with the highest downlink capacity within h-hop neighborhood. In case multiple proxies with the highest capacity exist, DSR+ favors the proxy with the shortest route (having minimum hop count). In case multiple routes with the same minimum hop count to proxies with the same highest capacity are available, then DSR+ chooses the proxy and corresponding route randomly. Routes discovered by RSNG, DSR+, and Greedy become invalid when a 1-second timeout expires. For the above four protocols, related probing mechanism, table exchanges, and routing overhead are implemented in order to reflect the net downloading throughput. Table 3 summarizes the implemented overhead involved with respective routing protocol.

TABLE 3
Summary of Overhead Incurred by Respective Routing Strategy

Routing Mechanism	Involved Overhead
RSNG	Probing Packet, RREQ, RREP, FREQ, GREQ
SNG	Probing Packet, C-Table Exchange, FREQ, GREQ
DSR+	RREQ, RREP, GREQ
Greedy	RREQ, RREP, GREQ

5.1 Impact of Network Information Scope

In this section, we investigate the impact of parameter h, and compare different routing strategies with respect to obtained downloading throughput. Fig. 6 shows the simulated network environment. Colored nodes represent potential gateways, which are classified into five levels in terms of downlink capacities. Channel (interface) configurations and estimated link rates are also illustrated in Fig. 6. In order not to further complicate the network environment, symmetric links (with equal link rate in both directions) are modeled in the simulations (note that this simplification does not affect the performance justification). We generate a mobile client A, which has two IEEE 802.11 interfaces operating on Channel 1 and 2 respectively and one cellular interface connecting to the IP

2. ns-2 version 2.29 with multi-channel multi-interface extension has been used for conducting our experiments in this section. This version has fixed unrealistic channel propagation model and IEEE 802.11 bugs at the MAC layer. The well-known problem of ignored accumulated interference power calculation has also been patched in the new version.

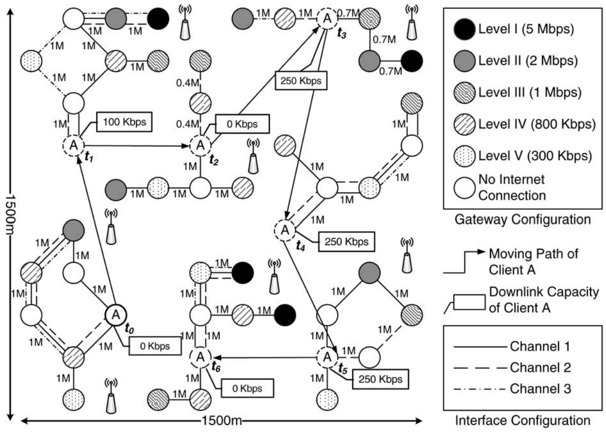


Fig. 6. Simulation environment with 41 nodes deployed in a $1500\ m \times 1500\ m$ network topology, along with corresponding Internet connection status, interface (channel) configurations, and ad hoc wireless link rates.

network, to observe the attainable downloading throughputs at different time snapshots (t_0-t_6). As client A roams across the network, its cellular channel qualities vary depending on its locations. We implement five approaches for client A to obtain downlink services: Without Relay (using its own cellular connections), Greedy, DSR+, SNG, and RSNG.

Fig. 7 displays the obtained throughput and overhead ratio produced by respective routing method at different time snapshot. The overhead ratio, defined as consumed routing overhead bits per successfully transmitted bit, is used to quantify the overhead percentage that accompanies effective throughput. In this relatively static environment, overhead ratio incurred by SNG protocol stays very low (around 0.01) due to infrequent C-Table exchange traffic, whereas RSNG requires a bit higher overhead cost that is still nicely kept below 0.08. The throughput result shows that our SNG and RSNG outperform other strategies, while Without Relay provides the lowest downloading throughput due to no cooperation with other clients. The Ideal value at each time snapshot indicates the theoretically expected throughput attainable under perfect transmission scheduling and without routing overhead, which is used as a reference upper bound. We also experiment on SNG ($h = 4$, $h = 5$) and RSNG ($h_{min} = 4$, $h_{min} = 5$), but the throughput improvement is insignificant, thus omitted from the figure. The results indicate that SNG and RSNG only need a moderate amount of neighborhood knowledge to outperform other strategies. In order to have a better understanding of how each routing mechanism determines the best route, we compile the downloading paths selected by respective protocol at each time snapshot (t_0-t_6) in Fig. 8. Interestingly, the best route (leading to the highest downloading throughput) does not always involve the best gateway (t_0, t_2, t_3), or the shortest hop distance (t_1, t_4, t_5, t_6). Moreover, at t_2, Greedy gives lower throughput with shorter route (2-hop) than the DSR+ strategy with longer route (3-hop), due to weak link quality over Ch2 (0.4 Mbps) used by Greedy. This phenomenon also reveals that the hop distance factor alone cannot act as the single

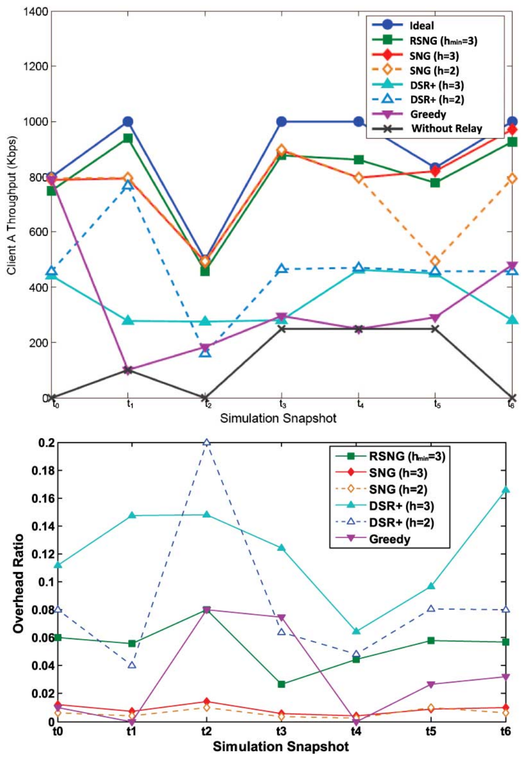


Fig. 7. Throughput performance obtained (left) and routing overhead incurred (right) by using different routing strategies as client A roams across the simulated network.

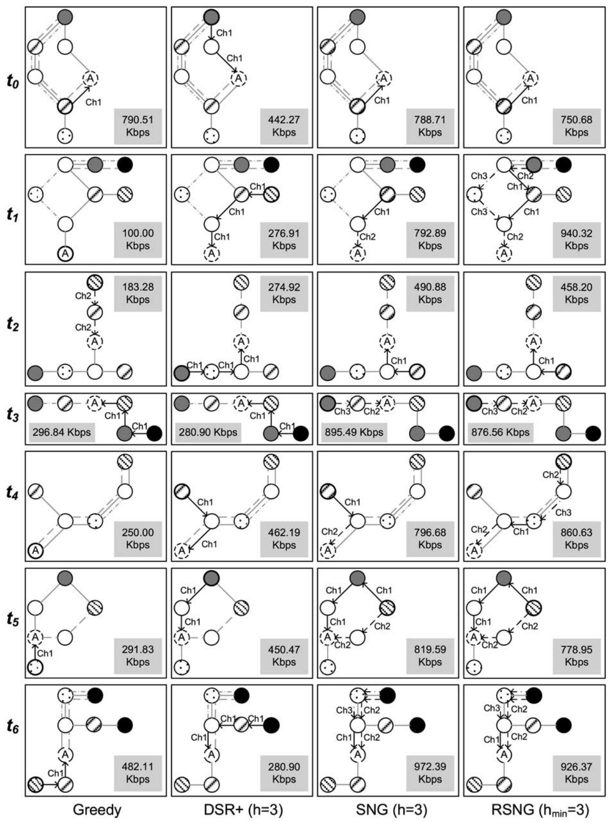


Fig. 8. Downloading flow paths selected by different routing strategies at respective time snapshot. Note that selecting Internet proxy (gateway) with the highest downlink rate does not necessarily yield the best throughput for client A, since the communication bottleneck may exist in the ad hoc network domain.

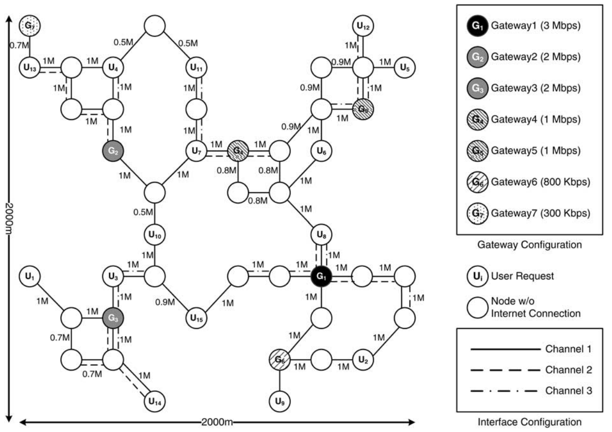


Fig. 9. Network configuration with 15 clients and 7 potential proxies (gateways) placed in a $1500\ m \times 1500\ m$ topology.

metric for a good route. In addition, downloading throughput can be further increased by enabling multi-path packet delivery (SNG at t_5, t_6 and RSNG at t_1, t_5, t_6). Note that at t_1, the best gateway actually exists at 4 hops away, which is beyond the reach of SNG with h set to 3. In contrast, RSNG is capable of discovering the best gateway because its RREQ can travel more than 3 hops as far as the throughput bottleneck of the ad hoc domain has not been reached. Similar phenomenon occurs at t_4, when the best gateway is located 4 hops away. Consequently, at t_1 and t_4, our RSNG performs better than SNG due to better gateways, thus better downloading routes, discovered. For other snapshots (t_0, t_2-t_3, t_5-t_6), since the best gateway exists within 3 hops, SNG and RSNG both discover the same downloading routes. However, our SNG performs slightly better than RSNG due to less routing overhead entailed by SNG in a relatively static environment. From the above observations, we conclude that a good routing protocol in such a hybrid network should take various factors into a unified consideration, thus validating the SNG and RSNG design philosophy.

5.2 Importance of System Load Balancing Among Users

Another essential problem for the multi-hop network is the system capacity distribution. We investigate this issue by observing the aggregate throughput produced by respective routing technique. Fig. 9 shows the simulated network, with seven potential gateways, and up to 15 user requests made in order. The aggregate network throughput (along with corresponding overhead cost) under different routing strategies is plotted in Fig. 10. Both SNG and RSNG protocols are able to yield significantly higher aggregate downloading throughput than the other two routing mechanisms. Since SNG and RSNG refresh gateway capacities and ad hoc link rates to reflect current bandwidth occupied by existing users, the routing process has better knowledge to distribute the downloading requests adequately. Overall, SNG performs slightly better than RSNG due to less routing overhead imposed on SNG under this static simulation environment. In addition, we analyze the gateway utilization status, shown in Fig. 11, under different routing strategies. With the same user requesting patterns (U_1 request, followed by U_2 request, followed by U_3 request, and so on), we observe that G_3 is under-utilized in

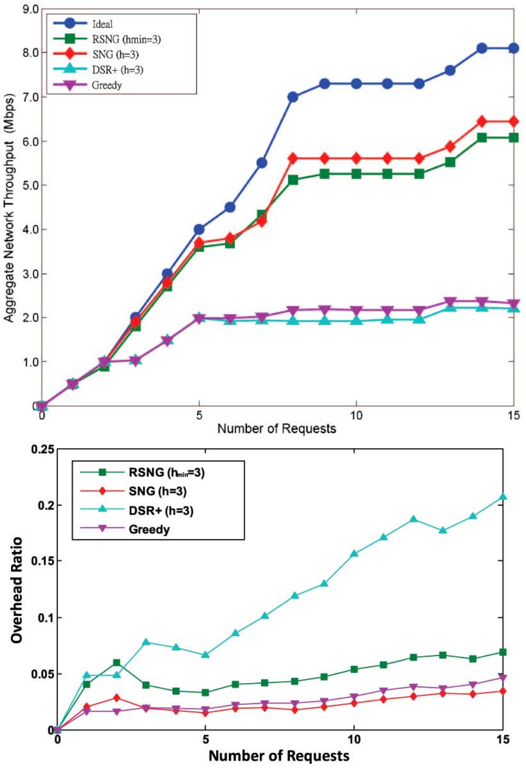


Fig. 10. Aggregate network throughput increases as user demands grow, and saturates at varying points under different routing strategies.

Greedy and DSR+. In fact, G_3 is selected by several users, such as U_1, U_3, and U_{14} as downlink gateway. However, because of ad hoc channel contentions, U_3 and U_{14} seldom get the chance to use G_3 capacity. On the flip side, SNG and RSNG avoid such adverse effect by distributing traffic to other non-interfering channels, leading to high G_3 utilization. We further analyze the distributions of per-user traffic flows under different routing mechanisms in Fig. 12, which illustrates the individual flow occupancy. Though similar numbers of user (10-11 users) are supported by the four approaches, our SNG and RSNG yield significantly higher average per-user throughput than the other two strategies

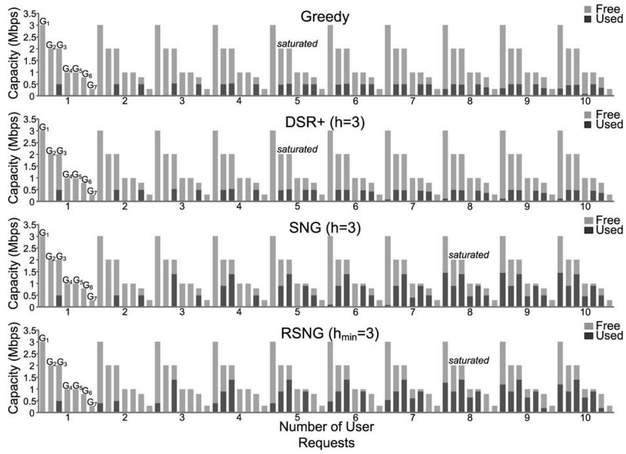


Fig. 11. Proxy (gateway) capacity utilization status under Greedy, DSR+, SNG, and RSNG routing strategies, respectively.

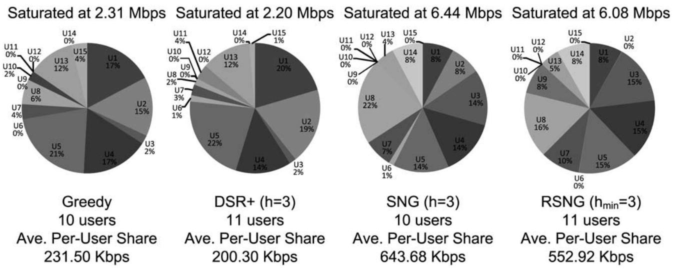


Fig. 12. Per-user traffic flow occupancy among saturated throughput obtained by respective routing strategy.

due to better balanced bandwidth allocations. As we can see from the figure, U_1, U_3 and U_{14} get to enjoy G_3 capacity with reasonable throughput share when SNG and RSNG are used, whereas U_3 and U_{14} occupy little or none of G_3 capacity when Greedy and DSR+ mechanisms are exercised. By wisely selecting proxies (gateways) and generating (single- or multi-path) routes around congested spots in the ad hoc network domain, SNG and RSNG effectively expand the aggregate downloading throughput. Two important design principles are revealed from the above experiments. First, both gateway and ad hoc link capacities should be refreshed to reflect up-to-date bandwidth allocation. Second, channel diversity should be leveraged to enable concurrent transmissions and further distribute the traffic.

5.3 Influence of Mobility and Inaccurate Link Capacity Estimate

Finally, we extend the environment in Fig. 6 to include 10 mobile clients such that the network dynamics in a realistic setting can be simulated, as depicted in Fig. 13. The interface configuration status for each mobile client is also shown in this figure, where NIC indicates the channel binding for respective IEEE 802.11 Network Interface Card of a mobile client. For instance, client A is equipped with two IEEE 802.11 interface cards binding to Channel 1 and 2 separately, while client B has only one IEEE 802.11 radio interface binding to Channel 1. In addition to IEEE 802.11 interfaces, all mobile clients have their

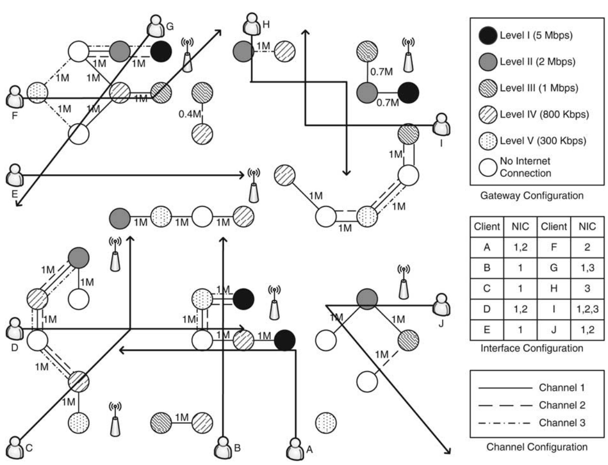


Fig. 13. Simulation environment with 41 static nodes deployed in a $1500\ m \times 1500\ m$ network topology, where 10 mobile clients (client A-J) roam around following pre-configured moving paths (there are 11 stops equally distributed along each path).

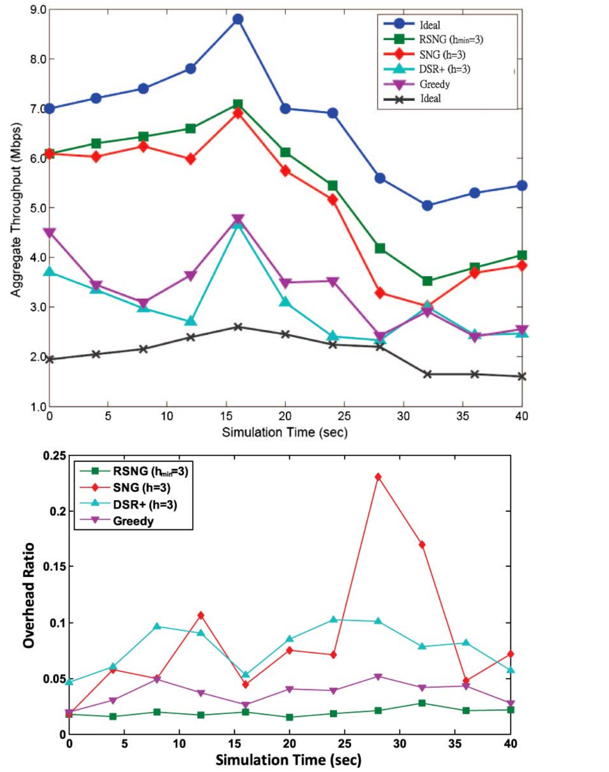


Fig. 14. Aggregate network throughput (left) and consumed routing overhead ratio (right) over time as 10 mobile clients roam across the simulated environment.

own cellular Internet connections, with downlink qualities inversely proportional to their distances away from the base stations. Based on 10 randomly generated moving paths, all mobile clients roam in this environment performing downloading during the simulated 40-second period. Under such a dynamic environment, the channel probing mechanisms adopted by SNG and RSNG are expected to produce inaccurate link capacity estimates due to varying surroundings. We conduct this set of experiments to investigate the adverse effect of mobility on the system throughput. Fig. 14 provides the simulation results on aggregate throughput and incurred overhead ratio. Although SNG and RSNG still give the best performance, the aggregate throughput improvements are not as pronounced as those in Fig. 10 (where a static environment is simulated) due to increased routing overhead and inaccurate link capacity evaluations. Nonetheless, in average, our RSNG and SNG have the potential to boost the downloading speeds for 2.6 and 2.4 times with respect to Without Relay, compared to Greedy's 1.6 and DSR+'s 1.4 times of rate increases. These results corroborate that RSNG and SNG can operate effectively even when their probing mechanisms do not function perfectly. Furthermore, as we can observe from Fig. 14, RSNG performs slightly better than SNG at all times. The reason mainly attributes to the increased C-Table exchanges executed by SNG (with the overhead ratio rising up to 0.23 at 28 sec simulation time), even though there may not necessarily be needs for some C-Table entries. In contrast, RSNG activates route discoveries only when traffic actually arrives. Collectively, SNG incurs more overhead than RSNG when mobility dominates the environment. As a result, we suggest to adopt RSNG in a relatively dynamic setting.

Based on a series of simulative experiments targeted on the multi-hop networking environment with heterogeneous Internet gateways, among the four implemented routing techniques, our proposed SNG and RSNG protocols are demonstrated to be capable of offering significantly better downloading rates through their optimized routing intelligence.

6 Conclusion

In this paper, we propose a synergized framework (SF), and design routing approaches to enable the suggested communication model. Our SNG protocol determines the best route for a downloading request in an optimized manner. Simulations show that the SNG mechanism is able to outperform other routing strategies, based on moderate amount of neighborhood information. On the other hand, the RSNG routing protocol is introduced to limit the information collection within one-hop communication scope and perform route discoveries in a reactive manner. According to the simulation results, our RSNG gives the best performance in a dynamic (mobile) setting.

Acknowledgment

This research was co-sponsored in part by the NSC of Taiwan under Grant 102-2221-E-009-014, and in part by the MoE Program Aiming for the Top University and Elite Research Center Development Plan (ATU Plan).

References

[1] *iPhone: Connecting to the Internet*, 2013. [Online]. Available: http://support.apple.com/kb/ht1695.

[2] *lp-Solve: A Mixed Integer Programming (MIP) Solver*, 2013. [Online]. Available: http://sourceforge.net/projects/lpsolve/.

[3] *Must-Have Reference for Wireless Communication—Understanding Wireless Telecom and Data Communications Terminology* [Online]. Available: http://www.eu.anritsu.com/musthaveguides, Dec. 2008.

[4] *Xperia Enterprise Connectivity* [Online]. Available: http://dl-www.sonyericsson.com/cws/download/1/984/193/1304320827/Xperia_Enterprise_Connectivity.pdf, Apr. 2011.

[5] G. N. Aggelou and R. Tafazolli, "On the relaying capability of next-generation GSM cellular networks," *IEEE Pers. Commun.*, vol. 8, no. 1, pp. 40–47, Feb. 2001.

[6] F. O. Akgül and M. O. Sunay, "Enhancing 3G high speed wireless data performance through utilization of 3G—WLAN system cooperation: A cross-network, cross-layer approach," *Wireless Pers. Commun.*, vol. 43, no. 1, pp. 89–106, Oct. 2007.

[7] R. Baumann, S. Heimlicher, and B. Plattner, "Routing in large-scale wireless mesh networks using temperature fields," *IEEE Netw.*, vol. 22, no. 1, pp. 25–31, Jan./Feb. 2008.

[8] J. Broch, D. Johnson, and D. Maltz, "The dynamic source routing protocol for mobile ad hoc networks," *IETF Internet Draft*, Oct. 1999.

[9] D. Cavalcanti, D. Agrawal, C. Cordeiro, B. Xie, and A. Kumar, "Issues in integrating cellular networks, WLANs, and MANETs: A futuristic heterogeneous wireless network," *IEEE Wireless Commun.*, vol. 12, no. 3, pp. 30–41, June 2005.

[10] L.-J. Chen, T. Sun, G. Yang, M. Y. Sanadidi, and M. Gerla, "Ad Hoc probe: Path capacity probing in wireless ad hoc networks," in *Proc. IEEE Int. Conf. Wireless Internet (WICON)*, July 2005, pp. 156–163.

[11] Z. Dziong, F. Khan, K. Medepalli, and S. Nanda, "Wireless internet access using IS-2000 third generation system: A performance and capacity study," *Wireless Netw.*, vol. 8, no. 4, pp. 325–336, July 2002.

[12] H.-Y. Hsieh and R. Sivakumar, "Performance comparison of cellular and multi-hop wireless networks: A quantitative study," in *Proc. ACM Special Interest Group Meas. Eval. (SIGMETRICS)*, June 2001, pp. 113–122.

[13] H.-Y. Hsieh and R. Sivakumar, "On using the ad-hoc network model in cellular packet data networks," in *Proc. ACM Int. Symp. Mobile Ad Hoc Netw. Comput. (MobiHoc)*, June 2002, pp. 36–47.

[14] I. Ioannidis, B. Carbunar, and C. Nita-Rotaru, "High throughput routing in hybrid cellular and ad hoc networks," in *Proc. IEEE Int. Symp. World Wireless Mobile Multimedia Netw. (WoWMoM)*, 2005, pp. 171–176.

[15] K. Jang, M. Han, S. Cho, H.-K. Ryu, J. Lee, Y. Lee, and S. B. Moon, "3G and 3.5G wireless network performance measured from moving cars and high-speed trains," in *Proc. ACM Int. Workshop Mobile Internet Through Cellular Netw. (MICNET)*, Sep. 2009, pp. 19–24.

[16] S. Jung, M. Kserawi, D. Lee, and J.-K. K. Rhee, "Distributed potential field based routing and autonomous load balancing for wireless mesh networks," *IEEE Commun. Lett.*, vol. 13, no. 6, pp. 429-431, June 2009.

[17] X. Li and L. Shen, "Performance Analysis in C3G-A System," *Proc. ACM Int'l Conf. Wireless Comm. Mobile Computing (IWCMC)*, June 2009, pp. 916–920.

[18] X. J. Li and P. H. J. Chong, "Performance analysis of multihop cellular network with fixed channel assignment," *Wireless Netw.*, vol. 16, no. 2, pp. 511–526, Feb. 2010.

[19] T.-Y. Lin, W.-H. Tam, K.-L. Fan, and Y.-C. Tseng, "Resource planning and packet forwarding in multi-radio, multi-mode, multi-channel, multi-rate (M4) wireless mesh networks," *Elsevier Comput. Commun.*, vol. 31, no. 7, pp. 1329–1342, May 2008.

[20] Y.-D. Lin and Y.-C. Hsu, "Multihop cellular: A new architecture for wireless communications," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Nov. 2000, pp. 1273–1282.

[21] H. Luo, R. Ramjee, P. Sinha, L. E. Li, and S. Lu, "UCAN: A unified cellular and ad-hoc network architecture," in *Proc. ACM Int. Conf. Mobile Comput. Netw. (MobiCom)*, Sep. 2003, pp. 353–367.

[22] P. Maillé, "Allowing multi-hops in cellular networks: An economic analysis," in *Proc. ACM Annu. Int. Conf. Model. Anal. Simul. Wireless Mobile Syst. (MSWiM)*, Oct. 2005, pp. 12–19.

[23] B. S. Manoj, K. J. Kumar, C. D. Frank, and C. S. R. Murthy, "On the use of multiple hops in next generation wireless systems," *Wireless Netw.*, vol. 12, pp. 199–221, 2006.

[24] M. Tanno. (2009, Aug.). *Standardizations for LTE/LTE-Advanced and Future Broadband* [Online]. Available: http://apwcs2009.yu.ac.kr/.

[25] T. Rouse, S. McLaughlin, and H. Haas, "Coverage-Capacity analysis of opportunity driven multiple access (ODMA) in UTRA TDD," in *Proc. IEEE Int. Conf. 3G Mobile Commun. Technol.*, Mar. 2001, pp. 252–256.

[26] S. Tanaka, H. Ishii, T. Sao, U. Iizuka, and T. Nakamori, "HSDPA throughput performances using an experimental HSDPA transmission system," *NTT DoCoMo Tech. J.*, vol. 6, no. 4, pp. 19–28, Mar. 2005.

[27] H.-Y. Wei and R. D. Gitlin, "Two-hop-relay architecture for next-generation WWAN/WLAN integration," *IEEE Wireless Commun.*, vol. 11, no. 2, pp. 24–30, Apr. 2004.

[28] H. Wu, C. Qiao, S. De, and O. Tonguz, "Integrated cellular and ad hoc relaying systems: iCAR," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 10, pp. 2105–2115, Oct. 2001.

[29] X. Wu, S.-H. G. Chan, and B. Mukherjee, "MADF: A novel approach to add an ad-hoc overlay on a fixed cellular infrastructure," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, 2000, pp. 549–554.

[30] X. Wu, S.-H.G. Chan, B. Mukherjee, and B. Bhargava, "MADF: Mobile-assisted data forwarding for wireless data networks," *J. Commun. Netw.*, vol. 6, no. 3, pp. 216–225. Sep. 2004.

[31] Y. Wu, K. Yang, and J. Zhang, "An adaptive routing protocol for an integrated cellular and ad-hoc network with flexible access," in *Proc. ACM Int. Conf. Wireless Commun. Mobile Comput. (IWCMC)*, 2006, pp. 263–268.

[32] A. N. Zadeh, B. Jabbari, R. Pickholtz, and B. Vojcic, "Self-Organizing packet radio ad hoc networks with overlay (SOPRANO)," *IEEE Commun. Mag.*, vol. 40, no. 6, pp. 149–157, June 2002.

Ting-Yu Lin received the PhD degree in computer science and information engineering (CSIE) from the National Chiao Tung University, Hsinchu, Taiwan. She is currently an associate professor with the Department of Electrical and Computer Engineering, National Chiao Tung University. Her research interests include wireless communications and mobile computing. She is a member of ACM.

Tai-Yi Huang received the BS and MS degrees in communications engineering from the National Chiao Tung University, Hsinchu, Taiwan, in June 2008 and October 2010, respectively. He is currently working as a software engineer at Realtek Semiconductor Corporation, Hsinchu, Taiwan. His research interests include next-generation wireless and cooperative networks.

Chia-Fu Hsu received the BS degree in electrical engineering and MS degree in communications engineering from the National Chiao Tung University, Hsinchu, Taiwan, in June 2010 and October 2012, respectively. He is currently working as a software engineer at MediaTek Incorporation Hsinchu, Taiwan. His research interests include next-generation wireless and cooperative networks.