Proceedings of 2009 APSIPA Summit and Conference, Sapporo, Japan, October 4-7, 2009

Rippling Broadcast (RIP-cast) for Wireless Ad Hoc Networks

Shan-Chi Yang, Ling-San Meng, and Ping-Cheng Yeh
Department of Electrical Engineering and
Graduate Institute of Communication Engineering
National Taiwan University
Taipei, Taiwan

***Abstract*—Broadcasting service is an important application in wireless networks. The goal is to distribute data packets from one source node to all other nodes in the network. Delivery ratio and reception delay are the two primary measures for broadcasting performance. Due to the wireless channel fluctuations, packet loss is often seen in wireless transmissions. However, the existing broadcast schemes have difficulty achieving good performance under severe packet loss. In this paper, we propose the rippling broadcast (RIP-cast) scheme which is able to achieve reliable and efficient transmissions in wireless ad hoc networks. The nodes in the network are designed to distribute data packets cooperatively in a special manner. The broadcast scheme also adopts the use of fountain codes to further enhance the reliability as well as the diversity of data transmissions. It is observed from the numerical experiments that the proposed scheme significantly enhances the delivery ratio and reduces the reception delay under packet loss compared to the existing broadcast schemes. The results suggest that RIP-cast is highly suitable for broadcasting service in wireless networks.**

I. Introduction

Wireless ad hoc networks consist of nodes that communicate in a distributed manner without the coordination of a central control station. Typical applications can be found in situations where built-up infrastructure is not available or impractical, such as disaster rescue or military purpose. Commercial deployment of ad hoc networks also emerges in the recent years [1].

Due to the highly dynamic property of the ad hoc networks, end-to-end route establishment can be a challenging task. The problem mainly lies in the lack of the knowledge of the network topology. To acquire such information, the technique of broadcast is often incorporated as part of the routing algorithms, e.g., the ad hoc on-demand distance vector (AODV) [2] and dynamic source routing (DSR) [3]. In addition to the task of route finding, broadcast is also needed for multimedia distribution, software update, and emergency notification. Since ad hoc networks are typically limited in hardware resource, identifying efficient broadcast schemes is of practical importance. Concerning the impairment introduced by the wireless channel, the reliability of packet delivery has also to be taken into account when developing broadcast protocols [4], [5].

Traditional broadcast schemes require each node rebroadcast the received packets directly, thus causing inevitable redundancy. The protocol design focus is usually on the reduction of such overhead while achieving fast and comprehensive message dissemination. The issue of reliability is less discussed. Flooding [6] is the simplest broadcast technique. As the name suggests, it requires that every node rebroadcast upon receiving a new packet. Aiming at reducing the redundancy, the probability-based and counter-based methods are both proposed in [7]. In the probability-based flooding scheme, upon receiving a new packet, a node rebroadcasts with a predefined probability p. One can thus trade the messaging overhead for the delivery ratio by adjusting p. In the counter-based scheme, a node initiates a timer upon receiving a previously unseen packet. When the timer expires, the node rebroadcasts if the number of duplicated packets received is less than a threshold value η.

In [8]–[11], more advanced broadcast schemes are proposed provided that the knowledge of the location information or the knowledge of the neighbors are available to the nodes. For example, the scalable broadcast algorithm (SBA) [11] requires that the knowledge of 2-hop neighbors be available to the nodes. The concept of the *broadcast cover set* is then applied for making rebroadcast decision at each node on receiving packets. However, acquiring the location information is generally a difficult task, especially for low-power and low-cost sensor networks. The use of neighbor knowledge could also lead to problems when the network topology changes too fast [12]. Finally, as noted above, the factor of reliability for packet delivery is not considered in all of the aforementioned protocols. In the fountain broadcast scheme (FBcast) [5], the broadcast messages are partitioned and encoded using fountain codes [13]. By virtue of the decoding mechanism of the fountain codes, both the broadcast efficiency and reliability can be improved. However, it is noted that the disseminated packets in FBcast are still duplication of the original coded segments. The problem of redundancy thus remains.

In this paper, a novel broadcast scheme based on fountain codes is proposed. We refer to this technique as the rippling broadcast scheme (RIP-cast). The nodes in the network are designed to distribute data packets cooperatively in a special manner. The broadcast message is partitioned into small segments and encoded using fountain codes. Every node in the network having received enough number of coded segments could perform fountain encoding to generate new coded segments and serve as another message source. The message spreading process resembles the phenomenon of ripple effect, hence the name rippling broadcast. RIP-cast distinguishes itself from the traditional broadcast schemes in that the disseminated packets in the networks are distinct rather than duplicated. Transmission reliability can thus be improved as a result of diversity. Simulation results show that the proposed scheme significantly enhances the delivery ratio and reduces the reception delay under packet loss compared to the existing broadcast schemes. The results suggest that RIP-cast is highly suitable for broadcasting service in wireless networks.

The rest of this paper is organized as follows. The system model is described in Section II. We present the proposed rippling broadcast scheme in Section III. The simulation results are given in Section IV. Finally, the conclusions are drawn in Section V.

II. Model Description

In this work, we consider a wireless ad hoc network as shown in Fig. 1. The network consists of nodes that communicate in a distributed manner without the existence of a central control station. It is assumed that each node is equipped with an isotropic antenna for transmission and reception, and all nodes use the same transmit power P_t. Given a transmitting node i in Fig. 1, the received power $P_{r,j}$ of the receiving node j is expressed as [14]

$$P_{r,j} = P_t G_t G_r \left(\frac{\lambda_0}{4\pi r_{ij}}\right)^2, \tag{1}$$

where

- G_t denotes the transmitting antenna gain,
- G_r denotes the receiving antenna gain,
- λ_0 denotes the free-space wavelength,
- r_{ij} denotes the distance between i and j.

Depending on the received SNR and the PHY setting used, the transmitted packet will have different packet error rate that might cause the packet loss.

The goal of this work is to design a broadcast protocol for a node to broadcast its data to the other nodes. All nodes are assumed to be capable of conducting encoding/decoding operations of fountain codes. The encoding procedure is described in [13]. The node first partitions the broadcast message into K data segments. It then randomly picks a number d following a probability mass function $\rho(d)$. Then the node randomly chooses d data segments and generates the bit-wise exclusive OR (XOR) result of these d data segments. The XOR result is regarded as a fountain coded segment with degree d and sent as a packet to the receiver. After the receiver successfully receives fountain coded segments, the receiver starts fountain decoding using the sum-product algorithm to reconstruct the data segments and thus the transmitted message. Details of the sum-product algorithm can be found in [15].

The receiving buffer of each node is assumed to be sufficiently large to save all of the received fountain coded segments for the decoding procedure. The medium access control (MAC) protocol run on each node follows the IEEE 802.11 standard [16]. Each node utilizes the carrier sense multiple access/collision avoidance (CSMA/CA) procedure to access the channel. It is worth noting that the IEEE 802.11 MAC does not use RTS/CTS handshake procedures for broadcast. Only the CSMA/CA procedure is used, and there is no MAC-level recovery on the data broadcast.

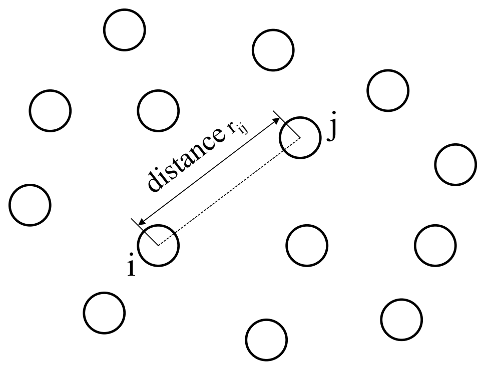

Fig. 1. An example of the wireless ad hoc network.

III. Rippling Broadcast Scheme

Here we propose the RIP-cast scheme for broadcasting service in wireless ad hoc networks. The operations of the source node and the relay (non-source) node are different. The source node first partitions the broadcast message into several equal-size data segments and broadcasts these data segments without fountain encoding the data. When a relay node receives data segments from the source node or fountain coded segments from other relay nodes, it conducts fountain decoding to reconstruct the data segments. To avoid same segments frequently rebroadcasted by the relay nodes, each relay node is designed to generate a new fountain coded segment from its reconstructed data segments. When a node finishes reconstructing all of the data segments, it attaches 1-bit acknowledgement (ACK) information to the fountain coded segment it rebroadcasts. The ACK information is to notify its neighbors that it has reconstructed all of the data segments. Each relay node keeps track of the percentage of its neighbors that have transmitted the ACK information. When the percentage is higher than a certain threshold, it indicates that a large proportion of its neighbors have reconstructed the original message. The node then stops broadcasting once itself also finishes reconstructing all data segments. Detailed description of RIP-cast is given as follows.

A. Broadcasting Operation on the Source Node

The operation of the RIP-cast run on the source node is listed in Algorithm 1. In our design, the source node u partitions the broadcast message into K equal-size data segments $s_1, s_2, ..., s_K$. Node u then broadcasts s_1 to s_K one by one, repeatedly until $R_{ACK}(u)$, the percentage of its neighbors that have reconstructed all of the data segments, is greater than a predetermined threshold R^{th}_{ACK}.

B. Broadcasting Operation on the Relay Node

The main complexity of RIP-cast is on the relay node, and the operations are listed in Algorithm 2. When a relay node v receives a new fountain coded segment or a data segment from other nodes, it performs the decoding procedure using the received segments. After the decoding procedure, node v saves the data segments it has reconstructed in its reconstructed segment set, denoted as $RSSet(v)$. Note that the cardinality $|RSSet(v)|$ of $RSSet(v)$ gives the number of the data segments node v has reconstructed.

If $|RSSet(v)| = 0$, it implies that node v has not yet reconstructed any data segment, and thus node v does not participate in the broadcast process. If $0 < |RSSet(v)| < K$, it implies node v has reconstructed some of the data segments. To avoid rebroadcasting the same fountain coded segments, it is designed that relay node v generates a new fountain coded segment from $RSSet(v)$ for broadcast. In addition, node v maintains a coded segment set, denoted as $CSSet(v)$, which consists of the data segments that can be reconstructed from the fountain coded segments previously sent by node v. In our design, to ensure neighbors of node v can receive a new fountain coded segment from node v that contains information about the data segment that has not been involved in the previous coded segments generated by node v, node v first randomly chooses a segment s_k from $RSSet(v) - CSSet(v)$. Node v then randomly chooses $d-1$ segments from $RSSet(v) - s_k$ to generate a degree-d fountain coded segment. As an illustrative example in Fig. 2, $RSSet(v) = \{s_1, s_2, s_3, s_4, s_5\}$ and $CSSet(v) = \{s_2, s_3, s_5\}$. Assume node v is about to generate a degree-d fountain coded segment. Node v first chooses s_1 from $\{s_1, s_4\}$. It then randomly chooses another 2 segments from $RSSet(v) - s_1$. On the other hand, if $RSSet(v) - CSSet(v) = \phi$, it means no matter how node v encodes, the fountain coded segment can not help its neighbors to reconstruct any new data segment. Node v then stops rebroadcasting.

For a node with a larger $|RSSet(v)|$, it indicates this node has reconstructed more data segments than other nodes. Hence, the fountain coded segment generated by this node is more likely to carry information about the data segment that has not yet been reconstructed by its neighbors. In order to give such node higher priority to rebroadcast, we use $\frac{|RSSet(v)|}{K}$ as the rebroadcast probability of the node. This further reduces the channel access and increases the broadcast efficiency of RIP-cast.

If $|RSSet(v)| = K$, it implies node v has reconstructed all of the data segments. There are two cases. If $RSSet(v) - CSSet(v) \neq \phi$, node v uses the same method to generate the fountain coded segment as described in the case of $0 < |RSSet(v)| < K$. If $RSSet(v) - CSSet(v) = \phi$, node v randomly chooses d segments from $RSSet(v)$ to generate new fountain coded segment and rebroadcasts with a predetermined probability P^t_{ACK}. In both cases, node v attaches 1-bit ACK information to the fountain coded segment. This 1-bit ACK is to inform its neighbors about the success of reconstruction of all data segments.

Algorithm 1: RIP-cast on the Source Node

Given the source node u,
Partition the broadcast message into K data segments.
Broadcast s_1 to s_K repeatedly.
if *($R_{ACK}(u) > R^{th}_{ACK}$)* **then**
    Stop broadcasting.

Algorithm 2: RIP-cast on the Relay Node

Given the relay node node v,
Decode the received fountain coded segments.
Update $RSSet(v)$.
if *($|RSSet(v)| = 0$)* **then**
    Do nothing.
if *($0 < |RSSet(v)| < K$)* **then**
    Update $CSSet(v)$.
    if *($RSSet(v) - CSSet(v) \neq \phi$)* **then**
        Use $RSSet(v)$ and $CSSet(v)$ to generate a fountain coded segment.
        Use $\frac{|RSSet(v)|}{K}$ as the rebroadcast probability.
        Record the rebroadcasted fountain coded segment.
    else
        Do nothing.
if *($|RSSet(v)| = K$)* **then**
    if *($R_{ACK}(v) > R^{th}_{ACK}$ and v has sent the ACK information)* **then**
        Stop broadcasting.
    else
        Update $CSSet(v)$.
        if *($RSSet(v) - CSSet(v) \neq \phi$)* **then**
            Use $RSSet(v)$ and $CSSet(v)$ to generate a fountain coded segment.
            Attach the ACK information to the fountain coded segment.
            Rebroadcast the fountain coded segment.
            Record the rebroadcasted fountain coded segment.
        else
            Use $RSSet(v)$ to generate a fountain coded segment.
            Attach the ACK information to the fountain coded segment.
            Use P^t_{ACK} as the rebroadcast probability.

C. Stopping Criterion

In this work, it is assumed that each node is aware of its neighbors through the "hello" messages exchange procedures described in [11]. Consider a node w (either source or relay), it keeps track of the percentage of its neighbors that have transmitted the ACK information, $R_{ACK}(w)$. If $R_{ACK}(w)$ is larger than a predetermined threshold R_{ACK}^{th}, it means a large portion of its neighbors have reconstructed all of the data segments. Hence, it has little need for node w to rebroadcast. As a result, node w stops broadcasting if $R_{ACK}(w) > R_{ACK}^{th}$ provided that it has sent the ACK information.

IV. Simulation Results

In this section, we evaluate the performance of the proposed RIP-cast scheme using computer simulation. The nodes are assumed to be uniformly distributed in a square area of 2500×2500 meter^2 with the source node situated in the center. The size of each broadcast message is set to 1000 bytes, which is then partitioned into 10 data segments, e.g., K=10. For isotropic antennas, we have $G_t = G_r = 1$. The free-space wavelength λ_0 is set to be 0.124 meters. The transmit power P_t is 8 mW, which gives a transmission range of approximately 500 meters. The IEEE 802.11b standard is adopted for the PHY setting. Following (1), the received power and hence the SNR can be calculated given the distance between the transmitter and the receiver. The resulting packet error rate is then determined by the received SNR and the PHY setting used. For comparisons, four other broadcast techniques are also simulated. They are the probability-based broadcast scheme (PB-cast), the counter-based broadcast scheme (Counter-cast), the SBA, and the FBcast scheme.

The delivery ratio, defined as the percentage of nodes successfully reconstructing the original message, is one of the primary measures for broadcasting performance. We plot the accumulated delivery ratio versus the simulation time in Fig. 3 ∼ 6 with increasing number of nodes. For convenience, we have adopted the notations of "RIP-cast, α, β" to denote the RIP-cast scheme with $R_{ACK}^{th} = \alpha$ and $P_{ACK}^{t} = \beta$, "PB-cast, α" to denote the PB-cast scheme with rebroadcast probability $p = \alpha$, and "Counter-cast, α" to denote the Counter-cast scheme with threshold value $\eta = \alpha$. It should be noted that "PB-cast, 1.0" corresponds to the case of simple flooding. It is observed that in the presence of severe packet loss, the traditional broadcast schemes have rather poor performance, especially when the network is sparse. We can also observe that in general, flooding outperforms the PB-cast scheme with $p = 0.8$, and the Counter-cast scheme with $\eta = 10$ outperforms that with $\eta = 5$. This is due to the fact that under severe packet loss, cutting redundancy is unfavorable to the process of message dissemination, since most of the packets are dropped as a result of corruption. This also explains why the SBA have such unsatisfactory performance. On the other hand, we can observe that the proposed RIP-cast scheme achieves both fast and reliable message dissemination, and bears only minor performance degradation when the network is sparse. The results also indicate that under the considered network model, the performance of RIP-cast is affected, but not critically determined by P_{ACK}^{t}. This provides us insight for the protocol optimization in the future.

To quantify the overhead associated with each message broadcast, we propose the use of the *average transmitted*

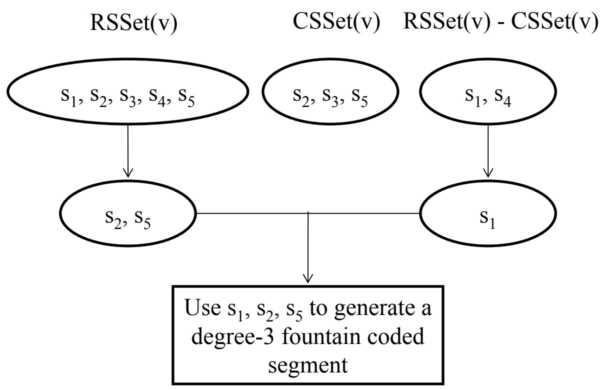

Fig. 2. An Example of the segment selection.

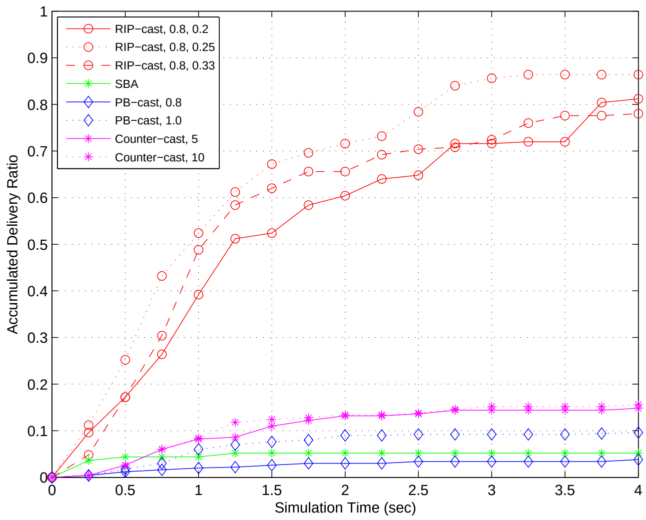

Fig. 3. The accumulated delivery ratio along the simulation time for a network consisting of 50 nodes.

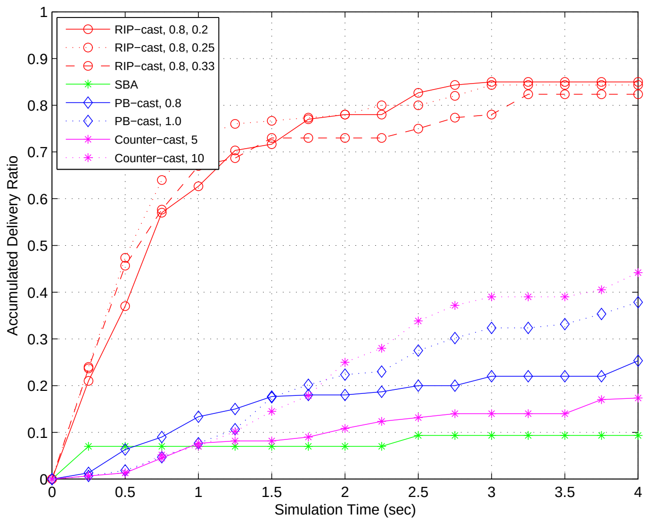

Fig. 4. The accumulated delivery ratio along the simulation time for a network consisting of 60 nodes.

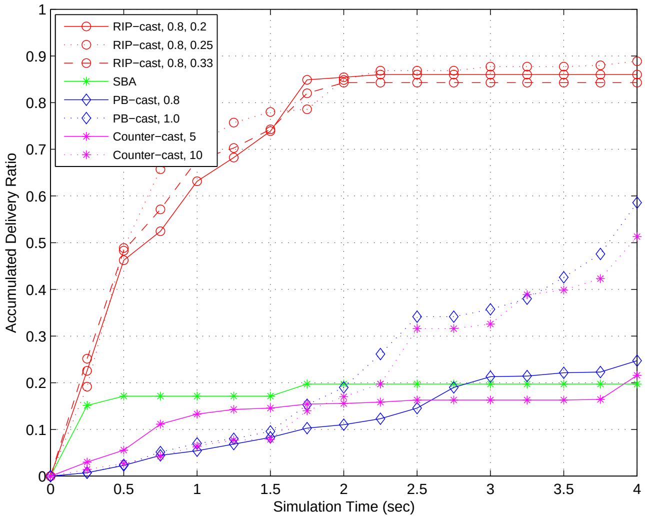

Fig. 5. The accumulated delivery ratio along the simulation time for a network consisting of 70 nodes.

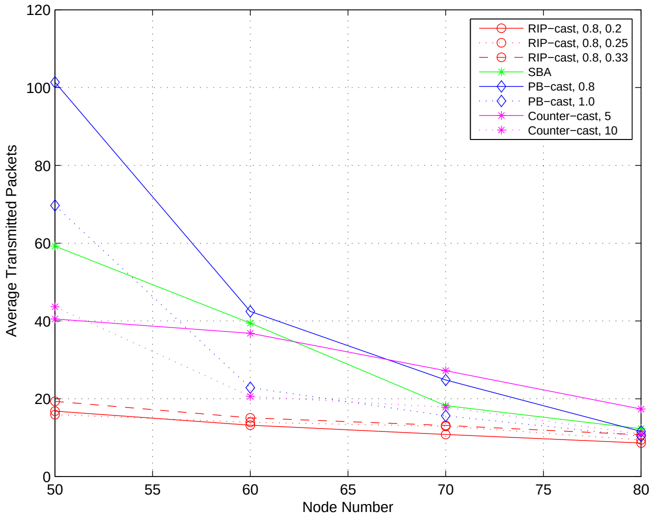

Fig. 7. The average transmitted packets versus the number of nodes in the network.

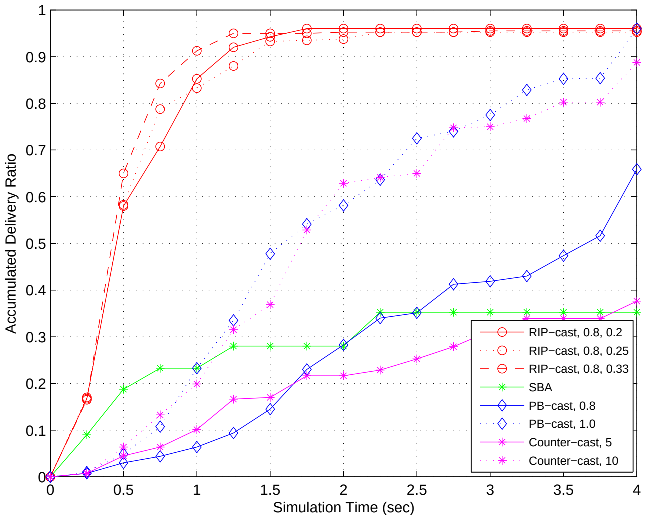

Fig. 6. The accumulated delivery ratio along the simulation time for a network consisting of 80 nodes.

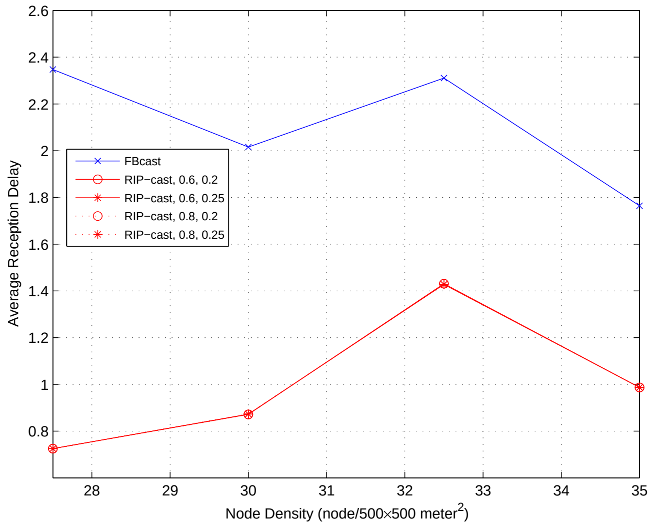

Fig. 8. The average reception delay versus the node density.

packets, which is defined as the ratio of the total number of rebroadcasted data/coded segments to the number of nodes successfully reconstructing the original message. In Fig. 7, we plot the average transmitted packets as a function of the number of nodes in the network. It is interesting to note that the overhead associated with PB-cast with $p = 1.0$ (flooding) is lower than that with $p = 0.8$, and the overhead associated with Counter-cast with $\eta = 10$ is lower than that with $\eta = 5$, since traditionally the former ones are regarded as causing more redundancy. As can be observed, the proposed RIP-cast scheme again achieves the best performance in terms of the induced message overhead.

In Fig. 8, we plot the average message reception delay versus the network node density for both the proposed scheme and FBcast. The average reception delay is calculated as the average time delay for a node to reconstruct the original message from the coded segments. For a direct comparison, the unit of node density in [5] has been adopted here, which is defined as the number of nodes per area of 500×500 meter^2. The results indicate that the proposed RIP-cast scheme has significantly smaller reception delay compared with FBcast to a considerable proportion at all tested network node density. We compare the average transmitted packets of the RIP-cast scheme against the FBcast scheme in Fig. 9. It is observed that for RIP-cast with $R_{ACT}^{th} = 0.8$ and $R_{ACT}^{th} = 0.6$, the message overhead is lowered by a factor of approximately 60% and 70%, respectively. We thus conclude that the proposed RIP-cast scheme is highly suitable for broadcasting service in the wireless environments.

V. Conclusions

An novel broadcast scheme, RIP-cast, has been proposed in this paper. The proposed RIP-cast scheme adopts the use

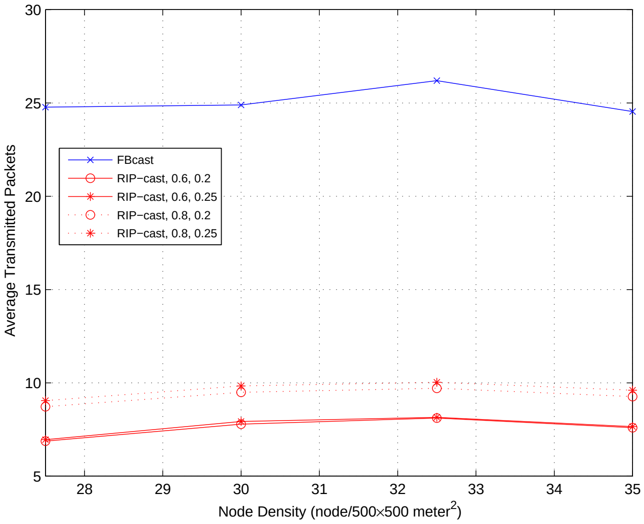

Fig. 9. The average transmitted packets versus the node density.

of fountain codes to achieve both reliable and efficient message broadcasting. The nodes in the network are designed to work cooperatively to generate new fountain coded segments. The proposed scheme distinguishes itself from the traditional broadcast schemes in that the disseminated packets in the networks are distinct rather than duplicated. Transmission reliability can thus be improved as a result of diversity. Simulation results show that the proposed scheme significantly enhances the delivery ratio and reduces the reception delay under packet loss compared with the existing broadcast techniques.

REFERENCES

[1] E. Huang, W. Hu, J. Crowcroft, and I. Wassell, "Towards commercial mobile ad hoc network applications: A radio dispatch system," in *Proc. ACM MobiHoc'05*, May 2005, pp. 355 – 365.

[2] C. E. Perkins and E. M. Royer, "Ad hoc on-demand distance vector routing," in *Proc. IEEE Workshop on WMCSA*, 1999, pp. 90–100.

[3] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," *IEEE Trans. Mobile Computing*, pp. 153–181, 1996.

[4] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, and S. Wicker, "An empirical study of epidemic algorithms in large scale multihop wireless networks," Intel Research, Tech. Rep., 2002.

[5] R. Kumar, A. Paul, and U. Ramachandran, "Fountain broadcast for wireless networks," in *Proc. IEEE INSS*, 2005.

[6] C. Ho, K. Obraczka, G. Tsudik, and K. Viswanath, "Flooding for reliable multicast in multi-hop ad hoc networks," in *Proc. International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM)*, 1999, pp. 64–71.

[7] S. Y. Ni, Y. C. Tseng, Y. S. Chen, and J. P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proc. ACM MobiCom*, 1999, pp. 151–162.

[8] W. Peng and X. Lu, "AHBP: An efficient broadcast protocol for mobile ad hoc networks," *Journal of Computer Science and Technology*, vol. 16, no. 2, pp. 114–125, 2001.

[9] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying for flooding broadcast message in mobile wireless networks," in *Proc. Hawaii International Conference on System Sciences (HICSS)*, 2002, pp. 3866–3875.

[10] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc networks," in *Proc. ACM MSWiM*, 2000, pp. 61–68.

[11] W. Peng and X. C. Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks," in *Proc. ACM MobiHoc*, 2000, pp. 129–130.

[12] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proc. ACM Mobihoc'02*, June 2002.

[13] D. J. C. MacKay, "Fountain codes," in *Proc. IEE Commun.*, vol. 152, May 2005, pp. 1062–1068.

[14] K. Chang, *RF and Microwave Wireless Systems*. Wiley, 2002.

[15] D. J. C. Mackay, *Information theory, inference, and learning algorithms*. Cambridge University Press, 2003.

[16] Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. IEEE Std 802.11, 1999.