Adaptive Multi-Copy Routing for Intermittently Connected Mobile Ad Hoc Networks

Zhuoqun Li, Lingfen Sun and Emmanuel C. Ifeachor
School of Computing, Communications and Electronics
University of Plymouth
Drake Circus, Plymouth, PL4 8AA, United Kingdom
{Zhuoqun.Li, L.Sun, E.Ifeachor}@plymouth.ac.uk

Abstract— **Intermittently Connected Mobile Ad Hoc Networks (ICMANs) are mobile networks in which complete source-to-destination paths do not exist most of the time because of their sparse topology. Such networks enable communications among sparsely distributed moving objects, e.g. a network of ambulances. Traditional routing schemes are unsuitable in ICMANs because they only forward packets in connected paths. Recently, *store-and-forward* based routing (e.g. multi-copy relaying) have been proposed to address this. However, existing multi-copy strategies do not adapt to the frequent variations in network conditions as they rely heavily on the source to control the relaying process. In this paper, we propose an efficient *store-and-forward* based scheme, Adaptive Multi-Copy Routing (AMR), for packet delivery in ICMANs. In this scheme, instead of relying on the source, each relay node independently decides whether to forward a packet according to the delivery delay target and current network conditions. Simulation results show that the proposed scheme always adapts to the varying network mobility and/or delivery probability that cannot be anticipated in the conventional source-defined strategies and delivers packets within specific delay targets with minimum traffic overhead.**

I. INTRODUCTION

Intermittently Connected Mobile Ad-hoc Networks (ICMAN) are mobile ad-hoc networks in which fully connected source-to-destination paths do not exist most of the time because of their sparse topology. Such networks enable communications among sparsely distributed moving objects, e.g. a network of moving ambulances in the scenario of e-healthcare emergency. As traditional routing protocols can only deliver packets over connected end-to-end paths, they would fail in such networks. To perform communications within an ICMAN, a *store-and-forward* routing paradigm may be used and, over time, packets would eventually reach their destination. This means that an intermediate node may have to carry received packets for a long time before they are forwarded to the next hops because of the intermittently available links.

A number of routing strategies, based on the concept of *store-and-forward*, have recently been introduced for ICMANs. Epidemic routing [1] performs packet relaying in a way that is reminiscent the concept of flooding. In epidemic routing, a node that has packets to send or relay would store these messages in its buffers if it is not possible to forward the packets. Whenever the node meets another node, the two nodes exchange copies of packets that they do not have in common. Thus, as nodes continue to move around, the packets they are carrying would reach their destination over time. In practice, epidemic routing suffers from high usage of network bandwidth. A simple way to reduce the excessive traffic produced by epidemic routing is to only replicate packets to nodes that have higher delivery probability [2] or to those with similar mobility pattern to the destination [3]. A more aggressive way to reduce the traffic overhead is to use single-copy routing [4], i.e. each relay node forwards at most one copy per packet. Although these relaying strategies can substantially reduce the delivery overhead, they increase delivery delay and decrease the success rate of delivery. To address the delay-overhead dilemma, multi-copy relaying strategies are proposed in [5] [6] to limit the delivery overhead by only replicating exactly $\mathcal{R}$ (the *replication factor*) copies of each packet for the expected delivery delay. The $\mathcal{R}$ packet copies can be distributed through a process that can be represented as a balanced binary-tree [5] or the locally-optimal tree [6]. The EBEC scheme [7] is a more complex version of multi-copy routing. EBEC utilize erasure coding to generate $\mathcal{R} \times K$ message blocks per packet for enhanced redundancy. These message blocks are selectively distributed among nodes according to their probability of reaching the destination for less delivery delay.

A common problem in conventional multi-copy routing protocols [5] [6] [7] is that they rely heavily on the source to control the relaying process. A source determines the relaying process by setting the *replication factor* before it initiates packet relaying. However, it is normally difficult and very complicated for the source to estimate a suitable $\mathcal{R}$ based on its limited knowledge about the network. If network conditions (e.g. network mobility or delivery probability) change during the relaying process, the predefined *replication factor* would no longer be appropriate leading to a degraded performance.

In this paper, we propose an efficient *store-and-forward* based scheme, Adaptive Multi-Copy Routing (AMR), for packet delivery in ICMANs. In this scheme, instead of using a source-defined *factor*, we let each of the intermediate relay nodes independently decide whether to further replicate a packet based on the current network conditions and the end-to-end delay target. Thus, the scheme is *replication-factor-free* and is more efficient in coping with unpredictable variations in the network conditions than source-defined relaying strategies. Therefore it is more suitable for highly dynamic ICMANs. Simulation results show that the proposed scheme always

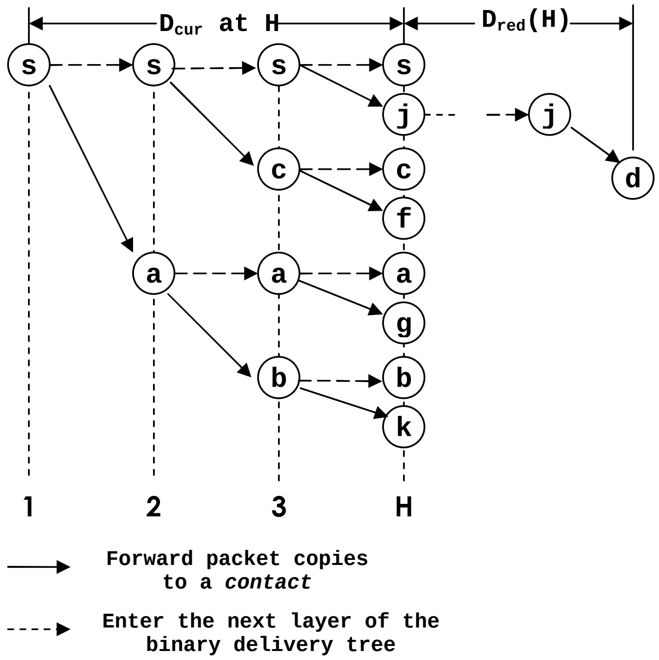


Fig. 1. The Binary Relaying Tree

adapts to the varying network mobility and/or delivery probability which cannot be anticipated in the conventional source-defined multi-copy strategies and delivers packets within specific delay targets with minimum traffic overhead.

The remainder of this paper is organized as follows: In section II we introduce the system model and the mechanisms of the proposed scheme of Adaptive Multi-copy Routing (AMR). The simulation based performance evaluation of the AMR scheme is presented in Section III. Section IV concludes this paper.

II. MULTI-COPY RELAYING WITH DISTRIBUTED ADAPTATIONS

In this section, we introduce the proposed Adaptive Multi-copy Routing (AMR) scheme. The relaying process in AMR is partially inspired by the binary spray&wait routing [5]. As shown in Fig.1, the relaying process of a packet in AMR starts from the source (node s) and finishes at the destination (node d). When s meets[1] node a, a copy of the packet is forwarded to a. Having passed a, s keeps moving on and giving a copy of the packet to each node it meets on the way. Nodes given a copy of the packet (including the source) would further replicate the copy to those that they meet on their way until either they think that they have spread enough packet copies or they meet the destination. Such a replication process of a packet can be represented as a binary tree with the source of the packet as its root. The delivery delay and overhead of a packet is determined by the depth of the binary delivery tree.

Unlike the conventional multi-copy protocols in which the source chooses a suitable depth of the delivery tree for a delay target by specifying the *replication factor* $\mathcal{R}$. The AMR scheme let the relay nodes independently decide the depth of the relaying process for each packet they are carrying. As a packet has spent some time (referred as $\mathcal{D}_{cur}$, see Fig.1) to reach a relay node, the purpose of limiting the depth of the delivery tree at a relay node is to limit the residual delivery delay (referred as $\mathcal{D}_{red}$, see Fig.1), instead of the end-to-end delivery delay, for a delay target. Such a distributed adaptation of the relaying process has two major benefits: firstly, the relay nodes have fresher knowledge of the networking conditions than the source hence their estimations of the delivery delay is more accurate. Secondly, estimating the residual delivery delay of a packet at the relay nodes is simpler than to estimate the end-to-end delivery delay at the source, i.e. $\mathcal{D}_{red}$ is independent of the number of nodes in the network. The details of the algorithms and mechanisms of the AMR scheme is given in the following subsections.

A. Distributed Estimation of the Residual Delivery Delay

Distributed estimation of the residual delivery delay $\mathcal{D}_{red}$ of a packet is the key mechanism of the AMR scheme. $\mathcal{D}_{red}$ is a function of both $\mathcal{R}$ and the mean intermeeting times[2] $\bar{\mathcal{T}}_{int}$. Recent study [5] has suggested that $\frac{\bar{\mathcal{T}}_{int}}{\mathcal{R}}$ is a good approximation of the residual delivery delay when all nodes in the network perform IID random walks. However, the estimation of the mean intermeeting times $\bar{\mathcal{T}}_{int}$ in [5] does not reflect the impact of network mobility. Groenevelt in [8] has proposed mathematic models to estimate the mean intermeeting times between nodes moving at a specific speed for both Random Waypoint model (RWP) and Random Direction Model (RD) at their steady states. Suppose that all nodes in the network are constant moving in the RWP model, the mean intermeeting times is given by [8]:

$$\bar{\mathcal{T}}_{int} = \frac{\mathcal{L}^2}{2\omega r \bar{v}} \tag{1}$$

where $\omega \approx 1.3683$ is a specific constant for the RWP model, r is the radio radius, $\bar{v}$ is the mean relative speed between nodes of the network, and $\mathcal{L}$ is the side length of the square geographical network area.

If a node does not have the knowledge of one or more parameters listed in Eq.1, it is still possible to estimate the mean of intermeeting times $\bar{\mathcal{T}}_{int}$ using the mean of the intermeeting times samples that it has collected. As the size of the network area, the radio radius and the model of the nodal movements are normally fixed, it is reasonable to assume that only the mean relative speed $\bar{v}$, which varies with time or the type of mobile devices, is unknown to a node. It is easier for a node to sample relative speeds from any *contacts* than to sample intermeeting times from those that it has met before. A single node can estimate the time-varying $\bar{v}$ by:

---

[1] by a meets b we mean $d_{ab} < r$, where d_{ab} is the distance between a and b and r is the radio propagation radius of a mobile node. We call a (b) a ***contact*** of b (a) when a meets b.

[2] Defined as the duration between the time when a node meet the other node and the next time when they meet again

$$\bar{v} \approx \frac{1}{|\mathcal{V}^{\tau}|} \sum_{\hat{v}_i \in \mathcal{V}^{\tau}} \hat{v}_i \tag{2}$$

where $\hat{v}_i$ is the relative speed sample that a node measured from a passing *contacts i* and $\mathcal{V}^{\tau}$ is the set of speed samples that a node collected during past τ seconds. τ should be designed to enable a node to collect enough samples (e.g. over 30 [9]) for an accurate estimation of $\bar{v}$ and to make the estimate emphasize on recent changes in the network mobility.

If the relaying process of a packet can be characterized by the binary delivery tree as depicted in Fig.1, a node can estimate the number of copies that have been replicated for the packet as $2^{\mathcal{H}-1}$ provided that it knows the current depth $\mathcal{H}$ of the delivery tree. Therefore, a single relay node could estimate the residual delivery delay $\mathcal{D}_{red}$ when the depth of the binary delivery tree is $\mathcal{H}$ by:

$$\mathcal{D}_{red}(\mathcal{H}) = \frac{\bar{\mathcal{T}}_{int}}{2^{\mathcal{H}-1}} \tag{3}$$

B. Estimating the Delivery Probability

The estimation of $\mathcal{D}_{red}$ in Eq.(3) is based on the assumption that a node could deliver a packet to a *contact* with a probability of 1. However, the probability of successful one-hop delivery in reality is susceptible to a variety of factors (e.g. Signal-to-Noise Ratio (SNR), contention or link duration). As packet losses due to low SNR is normally addressed in the link layer and contention rarely happens in the sparse topology, in this paper we only discuss the delivery probability with limited link duration, i.e. the probability that the transmission of a packet can be finished before the link breaks.

As illustrated in Fig.2, node *j* is passing by its *contact d* with a relative speed $\hat{v}_{jd}$. Let *x* denote the closest distance between *j*'s trajectory and node *d* and $\mathcal{T}_{lk}$ be the link duration. We have $\mathcal{T}_{lk} = \frac{2\sqrt{r^2-x^2}}{\hat{v}_{jd}}$. Let φ_i be the time required for transmitting the i_{th} packet to *d* and φ_{i-} be the amount of time that packet *i* has to wait until its transmission. We have $\varphi_{i-} = \sum_{k=0}^{i-1} \varphi_k + \varphi_{ctrl}$ $(i > 0)$ and $\varphi_{i-} = \varphi_{ctrl}$ $(i = 0)$, where φ_{ctrl} is the constant time for setting up a connection between the two nodes. Therefore, for a given φ_i and a relative speed estimate $\bar{v}$ the delivery probability of packet *i* (denoted as $\mathcal{P}_i$) is the probability that *x* is small enough to make $\mathcal{T}_{lk} > \varphi_i + \varphi_{i-}$, i.e. $\mathcal{P}_i(\varphi_i) = P(x \leq \sqrt{r^2 - (\frac{(\varphi_i+\varphi_{i-})\bar{v}}{2})^2})$. As *x* is uniformly distributed over [0, *r*) when the nodal movements reaches the steady state (assuming a RWP mobility model) [10], node *j* can estimate $\mathcal{P}_i$ for the i_{th} packet being forwarded to *d* by:

$$\begin{aligned} \mathcal{P}_i(\varphi_i) &= \int_0^{\sqrt{r^2-(\frac{(\varphi_i+\varphi_{i-})\bar{v}}{2})^2}} \frac{1}{r}\, dx \\ &= \begin{cases} \sqrt{1-(\frac{(\varphi_i+\varphi_{i-})\bar{v}}{2r})^2} & \varphi_i \leq \frac{2r}{\bar{v}} - \varphi_{i-} \\ 0 & \varphi_i > \frac{2r}{\bar{v}} - \varphi_{i-} \end{cases} \end{aligned} \tag{4}$$

Eq.(4) is based on the assumption that the mean relative speed $\bar{v}$ is known. For estimation of the distribution of link duration with unknown relative speed except of its distribution, we refer the readers to [11] or [12]. Knowing the time for transmitting a packet *i* to a contact and the current depth $\mathcal{H}$ of the delivery tree, a relay can now estimate the residual delivery time as:

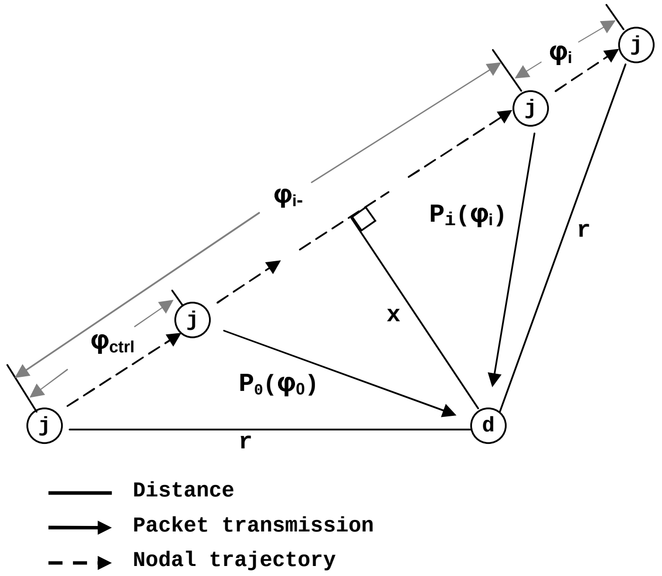


Fig. 2. Delivery probability with limited link duration

$$\mathcal{D}_{red}(\mathcal{H}, \varphi_i) = \frac{\bar{\mathcal{T}}_{int}}{2^{\mathcal{H}-1}\mathcal{P}_i(\varphi_i)} \tag{5}$$

C. Distributed Adaptation of the Relaying Process for a Delivery Delay Target

To allow relay nodes to independently adapt the relaying process, we add two new fields into the packet header, namely, the current depth of the binary delivery tree $\mathcal{H}$ and the delivery delay target $\mathcal{D}_{tg}$. $\mathcal{D}_{tg}$ is specified at the source according to the application layer requirement. $\mathcal{H}$ is increased for 1 at each relay node before being replicated to the next relay. The pseudo code for the implementation of the adaptation mechanisms at each relay node is listed in Fig.3. In the implementation, a relay can estimate the delivery probability of a packet to a *contact* using its knowledge of the packet size, the number of packets being forwarded to the *contact* and the bandwidth. From Fig.3 we can see that, based on the information embedded in a packet's header, a single node can make its own decision on whether to forward this packet to a *contact* or to wait until it meets the destination. On average, relay nodes in the same layer of the binary relaying tree of a packet would receive the packet around the same time (e.g. in a $\mathcal{N}$ nodes network the mean time that a packet spent for travelling from the $\mathcal{H}_{th}$ to the $\mathcal{H}+1_{th}$ layer of the binary delivery tree is about $\frac{\bar{\mathcal{T}}_{int}}{\mathcal{N}-2^{\mathcal{H}-1}}$). Assuming both the buffer size and the traffic load are constant, on average the communication traffic between the relay nodes and their *contacts* is constant hence they will draw the same conclusion on the delivery probability for the same packet. Therefore, the $\mathcal{H}_{th}$ layer relay nodes would make the same decision on whether to forward a packet. The forwarding process would continue until $\mathcal{H}$ is sufficient to achieve a residual delivery delay that is less than the delay target.

```
/* input: packet pkt, Contact_ID cID */
  AMR ( pkt, cID) ;

  if ( cID == pkt->dest ) {
            Relay_a_Packet ( pkt, cID ) ;
             return ;
  }

  if ( pkt->H  >= HEIGHT_LIMIT ) return ;

  $D_{cur}$ = CURRENT_TIME – pkt ->timestamp() ;

  $P_{delivery}$ = P( pkt->size, pkt->dest ) ;

  if ( $D_{cur}$ + $D_{red}$ ( pkt->H - 1, $P_{delivery}$ ) <= pkt->$D_{tg}$ ) {
            pkt->H = HEIGHT_LIMIT ;
            return ;
  }

  else {
            pkt->H ++ ;
            copy_packet ( pkt, pktcpy ) ;
            Relay_a_Packet ( pktcpy, cID ) ;
            return ;
  }
```

Fig. 3. Pseudo-code for the distributed adaptation of the relaying process of a packet

D. *Extra Traffic Due to the Imprecise Adaptation*

As the number of copies the AMR scheme produced for a packet tend to be powers of 2 with a specific delay target (e.g. if only 10 copies are needed for a delay target, the final number of copies produced by the AMR scheme would be $16 = 2^4 > 10$), we say such a binary tree based adaptation is imprecise. The extra copies $\Delta\mathcal{R}$ distributed for a packet would obviously bring more traffic than that are enough to achieve a certain delay target. However, we can show that the ratio of the extra copies over the network size $\frac{\Delta\mathcal{R}}{\mathcal{N}}$ becomes constant when the network size increases. Suppose that the number of nodes in the network grows to a very large size $\mathcal{N}$ ($2^{\mathcal{H}+1} > \mathcal{N} \geq 2^{\mathcal{H}}$), the biggest possible number of extra copies $\Delta\mathcal{R}_{max}$ produced by the imprecise adaptation is equal to $2^{\mathcal{H}} - 2^{\mathcal{H}-1} - 1$. We have $\frac{\Delta\mathcal{R}}{\mathcal{N}} \leq \frac{\Delta\mathcal{R}_{max}}{\mathcal{N}}$. Therefore, the extra copies per node produced for a packet due to the imprecise adaptation is limited by:

$$\frac{\Delta\mathcal{R}_{max}}{\mathcal{N}} = \frac{2^{\mathcal{H}} - 2^{\mathcal{H}-1} - 1}{2^{\mathcal{H}} + \varepsilon} \tag{6}$$

where $\varepsilon = \mathcal{N} - 2^{\mathcal{H}}$ and $\varepsilon \ll \mathcal{N}$. From Eq.(6), we can easily deduce that $\frac{\Delta\mathcal{R}}{\mathcal{N}}$ is bounded by 0.5 as the size of the network is growing to infinity.

III. SIMULATION RESULTS AND EVALUATION

In this section, we introduce our simulation model and give the simulation-based performance evaluation of the proposed AMR scheme.

A. *Simulation Model*

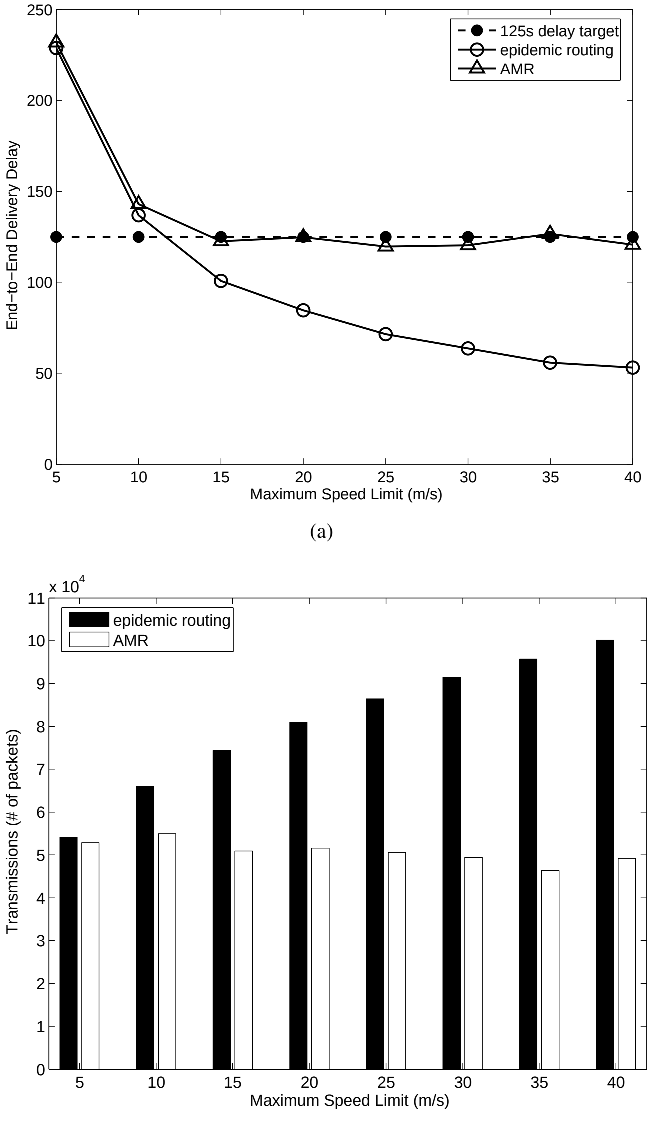
We evaluate the performance of the AMR scheme using the *ns-2* network simulator. The simulated ICMAN consists of $\mathcal{N} = 40$ nodes randomly distributed in a square area with the size of $\mathcal{L} \times \mathcal{L}$. Each node in the network utilizes IEEE 802.11b as the MAC protocol and is capable of transmitting data at the rate of 2 MBit/s. In our simulations, $\mathcal{L}$ is fixed at 600m while the radio radius of a node is set to be 50m resulting in a relatively sparse topology with a node density (given by $\frac{\pi r^2 \mathcal{N}}{\mathcal{L}^2}$) of 0.87. All the mobile nodes in the network move around according to the Random Waypoint Model (RWP). The pause time of the RWP model is kept at 0 to produce continuous movements. The movement speed of a mobile node is uniformly distributed over $[v_{min}, v_{max}]$, where v_{min} is fixed at 2.5m/s and v_{max} is varied from 5m/s to 40m/s to create simulation scenarios with different mobility. If not stated, the default value of v_{max} is 20m/s. Every node in the network is acting as the packet source, relay and receiver at the same time. However, no data traffic is produced in the 1100 seconds after the simulation starts to allow the mobility model to reach its steady state. After that, each node generate a packet destined to each of the rest of the network every 0.5 second for a duration of 500s. The simulation then continues for another 900 seconds to allow most of the packets to reach their destination. Each simulation is repeated 30 times with different random seeds to provide smoother results.

B. *Performance Evaluation*

The performance of the AMR scheme regarding its adaptability to varying network mobility with an ideal delivery probability of 1 are presented in Fig.4 and Fig.5 for 125s and 250s of delay targets, respectively. We can see from Fig.4(a) that the relay nodes in the AMR scheme keeps adapting to the increasing network mobility for the specified 125s delay target. Although it does not achieve the delay target when the network mobility is low (e.g. the maximum speed limit v_{max} is below 10m/s), the AMR scheme has achieved a delivery delay as quick as that are achieved by the fastest possible scheme, the epidemic routing[3]. When the network mobility climbs up, we can see that the AMR scheme quickly adjusts the depth of the delivery process at each relay node resulting in delivery delays that are just below the delay target. It seems that both epidemic routing and AMR can achieve the delivery target and the former is preferable to the latter as it is simpler. However, as shown in Fig.4(b), by adapting the depth of delivery process to the varying network mobility, in general the AMR scheme minimizes the traffic overhead to achieve the given delay target.

---

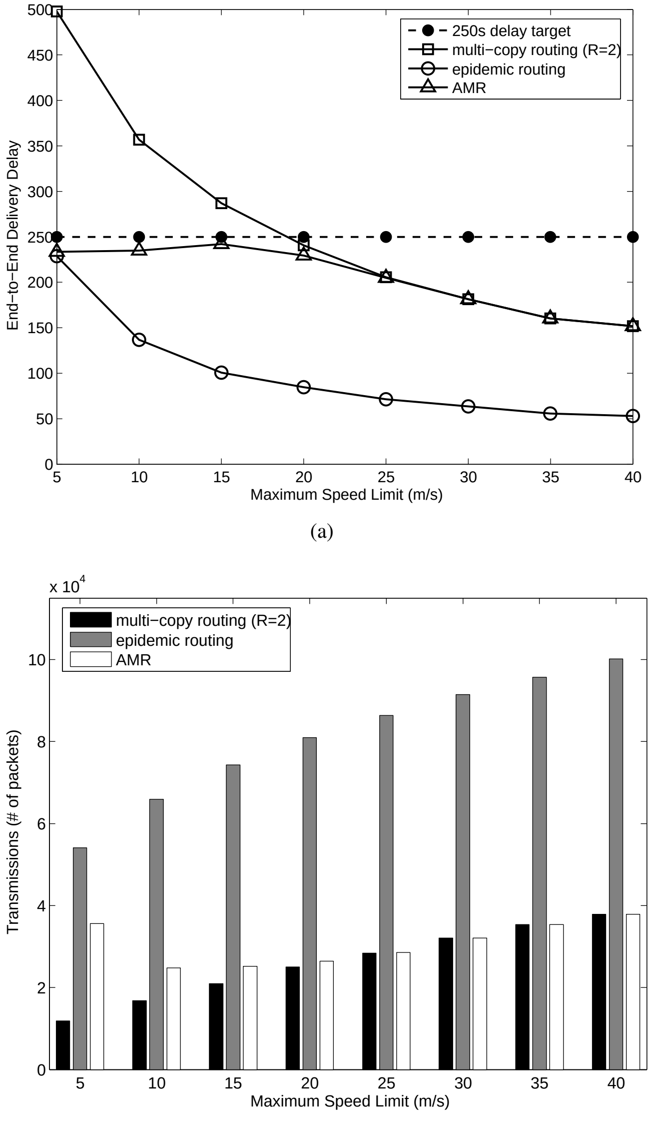
[3] By limiting the traffic generation rate, we can achieve a so low contention probability for epidemic routing that makes it become the fastest solution.



(a)

(b)

Fig. 4. Performance comparison with delay target 125 seconds (a) Delivery delay (b) Total packet transmissions



(a)

(b)

Fig. 5. Performance comparison with delay target 250 seconds (a) Delivery delay (b) Total packet transmissions

Similar results can be found in Fig.5, where the AMR scheme is still able to adapt to the varying network mobility to meet the relaxed delay target of 250s. For this relaxed delay constraint a multi-copy strategy with a fixed $\mathcal{R} = 2$ is preferable than the epidemic routing, as in this case 2 copies per packet is enough to keep the delivery delay below the target with a median degree of mobility (e.g. $v_{max} \geq 20m/s$). It is also shown in Fig.5(b) that overall the 2-copy multi-copy strategy even performs less packet transmissions than the AMR scheme. However, the extra traffic is utilized by the AMR scheme for scenarios of $v_{max} \leq 20m/s$ to achieve close-to-target delivery delays resulting in an overall optimal balance between the constraints of delay and bandwidth consumption.

The adaptability of the AMR scheme to a variety of delay targets with a constant delivery probability of 1 is demonstrated in Fig.6. The curve of the AMR scheme in Fig.6 sticks to that of the delay targets spanning from 90 seconds to 250 seconds, which confirms that the performance of the AMR scheme is stable over the wide range of delay targets. Fig.7 shows the performance of the AMR scheme in comparison to a 2-copy multi-copy scheme against varying delivery probabilities given a fixed delay target of 250 seconds and a constant $v_{max} = 20m/s$. The 2-copy multi-copy scheme has been shown in Fig.5 where the delivery probability is 1 to have a similar performance with the AMR scheme for the given delay target and network mobility. However, when the delivery probability decreases the fixed 2-copy multi-copy scheme misses the delay target due to its lack of adaptivity. On the contrary, Fig.7 shows that the varying delivery probabilities do not stop the AMR scheme from reaching the delay target.

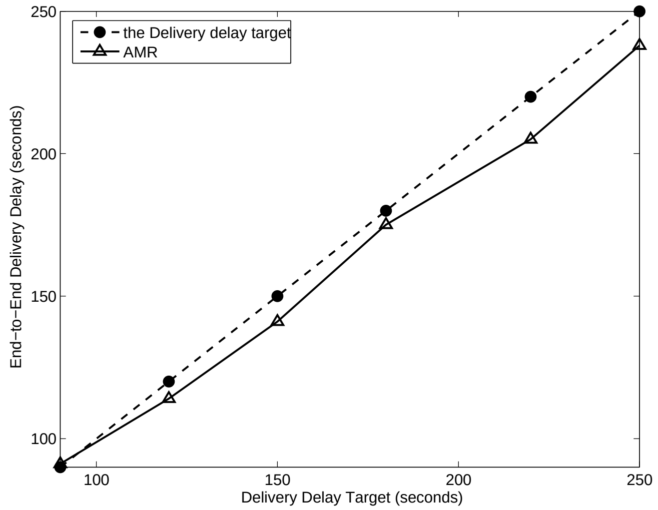


Fig. 6. the Delay targets Vs. the delivery delays achieved by the AMR scheme

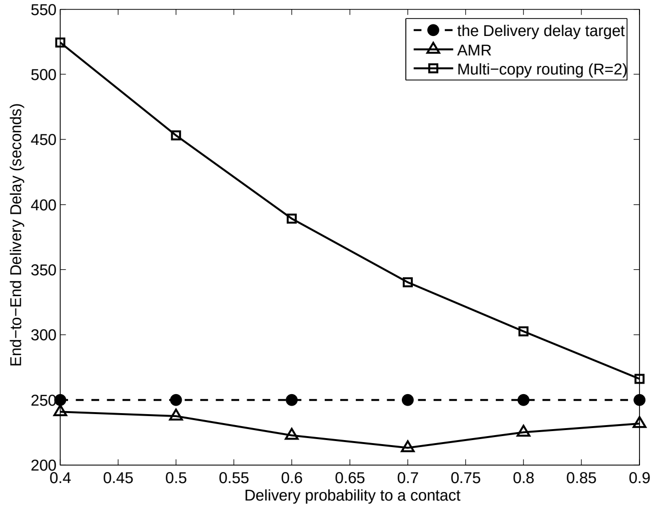


Fig. 7. the Delivery probability Vs. the end-to-end delivery delay

There are bigger gaps between the delivery delay of the AMR scheme and the delay target when the delivery probability is around 0.7. These gaps are resulted from the imprecise adaptation as explained in Section II.D.

The delivery ratios of the AMR scheme and those of other strategies being compared are not presented here. As we designed a big enough relay buffer at each node and allocated enough time for the multi-hop packet relay, all the protocols investigated achieved a delivery ratio of over 98% in all scenarios.

IV. CONCLUSION

Store-and-forward based routing (e.g. multi-copy relaying) has been recently proposed for Intermittent Connected MANETs (ICMANs) to replace traditional routing technologies that were designed for fully connected paths. However, such multi-copy relaying strategies do not adapt to the frequent variations of network conditions as they rely on the source to control the relaying process. The *store-and-forward* based scheme, the Adaptive Multi-Copy Routing (AMR), addresses this issue and provides an efficient way for packet delivery in ICMANs. Instead of using a source-defined *replication factor*, each relay node independently decides whether to forward a packet based on the delivery delay target and current network conditions. Simulation results show that the proposed scheme always adapt to the varying network mobility and/or delivery probability which cannot be anticipated in the conventional source-defined strategies and delivers packets within specific delay targets with minimum traffic overhead.

ACKNOWLEDGMENT

The work reported here is supported in part by the BIOPATTERN EU Network of Excellence (EU Contract 508803).

REFERENCES

[1] A. Vahdat and David Becker, “Epidemic routing for partially-connected ad hoc networks,” *Technical Report, Duke University*, April 2003.

[2] A. Lindgreny, A. Doria, and O. Scheleny, “Probabilistic routing in intermittently connected networks,” *SIGMOBILE Mobile Computng Communication Review*, 2003.

[3] J. Leguay, T. Friedman, and V. Conan, “Evaluating mobility pattern space routing for dtns,” *in Proc. IEEE Infocom*, April 2006.

[4] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, “Single-copy routing in intermittently connected mobile networks,” *in Proc. of IEEE SECON*, October 2004.

[5] ——, “Multi-copy routing in intermittently connected mobile networks,” *Technical Report, University of Southern California*, 2004.

[6] T. Small and Z. J. Hass, “Resource and performance tradeoffs in delay-tolerant wireless networks,” *in Proc. of WDTN Workshop in association with ACM SIGCOMM*, August 2005.

[7] Y. Liao, K. Tan, Z. Zhang, and L. Gao, “Estimation based erasure-coding routing in delay tolerant networks,” *Microsoft Technical Report*, 2006.

[8] R. Groenevelt, “Stochastic models for mobile ad hoc networks,” *PhD Thesis, INRIA*, April 2005.

[9] D. C. Montgomery and G. C. Runger, “Applied statistics and probability for engineers (second edition),” *John Wiley*, 1999.

[10] W. Navidi and T. Camp, “Stationary distributions for the random waypoint mobility model,” *IEEE Transactions on Mobile Computing*, vol. 3, no. 1, January-March 2004.

[11] S. Cho and J. P. Hayes, “Impact of mobility on connection stability in ad hoc networks,” *in Proc. of IEEE WCNC*, March 2005.

[12] Y. Han, R. J. La, and A. M. Makowski, “Distribution of path durations in mobile ad-hoc netwroks - palm’s theorem at work,” *in Proc. of the 16th ITC Specialist Seminar*, August 2004.