Wireless Netw (2012) 18:679–695
DOI 10.1007/s11276-012-0427-2

The fundamental limits of broadcasting in dense wireless mobile networks

Giovanni Resta · Paolo Santi

Published online: 8 March 2012

Abstract In this paper, we investigate the fundamental properties of broadcasting in *mobile* wireless networks. In particular, we characterize broadcast capacity and latency of a mobile network, subject to the condition that the stationary node spatial distribution generated by the mobility model is uniform. We first study the intrinsic properties of broadcasting, and present the RippleCast broadcasting scheme that simultaneously achieves asymptotically optimal broadcast capacity and latency, subject to a weak upper bound on maximum node velocity and under the assumption of static broadcast source. We then extend RippleCast with the novel notion of center-casting, and prove that asymptotically optimal broadcast capacity and latency can be achieved also when the broadcast source is mobile. This study intendedly ignores the burden related to the selection of broadcast relay nodes within the mobile network, and shows that optimal broadcasting in mobile networks is, in principle, possible. We then investigate the broadcasting problem when the relay selection burden is taken into account, and present a combined distributed leader election and broadcasting scheme achieving a broadcast capacity and latency which is within a $\Theta((\log n)^{1+\frac{2}{\alpha}})$ factor from optimal, where n is the number of mobile nodes and $\alpha > 2$ is the path loss exponent. However, this result holds only under the assumption that the upper bound on node velocity converges to zero (although with a very slow, poly-logarithmic rate) as n grows to infinity.

G. Resta · P. Santi (✉)
IIT-CNR, Via G. Moruzzi 1, 56124 Pisa, Italy
e-mail: paolo.santi@iit.cnr.it

G. Resta
e-mail: giovanni.resta@iit.cnr.it

Keywords Wireless networks · Mobile networks · Broadcast capacity · Broadcast latency SINR interference model

1 Introduction

Investigation of fundamental properties of wireless networks has received considerable attention in the research community, starting from the seminal Gupta and Kumar [7] work that characterized the capacity of a wireless multi-hop network for unicast transmissions. Since then, fundamental properties of wireless multi-hop networks have been investigated for a variety of communication patterns including unicast [6, 18, 20, 26], broadcast [9, 22, 28], multicast [14, 25], and convergecast [15, 16]. It has been shown that wireless multi-hop network scaling laws significantly change depending on network parameters such as node deployment (e.g., random vs. arbitrary), traffic pattern, and node mobility. Node mobility in particular has been shown to have considerable effects on wireless network scaling laws: for instance, per-node capacity of unicast transmission has been shown to be asymptotically vanishing with the number n of network nodes independently of the node deployment (see [7]), but to become *constant* (i.e., asymptotically optimal) in case network nodes are mobile [6] (under the assumption that very large delays in packet delivery can be tolerated). The reason of the beneficial effect of node mobility on per-node capacity is that what limits per-node unicast capacity in a static wireless multi-hop network is the relaying burden, i.e., the fact that the same packet has to be sent several times before it can reach the destination.[1] If nodes are mobile, the relay burden can be avoided (or at least significantly reduced) by exploiting a "wait and deliver" strategy:[2] since nodes move randomly, there is a high probability that the sender and the destination eventually come into each others reach, and the packet can be delivered to the destination with no (or only few) retransmission(s).

To the best of our knowledge, none of the existing papers have investigated the effect of mobility on *broadcasting* scaling laws. Broadcasting scaling laws have been recently characterized in a series of papers [9, 28], including our work [22, 23] showing that, contrary to what happens for unicast transmission, asymptotically optimal capacity *and* latency can be achieved simultaneously for broadcast communication. However, all these results are based on the assumption that network nodes are static. An implicit consequence of this assumption is that the communication burden induced by the need of selecting broadcast relaying nodes within the network (called the *coordination burden* in the following) is consistently ignored in the analysis. This is acceptable in a static network, since the selection of broadcast relaying nodes can be assumed to be done once and for all at the beginning of the broadcasting session, implying that the coordination burden can be safely ignored in the analysis as long as the duration of the broadcasting session is sufficiently long. However, if relay nodes are mobile, a change in their position might cause an incomplete coverage of the broadcast packets, which must be received by *all* network nodes. Thus, the role of broadcast relay node must be continuously rotated amongst network nodes in a mobile network, in order to ensure broadcast coverage in spite of node mobility. Given this, evaluating the coordination burden cost becomes an integral part of the characterization of broadcasting scaling laws in mobile networks.

Note that, when mobility comes into play, the issue of asymptotic node density under which broadcasting scaling laws are investigated becomes relevant. In fact, in case of static networks, broadcast can be successfully completed only if the graph representing all possible communication links in the network is connected. On the other hand, this requirement is no longer needed in case of mobile networks: even if the network is never connected at any specific instant of time, broadcasting can be completed exploiting a "wait and deliver" strategy similar to that proposed by Grossglauser and Tse for unicast transmissions [6]. In general, two scenarios can be considered when approaching the study of broadcasting scaling laws in mobile networks: the *sparse* and the *dense* scenario[3]. In the sparse scenario, node density is not sufficient to ensure full network connectivity at any instant of time, and broadcasting can be achieved only through a "wait and deliver" strategy enabled by node mobility. Conversely, in the dense scenario node density is sufficient to ensure full network connectivity at any instant of time, and node mobility is no longer *necessary* for completing the broadcasting task. Clearly, in the sparse scenario the speed of propagation of broadcast packets within the network is dominated by the physical node velocity, which is several order of magnitudes smaller than the speed of propagation of packets in the air (this is true also when MAC layer processing time is considered). Hence, the only type of broadcasting possible in sparse mobile networks is one in which very large latencies can be tolerated, i.e., a *delay tolerant* broadcast. Scaling laws of broadcast latency in delay tolerant networks have been studied, e.g., in [8, 11]. To the best of our knowledge, ours is the first study in the literature investigating broadcast capacity and latency scaling laws in *dense* mobile networks. More specifically, our goal in this paper is to gain a better understanding of the effect of mobility on the broadcasting communication paradigm in a dense network, in order to understand whether, e.g., broadcasting of multimedia or real-time information is still possible in a dense, mobile network environment.

We first show that *broadcasting is not inherently capacity nor latency limited by node mobility*: we present a simple cell-based broadcasting scheme, called RippleCast, that simultaneously achieves optimal broadcast capacity and latency under the assumption that: (1) the broadcast source is static; (2) nodes move in a bounded region according to a mobility model whose stationary node spatial distribution is uniform; and (3) maximum node velocity is upper bounded by a (very large) constant. We then extend this result to the case of mobile broadcast source, by combining RippleCast with the novel notion of center-cast. However, when the cost related to the coordination burden is taken into account the picture changes considerably: broadcasting capacity and latency degrades by a factor $\Theta((\log n)^{1+\frac{2}{\alpha}})$ with respect to optimal—n is the number of network nodes and $\alpha > 2$ is the path loss exponent-, and the upper bound on maximum node velocity becomes asymptotically vanishing as $n \to \infty$. We thus formally prove that what limits broadcast performance in a dense, mobile network *are not* the inherent properties of broadcast communication, but the *coordination burden*

[1] This is true unless the destination is the vicinity of the sender, which occurs with vanishingly probability in a sufficiently large network with randomly selected source/destination pairs.

[2] This strategy has become the fundamental communication paradigm in delay tolerant networks [4].

[3] A formal definition of sparse and dense mobile networks will be given in Sect. 3.

induced by the need of frequent re-selection of relay nodes within the network.

The rest of this paper is organized as follows. In Sect. 2, we survey and critically discuss related work. In Sect. 3, we introduce the network model and some preliminary definition, including formal definitions of broadcast capacity and latency in a dense, mobile wireless network. In Sect. 4, we present a trivial upper bound on broadcast capacity, and a less trivial lower bound on broadcast latency for dense, mobile wireless networks. We then proceed in Sect. 5 presenting the first technical contribution of this paper, namely a broadcast algorithm called RippleCast, which is shown to simultaneously achieve asymptotically optimal broadcast capacity and latency under the assumption that broadcast relay nodes are "magically" selected. In Sect. 6, we re-visit the result presented in the previous section by explicitly taking into account the communication burden produced by the broadcast relay nodes selection process. The results presented in Sects. 5 and 6 hold under the assumption that the broadcast source is a static node. In Sect. 7, we relax this assumption, and show that, if a suitable upper bound on source velocity holds, asymptotically optimal broadcast capacity and latency can still be achieved (under the assumption of "magically" selected relay nodes) by combining RippleCast with the novel notion of center-cast. Finally, Sect. 8 presents some final considerations and possible ways of extending our work.

2 Related work

The fundamental properties of broadcasting in wireless multi-hop networks have been investigated only very recently. In [28], Zheng investigated the broadcast capacity of random networks with single broadcast source under the generalized physical interference model, and presented a broadcast scheme providing asymptotically optimal capacity. The author also presented a different broadcast scheme, and proved its asymptotically optimal performance with respect to information diffusion rate, which is closely related to latency. The authors of [9] confirmed that optimal broadcast capacity can be achieved in a more general network model, in which arbitrary node positions are allowed, an arbitrary subset of the network nodes is assumed to generate broadcast packets, and accurate SINR-based interference models are used. In [22], we have shown that asymptotically optimal broadcast capacity and latency can be simultaneously achieved in a static network, under the assumption of single broadcast source. This result has been recently extended to the case of an arbitrary number of broadcast sources in [23].

While several papers have proposed broadcasting schemes for mobile networks (see, e.g., [19, 21]), to the best of our knowledge none of them attempted at characterizing the fundamental properties of broadcasting in mobile networks. The work that is closest to our is [2], where the authors present a location-based broadcasting protocol for mobile ad hoc networks, and formally characterize the number of communication steps needed to deliver a broadcast packet to all network nodes. Similarly to our approach, the authors propose selecting broadcast relay nodes based on their position, and present theoretical results that hold under the assumption that node velocity is upper bounded by certain constants. However, the authors in [2] are concerned with delivering a *single* broadcast packet, while in this paper we are interested in characterizing the maximum *rate* at which broadcast packets can be sent by the source. Furthermore, the results of [2] are valid under a simplistic interference model based on the notion of conflict graph, while ours hold under the more realistic, SINR-based physical interference model.

A related area of research is that investigating the speed of information propagation in sparse, mobile networks. For instance, in [11] the authors consider a network in which nodes move according to i.i.d. mobility and Brownian motion models, and showed that if node density is not sufficient to ensure full network connectivity (sparse network), the latency in delivering packets scales linearly with the Euclidean distance between the sender and the receiver, while it scales sub-linearly in case node density is sufficient to ensure full network connectivity (dense network). Note that, although with a different network model, our results about broadcasting latency confirm the findings of [11]: we in fact prove that broadcasting latency is sub-linear—more specifically, $\Theta\left(\sqrt{\frac{n}{\log n}}\right)$—in a dense, mobile network with n nodes. The study reported in [8] considers sparse mobile networks, and prove results similar to those presented in [11] using more general mobility models and providing accurate upper bounds on information propagation speed within the network. Differently from our work, the studies reported in [8, 11] consider only latency in packet delivery, in a scenario in which a *single* packet is generated and propagated within the network. Furthermore, the focus in these works is mostly [11] or entirely [8] on the sparse mobile network scenario.

Summarizing, to the best of our knowledge this paper is the first studying broadcast scaling laws in terms of both capacity and latency in a dense, mobile network.

3 Network model and preliminaries

We consider a wireless network composed of $n+1$ wireless nodes distributed in a square region R of side $L = L(n)$. One of the nodes is stationary, and is located in the center of the deployment region. This node, denoted s in the following, is the broadcast source. The remaining n nodes are mobile, and move within R according to some continuous-time mobility model $\mathcal{M}$. Model $\mathcal{M}$ is such that the induced stationary node spatial distribution (which is assumed to exist) is *uniform*. In other words, a snapshot taken at time t of the positions of n nodes moving according to $\mathcal{M}$, for a sufficiently large t, is statistically equivalent to a uniform random distribution of n nodes into R. Examples of mobility models satisfying this assumption are random walks, Brownian motion, random direction model with proper border rules, etc (see [13] and references therein).

We assume nodes communicate through a shared wireless channel of a certain, constant capacity W, and that the nodes transmission power is fixed to some value P. Correct message reception at a receiver node is subject to an SINR-based criterion, also known as *physical interference model* [7]. More specifically, a packet sent by node u is correctly received at a node v (with rate W) if and only if

$$\frac{P_v(u)}{N + \sum_{i\in\mathcal{T}} P_v(i)} \geq \beta,$$

where N is the background noise, β is the capture threshold, $\mathcal{T}$ is the set of nodes transmitting concurrently with node u, and $P_v(x)$ is the received power at node v of the signal transmitted by node x.

We also make the standard assumption that radio signal propagation obeys the log-distance path loss model, according to which the received signal strength at distance d from the transmitter (for sufficiently large d, say, $d \geq 1$) equals $P \cdot d^{-\alpha}$, where α is the path loss exponent. In the following, we make the standard assumption that $\alpha > 2$, which is often the case in practice. We then have[4] $P_v(x) = P \cdot d(x,v)^{-\alpha}$, where $d(x, v)$ is the Euclidean distance between nodes v and x, and the SINR value at node v can be rewritten as follows

$$SINR(v) = \frac{d(u,v)^{-\alpha}}{\frac{N}{P} + \sum_{i\in\mathcal{T}} d(i,v)^{-\alpha}}.$$

For given values of P, β, α, and N, we define the transmission range r_{max} of a node as the maximum distance up to which a receiver can successfully receive a message *in absence of interference*. From the definition of physical interference model, we have $r_{max} = \sqrt[\alpha]{P/(\beta N)}$.

The *maximal communication graph at time t* is a graph $G(t) = (\mathcal{V}, \mathcal{E}(t))$ representing all possible communication links in the network at time t; i.e., $\mathcal{V}$ is the set of the $n + 1$ nodes, and (undirected) edge $(u,v) \in \mathcal{E}(t)$ if and only if $d(u, v, t) \leq r_{max}$, where $d(u, v, t)$ is the Euclidean distance between u and v at time t. Given that existence of a link in $G(t)$ depends only on distance between nodes, graph $G(t)$ is equivalent to a unit disk graph, which has well-known limitations in modeling wireless networks [12]. However, up to straightforward technical details, the results presented in this paper can be extended to the more realistic cost-based radio propagation model of [24], which is shown to closely resemble log-normal shadowing propagation.

We define *sparse* and *dense* mobile networks depending on the asymptotic properties of graph $G(t)$. More specifically, assume t is large enough so that the node spatial distribution converged to the stationary distribution of mobility model $\mathcal{M}$. We say that the mobile network is *dense* if and only if $Prob(G(t) \text{is connected}) \to 1$ as $n \to \infty$; conversely, we say that the mobile network is *sparse* if and only if $Prob(G(t) \text{is connected}) \to 0$ as $n \to \infty$. Unless otherwise stated, in the following we restrict our attention to the case of dense, mobile networks.

We define the *broadcast capacity* of the network as the maximum possible rate $\lambda(n)$ such that all packets generated by source s are received by the remaining n nodes within a certain time T_{max}, with $T_{max} < \infty$. The *broadcast latency* of the network is the *minimal* time $T(n)$ such that the packet generated by s at time t is received by *all* the n nodes within time $t + T(n)$. Given our focus on dense, mobile networks, the assumption of connected maximal communication graph is made throughout this paper. More specifically, we assume that graph $G(t)$ is connected w.h.p. under the assumption that nodes are distributed according to the asymptotic node spatial distribution resulting from mobility model $\mathcal{M}$ which, we recall, is assumed to be uniform.[5]

Given the assumption of stationary uniform node spatial distribution of the mobility model $\mathcal{M}$, the *critical transmission range* for connectivity of graph $G(t)$ is [3]:

$$ctr(n) = \Theta\left(L(n)\sqrt{\frac{\log n}{n}}\right).$$

We recall that the critical transmission range for connectivity is the minimal common value of the transmission range such that the resulting maximal communication graph is connected.

Assume the deployment region R is divided into non-overlapping square cells of side l, with $l = \frac{r_{max}}{2h\sqrt{2}}$, for some constant $h > 1$. In turn, each of these cells is partitioned into 9 square mini-cells of side $\frac{l}{3}$ (see Fig. 1). The following proposition defines a value of $L(n)$ such that several properties of the resulting node deployment hold, w.h.p.

[4] To simplify notation, in the following we assume that the product of the transmitter and receiver antenna gain is 1.

[5] Given the probabilistic characterization of mobile node positions assumed in this paper, most of the properties proved in this paper hold with high probability (w.h.p.), i.e., with probability at least $1 - \frac{1}{n}$.

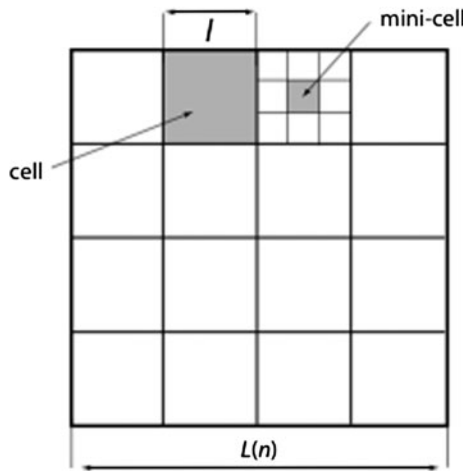

Fig. 1 Cell subdivision of the deployment region

Proposition 1 *Assume* $L(n) = \frac{r_{max}}{6h\sqrt{2}}\sqrt{\frac{n}{\log n}}$ *for some constant* $h > 1$, *and assume n nodes are distributed uniformly at random in a square region of side* $L(n)$. *Then, the following properties hold w.h.p.:*

(a) *the minimally occupied mini-cell contains at least one node;*
(b) *the maximally occupied mini-cell contains* $\Theta(\log n)$ *nodes;*
(c) *the maximum transmission range* r_{max} *is asymptotically minimal to ensure network connectivity.*

Proof To prove (*a*), we observe that when $L(n) = \frac{r_{max}}{6h\sqrt{2}}\sqrt{\frac{n}{\log n}}$, the total number C of mini-cells in the deployment region is

$$C = \left(\frac{L(n)}{l/3}\right)^2 = \left(\frac{\frac{r_{max}}{6h\sqrt{2}}\sqrt{\frac{n}{\log n}}}{\frac{r_{max}}{6h\sqrt{2}}}\right)^2 = \frac{n}{\log n}.$$

It follows that the ratio η between the number of nodes and the number of cells is $\log n$. Theorem 5, page 111 of [10] states that, when $\eta = \log n$, the number of nodes in the minimally occupied cell is greater than zero w.h.p., which implies the result when $L(n) = \frac{r_{max}}{6\sqrt{2}}\sqrt{\frac{n}{\log n}}$.

The proof of (*b*) follows directly from Lemma 1 of [15].

The proof of (*c*) follows by observing that the critical transmission range for connectivity when n nodes are distributed uniformly at random in a square of side $L(n)$ is [3]

$$\Theta = \left(L(n) \cdot \sqrt{\frac{\log n}{n}}\right) = \left(\frac{r_{max}}{6h\sqrt{2}} \cdot \sqrt{\frac{n \log n}{n \log n}}\right) = \Theta(r_{max}).$$

Finally, we introduce the notion of cell distance, which will be extensively used in the following. Given any two cells A and B in the deployment region, the cell distance between A and B, denoted $d(A, B)$, is the minimum number of adjacent cells (horizontal, vertical, and diagonal adjacency) that must be traversed to reach A starting from B (and viceversa). □

4 Bounds on broadcast capacity and latency

The following upper bound on the broadcast capacity trivially follows by observing that the maximum rate at which any receiver can receive broadcast packets is W [9]. The bound holds for an arbitrary network.

Claim In any network with n nodes, we have $\lambda(n) \leq W$.

Define $D(n)$, the *diameter* of the network (relative to the broadcast source), as the maximum Euclidean distance between a network node u and the source s. Given that nodes are mobile, the diameter of the network changes over time. However, Proposition 1 implies an invariant property of network diameter under our deployment assumptions, as stated in the following proposition:

Proposition 2 *Let* $D(n, t)$ *be the network diameter at time t. If t is sufficiently large,* $L(n) = \frac{r_{max}}{6h\sqrt{2}}\sqrt{\frac{n}{\log n}}$ *for some constant* $h > 1$, *n nodes move according to a mobility model with stationary uniform node spatial distribution in a square region of side* $L(n)$, *and the source node is located in the center of the deployment region, then* $D(n) \geq \frac{\sqrt{2}}{2}(L(n) - \frac{2}{3}l) = \Omega(L(n))$, *w.h.p.*

Proof The proof follows immediately by observing that, by Proposition 1, every mini-cell in the deployment region (and in particular those at the corners) contains at least one node, w.h.p. □

We are now ready to prove a lower bound for broadcast latency in mobile networks, subject to an upper bound on node velocity.

Theorem 1 *Suppose the same assumptions of Proposition 2 hold, and the maximum node velocity is* $\tilde{v} = \frac{r_{max}}{\tau}$, *where* τ *is the (constant) time required to send and correctly receive a packet. Then, the broadcast latency is* $\Omega\left(\sqrt{\frac{n}{\log n}}\right)$, *w.h.p.*

Proof By Proposition 2, the packet generated by the source at time t has to travel distance at least $\frac{\sqrt{2}}{2}(L(n) - \frac{2}{3}l)$, w.h.p., to reach the nodes that were in the corner mini-cells at time t. Consider one such node u, and consider the segment $\overline{us}$ connecting u to s. Since the progress of the packet generated at time t towards node u is at most r_{max} at each communication step of duration τ, and node u in the best case travels along $\overline{us}$ directed towards s with speed at most $\tilde{v} = \frac{r_{max}}{\tau}$, it is easy to see that at least $\frac{\sqrt{2}}{4}(L(n) - \frac{2}{3}l)$ communication steps (each of duration τ) are required for the packet to reach node u. Observing that τ is a constant, we can conclude that $T(n) = \Omega(L(n)) = \Omega\left(\sqrt{\frac{n}{\log n}}\right)$. □

Notice that the upper bound $\tilde{v}$ on node velocity is comparable to the speed of radio signal propagation in the air; even considering MAC processing time, this speed is very large: it is about $4.76 \cdot 10^6$ m/s with typical values of IEEE 802.11a/g technology (see end of Sect. 5).

5 Matching capacity and latency bounds

In this section we present a broadcasting algorithm achieving asymptotically optimal capacity and latency bounds in mobile networks, under the assumption that broadcast relaying nodes are somewhat magically selected within the network. This assumption, although admittedly not realistic, is made with the purpose of separately studying the fundamental properties of *broadcasting* in mobile networks from those of *electing leaders* (i.e., relay nodes). While using specific relay nodes to forward broadcast packets is indeed the most common approach to broadcasting, strictly speaking leader election is a separate task from broadcasting, which in principle can be achieved also without explicit leader election (e.g., through cooperative communication).

5.1 Algorithm overview

While broadcasting in mobile networks is apparently a very complex task due to mobility of individual nodes, this apparent complexity can be tamed by observing that the identity of a specific node within the network is not relevant to a broadcasting scheme, as long as reception of each broadcast packet by each of the (mobile) nodes can be guaranteed. In other words, what is relevant to a broadcasting scheme *is not the identity of a node*, but *its position within the network*. Thus, instead of selecting specific nodes to relay broadcast packets, a smart broadcasting scheme for mobile networks should focus on invariant properties of the node spatial distribution generated by the mobility model, and use such properties to select relay nodes based on their location within the network.

The broadcasting scheme, which we call RippleCast, is based on the following assumptions:

- a spatial TDMA approach is assumed at the MAC layer: time is divided into transmission slots, and a carefully chosen set of links (transmission set) is activated in each slot. The duration of a slot is sufficient to transmit a packet from the sender to the receiver, including propagation time;
- the deployment region is divided into cells and mini-cells, as described in Sect. 3. Cell subdivision is used to virtualize the broadcasting task from a node-related process to a cell-related process. In particular, broadcast relaying nodes (leaders) are chosen within the *central* mini-cell of each cell, and the broadcasting process becomes one of propagating broadcast packets between cells. Without loss of generality, we assume that the source node s is in the central cell.

RippleCast is based on a cell coloring scheme, as in Fig. 2, composed of a constant number $\bar{k}^2$ of colors, which is used to spatially separate simultaneously active transmissions. In particular, the coloring scheme ensures that, under the assumption that at most one transmitter is active in each cell with the same color, all transmitted packets are correctly received by *all* the nodes located in the cells adjacent to the transmitter cell. A *round* of transmission is composed of $\bar{k}^2$ transmission slots, one for each color. The color of a cell A is denoted $col(A)$ in the following. Similarly, $col(u)$ denotes the color of the cell to which node u belongs. With RippleCast, propagation of broadcast packets occurs along concentric "waves" (*ripples*, whence the name RippleCast): in the first round, a packet is transmitted to nodes located in cells at cell distance one from s; in the second round, the packet is propagated to nodes located in cells at cell distance two from s, and so on, till the packet is propagated to the furthest cells in the deployment region (see Fig. 3). Since a new packet is generated by source s at each round, the propagation proceeds in a pipelined fashion, and eventually at each round each ripple of leaders is propagating a different packet.

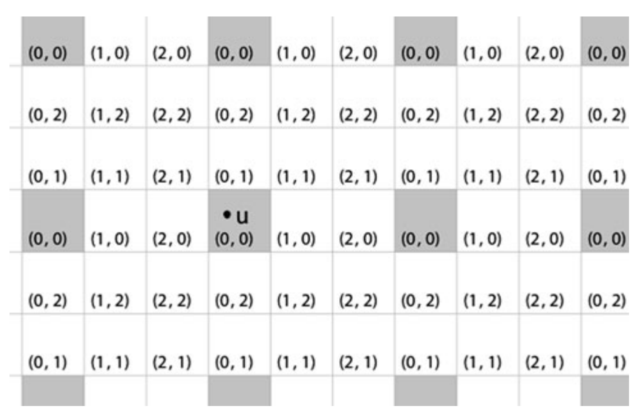

Fig. 2 Two-dimensional coloring of parameter $k = 3$

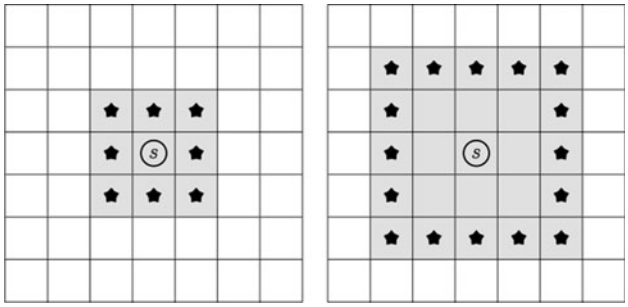

Fig. 3 The propagation front (*ripple*) of a broadcast packet. *Stars* represent cell leaders sending a certain packet p, and shaded cells are those which already received p. Propagation proceeds in a pipelined fashion, and eventually at each step each ripple is propagating a different packet

```
Algorithm  for source node s:
      Let i be the color of the current time slot; ID is the
      current packet ID
      1. if col(s) = i then
      2.     transmit new packet; ID = ID + 1
Algorithm  for a  generic node v:
      Let i be the color of the current time slot; if v is a leader node,
      let j be the ID of the last packet received by node v
      1. listen to the channel
      2. if new packet arrive then
      3.     receive the packet
      4.     let j' be the ID of the received packet
      5.     if (j' = j + 1) then
      6.         store packet in transmit buffer
      7. if (col(v) = i) and cellLeader(v) then
      8.     if buffer(v) is not empty then
      9.         transmit packet and empty transmit buffer
```

Fig. 4 The RippleCast broadcasting scheme

5.2 RippleCast

The RippleCast algorithm is reported in Fig. 4. The algorithm for the source node is very simple: when the transmission slot correspondent to *col*(s) is scheduled, the source node transmits a new packet, and increments the packet ID by one. Any non-source node v acts as follows. Independently of the color of the scheduled slot, node v listens to the channel, and receives new packets. Note that a node in general receive packets with the same ID several times; only *new* packets are received at step 3. of the algorithm. If the ID of the new received packet equals the ID of the most recently received packet increased by one, then the new packet is stored in the transmit buffer. If the color of the current slot equals *col*(v), v is the cell leader, and the transmit buffer is not empty, the packet is transmitted and the transmit buffer emptied.

Function *cellLeader*() at step 7. checks whether node v is currently a leader node. Leader selection obeys the following rules. During round t, a node (call it v) currently (more specifically, at the beginning of the round) located within a central mini-cell is selected[6] as leader node for that cell (call it A) for that round. More specifically, during round t node v will be in charge of transmitting the packet received by the cell it belonged to at round $t-1$. As we shall see in the next section, the fact that leader nodes are selected amongst the nodes in the central-mini cell, coupled with an upper bound on node velocity, ensures that node v was in cell A also during the entire round $t-1$, thus guaranteeing a correct propagation of broadcast packets. If node v is still in the central mini-cell of cell A at the beginning of round $t+1$, it keeps the leader role also in the next round, otherwise a new node amongst the ones currently present in the central mini-cell is selected as leader for round $t+1$.

5.3 Analysis

We start borrowing a result from [22], which shows that cells can be colored using $\bar{k}^2 = \Theta(1)$ colors, in such a way that the packet transmitted by a leader node is correctly received by *all* nodes located in neighboring cells (horizontal, vertical, and diagonal adjacency), under the assumption that at most one node per cell with the same color is transmitting—similar results about node coloring can be found, e.g., in [9]. The coloring scheme depicted in Fig. 2 assigns the same color to cells at cell distance $\bar{k}$ along the horizontal and vertical direction (details can be found in [22]). The following result has been proved in [22].

Proposition 3 *Given a deployment region divided into square cells of side* $l = \frac{r_{max}}{2h\sqrt{2}}$, *for some constant* $h > 1$, *it is possible to devise a coloring scheme with* k^2 *colors, where* $k \geq \bar{k} = \left\lceil 2 + 2^{\frac{3}{2}+\frac{4}{\alpha}}(\beta\zeta(\alpha-1)h^{\alpha}/(h^{\alpha}-1))^{\frac{1}{\alpha}} \right\rceil$, *and* ζ *is the Riemann's zeta function, such that the packets transmitted by leader nodes with the same color are received by all nodes located in cells adjacent to the cell of a transmitter node (horizontal, vertical, and diagonal adjacency), under the assumption that at most one node per cell with the same color is transmitting.*

Note that h, α and β being constants, the number of colors $\bar{k}^2$, which coincides with the number of transmission slots in a round, is $\Theta(1)$.

The next Lemma, whose straightforward proof is omitted, states that source node s generates new packets at rate $\frac{W}{\bar{k}^2} = \Omega(W)$, which is asymptotically optimal.

[6] The actual rule used to selected leaders in case more than one nodes are present in a mini-cell is irrelevant.

Lemma 1 *Assume algorithm* RippleCast *is used to broadcast packets in the network. The source node s generates packets at rate* $\frac{W}{\bar{k}^2} = \Omega(W)$.

We next show that each packet generated by the source is correctly received by *all* network nodes within time $T(n) = O\left(\sqrt{\frac{n}{\log n}}\right)$.

Lemma 2 *Assume n nodes move within a square region of side* $L(n) = \frac{r_{max}}{6h\sqrt{2}}\sqrt{\frac{n}{\log n}}$ *according to a mobility model* $\mathcal{M}$ *with:* (1) *uniform stationary node spatial distribution, and* (2) *maximum node velocity equal to* $\bar{v} = \frac{l}{3\bar{k}^2\tau}$, *where* τ *is the duration of a transmission slot and l is the side of a cell. Furthermore, assume algorithm* RippleCast *is used to broadcast packets. Then, a packet generated by the source node at round t is received by all network nodes within round* $t + O\left(\sqrt{\frac{n}{\log n}}\right)$, *w.h.p.*

Proof See Appendix. □

Is the upper bound on node velocity imposed by Lemma 2 restrictive? The answer, for typical values of the network parameters, is *no*, owing to the very high packet propagation speed within the network. For instance, assuming an outdoor propagation environment with path-loss $\alpha = 3$, channel parameters typical of an 802.11a/g network with 54 Mbs data rate (more specifically, $\beta = 22$ dB, $P =$ 100 mW, and $N = -90$ dBm), a packet size of 1 KB, and setting $h = 2$ in the cell partitioning scheme, we have that $\bar{k} = 50$, $r_{max} = 858$ m, $l = 151$ m, $\tau = 180\,\mu$s (leaving ade quate margin for radio signal propagation time), and uniform stationary node spatialthe upper bound on velocity is $\bar{v} = 111.852$ m/s ≈ 403 km/h. Observe that more stringent upper bounds on node velocity would be obtained if lower data rates admitted by the IEEE 802.11a/g standard would be considered in the above calculation. However, technology improvements are projected to deliver faster data rates in the near future (think about the 600 Mbs raw data rate which will be soon achieved by IEEE 802.11n), so the velocity upper bound needed for our result to hold will actually become even less stringent as technology advances.

We are now ready to prove the main result of this section:

Theorem 2 *Assume n nodes move within a square region of side* $L(n) = \frac{r_{max}}{6h\sqrt{2}}\sqrt{\frac{n}{\log n}}$ *according to a mobility model* $\mathcal{M}$ *with:* (1) *uniform stationary node spatial distribution, and* (2) *maximum node velocity equal to* $\bar{v} = \frac{l}{3\bar{k}^2\tau}$, *where* τ *is the duration of a transmission slot and l is the side of a cell. Algorithm* RippleCast *provides asymptotically optimal broadcast capacity and latency.*

Proof The proof is a straightforward consequence of lemmas 1 and 2, and of the observation that the duration of a round (which is composed of $\bar{k}^2$ transmission slots, each of constant duration τ) is $\Theta(1)$. □

6 Broadcasting with leader election

In this section, we revisit the broadcasting problem taking into account the burden incurred by leader election, which was intendedly ignored in the previous section. Distributed leader election is one of the most investigated problems in the distributed computing literature. Though, the leader election problem we face is non-standard: although each single leader election in a mini-cell corresponds to the classical single-hop leader election problem [17], we have to perform several such elections: one for each of the $\frac{n}{\log n}$ mini-cells in the deployment region. Since sequentially executing these elections would considerably impact both broadcast capacity and latency, we propose running as many simultaneous leader elections as possible, subject to the condition that simultaneously active leader elections do not corrupt each other.

The approach we pursue to tackle the problem at hand is a combination of the ID-based leader election scheme proposed in [1] for network-wide election of a single leader node in a wireless multihop network, and of the carrier sense based technique used in [24] to distributedly build a dominating set in a wireless multihop network. The main idea is to run parallel leader elections in the cells colored with the same color. As we shall see, in order to ensure that mutual interference does not corrupt concurrent leader elections, we have to use a relatively larger (and non-constant) number of colors, which leads to a poly-logarithmic broadcast capacity and latency degradation with respect to optimal. Even worse, using a non-constant number of colors leads to an asymptotically vanishing upper bound on node maximum velocity.

The leader election process in a cell is performed as follows. Each of the n mobile nodes in the network is assigned with a unique binary ID. Let $ID(u)$ denote the ID of node u. Similarly to [1], the binary representation of node IDs is used to elect leader nodes: at the end of the election process, the leader node for a certain cell A is the node with highest ID among the nodes within the central mini-cell of A at the beginning of the leader election process. The leader election process, reported in Fig. 6, is divided into $\log n$ phases of constant duration, where the duration of a phase is sufficient to transmit a single bit of information on the channel. During phase i, node u, if still active, checks whether the i-th most significant bit of its ID is 1, and in that case transmits a "1" bit on the channel. Otherwise, it listens to the channel, and becomes inactive if the signal sensed on the channel exceeds a certain threshold T_s. As we shall see, threshold T_s is set in such a way that the following two properties are satisfied:

(a) if at least one node within the same mini-cell of u is transmitting, then the sensed signal at u is $>T_s$.
(b) if no node within the same mini-cell of u is transmitting, then the sensed signal at u is $<T_s$.

Properties (a) and (b) ensure that threshold T_s on the sensed channel value ch can be used to discriminate between two different situations: no node within the same mini-cell is transmitting ($ch < T_s$), or at least one node within the same mini-cell is transmitting ($ch > T_s$). Thus, if node u still competing for becoming leader detects that $ch > T_s$ at step 7 of the leader election algorithm, it leaves the competition, since condition $ch > T_s$ ensures that at least another node in the same mini-cell has a higher ID than that of node u.

Note that, in order for such threshold T_s to exist, we must be able to upper bound the aggregate power received at u generated by nodes in other mini-cells; furthermore, in order for (a) and (b) to simultaneously hold, this upper bound must not depend on n. We now show that a threshold T_s satisfying properties (a) and (b) above can actually be defined if we use $(k^*)^2$ colors, where $k^* = \Theta(\log n)$.

We first prove the following technical lemma, which provides a bound on the amount of power received by a node from nodes in cells with the same color.

Lemma 3 *Let us assume a cell coloring with k^2 colors as defined in* Fig. 2 *and that each cell contains at most m nodes. Let us fix an arbitrary node u in an arbitrary cell C. If $k \geq 2$ and all the nodes in all the cells (apart C) with the same color as C transmit simultaneously, then the interference P_I experienced by u satisfies*

$$P_I < m\frac{16P\zeta(\alpha-1)}{(k-1)^{\alpha}l^{\alpha}}, \qquad (1)$$

where ζ is the Riemann's zeta function.

Proof See Appendix. □

Lemma 4 *Assume the cell coloring scheme is composed of k^2 colors, with*

$$k > k^* = 1 + \frac{2^{4/\alpha}\cdot\sqrt{2}\cdot\zeta(\alpha-1)^{1/\alpha}}{3}\cdot(c\log n)^{1/\alpha},$$

for some constant $c > 1$. Then, there exists a (constant) threshold T_s such that properties a) and b) above are satisfied.

Proof We first lower bound the intensity P_T of the signal received by a node u within a mini-cell when another node within the same mini-cell is transmitting. Given the assumed radio propagation model, we have

$$P_T \geq P\left(\frac{l}{3}\sqrt{2}\right)^{-\alpha} = T'_s,$$

which implies that we must have $T_s < T'_s$. □

We now upper bound the intensity of the signal received at node u generated by nodes belonging to other mini-cells with the same color within the network. Observing that the maximally occupied mini-cell contains at most c log n nodes, for some constant $c > 1$ (see Proposition 1), and letting $m = c\log n$ and $k = k^*$ in Lemma 3, we obtain that the aggregate power P_I at node u generated by nodes within mini-cells with the same color is upper bounded by:

$$P_I < (c\log n)\frac{16P\zeta(\alpha-1)}{(k-1)^{\alpha}l^{\alpha}} = T''_s,$$

for some constant $c > 1$, where k is the step of the coloring scheme (i.e., we have k^2 colors in total). Thus, if we set $T_s > T''_s$, we are guaranteed to satisfy property (b). The proof of the Lemma follows by observing that, when $k > k^*$, we have $T'_s > T''_s$, and a threshold satisfying both properties (a) and (b) above can be obtained by choosing any value T_s such that $T''_s < T_s < T'_s$.

We are now ready to introduce the broadcasting scheme, which is a combination of RippleCast with the leader election scheme presented above. A round of the broadcast scheme is composed of two steps (see Fig. 8): in the first step, leader nodes *for each cell* are elected according to the leader election algorithm described above; in the second step, RippleCast is executed using leader nodes elected in the first step to propagate broadcast packets.

We are now ready to characterize the asymptotic properties of this combined broadcasting scheme.

Lemma 5 *Assume n nodes move within a square region of side $L(n) = \frac{r_{max}}{6h\sqrt{2}}\sqrt{\frac{n}{\log n}}$ according to a mobility model $\mathcal{M}$ with:* (1) *uniform stationary node spatial distribu tion, and* (2) *maximum node velocity equal to $v^* = \frac{l}{3(\tau'\cdot(k^*)^2\cdot\log n+k^2\tau)}$, where τ′ is the duration of a phase of the leader election process, τ is the duration of a transmission slot, and l is the side of a cell. Furthermore, the above described combined leader election and broadcasting scheme is used to broadcast packets. Then, a packet generated by the source node at round t is received by all network nodes within round $t + O\left(\sqrt{\frac{n}{\log n}}\right)$, w.h.p.*

Proof Similarly to Lemma 2, we can show that the packet transmitted by the source during round t is received by each network node within round $t + O\left(\sqrt{\frac{n}{\log n}}\right)$. However, the upper bound on node velocity must take into account the longer duration of a communication round, which comprises also the leader election step. The leader election step lasts for $(k^*)^2 \cdot \log n$ phases overall (leader election processes, each lasting log *n* phases, are performed in parallel for each of the $(k^*)^2$ colors). Note that the duration τ' of a phase should be sufficient to send a single bit of information of the channel, i.e., $\tau' \ll \tau$. The total duration of a communication round is then $\tau' \cdot (k^*)^2 \cdot \log n + \bar{k}^2\tau$. Similarly to Lemma 2, the maximum node velocity must be set in such a way that the maximal traveled distance within a communication round equals $\frac{l}{3}$, from which we derive $v^* = \frac{l}{3(\tau' \cdot (k^*)^2 \cdot \log n + \bar{k}^2\tau)}$. □

Theorem 3 *Assume n nodes move within a square region of side* $L(n) = \frac{r_{max}}{6h\sqrt{2}}\sqrt{\frac{n}{\log n}}$ *according to a mobility model* $\mathcal{M}$ *with:* (1) *uniform stationary node spatial distribution, and* (2) *maximum node velocity equal to* $v^* = \frac{l}{3(\tau' \cdot (k^*)^2 \cdot \log n + \bar{k}^2\tau)}$, *where* τ' *is the duration of a phase of the leader election process,* τ *is the duration of a transmission slot, and* l *is the side of a cell. The above described combined leader election and broadcasting scheme provides broadcast capacity and latency within a factor* $\Theta((\log n)^{1+\frac{2}{\alpha}})$ *from optimal.*

Proof The duration of step 1 (leader election) in each round is $\Theta((\log n)^{1+2/\alpha})$. In fact, leaders must be elected for each cells, which are divided into $(k^*)^2 = \Theta((\log n)^{2/\alpha})$ groups. The leader election process, which lasts log *n* time, goes on in parallel for all the cells in a group, implying that the overall duration of step 1 in a round is $\Theta(\log n \cdot (\log n)^{2/\alpha}) = \Theta((\log n)^{1+2/\alpha})$. Even if step 2 of a round has a constant time duration (this is because the required number of colors $\bar{k}^2$ for RippleCast is a constant), the overall duration of a round of communication is $\Theta((\log n)^{1+2/\alpha})$. Since the source transmits a new broadcast packet at each round, we have that the broadcast rate is $\Theta\left(\frac{W}{(\log n)^{1+\frac{2}{\alpha}}}\right)$, which, according to Claim 4, is within a factor $\Theta((\log n)^{1+\frac{2}{\alpha}})$ from optimal. By Lemma 5, the packet generated by the source at round *t* is received by each network node within round $t + O\left(\sqrt{\frac{n}{\log n}}\right)$. Given that the duration of a round is $\Theta((\log n)^{1+\frac{2}{\alpha}})$ and Claim 1, we have that the broadcast latency achieved by our scheme is also within a factor $\Theta((\log n)^{1+\frac{2}{\alpha}})$ from optimal. □

Comparing theorems 2 and 3, we observe a polylogarithmic performance degradation with respect to both capacity and latency when the burden for leader election is taken into account. Most importantly, the burden related to the leader election process considerably strengthen the upper bound on node velocity, *which becomes asymptotically vanishing as n grows to infinity*. Thus, the larger the network, the more stationary the nodes must be in order to achieve near-optimal broadcast capacity and latency. However, owing to the orders of magnitude smaller value of τ' as compared to τ (we recall that τ' is the time necessary to transmit a single bit of information, instead of an entire packet) and logarithmic dependence on *n*, the actual bound on maximal node velocity is only marginally influenced by the number of network nodes. For instance, the upper bound v^* on node velocity is $v^* = 111.816$ m/s when $n = 2^{10} = 1{,}024$ (assuming the same parameters as in Sect. 5.3, and setting $\tau' = \frac{\tau}{1{,}000}$), which should be compared to $\bar{v} = 111.852 m/s$ when the leader election burden is ignored. When $n = 2^{50}$ (far above the size of any practical network), the upper bound becomes $v^* = 111.415$ m/s, which is only marginally smaller than $\bar{v}$.

7 Mobile broadcast source

In the previous sections, we have assumed that the source node *s* does not change its position over time. We now relax this assumption, and assume that *s* can move within *R* with a properly upper bounded velocity v_s. For convenience, in the following we denote by v_{ns} the speed[7] of mobile nodes, which will also be properly upper bounded. For clarity of presentation, we initially assume that, similarly to Sect. 5, broadcast relay nodes are "magically" selected within the network. We will bring leader election into the picture at the end of the section. Furthermore, in the following we retain the hypotheses about node deployment and mobility pattern as in the previous sections, yielding in particular the (mini-)cell occupancy properties. To simplify presentation, in the statement of lemmas and theorems we will omit re-stating properties of the deployment and mobility model. Similarly, occupancy arguments are not mentioned in the proofs, and assumed to implicitly hold.

We first present a high-level, intuitive description of our capacity and latency achieving broadcast scheme. The main problem to be faced when the source node is mobile is that the "ripples" along which broadcast packets are propagated in the network move along with the source. Since broadcast packets generated by the source in different rounds might actually start from different locations (cells), in absence of careful scheduling decisions these

[7] Different non-source nodes are allowed to have different velocity, as long as a common upper bound on node velocity is not impaired.

packets could actually collide at some cell, possibly leading to suboptimal broadcast latency due to the need of buffering packets. To circumvent this problem, we introduce the novel notion of center-cast. The main idea is to split the broadcasting process into two inter-leaved phases (see Fig. 9): a 'center-cast' phase, and the RippleCast phase. During 'center-cast', the source node, independently of its location within R, sends its packets towards (any node located within) the center cell C_c in R; in the RippleCast phase, the leader node of cell C_c (called the *virtual source* in the following) broadcasts the packets received during the center-cast phase. Center-cast and RippleCast phases are designed such that (a) broadcast packets are never buffered for more than one round during the center-cast phase, and (b) a new broadcast packet is received by the virtual source at least every other round, which, combined with the fact that rounds have constant time duration, yield asymptotically optimal broadcast capacity and latency.

We now present the center-cast phase of the broadcasting scheme; the RippleCast phase is very similar to the one described in Sect. 5, with only two differences: (1) the broadcast source is the leader node of cell C_c (virtual source), the central cell in R; and (2) each of the k^2 cells is allocated two, instead of one, transmission opportunities. As for (1), observe that our approach and presented bounds remain valid independently of the actual location of C_c within R, with the only requirement that the location of C_c does not change over time. For definiteness, in the following we assume C_c is the cell (or any of the cells) located at (close to) the center of R. As for (2), we remark that 2 instead of 1 transmission opportunities per round are needed to account for situations (which, as we shall see, might be a consequence of source mobility) in which more than one new packet is received by the virtual source during the center-cast phase. Note that, since as many as two packets can be stored in the virtual source and leader node RippleCast buffers at some round, a prioritization of older over newer packets is applied in order to preserve packet ordering and minimize latency. For details on how prioritization is achieved, see description of the similar mechanism used during the center-cast phase presented below. Summarizing, the duration of the RippleCast phase in a round is $2k^2$ slots, with $k \geq \bar{k}$ and $\bar{k}$ defined as in Proposition 3.

Differently from the RippleCast phase, the center-cast phase is composed of $2k^2 + 1$ slots, with $k \geq \bar{k}$. This is to allow every cell to have *two* transmission opportunities during a center-cast phase. As we shall see, if source velocity is properly upper bounded, two transmission opportunities in a phase are *sufficient* for property a) to hold. The additional slot, at the *end* of the center-cast phase, is assigned to the source node, which hence has a (single) transmission opportunity during each center-cast phase. Summarizing, a broadcast round with mobile source is composed of $4k^2 + 1$ slots overall, with $2k^2 + 1$ slots used in the center-cast phase, and $2k^2$ slots used in the RippleCast phase (see Fig. 9).

The CenterCast algorithm used during the center-cast phase is reported in Fig. 10. The algorithm for the source node is very simple: the node simply checks whether the current color is the one reserved for source node transmission, in which case it generates and transmits a new packet, and increases the packet ID counter. Note that, differently from RippleCast, the generated packet also contains the cell-ID of the next cell on the path from current s position to the center cell C_c, which is computed at step 2. Non-source nodes continuously listen to the channel; when a packet is received, a decision is taken on whether the packet should be stored in the transmit buffer. The packet is stored in the transmit buffer only if the following conditions are fulfilled: (1) the currently received packet is more recent than the last received packet; (2) the node is leader for the current round; (3) the *cellID* of the packet equals the cell to which the node belonged at the beginning of the round. As we shall see, this mechanism, coupled with periodic leader re-election performed *at the beginning* of each broadcast round, ensures latency optimal flowing of packets generated by the source towards the center cell C_c. Packets in the transmit buffer are transmitted whenever the cell color is active (step 6.). Note that, differently from the case of RippleCast, node buffers have two positions instead of one. Hence, when a transmission opportunity arises (step 6.), the node extracts from the buffer the *older packet* (step 8.), transmits it, and removes it from the buffer. As we shall see, prioritizing older packets with respect to newer ones is necessary to guarantee optimal broadcasting latency. Note that, before transmitting a packet extracted from the buffer, node v substitutes the cell-ID included in the buffered packet with a new one, which is recomputed (step 7.) based on current position of node v.

We now prove some technical lemmas needed to derive the main result of this section.

Lemma 6 *If the velocity v_s of the source node is at most $\bar{v}_s = \frac{l}{2(4k^2+1)\tau}$, then the source node can move only between adjacent cells (horizontal, vertical, and diagonal adjacency) during two broadcast rounds.*

Proof The proof is based on a simple geometric argument. The minimal distance to be traversed to go from a cell C to a non-adjacent cell C' is $l + \epsilon$, where l is the cell side and $\epsilon > 0$ is an arbitrarily small positive constant. In order to cover such distance in time at most $2(4k^2 + 1)\tau$

(the duration of two broadcast rounds), v_s must be at least $\frac{l+\epsilon}{2(4k^2+1)\tau} > \frac{l}{2(4k^2+1)\tau}$, and the lemma follows. □

Note that the upper bound on source node velocity established by Lemma 6, using typical IEEE 802.11a/g settings (cfr. end of Sect. 5), is about 84 m/s ≈ 302 km/h, which is lower than the bound on node velocity needed for achieving optimal broadcasting performance, but it is still quite high.

Definition 1 (*Source cell*) Let t_i be the time at which round i begins. The *source cell* at round i, denoted $C(s, i)$, is the cell to which node s belongs at time $t_i + 2k^2\tau$.

In the following, we denote by $L(C, i)$ the leader node for cell C at round i. We recall that leader election is performed at time t_i, i.e., at the beginning of the round, and that leader nodes are selected among those in central mini-cells.

Lemma 7 *Assume non-source nodes move with velocity at most* $\bar{v}_{ns} = \frac{l}{6(4k^2+1)\tau}$. *Then, node* $L(C, i)$ *is still located within cell* C *by the end of round* $i + 1$.

Proof The proof is again based on a simple geometric argument: since leader nodes are elected amongst nodes within central mini-cells, we have that the minimum distance node $L(C, i)$ has to cover to exit from cell C is $\frac{l}{3} + \epsilon$, for some constant $\epsilon > 0$. If node velocity is at most $\frac{l}{6(4k^2+1)\tau}$, such distance cannot be covered during the duration of two broadcast rounds. □

Note that velocity bound $\bar{v}_{ns}$ is more than a factor of 6 smaller than that needed for Theorem 2.

Lemma 8 *Assume* $v_s \leq \bar{v}'_s = \frac{r_{max}}{\tau} \cdot \left(1 - \frac{1}{h} - \frac{1}{6h\sqrt{2}k^2}\right)$, *non-source node move with velocity at most* $\bar{v} = \frac{l}{3k^2\tau}$, *and let* t_i *be the time at which round* i *begins. Then, the packet* p_i *transmitted by* s *during round* i *is correctly received by all nodes residing in the cells adjacent to cell* $C(s, i)$ *at time* $t_i + 2k^2\tau$.

Proof In order for packet p_i to be correctly received by a node u, node u must remain within s's transmission range for the whole duration of packet transmission. We observe that the duration of packet transmission is at most τ, that the maximum distance between s and a node in a cell adjacent to $C(s, i)$ is $2l\sqrt{2}$, and that the maximum velocity of node u is $\bar{v} = \frac{l}{3k^2\tau}$. Hence, p_i can be correctly received by any node u that resided in a cell adjacent to $C(s, i)$ at time $t_i + 2k^2\tau$ if and only if the following inequality is satisfied:

$$2l\sqrt{2} + v_s\tau + \bar{v}\tau \leq r_{max} \tag{2}$$

The lemma then follows by substituting $l = \frac{r_{max}}{2h\sqrt{2}}$ and $\bar{v} = \frac{l}{3k^2\tau}$ into (2), and solving for v_s. □

Note that the upper bound on source node velocity required in this lemma is much looser than that necessary for Lemma 6; using typical parameters of IEEE 802.11 a/g technology (cfr. end of Sect. 5), we obtain $\bar{v}'_s \approx 2.38 \cdot 10^6$ m/s.

Lemma 9 *Assume* $v_s \leq \min\{\bar{v}_s, \bar{v}'_s\}$, $v_{ns} \leq \bar{v}_{ns}$, *and let* t_i *be the time at which round* i *begins. Then, the packet transmitted by source node* s *at round* i *is received by the virtual source at round* $i + d(s, i)$, *where* $d(s, i)$ *is the cell distance between* $C(s, i)$ *and the center cell* C_c.

Proof See Appendix. □

Lemma 10 *Assume* $v_s \leq \min\{\bar{v}_s, \bar{v}'_s\}$, *and* $v_{ns} \leq \bar{v}_{ns}$. *Then, the virtual source transmits a new broadcast packet at rate* $\frac{W}{2(4k^2+1)} = \Omega(W)$.

Proof Note that the virtual source has two transmission opportunities during the RippleCast phase at each round, yielding a broadcast rate of up to $\frac{2W}{4k^2+1}$. However, this rate can be achieved only if two new broadcast packets are received by the virtual source at every round. Due to source node mobility, this is not always possible, which brings to a factor of 4 rate degradation with respect to $\frac{W}{4k^2+1}$. To understand why, we first observe that by Lemma 9, packet p_i generated by source node at round i is received by the virtual source at round $i + d(s, i)$, where $d(s, i)$ is the cell distance between cell $C(s, i)$ and cell C_c. This would ensure a continuous flow of new packets to the virtual source if the source were static. Since source node is mobile, though, cells distances $d(s, i)$ change over time, and continuos flows of packets toward the virtual source is not always guaranteed: in fact, if $C(s, i + 1) = C(s, i) + j$ for some $j > 0$, we have j rounds during which no new packet is received at the virtual source. However, Lemma 6 ensures that $|C(s, i) - C(s, i + 1)| \leq 1$ for each i, implying that a new packet is received by the virtual source at least every other round, which completes the proof. □

Lemma 11 *Assume* $v_s \leq \min\{\bar{v}_s, \bar{v}'_s\}$, *and* $v_{ns} \leq \bar{v}_{ns}$. *Then, the packet generated by the source at round* i *is received by all network nodes within round* $i + O\left(\sqrt{\frac{n}{\log n}}\right)$, *w.h.p.*

Proof By Lemma 9, packet p_i is received by the virtual source at round $i + d(s, i)$, where $d(s, i)$ is clearly $O\left(\sqrt{\frac{n}{\log n}}\right)$. Once virtual source is reached, Algorithm RippleCast ensures p_i is received by all nodes in the network within further $O\left(\sqrt{\frac{n}{\log n}}\right)$ rounds. This is implied by the fact that, from Lemma 6, the virtual source can receive at most two new broadcast packets during any center-cast phase, and that the virtual source (and every other leader node in the network) has two transmission opportunities during the RippleCast phase. This, coupled with the priority rule used to manage RippleCast transmit buffers, guarantees optimal broadcasting latency. □

Theorem 4 *Assume* $v_s \leq \min\{\bar{v}_s, \bar{v}' s\}$, and $v_{ns} \leq \bar{v}_{ns}$. *The combined* CenterCast *and* RippleCast *Algorithm provides asymptotically optimal broadcast capacity and latency.*

Proof The proof is a straightforward consequence of lemmas 10 and 11, and of the observation that the duration of a round (which is composed of $4\bar{k}^2 + 1$ transmission slots, each of constant duration τ) is $\Theta(1)$. □

Broadcasting with leader election in case of mobile source can be achieved by compounding an initial leader election phase (see Sect. 6) with the center-cast and RippleCast phase, yielding an overall duration of $\tau' \cdot (k^*)^2 \cdot \log n + (4\bar{k}^2 + 1)\tau$ for the broadcast round. This leads to the same performance bounds as in the case of static source node (cfr. Sect. 6). In the interest of brevity, formal proofs are not reported.

8 Discussion and future work

In this paper, we have investigated the fundamental limits of broadcasting in dense, mobile wireless networks, and we have shown that, while broadcasting is not inherently limited (in terms of both capacity and latency) by neither source nor node mobility, the coordination burden caused by the need of repeatedly selecting broadcast relay nodes does indeed reduce broadcast performance of a poly-logarithmic factor. Our results hold under a set of assumptions: nodes move within a square region according to a mobility model with stationary uniform node spatial distribution, and node velocity is upper bound by a constant (which becomes an asymptotically vanishing function of n when the coordination burden is taken into account).

We first observe that some generalizations of our results are straightforward: up to tedious technical details, our findings can be extended to deployment regions of different compact shapes, as long as broadcast ripples are still "closed curves". Extension to mobility models whose stationary node spatial distribution is "almost uniform" is also straightforward; by "almost uniform", we mean that the ratio between the larger and smaller value of the two-dimensional probability density function describing stationary node positions within the deployment region is an arbitrary *positive* constant.

What are the implications of our findings for the design of practical broadcasting protocols for mobile networks? The main implication is that network designers should focus their design on identifying *invariant* properties of the mobile network (e.g., node spatial distribution), and then build their protocol exploiting these properties. Clearly, location-awareness is likely to be a key feature in designing efficient broadcasting protocols for mobile wireless networks. The analysis of the mobile source case brings to the attention another interesting hint for the design of practical broadcasting protocols, namely the need of prioritizing propagation of packets within the network. The novel notion of center-cast can also prove very useful in the design of broadcasting protocols with mobile source.

It is interesting also to discuss the relative effect of node mobility in case of unicast and broadcast communications: in unicast communication—under the assumption that arbitrarily high delays can be tolerated-, node mobility can be used as a mean to suppress (or considerably reduce) the relaying burden, thus bringing capacity up to the optimal value; on the contrary, in case of broadcast, node mobility introduces the need of frequently re-selecting broadcast relay nodes, thus inducing a coordination burden which causes a poly-logarithmic capacity and latency degradation with respect to optimal. However, it is important to observe that this performance degradation is not inherently due to the broadcast communication pattern, but rather to a "common practice" of performing broadcast communications based on the selection of broadcast relay nodes. Hence, a promising research direction is to investigate whether alternative broadcasting approaches can be used to reach the capacity and latency limits. In particular, we intend to explore cooperative communications, which have already been successfully used to improve capacity limits for unicast communications (see, e.g., [5, 20]). Another interesting direction for future work stems from the observation that leader nodes in the broadcast process are a dominating set for network nodes. Thus, the results presented in this paper could be used to investigate performance of algorithms for building/maintaining dominating sets in a mobile, dense environment, such as the ones proposed in [27].

Appendix

Proof of Lemma 2. Let us call the cells at cell distance i from the cell containing the source node the i-th ripple. We start showing that: *a*) for each cell A in the i-th ripple, the leader node of cell A transmits during round $t + i$ the packet generated by the source node at round t. The proof is by induction on i. Property *a*) trivially holds when $i = 0$. Assume now property *a*) holds for each $j < i$. In order for *a*) to hold also for i, we need to show that, for any cell A in the i-th ripple, the node selected as leader for A during round $t + i$, which is going to transmit during round

$t + i$, has received the packet generated by the source at round t *before* its transmission opportunity during round $t + i$. Given that *a*) is assumed to hold for $j < i$, we have that the leader node of cell B, where B is any of the cells in the $(i - 1)$-th ripple adjacent to A (note that at least one such cell always exists), has transmitted during round $t + i - 1$ the packet generated by source node at round t. Given the rules for selecting leader nodes, we have that the leader node at round $t + i - 1$ for cell B is selected amongst the nodes located in the central mini-cell of B at the beginning of round $t + i - 1$. By Proposition 1, we have that at least one such node exists, w.h.p. Furthermore, the upper bound $\bar{v}$ on node velocity guarantees that a node travels at most $\bar{v}k^2\tau = \frac{l}{3}$ in the time elapsing between the beginning of round $t + i - 1$ and the beginning of round $t + i$. Since the leader node of cell B was within the central mini-cell of cell B at the beginning at round $t + i - 1$ and given the above observation about the distance traveled by nodes, we have that the leader node of cell B is still within cell B when it is scheduled for transmission during round $t + i - 1$. Hence, by Proposition 3, we have that the packet transmitted by the leader node of cell B during round $t + i - 1$, which by induction is the packet generated by the source node at round t, is correctly received by all nodes within cell A at the time of transmission. In particular, the leader node w for cell A at round $t + i$ is within the central mini-cell of A at the beginning of round $t + i$, which given the above observation about maximum traveled distance, ensures that w was within cell A also during the entire round $t + i - 1$. Thus, node w can correctly receive the packet sent by the leader node of cell B during round $t + i - 1$, and can forward it in the network when scheduled for transmission at round $t + i$, which implies property *a*).

Let us now define the set of *covered cells* $Cov(p)$ for a certain packet p as the set of cells such that their respective leader nodes have already transmitted packet p. By property *a*), and assuming packet p is generated by the source at round t, we have that $Cov(p)$ at round $t + i$ is the union of all the cells in ripples $0, \ldots, i$. Given the assumption on the size $L(n)$ of the deployment region, we have that $Cov(p)$ contains *all* the cells in the deployment region at round $t + \frac{L(n)}{2l} = t + O(\sqrt{\frac{n}{\log n}})$. Let us now consider an arbitrary mobile node u, and assume by contradiction that node u has not received packet p by the end of round $t + \frac{L(n)}{2l}$. Since $Cov(p)$ contains all the cells in the deployment region by then, and considering that each of the ripples propagating packet p is a "closed curve"[8], the only possible way for node u to avoid receiving p is to cut through the ripple propagation front during round j, for some $0 < j < \frac{L(n)}{2l}$. However, for this to be possible, node u should travel distance at least $2l$ between two successive rounds (see Fig. 5), which is possible only if node velocity is at least $v' = \frac{2l}{2k^2\tau} > \bar{v}$. Thus, the assumption about maximum node velocity is contradicted, and the Lemma is proved. □

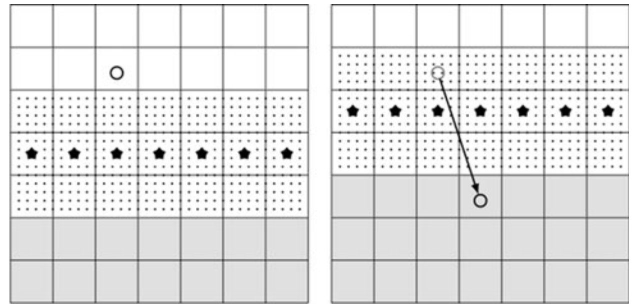

Fig. 5 Assume the source s is somewhere south of the diagrams, and the propagation front of packet p moves northward. *Stars* represents cell leaders active in a certain round, and the checkered region is the region covered by them. The *white area* has not yet been covered by packet p, while the *gray area* represents cells in $Cov(p)$ during a certain round. On the left, a *circle* represents a node lying in the *white area* which has not yet received p at a certain round $t + j - 1$. To avoid reception of packet p, the node must cut the propagation front and reach the gray area during round $t + j$ (right), where p is no longer transmitted. Thus, the node should travel distance at least $2l$ between the two consecutive rounds

[8] For the sake of simplicity, we use the intuitive notion of "closed curve" when referring to a ripple, although the ripple is not a curve in standard geometric sense.

Proof of Lemma 3. Let us consider now the interference experienced by u under the condition that in each cell with the same color there are at most m nodes. Assume w.l.o.g. that $cell(u)$ has coordinates (0,0). Given the coloring scheme, interferers lie in the cells with bottom left corner at $(x \cdot k \cdot l, y \cdot k \cdot l)$ with $x, y \in \mathbf{Z}$ and $(x, y) \neq (0, 0)$ (shaded cells in Fig. 2).

The distance $d(x, y)$ between u and an interferer located in cell $(x \cdot k \cdot l, y \cdot k \cdot l)$, with $x, y \neq 0$, can be lower bounded as follows:

$$d(x, y) \geq \sqrt{(|x|kl - l)^2 + (|y|kl - l)^2}, \tag{3}$$

where the term $-l$ depends on the actual positions of u and I inside their respective cells.

Since $a^2 + b^2 \geq (\max\{a, b\})^2$, from (3) we obtain the following lower bound on $d(x, y)$:

$$d(x, y) \geq \max\{|x|, |y|\}kl - l = l(k \max\{|x|, |y|\} - 1) \geq \\ \geq (k - 1)l \max\{|x|, |y|\}.$$

Note that the last bound is always strictly positive, since we are assuming $k \geq 2$ and $|x|$, $|y|$ are not both 0.

The interference received by u thus satisfies

$$P_I < m \sum \frac{P}{((k-1)l \max\{|x|,|y|\})^\alpha} = \\ = m \frac{P}{(k-1)^\alpha l^\alpha} \sum \frac{1}{\max\{|x|,|y|\}^\alpha},$$

where the sum is extended over all the pairs $(x, y) \neq (0, 0)$, with $x, y \in \mathbf{Z}$.

Counting twice the contributions along $x = 0$, $y = 0$, and $|x| = |y|$, we have

$$\sum_{(x,y)\neq(0,0)} \frac{1}{\max\{|x|,|y|\}^\alpha} < 8 \sum_{x=1}^{\infty} \sum_{y=0}^{x} \frac{1}{x^\alpha}$$

due to the eightfold symmetry of the summation shown in Fig. 7. Collecting the values for which $\max(x, y) = x$ we obtain

$$8 \sum_{x=1}^{\infty} \sum_{y=0}^{x} \frac{1}{x^\alpha} = 8 \sum_{x=1}^{\infty} \frac{x+1}{x^\alpha} < 16 \sum_{x=1}^{\infty} \frac{1}{x^{\alpha-1}} = 16\zeta(\alpha-1),$$

where $\zeta(\cdot)$ is the Riemann's zeta function and summarizing we obtain formula (1). □

Proof of Lemma 9. Let P_s denote a minimum-hop cell path connecting cell $C(s, i)$ with C_c. In other words, P_s is a minimum-length sequence of cells $C(s,i) = C_0, \ldots, C_j = C_c$ such that, for each $q = 0, \ldots, j-1$, cells C_q and C_{q+1} are adjacent. We will prove that the packet generated by s at round i is propagated through P_s till it reaches C_c, with the packet progressing one cell at each round. The proof is by induction on the cell distance q from $C(s, i)$. More specifically, we want to prove the following property: *a*) packet p_i is correctly received by node $L(C_{q+1}, i + q)$ during round $i + q$, for any $0 \le q \le j - 1$. Property *a*) implies the lemma by observing that $d(C(s, i), C_c) = d(s, i)$ implies existence of a cell path of length $d(s, i)$ connecting cells $C(s, i)$ and C_c.

To prove the base case $q = 0$, we observe that Lemma 8 ensures that p_i is correctly received by all nodes that are located in cells adjacent to $C(s, i)$—including cell C_1—at time $t_i + 2k^2\tau$. On the other hand, Lemma 7 ensures that node $L(C_1,i)$ remains within cell C_1 during the whole duration of rounds i and $i + 1$. Thus, node $L(C_1,i)$ is guaranteed to be in cell C_1 at time $t_i + 2k^2\tau$, and to correctly receive p_i.

```
Algorithm  for any non source node u:
Time is divided into log n phases of constant duration
    1. if u is not in a central mini-cell then exit //not a leader
    2. for i = 1 to log n do //phase i
    3.     if bit(ID(u), (log n − i)) = 1 then
    4.         transmit a "1" bit
    5.     else
    6.         listen to the channel
    7.         if sensed signal > T_s then exit //not a leader
    8. set leader = true //node u is leader
```

Fig. 6 The leader election algorithm

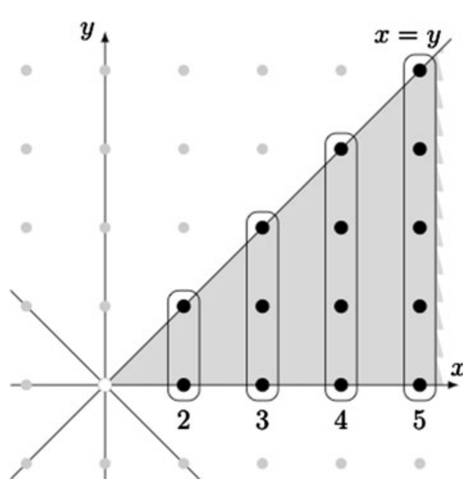

Fig. 7 Eightfold symmetry in the derivation of the upper bound to the total interference

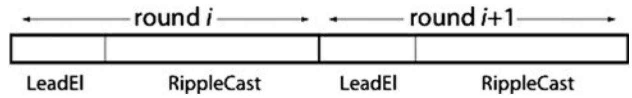

Fig. 8 The broadcast scheme with leader election

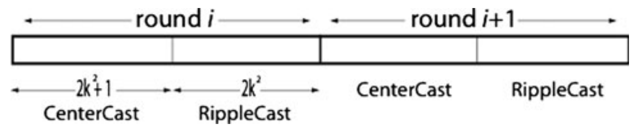

Fig. 9 Broadcasting with mobile source

```
Algorithm  for source node s:
    Let i be the color of the current time slot;
    let c(s) be the color of the reserved source slot;
    ID is the current packet ID
    cellID is the ID of the next cell to be visited by the packet
    1. if c(s) = i then
    2. based on own location, compute cellID
    3. transmit new packet (cellID,ID); ID = ID + 1
Algorithm  for a generic node v:
    Let i be the color of the current time slot;
    let j be the ID of the last packet received by node v
    let cell(v) be the cell to which node v belongs
    at beginning of the round
    1. listen to the channel
    2. if new packet arrive then
    3.     receive packet (cellID,j')
    4.     if (j' ≥ j + 1) and cellLeader(v) and cellID=cell(v) then
    5.         store packet (cellID,j') in transmit buffer
    6. if (col(v) = i) and buffer(v) is not empty then
    7.     based on own location, re-compute cellID
    8.     extract from buffer the packet (xxx, yy) with lowest ID;
    9.     transmit packet (cellID,yy) and
           remove packet from transmit buffer
```

Fig. 10 The CenterCast phase of the broadcasting scheme with mobile source

Assume now that property a) holds for any $q' < q$. By induction hypothesis, node $L(C_q, i + q - 1)$ has correctly received p_i at the end of round $i + q - 1$. We have to prove that p_i will be transmitted by node $L(C_q, i + q - 1)$ during the next broadcast round $i + q$. We first observe that, in order for a packet to be transmitted by a non-source node, the packet had to be stored in the transmit buffer upon reception. This happens if and only if the following three conditions are fulfilled: (1) the node is the leader of its cell for the current round; (2) the *cellID* in the packet equals the cell to which the node belongs; and (3) packet ID is larger than that of the last received packet. It is easy to see that condition (1) is fulfilled at node $L(C_q, i + q - 1)$. Condition (2) is fulfilled under the assumption, which holds without loss of generality, that the next cell selected by node $L(C_{q-1}, i + q - 2)$ when transmitting p_i was set to C_q. As for condition (3), we have to prove that packet ordering is preserved when forwarding packets generated by the source towards cell C_c. More specifically, we have to prove that packets p_j with $j > i$ cannot be received by $L(C_q, i + q - 1)$ before packet p_i. Note that there are two possible ways in which a packet p_j with $j > i$ can be propagated "faster" than packet p_i:—(i) packet p_j is a new packet generated by s, and s has moved along P_s in the direction of C_c; and—(ii) packet p_j is a packet relayed by a leader node on its route to cell C_c. As for case i), we observe that, by Lemma 6, node s can only move to adjacent cells during a broadcast round, hence its speed cannot exceed the speed of packet p_i propagation towards C_c. Yet, it is possible that, say, node s is located in cell C_1 at round $i + 1$, possibly leading to impaired packet ordering in case packet p_{i+1} is transmitted by s before packet p_i is forwarded to C_2 by node $L(C_1, i)$. However, the transmission slot reserved for source transmission is located *after* the $2k^2$ slots allocated for forwarding pending packets to the center cell C_c, hence packet ordering is preserved in case (i). As for case (ii), we observe that it is possible for a leader node to have *two* packets in the transmit buffer during a certain round (due to, e.g., mobility of the source in the direction of the center cell as explained above). However, in case multiple packets are present in the transmit buffer, older packets are prioritized over newer ones, thus preserving packet ordering also in this case. We have then proved that packet p_i is stored in the buffer of node $L(C_q, i + q - 1)$ during round $i + q - 1$. We now have to prove that node $L(C_q, i + q - 1)$ will transmit this packet during round $i + q$, and that node $L(C_{q+1}, i + q)$ correctly receives the packet. As for the first part, observe that nodes in cell C_q have two transmission opportunities during round $i + q$; since node $L(C_q, i + q - 1)$ has at most two packets in its transmit buffer (packet p_i, and possibly packet p_j with $j = i - 1$ or $j = i + 1$), two transmission opportunities are sufficient for node $L(C_q, i + q - 1)$ to transmit all packets in the buffer, including packet p_i. To see why at most two packets can be stored in a node's transmit buffer, it is sufficient to observe that multiple packets are sent by a cell only when the source is traveling, say, from cell C_k to cell C_{k+1} in a round j, in which case two packets will be transmitted by cell C_{k+1} during round $j + 1$ (the packet p_j sent by the source at round j, and the new packet p_{j+1} generated by the source at round $j + 1$). If the source would be allowed to move to cell C_{k+2} during round $j + 2$, we would have cell C_{k+2} transmitting 3 packets at round $j + 2$ (packets p_j, p_{j+1}, and the new packet p_{j+2} generated by the source at round $j + 2$). However, Lemma 6 implies that the source *cannot* move to cell C_{k+2} during round $j + 2$, since it can cross at most one cell during the duration of two broadcast rounds. This implies that every cell transmits at most two packets during any center-cast phase of the broadcast round, which in turns implies that at most two packets can be stored in a node's transmit buffer.

We are now left to show that node $L(C_{q+1}, i + q)$ correctly receives packet p_i. By induction hypothesis, by the fact that leader nodes are unique in a cell, and by Lemma 7, we have that the only node in cell C_q that has non-empty transmit buffer during round $i + q$ is $L(C_q, i + q - 1)$, implying that there is no conflicting transmission from other nodes in C_q during node $L(C_q, i + q - 1)$ transmission(s). In order to prove that node $L(C_{q+1}, i + q)$ correctly receives p_i, it is sufficient to observe that, by Lemma 7, node $L(C_q, i + q - 1)$ is still within cell C_q during round $i + q$, which implies that packets sent by this node during round $i + q$ are correctly received by all nodes in adjacent cells, including node $L(C_{q+1}, i + q)$. □

References

1. Brar, G., Blough, D., & Santi, P. (2008). The SCREAM approach for efficient distributed scheduling with physical interference in wireless mesh networks. In *Proceedings of IEEE ICDCS*, pp. 214–224.
2. Chen, Y., & Welch, J. L. (2005). Location-based broadcasting for dense mobile ad hoc networks. In *Proceedings of ACM MSWiM*, pp. 63–70.
3. Ellis, R. B., Martin, J. L., & Yan, C. (2007). Random geometric graph diameter in the unit ball. *Algorithmica, 47*(4), 421–438.
4. Fall, K. (2003). A Delay-Tolerant network architecture for challenged internets. In *Proceedings of ACM SIGCOMM*, pp. 27–34.
5. Franceschetti, M., Migliore, M. D., & Minero, P. The capacity of wireless networks: Information-theoretic and physical limits. Available at http://circuit.ucsd.edu/~massimo/papers.htm.
6. Grossglauser, M., & Tse, D. N. C. (2001). Mobility increases the capacity of ad-hoc wireless networks. In *Proceedings of IEEE Infocom*, pp. 1360–1369.
7. Gupta, P., & Kumar, P. R. (2000). The capacity of wireless networks. *IEEE Transactions on Information Theory, 46*(2), 388–404.
8. Jacquet, P., Mans, B., & Rodolakis, G. (2009). Information propagation speed in mobile and delay tolerant networks. In *Proceedings of IEEE Infocom*, pp. 244–252.

9. Keshavarz-Haddad, A., & Riedi, R. (2007). On the broadcast capacity of multihop wireless networks: Interplay of power, density and interference. In *Proceedings of IEEE SECON*, pp. 314–323.
10. Kolchin, V. F., Sevast'yanov, B. A., & Chistyakov, V. P. (1978). *Random allocations*. Washington, DC: V.H. Winston and Sons.
11. Kong, Z., & Yeh, E. (2008). On the latency of information dissemination in mobile wireless networks. In *Proceedings of ACM MobiHoc*, pp. 139–148.
12. Kuhn, F., Wattenhofer, R., & Zollinger, A. (2008). Ad hoc networks beyond unit disk graphs. *Wireless Networks, 114*(5), 715–729.
13. LeBoudec, J.-Y., & Vojnovic, M. (2005). Perfect simulation and stationarity of a class of mobility models. In *Proceedings of IEEE Infocom*, pp. 2743–2754.
14. Li, X. Y., Tang, S .J., & Frieder, O. (2007). Multicast capacity for large scale wireless ad hoc networks. In *Proceedings of ACM Mobicom*, pp. 266–277.
15. Liu, B., Towsley, D., & Swami, A. (2008). Data gathering capacity of large scale multihop wireless networks. In *Proceedings of IEEE MASS*, pp. 124–132.
16. Marco, D., Duarte-Melo, E. J., Liu, M., & Neuhoff, D. L. (2003). On the many-to-one transport capacity of a dense wireless sensor network and the compressibility of Its data. In *Proceedings of the IPSN, LNCS 2634*, 1–16.
17. Nakano, K., & Olariu, S. (2002). A survey on leader election protocols for radio networks. In *Proceedings of ISPAN*, pp. 71–77.
18. Neely, M. J., & Modiano, E. (2005). Capacity and delay tradeoffs for ad hoc mobile networks. *IEEE Transactions Information Theory, 51*(6), 1917–1937.
19. Ni, S., Tseng, Y., Chen, Y., & Sheu, J. (1999). The broadcast storm problem in a mobile ad hoc network. In *Proceedings of ACM Mobicom*, pp. 151–162.
20. Ozgur, A., Leveque, O., & Tse, D. (2007). Hierarchical cooperation achieves linear capacity scaling in ad hoc networks. In *Proceedings of IEEE Infocom*, pp. 382–390.
21. Pleisch, S., Balakrishnan, M., Birman, K., & van Renesse, R. (2006). MISTRAL: Efficient flooding in mobile ad-hoc networks. In *Proceedings of ACM MobiHoc*, pp. 1–12.
22. Resta, G., & Santi, P. (2009). Latency and capacity optimal broadcasting in wireless multi-Hop networks. In *Proceedings of IEEE ICC*.
23. Resta, G., & Santi, P. (2011). Latency and capacity optimal broadcasting in wireless multi-hop networks with arbitrary number of sources. *IEEE Transaction on Information Theory, 57*(12), 7746–7758.
24. Scheideler, C., Richa, A., & Santi, P. (2008). An *O*(*log n*) dominating set protocol for wireless ad hoc networks under the physical interference model. In *Proceedings of ACM MobiHoc*, pp. 91–100.
25. Shakkottai, S., Liu, X., & Srikant, R. (2007). The multicast capacity of large multihop wireless networks. In *Proceedings of ACM MobiHoc*, pp. 247–255.
26. Sharma, G., Mazumdar, R., & Shroff, N. (2006). Delay and capacity trade-offs in mobile ad hoc networks: A global perspective. In *Proceedings of IEEE Infocom*, pp. 1–12.
27. Wu, Y., & Li, Y. (2008). Construction algorithms for k-connected m-dominating sets in wireless sensor networks. In *Proceedings of ACM Mobihoc*, pp. 83–90.
28. Zheng, R. (2006). Information dissemination in power-constrained wireless networks. In *Proceedings of IEEE Infocom*, pp. 1–10.

Author Biographies

Giovanni Resta received the MS degree in computer science from the University of Pisa, Italy, in 1988. In 1996, he became a researcher at the Istituto di Matematica Computazionale of the Italian National Research Council (CNR), Pisa. He is now a senior researcher at the Istituto di Informatica e Telematica (CNR) in Pisa. His research interests include computational complexity (especially in relation to linear algebra problems), parallel and distributed computing, and the study of structural properties of wireless ad hoc networks.

Paolo Santi received the Laura Degree and Ph.D. degree in computer science from the University of Pisa in 1994 and 2000, respectively. He has been with the Istituto di Informatica e Telematica del CNR in Pisa, Italy, since 2001, first as reasearcher and now as senior researcher. During his career, he visited Georgia Institute of Technology in 2001, and Carnegie Mellon University in 2003. His research interests include fault-tolerant computing in multiprocessor systems (during Ph.D. studies), and, more recently, the investigation of fundamental properties of wireless multihop networks such as connectivity, lifetime, capacity, mobility modeling, and cooperation issues. He has contributed more than 60 papers and a book in the field of wireless ad hoc and sensor networking, he is Associate Editor of IEEE Trans. on Mobile Computing and IEEE Trans. on Parallel and Distributed Systems, he has been General Co-Chair of ACM VANET 2007 and 2008, and he is involved in the organizational and technical program committee of several conferences in the field. He is a member of IEEE Computer Society and a senior member of ACM and SIGMOBILE.