Abstract—This paper characterizes the throughput-delay tradeoffs in mobile ad hoc networks (MANETs) with network coding, and compares results in the situation where only replication and forwarding are allowed in each node. The schemes/protocols achieving those tradeoffs in an effective and decentralized way are proposed and the optimality of those tradeoffs is established. The scenarios in which network coding can provide significant improvement on network performance are identified under different node mobility patterns (fast and slow mobility). The insights on when and how information mixing is beneficial for MANETs with multiple unicast and multicast sessions are provided. As far as we know, this is the first work characterizing scaling laws of throughput and delay of MANETs with network coding.

I. INTRODUCTION

One distinct characteristic of wireless mobile ad hoc networks (MANETs) is that, besides transporting data through multi-hop connected paths between the source and destination, packets can also be delivered through the physical mobility of relay nodes which is called store-carry-and-forward paradigm in the literature [1]. Grossglauser and Tse [2] have shown that significant gains in per-node throughput can be obtained by exploiting this paradigm. In particular, they proposed a 2-hop relaying scheme, and showed that it can achieve a constant per-node throughput. The scheme overcomes the throughput bound of $O(1/\sqrt{n \log n})$ originally established by Gupta and Kumar [3] for a static wireless network, where $n$ is the number of nodes for definitions of the standard asymptotic notation used throughout the paper). Although heavy use of relaying through node mobility allows for higher throughput, it also bears negative side-effects: increased delay. It has been shown in [4], [5] that the 2-hop relaying scheme in [2] yields an extremely large average delay of $\Omega(n)$. Since both throughput and delay are important network performance metrics from the perspective of an application, significant effort in the last few years has been devoted to understand the throughput-delay relationship in MANETs (refer to Section II-A and the references therein) in the networking research community. An interesting work by Neely and Modiano [5] suggested to utilize redundant packets transmission through multiple opportunistic paths (which are composed of multiple opportunistic links) of a MANET to balance the conflicting requirements on throughput and delay. The basic idea is that the time required for a packet to reach the destination (i.e., end-to-end delay) can be reduced by repeatedly transmitting this packet to many relay nodes of the network, and thus improving the chances that some user holding an original or duplicate version of the packet reaches the destination node. Clearly, the cost of this approach is the decreased throughput since duplicate packets waste scarce opportunities of wireless transmissions. In particular, with i.i.d. mobility, it was shown that for per-node throughput $T(n) = O(1)$, the relaying strategies with replication could yield end-to-end delay $D(n)$ scaling as $\Theta(n \cdot T(n))$ [5].

Previous studies on the scaling laws of MANETs, as discussed above, are all based on the implicit assumption that each node can only perform traditional operations on packets, such as storage, replication and forwarding. Recently, network coding, first introduced by Ahlswede et al. [6] in 2000, has been widely recognized as a promising primitive operation besides simple replicating and forwarding incoming packets [7]. Using the paradigm of network coding, when a node is scheduled to transmit, it may transmit a “mixed” packet as a result of algebraic operations on its incoming packets to maximize the usefulness of this transmission to all receivers in its transmission range. It is worth noting that a particular useful form of network coding called Random Linear Coding (RLC) was proposed in the literature [8], [9] to independently and randomly mix incoming packets at each node with linear operations, which allows the nodes of the network to achieve the optimal performance in a decentralized fashion.

Intuitively, when RLC instead of replication is used to minimize the end-to-end delay, network congestion can be alleviated and the requirement on buffer size can be relaxed. Therefore, a better throughput-delay tradeoff is expected to be obtained. Since network coding was not taken into consideration in Grossglauser and Tse’s original work [2] and the related work [4], [5], [10] that followed, an interesting question raised naturally is how much benefit network coding can provide to the network performance of MANETs compared to when only simple replication and forwarding are allowed for relay nodes. Answering this question will help us better understand not only the benefits and limitations of network coding in wireless networks but also the fundamental tradeoffs determining MANET’s performance.

In this paper we conduct a thorough study on the scaling laws governing MANETs. We characterize the throughput-delay tradeoffs with respect to different node mobility patterns. We identify scenarios in which network coding can provide significant improvement on network performance. Note that our work differentiates MANETs from static wireless networks by the roles network coding plays, because previous work showed that network coding could only provide constant improvement on the throughput of static wireless networks (cf. Section II-C and the references therein). We also provide insights on when and how information mixing is beneficial and propose algorithms to show that these benefits can be achieved in an effective and decentralized fashion.
II. BACKGROUND AND RELATED WORK

A. Scaling Laws of MANETs without Network Coding

Seminal work of Gupta and Kumar [3] initiated the investigation on how the throughput of wireless networks scales with \( n \), the number of nodes. Under the assumption that nodes with common transmission range are randomly distributed, it is shown that per-node throughput for static wireless networks scales as \( \Theta(1/\sqrt{n \log n}) \). Note that [3] implicitly used a fluid model for establishing throughput scaling. Later work by Kulkarni and Viswanath [11] consolidated the result of [3] with an explicit constant packet size model. In [12], with percolation theory, Franceschetti et al. showed that the \( \Theta(1/\sqrt{n \log n}) \)-per-node throughput is achievable if each node can adjust its transmission range (through power control), however, the throughput vanishing problem for large-scale (\( n \to \infty \)) static wireless networks still remains. In [2], Grossglauser and Tse showed that the mobility of the nodes in a MANET can be exploited to overcome this problem. The 2-hop relaying scheme they proposed achieves a constant per-node throughput at the cost of a large delay on the order of \( n \) [4], [5]. This result reveals the possibility of trading larger delay for higher throughput or lower throughput for smaller delay in MANETs. Since then, a flurry of research activities have tried to characterize the throughput-delay relationship with respect to node mobility, e.g., [4], [5], [13]–[20].

In general, there are two ways to trade throughput for delay in the literature. Kleinrock et al. [21] may be the first to find that delay can be reduced by increasing the transmission radius of each relay node, at the expense of reducing the number of simultaneous transmissions the network can support, which leads to a lower throughput. Similar transmission radius scaling techniques have appeared in [4], [14]–[20]. Another approach, which improves delay via redundant packet transfers is considered in [5], [22]. In this paper, we follow this approach, adopting redundant strategy and comparing it with network coding for the following reasons:

- First of all, the assumption that transmission ranges can scale with \( n \), the number of nodes, is impractical for large-scale MANETs. To obtain the scaling law of MANETs, we usually require \( n \) tending to infinity, which is equivalent to assuming \( \sqrt{\lambda_n} \to \infty \) for extended network model, where \( \lambda_n \) is the area of the network (cf. Section III-A). In general, wireless device is power limited, rendering it impossible to require the transmission range reaching the order of \( \sqrt{\lambda_n} \).
- Second, tradeoffs theoretically analyzed using the first means mentioned above are mainly based on fluid model, in which the packets are allowed to be arbitrarily small as \( n \to \infty \) (e.g., [4], [14]–[17], [19], [20]). On the other hand, tradeoffs obtained through the second approach assume constant packet size model, where packet size remains constant, i.e., does not scale down with \( n \) (e.g., [5]). We prefer the constant packet size model since in reality, packet size does not change when more nodes are added to the network. Furthermore, fluid model cannot be applied to scenarios with network coding, since every coded packet includes a “code vector” of at least constant size for successful decoding.

Note that with the additional constraint that the packet size remains constant, the throughput-delay tradeoff can be no better than that in the fluid model, and the analysis of constant packet size model is much harder than that of fluid model [18]. Throughout the paper, our results on scaling laws of MANET with or without network coding are all based on constant packet size model (cf. Section III-A) for the rigor of theoretical analysis.

- Finally, in this paper we are interested in examining pure gains introduced by network coding in MANETs. Replication strategies can be replaced by network coding, which provides a good chance for comparison. Transmission radius scaling techniques, however, are orthogonal to network coding, and should be studied separately.

We would like to point out that all the results discussed above are based on the implicit assumption that only storage-and-forwarding (without network coding) is allowed in each node, while in this paper we seek to understand whether network coding indeed affects the throughput-delay tradeoffs in MANETs.

B. Network Coding Applications in Wireless Networks

At the very beginning, research on network coding mainly focused on multicast scenario. In their pioneering theoretical work, Alswede et al. [6] showed that the min-cut throughput of a multicast session on a directed graph can be achieved, provided that one allows network coding, i.e., encoding at the intermediate nodes of the network. Conversely, it is generally not possible to achieve this communication rate if one allows only routing or copying packets at the intermediate nodes of the network. Shortly afterwards, Li, Yeung, and Cai [23] showed that it is sufficient for the encoding functions at the interior nodes to be linear. Subsequent work by Jaggi et al. [8] and Ho et al. [9] showed that the linear encoding functions can be designed randomly and independently at each node, which leads to a particular useful form of network coding, the RLC. Since RLC operates in a decentralized fashion [24], [25] which is extremely suitable for MANETs where centralized control is almost impossible or costly, we concentrate on RLC throughout the paper. Performance of RLC for local and global broadcast in wireless networks has been extensively studied in the literature, e.g., [7], [26]–[30].

For practically more important case of multiple unicast, we can only ascertain that it is totally different from multicast cases. For example, Li and Li conjectured in [31] that for undirected network with multiple unicast sessions, network coding does not help much on throughput. A deep understanding on achievable throughput for multiple unicast sessions in a network is still an open problem. In general, it is not clear whether network coding should be performed, and if it should, what the strategy must be [7]. One of our main contributions in this paper is that we analytically address this problem and show that, although RLC still cannot improve the order of throughput in MANETs, it changes the achievable throughput-delay tradeoffs significantly, which we believe will help improve our understanding of the theoretical limits on the benefits of network coding and on how to achieve them for MANETs with multiple unicast sessions.

The idea that, when RLC is allowed in intermediate nodes, compared to replication strategies [5], [22], larger throughput can be achieved with the same delay and smaller sizes of node buffers, was perhaps first explicitly developed in the work [32] by Zhang et al., where a simulation-based study of the benefit of RLC in one unicast communication is also presented. The recent work by Lin and Li [33] gives a rigorous analysis of this idea based on ordinary differential equations. In this work, Alswede et al. [6] showed that the min-cut throughput in MANETs, it changes the achievable throughput-delay tradeoffs significantly, which we believe will help improve our understanding of the theoretical limits on the benefits of network coding and on how to achieve them for MANETs with multiple unicast sessions.

The idea that, when RLC is allowed in intermediate nodes, compared to replication strategies [5], [22], larger throughput can be achieved with the same delay and smaller sizes of node buffers, was perhaps first explicitly developed in the work [32] by Zhang et al., where a simulation-based study of the benefit of RLC in one unicast communication is also presented. The recent work by Lin and Li [33] gives a rigorous analysis of this idea based on ordinary differential equations. In this work, Alswede et al. [6] showed that the min-cut throughput in MANETs, it changes the achievable throughput-delay tradeoffs significantly, which we believe will help improve our understanding of the theoretical limits on the benefits of network coding and on how to achieve them for MANETs with multiple unicast sessions.

The idea that, when RLC is allowed in intermediate nodes, compared to replication strategies [5], [22], larger throughput can be achieved with the same delay and smaller sizes of node buffers, was perhaps first explicitly developed in the work [32] by Zhang et al., where a simulation-based study of the benefit of RLC in one unicast communication is also presented. The recent work by Lin and Li [33] gives a rigorous analysis of this idea based on ordinary differential equations. In this work, Alswede et al. [6] showed that the min-cut throughput in MANETs, it changes the achievable throughput-delay tradeoffs significantly, which we believe will help improve our understanding of the theoretical limits on the benefits of network coding and on how to achieve them for MANETs with multiple unicast sessions.

The idea that, when RLC is allowed in intermediate nodes, compared to replication strategies [5], [22], larger throughput can be achieved with the same delay and smaller sizes of node buffers, was perhaps first explicitly developed in the work [32] by Zhang et al., where a simulation-based study of the benefit of RLC in one unicast communication is also presented. The recent work by Lin and Li [33] gives a rigorous analysis of this idea based on ordinary differential equations. In this work, Alswede et al. [6] showed that the min-cut throughput in MANETs, it changes the achievable throughput-delay tradeoffs significantly, which we believe will help improve our understanding of the theoretical limits on the benefits of network coding and on how to achieve them for MANETs with multiple unicast sessions.

The idea that, when RLC is allowed in intermediate nodes, compared to replication strategies [5], [22], larger throughput can be achieved with the same delay and smaller sizes of node buffers, was perhaps first explicitly developed in the work [32] by Zhang et al., where a simulation-based study of the benefit of RLC in one unicast communication is also presented. The recent work by Lin and Li [33] gives a rigorous analysis of this idea based on ordinary differential equations. In this work, Alswede et al. [6] showed that the min-cut throughput in MANETs, it changes the achievable throughput-delay tradeoffs significantly, which we believe will help improve our understanding of the theoretical limits on the benefits of network coding and on how to achieve them for MANETs with multiple unicast sessions.

The idea that, when RLC is allowed in intermediate nodes, compared to replication strategies [5], [22], larger throughput can be achieved with the same delay and smaller sizes of node buffers, was perhaps first explicitly developed in the work [32] by Zhang et al., where a simulation-based study of the benefit of RLC in one unicast communication is also presented. The recent work by Lin and Li [33] gives a rigorous analysis of this idea based on ordinary differential equations. In this work, Alswede et al. [6] showed that the min-cut throughput in MANETs, it changes the achievable throughput-delay tradeoffs significantly, which we believe will help improve our understanding of the theoretical limits on the benefits of network coding and on how to achieve them for MANETs with multiple unicast sessions.
in [32] and [33] is modeled with meeting times of any pair of nodes, to simplify the analysis. The problem is that, the most important feature of wireless transmission, i.e., interference, is not included in their modeling. It is nevertheless still reasonable for [32] and [33], since the authors are mainly interested in delay tolerant networks, where nodes are assumed to be sparsely distributed and interference from simultaneous transmissions can be ignored. However, it is obviously not suitable for the study of general MANETs.

- Second, the traffic pattern considered in our paper is more practical. The number of unicast sessions supported in this paper is \( \Theta(n) \), while only one unicast or broadcast session is assumed in [32] and [33].

- Next, only epidemic routing and its replacement of network coding are considered in [32] and [33], while in our work, several alternatives are considered and different algorithms are developed, which achieve throughput-delay pairs on different orders of \( n \). Therefore, we obtain a complete characterization of tradeoffs in MANETs.

- Most importantly, explicit expressions of network performance or tradeoffs are obtained in our paper for the first time, which provides insights on the degree of scalability of MANETs with network coding.

C. Scaling Laws of Wireless Networks with Network Coding

Scaling laws governing wireless networks with network coding have only been investigated in the limited scenarios in the literature recently. The delay gains and reliability benefit (measured in the reduced number of transmissions) of network coding in unreliable wireless networks were characterized in [34], [35] and [36], respectively. However, these results are for one multicast session with one-hop transmission or stable network topology. For multiple unicast scenario, Liu et al. [37], [38] and Keshavarz-Haddad et al. [39] showed that for static wireless networks, network coding and broadcasting at most provide a constant-factoried improvement in the throughput, compared to Gupta and Kumar’s \( \Theta(1/\sqrt{n \log n}) \) per-node throughput [3]. As far as we know, the scaling laws for throughput and delay are still unexploited for MANETs in the literature. More importantly, our results show that, network coding can provide significant improvement on network performance when mobility is utilized, which is impossible in static wireless networks [37]–[39]. We believe it reveals the intrinsic difference between MANETs and static wireless networks.

III. MANET MODELS AND DEFINITIONS

A. Network Models

**Random network model for MANETs:** Consider an ad hoc network where \( n \) nodes are distributed uniformly at random in a square area of \( A_n \). The square is assumed to be a torus\(^2\), i.e., the top and bottom edges are assumed to touch each other and similarly the left and right edges also are assumed to touch other. We consider a multiple \( (n) \) unicast scenario in which each node \( i \in \{1, 2, \ldots, n\} \) is a source node for one unicast session, and a destination node for another unicast session. Suppose that the source node \( i \) has data intended for destination node \( d(i) \). We assume that each source node has an infinite stream of packets to send to its destination.

\(^2\)We assume the torus to avoid edge effects, which otherwise complicates the analysis. We note, however, that the results in the paper hold for square, disk or any other shapes of practical interests.

The source-destination (S-D) association does not change with time, although the nodes themselves move.

**Mobility models:** The torus is divided into \( m = \Theta(n) \) square cells of area \( A_m/m \) each, resulting in a two-dimensional \( \sqrt{m} \times \sqrt{m} \) discrete torus\(^3\), see Fig. 1 for an illustration. The initial position of each node is equally likely to be any of the \( m \) possible cells independent of others. We assume the time is slotted and we study the following mobility models in this paper:

- **Fast mobility model (i.i.d. mobility model):** At each time slot, nodes randomly choose a new cell location independently and identically (i.i.d.) distributed over all cells in the network. This model captures the situation when mobile user moves so quickly that its position is almost independent from time to time. With this assumption, the network topology dramatically changes in every time slot, so that the network behavior cannot be predicted and fixed routing algorithms cannot be used. This mobility model is also used in [5], [14], [16], [19], [20].

- **Slow mobility model (random walk model):** Let a node be in cell \( (i, j) \in \{1, \ldots, \sqrt{m}\}^2 \) at time slot \( t \), then, at time slot \( t + 1 \), the node is equally likely to be in the same cell \( (i, j) \) or any of the four adjacent cells \( (i \pm 1, j), (i, j \pm 1), (i, j + 1) \), where addition and subtraction are modulo \( \sqrt{m} \). So each node in fact independently performs a simple random walk on the two-dimensional \( \sqrt{m} \times \sqrt{m} \) discrete torus. Note that this model implicitly sets an upper bound on the velocity of nodes as \( \sqrt{2A_m/m} \). Therefore, it is a suitable model for capturing real motion of nodes with slow mobility. Similar mobility model is also adopted in [4], [17]–[19].

**Model for successful transmission:** For characterizing the condition for a successful transmission, we adopt the protocol model as defined in [3]. We assume that all nodes use a common range \( r_c \) for their transmissions, and a transmission from node \( i \) to node \( j \) is successful if and only if \( d_{ij} \leq r_c \) and \( d_{kj} \geq (1 + \Delta)r_c \). For any other simultaneous transmitter, say node \( k \). Here, \( d_{ij} \) is the distance between nodes \( i \) and \( j \), and \( \Delta \) is a positive constant independent of \( n \). During a successful transmission, nodes send data at a constant rate of \( W \) bits per second. In the other common used model of successful transmission, namely, the physical model, a transmission is successful if the SINR is greater than some constant. It is well known [3], [4] that with a fading factor \( \alpha > 2 \), the protocol model is equivalent to the physical model. Therefore, we prefer the use of the protocol model in this paper for a cleaner presentation of the key ideas.

**Concurrently transmitting cells:** Now we define the transmission range and schedule. We choose \( r_c \) in such a way

\(^3\)For simplicity, assume \( \sqrt{m} \) is an integer.
B. Network Performance Metrics

Definition of throughput: A throughput $\lambda > 0$ is said to be feasible/achievable if every node can send at a rate of at least $\lambda$ bits per second to its chosen destination. We denote by $T(n)$, the maximum feasible throughput w.h.p. Given a scheme II, let $M_{II}(i, t)$ be the number of packets from source node $i$ that destination node $d(i)$ receives in $t$ timeslots under scheme II, for $1 \leq i \leq n$. Note that this could be a random quantity for a given realization of the network. Define the long term throughput of S-D pair $i$, denoted by $\lambda_{II}(n)$, to be

$$\lambda_{II}(n) = \liminf_{t \to \infty} \frac{M_{II}(i, t)}{t}.\]$$

Scheme II is said to have throughput $T_{II}(n)$ if

$$\lim_{n \to \infty} \mathbb{P} (\lambda_{II}(n) \geq T_{II}(n) \text{ for all } i) = 1.$$
When destination nodes receive packets from relays, they will first tell relays which packet they are looking for before the transmission begins (using the handshake).

- **Multi-hop relay with \(k_2\) replicas** is just another type of flooding scheme, which transmits \(k_2\) replicas of each packet, and places no constraints on the number of hops. At every timeslot in each active cell, the oldest packet will be selected to send to all nodes in the cell.

Now, we analyze the performance of the schemes described above. First of all, we give the lower bounds on delays under fast mobility model as follows [5].

**Theorem 1**: Under fast mobility model, \(D(n) = \Theta(\log n)\) for any scheme and \(D(n) = \Omega(\sqrt{n})\) for any 2-hop relay scheme.

**Remark 1**: To achieve the optimal throughput with the minimal delay given above, parameters like \(k_1\) and \(k_2\) in the proposed schemes should be carefully chosen under different mobility models. It has been shown in [5] that under fast mobility model, \(k_1 = \Theta(\sqrt{n})\) is enough for 2-hop relay scheme to achieve the minimal delay \(\Theta(\sqrt{n})\). Further increasing redundancy \(k_1\) will only reduce throughput without decreasing delay. Following the same argument, we have the optimal \(k_2 = \Theta(\log n)\) for flooding scheme. These lead to the following Theorem.

**Theorem 2**: Assuming infinite buffer space at each node, throughput-delay tradeoffs achieved by the three redundancy-based schemes proposed in [5] for MANETs under fast mobility model can be summarized in the following table.

<table>
<thead>
<tr>
<th>scheme</th>
<th>throughput</th>
<th>delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-hop relay without replicas</td>
<td>(\Theta(1))</td>
<td>(\Theta(n))</td>
</tr>
<tr>
<td>2-hop relay with (k_1) replicas</td>
<td>(\Theta(1/\sqrt{n}))</td>
<td>(\Theta(\sqrt{n}))</td>
</tr>
<tr>
<td>multi-hop relay with (k_2) replicas</td>
<td>(\Theta(1/n^2\log n))</td>
<td>(\Theta(\sqrt{n}\log n))</td>
</tr>
</tbody>
</table>

**Proof**: The proof of Theorems 1 and 2 is similar to the proof of Theorems 3, 6, 7 and 8 in [5], with minor differences caused by our use of the protocol model and cell scheduling scheme, ignoring queueing delays at source nodes. Due to space constraints, we do not repeat the proof here.

Next, we consider the throughput-delay tradeoffs under slow mobility model. We first show that the tradeoff with slow mobility is dramatically different from the one with fast mobility by the following theorem.

**Theorem 3**: Under slow mobility model, \(D(n) = \Omega(\sqrt{n})\) for any scheme and \(D(n) = \Omega(\sqrt{n}\log n)\) for any 2-hop relay scheme.

**Proof**: From random walk model, node speed is upper bounded by \(\sqrt{2A}/n = O(1)\) and the transmission range \(r_c = \Theta(1)\). Therefore, information propagation speed will be no larger than \(\Theta(1)\) per timeslot. It can be shown that the distance between the initial positions of S-D pair is \(\Omega(\sqrt{A}/n) = \Omega(\sqrt{n})\) w.h.p. [11]. Hence, the expected delay is at least \(\Omega(\sqrt{n})\) timeslots. For 2-hop relay cases, see our technical report [43] for a complete proof.

**Theorem 4**: Assuming infinite buffer space at each node, throughput-delay tradeoffs achieved by the three redundancy-based schemes proposed in [5] for MANETs under slow mobility model can be summarized in the following table.

<table>
<thead>
<tr>
<th>scheme</th>
<th>throughput</th>
<th>delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-hop relay without replicas</td>
<td>(\Theta(1))</td>
<td>(\Theta(n\log n))</td>
</tr>
<tr>
<td>2-hop relay with (k_1) replicas</td>
<td>(\Theta(1/\sqrt{n}\log n))</td>
<td>(\Theta(\sqrt{n}\log n))</td>
</tr>
<tr>
<td>multi-hop relay with (k_2) replicas</td>
<td>(\Theta(1/n^2\log n))</td>
<td>(\Theta(\sqrt{n}))</td>
</tr>
</tbody>
</table>

**Proof**: The proof for the performance of 2-hop relay without replicas can be found in [4], [17]. For the other two schemes are not reported in the literature. We complete the proof of this theorem in our technical report [43]. Note that the performance above is achieved with \(k_1 = \Theta(\sqrt{n}\log n)\) and \(k_2 = \Theta(\sqrt{n})\), respectively.

**V. THROUGHPUT-DELAY TRADEOFFS WITH NETWORK CODING: SCHEMES AND RESULTS**

We first review RLC used in our network coding based schemes. This bears exactly the same setup as in [25]. Then we describe the schemes developed for analyzing tradeoffs in MANETs with network coding, and identify scenarios in which RLC improves network performance of MANETs.

**A. Network Coding Operation**

Random linear coding (RLC) is applied to a finite set of \(k\) original packets (i.e., \(M = \{m_1, m_2, \ldots, m_k\}\)), that is called a generation. Each packet is viewed as an \(r\)-dimensional vector over a finite field, \(\mathbb{F}_q\), of size \(q\), i.e., \(m_i \in \mathbb{F}_q^r\), \(i = 1, 2, \ldots, k\). If the packet size is \(m\) bits, this can be done by viewing each packet as an \(r = [m/\log_q(q)]\)-dimensional vector over \(\mathbb{F}_q\) (instead of viewing each packet as an \(m\)-dimensional vector over binary field). Typically, \(\mathbb{F}_q\) (i.e., \(\mathbb{F}_{2^m}\)) is used. All the additions and multiplications in the following description are assumed to be over \(\mathbb{F}_q\). We assume that all the \(k\) packets in \(M\) are linearly independent. During the execution of a RLC based relay scheme, the destination node of \(M\) collects linear combinations of the packets in \(M\). Once there are \(k\) independent linear combinations at a node, it can recover all the original packets in \(M\) successfully.

Now, consider a certain timeslot \(t\). Let \(S_t(v)\) and \(S_{u}(v)\) denote the set of all the coded packets (each coded packet is a linear combination of the packets in \(M\)) at node \(v\) and \(u\), respectively, at the beginning of the timeslot \(t\). More precisely, if coded packet \(f_i \in S_t(v)\), where \(i = 1, 2, \ldots, |S_t(v)|\), then \(f_i \in \mathbb{F}_q^r\) has the form \(f_i = \sum_{j=1}^k a_{ij}m_j\), \(a_{ij} \in \mathbb{F}_q\). The scheme ensures that \(a_{ij}\)'s are known to node \(v\) by appending each packet \(f_i\) with a “code vector”, which will be explained a little later. Let \(S_t(v)\) and \(S_{u}(v)\) denote the subspaces spanned by the coded packets in \(S_t(v)\) and \(S_{u}(v)\), respectively. If \(S_t(v) \not\subseteq S_{u}(v)\), we say node \(v\) has useful information about \(M\) for \(u\). In timeslot \(t\), if node \(v\) is scheduled by the scheme to transmit a packet related to \(M\) to node \(u\), \(v\) first checks if it has useful information for \(u\). If so, \(v\) transmits a “random” coded packet with payload \(f_{\text{new}} \in \mathbb{F}_q^r\) to \(u\), where

\[
 f_{\text{new}} = \sum_{f_i \in S_t(v)} \beta_i f_i, \quad \beta_i \in \mathbb{F}_q, \quad \text{and} \quad \mathbb{P}(\beta_i = \beta) = \frac{1}{q}, \quad \forall \beta \in \mathbb{F}_q
\]

It is easy to check that \(f_{\text{new}}\) is still a linear combination of the \(k\) original packets, and can be written as \(f_{\text{new}} = \sum_{i=1}^k \theta_i f_i\), where \(\theta_i = \sum_{f_j \in S_t(v)} \beta_i \cdot a_{ij} \in \mathbb{F}_q^r\). For decoding purposes, the vector \((\theta_1, \theta_2, \ldots, \theta_k) \in \mathbb{F}_q^k\), called code vector, will be appended to \(f_{\text{new}}\), and sent as overhead. This overhead clearly requires a padding of additional \(k \log_2(q)\) bits. If the packet size \(m > \log_2(q)\), which would be the case under our constant packet size model, then the overhead required by the RLC based scheme can be ignored in our analysis.\(^3\)

\(^3\) More precisely, the constant packet size model for original packets means that the packet size scales as \(\Theta(\log n)\) bits, since it needs to carry the ID of the destination node with \(\Theta(\log n)\) bits. For a fair comparison, we require that \(k = O(\log n)\) for the coded packets throughout the paper. Therefore the overhead introduced by the code vector will not change the order of our results on \(T(n)\) and \(D(n)\) for RLC-based schemes.
We say that \( v \) sends an innovative coded packet \( f_{\text{new}} \) to \( u \), if \( f_{\text{new}} \) can increase the dimension of the subspace \( S_u(t^-) \), i.e., \( \dim(S_u(t^-)) \leq k \) in general and \( \dim(S_u(t^-)) = k \), node \( u \) can recover all the \( k \) original packets at once. We now recall the following key result about RLC, which says that \( f_{\text{new}} \) will be an innovative coded packet for \( u \) with probability no less than \( 1 - \frac{1}{q} \).

**Proposition 2:** (Lemma 2.1 in [25]) Let \( S_u(t^+ \cup \{f_{\text{new}}\} \) be the subspace spanned by the code vectors in \( u \) at the end of timeslot \( t \), i.e., after receiving a coded packet \( f_{\text{new}} \) from \( v \) according to the RLC based scheme described as above. Then,

\[
P\left( \dim(S_u(t^+)) > \dim(S_u(t^-)) \mid S_u(t^-) \nsubseteq S_u(t^-) \right) \geq 1 - \frac{1}{q}.
\]

**B. RLC-Based Relay Schemes**

In this subsection, we describe RLC-based relay schemes with different routing strategies, which will be used later to exploit throughput-delay tradeoffs in MANETs.

We first introduce the concept of a big generation. In what follows, when we say that the source node groups \( k = \omega(\log n) \) original packets into one big generation, we mean that separate \( k \) packets into \( k/\Theta(\log n) \) generations, each with \( \Theta(\log n) \) packets. When the destination node tries to decode one original packet, it first needs to collect \( \Theta(k) \) coded packets from the big generation, with \( \Theta(\log n) \) coded packets from each generation. Therefore the overhead introduced by RLC is ignorable in our analysis (cf. footnote 5).

**Schemes 1: 2-hop Relay with RLC**

1. \( k \) original packets in each source node will be grouped into one (big) generation. Each source will send \( m = (1+\epsilon)k \) coded packets for each generation, where \( \epsilon \) is a constant.

2. Coded packets for each generation will have the same timestamp \( t_p \). The value of \( t_p \) is the time the first coded packet of that generation leaves the source. All coded packets of a generation will be deleted from the relay buffer at the timeslot \( t \) if \( t - t_p > \theta_h \), where the threshold \( \theta_h \) depends on \( D(n) \) of the scheme and will be sufficiently larger than \( D(n) \).

3. Each cell becomes active once in every \( K^2 \) timeslots as discussed in Proposition 1. In an active cell, transmission is always between nodes within the same cell.

4. For an active cell with at least two nodes, a random transmitter-receiver pair is selected, with uniform probability over all possible node pairs in the cell. With probability \( 1/2 \), the transmitter is scheduled to operate in either “Source-to-Relay” or “Relay-to-Destination” mode, described as follows:

   - **Source-to-Relay Mode:** The transmitter sends a coded packet of its current generation, and does so upon every transmission opportunity while it is in source-to-relay mode until \( m \) coded packets have been delivered to distinct nodes. If all other nodes in the cell already have one coded packet for that generation, the source will begin to transmit coded packets from the next generation. Every node stores a single packet per S-D pair per generation. When the node receives a new packet, a relay linearly combines the incoming packet with the stored one, and replaces the stored packet with the result. Note that the nodes operate in broadcast mode, i.e., every node will hear every transmission in its range, and update the packet storage as described above.

   - **Relay-to-Destination Mode:** If the designated transmitter has a coded packet in its relay buffer for the destination node, and the rank of coded packets of that generation in the receiver is smaller than \( k \), the coded packet is transmitted to the designated receiver.

**Remark 2:** Since \( m > k \), we need a mechanism to stop unnecessary relay of coded packets of a generation when it is already decoded in the destination. Here we use a proactive stopping mechanism, i.e., the timestamp of each generation, since we can bound the delay of the scheme. In the analysis part presented later, we will show that \( k = \Theta(n) \), and \( D(n) \) for this scheme is also \( \Theta(n) \) for fast and slow mobility models. Therefore, \( \theta_h \) should be larger than \( \Theta(n) \). More complicated reactive stopping mechanisms (cf. [33] and the references therein) can be adopted to enhance the efficiency of the scheme in practice. However, we follow the simplest design for analytical tractability of the scheme.

**Schemes 2: Multi-hop Relay with RLC**

1. \( k \) original packets in each source node will be grouped into one (big) generation. Each source will send \( m = (1+\epsilon)k \) coded packets for each generation, where \( \epsilon \) is a constant.

2. Transmission is always between nodes within the same cell.

3. For an active cell with at least two nodes, perform the following: among all packets contained in at least one node of the cell and which have useful information for some other node in the same cell, choose the packet with the smallest generating time \( t_g \). If there are ties, choose the packet from the S-D pair \( i \) which maximizes \( (t_g + i) \mod n \). Transmit this packet to all other nodes in the cell. If the selected packet is in the source, then the source will transmit the linear combination of its \( k \) original packets of the same generation, instead of a particular packet belonging to that generation.

4. Every node stores a single packet per S-D pair per generation. When the node receives a new packet, a relay linearly combines the incoming packet with the stored one, and replaces the stored packet with the result.

**C. Main Results for RLC-Based Schemes**

In this subsection, we summarize the performance of the above schemes under different mobility models. Here, we
focus on the intuition and explanation of these results. Proofs of theses results will be given in the next Section.

**Theorem 5:** When 2-hop relay with RLC scheme is used and $k = \Theta(n)$, we have $T(n) = \Theta(1)$ and $D(n) = \Theta(n)$ for fast and slow mobility models.

**Remark 4:** Compare to Theorems 2 and 4, it is easy to see that, RLC provides delay improvement $\Theta(\log n)$ under slow mobility model. No gain is found under fast mobility model. It is not surprising, since 2-hop relay with RLC scheme is used to replace 2-hop relay without replicas, and we know that in the latter, there is no duplicated packets in order to maximize the throughput. Thus we cannot expect any gains when network coding is used. The gain $\Theta(\log n)$ of delay under slow mobility model comes from the lower information propagation speed, and the mixing of packets increase this speed by guaranteeing that every packet the destination received from relay nodes will contribute some information for the decoding of the packet from the same generation. For fast mobility model, this benefit vanishes since the information propagation speed is high enough, and the delay for waiting $k$ coded packets for decoding dominates the whole delay.

**Theorem 6:** When multi-hop relay with RLC scheme is used, under fast mobility model with $k = \Theta(\log n)$, we have $T(n) = \Theta(1/n)$ and $D(n) = \Theta(\log n)$. Under slow mobility model with $k = \Theta(\sqrt{n})$, we have $T(n) = \Theta(1/n)$ and $D(n) = \Theta(\sqrt{n})$.

**Remark 5:** Under fast and slow mobility models, multi-hop RLC-based schemes always provide significant gains compared to flooding schemes. We can see that the RLC-based scheme can achieve minimal delay, with an improved delay-constrained throughput. The intuition is that, when flooding is used, there exist enough opportunities to enhance performance by replacing replicas with more intelligent coding.

Fig. 3 compares timetables of 2-hop and multi-hop RLC-based relay schemes. It can be found that in 2-hop relay schemes, multiple sessions operate in a parallel fashion, while in multi-hop relay schemes, they operate in a sequential fashion. Therefore, at each timeslot, for 2-hop relay schemes, traffic pattern is still multiple unicasts. Recall our discussion in Section II-B, for multiple unicasts, we seldom find gains from network coding. While for multi-hop relay schemes, at each timeslot, traffic pattern looks more like one broadcast session, where gains from network coding are naturally expected.

**Remark 6:** Also notice that, multi-hop relay schemes can be divided into multiple phases, and in each phase, relaying for one generation from one S-D pair will dominate the network, which is in fact a type of information flooding in this phase (refer to Fig. 3(b) for illustration). The result is that in each phase, packets from one generation will be broadcasted to the whole network, and if the other $n-1$ nodes are receivers, they can all decode the original packets in that generation at the end of that phase. So it guarantees that multi-hop relay with RLC coding can support all-to-all traffic pattern ($n$ broadcast sessions) with the same performance. Note that this also means that the same network performance can be achieved for any $n$ multicast sessions (since receivers in this case are just a subset of receivers in the broadcast case). From Theorem 6, we can easily obtain the following corollary on the performance of multiple broadcasts and multicasts with network coding.

**Corollary 1:** For all-to-all communications or any multicasts with $n$ sources, when multi-hop relay with RLC scheme is used, under fast mobility model with $k = \Theta(\log n)$, we have $T(n) = \Theta(1/n)$ and $D(n) = \Theta(\log n)$. Under slow mobility model with $k = \Theta(\sqrt{n})$, we have $T(n) = \Theta(1/n)$ and $D(n) = \Theta(\sqrt{n})$.

In [30], Fragouli et al. designed an RLC-based scheme based on results from [25]. For all-to-all communications, they showed that their scheme achieves $T(n) = \Theta(1/n)$ and $D(n) = \Theta(n)$ under fast mobility model. Obviously, their scheme obtained the same throughput as ours at the cost of much larger delay. The basic idea of their scheme is that, $k$ packets from $k$ different sources will be grouped into one generation, and the relaying scheme is essentially the same as ours. The comparison here raises an interesting question—why in our RLC-based schemes we only mix packets from the same source? The reasons are the following: first of all, as shown in the above comparison, even for all-to-all communication scenarios, mixing packets from different sources is not a good choice. Second, for multiple unicast scenarios, we mix packets from different sources and these packets have different destinations. When one destination decodes a packet designated for another destination, this packet is in fact a duplicate at the first destination which will reduce the throughput. In our multi-hop relay with RLC, we also introduce redundancy for the same reason. However, the redundancy here is explicitly designed for decreasing the delay. While for the former case, it is purely a waste of network resource in multiple unicast scenarios. Finally, grouping packets from different sources requires coordinations. We are not sure about the cost for performing this coordination task, and we are interested in designing fully decentralized schemes, in which the operations from different nodes should be decoupled as much as possible.

**VI. THROUGHPUT-DELAY TRADEOFFS WITH NETWORK CODING: ANALYSIS**

In this section, we give outlines of proofs for the results on RLC-based relaying schemes discussed in the previous section. Intuitions behind these proofs are also provided.

**A. Preliminaries**

To facilitate the theoretical analysis, we need first investigate two critical delays for fast and slow mobility models: minimal delays for 2-hop relays and for flooding. Here, 2-hop relay represents any scheme with controlled redundancy on the number of hops (in the 2-hop relay case, the number of hops for each packet is 2, and other schemes with constant hop constrains will yield the critical delays on the same order of $n$), and flooding represents all schemes that remove this constraint totally.

Consider the following situation: initially, only one node’s color is red, which we call the source. All other nodes are blue. Whenever a source node encounters a blue node (in the same cell), the latter is colored red. The time for $\Theta(n)$ nodes to become red is called minimal 2-hop delay. If we change the rule slightly: whenever a red node encounters a blue node, the latter is colored red, then the corresponding time is named minimal flooding delay. Obviously, these two critical delays reflect the intrinsic properties of how mobility will facilitate information propagation. These two quantities
are scheme-independent, i.e., they hold for any scheme with or without replicas and with or without network coding.

For fast mobility model, the values of these two critical delays are available in the literature [5], and are included here for completeness.

**Lemma 1**: (Theorem 3 and Lemma 3 in [5]) The minimal 2-hop delay and the minimal flooding delay under fast mobility model are $\Theta(n)$ and $\Theta(\log n)$, respectively.

Next, we present the results for slow mobility model.

**Lemma 2**: The minimal 2-hop delay under slow mobility model is $\Theta(n)$.

**Proof**: Under slow mobility model, the joint position of two nodes due to independent random walks can be viewed as a difference random walk relative to the position of one node. Then the inter-meeting times are just the inter-visit times of cell $(1,1)$ for the difference random walk on a $\sqrt{n} \times \sqrt{n}$ torus. Let $\tau$ be the random variable representing the inter-meeting time defined as above. El Gamal et al. prove the following lemma in [17].

**Lemma 3**: $E[\tau] = n$ and $E[\tau^2] = \Theta(n^2 \log n)$. Let $N$ be the number of distinct nodes the red node has met in $n$ timeslots. Based on above results, we can obtain that $E[N] = (1 - e)n$, where $0 < e < 1$ is a constant, and $\sigma_N = O(n \log n)$. By Chebyshev inequality, for any $0 < \kappa < 1$, $P\{N \leq (1 - \kappa)E[N]\} \leq \frac{\sigma_N}{\kappa^2E[N]^2} = O(\frac{\log n}{n}) \rightarrow 0$, which means that $N = \Theta(n)$ w.h.p.

**Lemma 4**: The minimal flooding delay under slow mobility model is $\Theta(\sqrt{n})$.

**Proof**: Note that, Theorem 3 already shows that no scheme will obtain a delay better than $\Theta(\sqrt{n})$ under slow mobility model. We need just to show that this is achievable using flooding. We cite the following important result about rumor spreading on torus: Theorem 3 in [44] states that the following result is true, at timeslot $t$, there exists a sub-torus of size $\sqrt{k} \times \sqrt{k}$, where for each cell in this sub-torus, there exists at least one red node. Therefore, in $\Theta(\sqrt{n})$ timeslots, we can cover the whole torus of size $\sqrt{n} \times \sqrt{n}$ w.h.p.

The following lemma is useful in delay analysis, since it confirms that the effect of transmission scheduling only contributes a constant factor, which can be ignored in asymptotic analysis. Therefore, the time for two desired nodes to meet will dominate the delay of the scheme.

**Lemma 5**: In the schemes mentioned above, every node will be scheduled to transmit or receive a packet with a constant, non-vanishing probability that is independent of $n$.

**Proof**: This result can be obtained from Proposition 1. It only depends on the steady state node location distribution. Note that fast and slow mobility models have the same node location distributions in the steady state. Therefore, this result applies to both mobility models.

**B. Proof for Main Results**

**Proof outline for 2-hop relay with RLC (Theorem 5)**

We first prove the case for fast mobility model. Obviously, if we can prove that in $N = \Theta(n)$ timeslots, the destination can receive $\Theta(n)$ coded packets, then based on Proposition 2 the destination has enough coded packets to recover $k = \Theta(n)$ original packets w.h.p. Therefore, delay is upper bounded by $O(n)$ and the throughput is $T(n) = O(1)$.

From the description of the scheme, we know the source will send $m = \Theta(n)$ coded packets to the network. However, the destination may get $k'$ packets, which is fewer than $m$ packets for the following reasons:

1. The source node only delivers coded packets to $m_1 < m$ different nodes acting as relays in $N$ timeslots;
2. The Destination node only meets $m_2 < m$ relay nodes that have useful information about the generation the destination wants to decode in $N$ timeslots;
3. Furthermore, $k' < m_2$ since when the destination meets a relay, it will not always be scheduled to receive a packet from the relay. However, based on Lemma 5, the difference between $k'$ and $m_2$ caused by the above reasons is upper-bounded by a constant factor and we can assume that $\Theta(k') = \Theta(m_2)$.

Obviously, if there are $\Theta(n)$ nodes in the network which have useful information for the destination, the probability for the destination to meet a relay in each timeslot is a constant value, i.e., will not scale down with $m$. Therefore, $P(k' = \Theta(n)|m_1 = \Theta(n)) = 1$. Thus, we need to prove that $m_1 = \Theta(n)$, which is established in Lemma 1). Due to space constraints, we do not repeat the proof here.

The proof for the slow mobility case is similar to the argument above, and the differences are given as follows: Recall that Lemma 2 already shows that after $N_1 = \Theta(n)$ timeslots, $m_1 = \Theta(n)$. From [45], we know that the mixing time of a simple random walk on a $\sqrt{n} \times \sqrt{n}$ torus is also $\Theta(n)$. Therefore, there exist a constant $\epsilon$ such that after $N_2 = \epsilon n$ timeslots, these $m_1$ nodes with coded packets are uniformly distributed in the torus w.h.p. which means that each node in the network has coded packets with a constant probability. Instead of collecting coded packets as soon as possible, the destination nodes begin to collect packets after $N_1 + N_2$ timeslots. It can be proved that the timeslots $N_3$ required to collect $n$ coded packets is still $\Theta(n)$. Therefore the total delay $N = N_1 + N_2 + N_3 = \Theta(n)$ w.h.p.

In both mobility models, each source node sends $m = \Theta(n)$ coded packets for each big generation, and each big generation has $\Theta(n)$ original packets, then each coded packet contains $\Theta(1)$ information of original packets. Because every coded packet is transmitted twice, we have $T(n) = \Theta(1)$.

**Proof for multi-hop relay with RLC scheme (Theorem 6)**

The central problem here is still the following: can destination node get $\Theta(k)$ coded packets within $\Theta(k)$ timeslots? If it is the case, based on Proposition 2 the destination has enough coded packets to recover $k$ original packets w.h.p. Then the delay is upper bounded by $O(k)$. Replacing $k$ with $\Theta(\log n)$ and $\Theta(\sqrt{n})$ for fast and slow mobility cases, respectively, we obtain the results on delays. Since we get $k$ original packets in $\Theta(k)$ timeslots, the throughput for the phase when the transmissions of this $S$-P pair dominate the network is $\Theta(1)$. For fairness embedded in the scheme, this situation happens once for $\Omega(1/n)$ phases. Therefore, long-term throughput $T(n) = O(1)$, which completes the proof.

Next, we concentrate on an equivalent problem: how many timeslots do we need in order to receive at least $\Theta(k)$ coded packets at the destination? We denote it as $N$. Obviously, $E[N] \leq E[S_1] + E[S_2]$, where $S_1$ and $S_2$ respectively represent the timeslots required for $\Theta(n)$ nodes in the network to have one coded packet for that generation, and the time required for the destination to receive $\Theta(k)$ packets given that other $\Theta(n)$ nodes hold coded packets.

Note that Lemma 1 and Lemma 4 establish that $E[S_1] = \Theta(\log n)$ and $E[S_2] = \Theta(\sqrt{n})$ for fast and slow mobility models, respectively. They both agree with $\Theta(k)$ in respective schemes. In fact, it does not happen by coincidence but by...
how we choose the parameter for RLC. Thus we obtain $E[S_1] = \Theta(k)$. For $S_2$, since $\Theta(n)$ nodes in the network have the coded packet, every transmission happened in the destination’s cell will help obtain a coded packet w.h.p., which leads to $E[S_2] = \Theta(k)$. Summing $E[S_1]$ and $E[S_2]$ proves the result that $E[N] \leq \Theta(k)$. See our technical report [43] for detailed proof.

VII. CONCLUDING REMARKS

In this paper, we characterize the throughput-delay tradeoffs in mobile ad hoc networks (MANETs) with network coding, and compares with the scenarios where only replication and forwarding are allowed in each node. The schemes/protocols achieved in this paper are effective and decentralized way are proposed and the optimality of the tradeoffs is established. The scenarios in which network coding provides significant improvement on network performance are identified under different node mobility patterns (fast and slow mobility). The insights on when and how information mixing is beneficial for MANETs with multiple unicast and multicast sessions are provided.

REFERENCES


