Candidate Selection Algorithms in Opportunistic Routing based on Distance Progress

Amir Darehshoorzadeh[1], Llorenç Cerdà-Alabern[1], and Vicent Pla[2]

[1]Univ. Politècnica de Catalunya, Computer Architecture Dep., Jordi Girona 1-3, E-08034, Barcelona, Spain
[2]Univ. Politècnica de Valencia, Communications Dep., ETSIT, Camino de Vera s/n, E-46022, Valencia, Spain

***Abstract*—Opportunistic Routing (OR) is a new class of routing protocols that selects the next-hop forwarder on-the-fly. In contrast to traditionally routing, OR does not select a single node as the next-hop forwarder, but a set of forwarder candidates. When a packet is transmitted, the candidates coordinate such that the best one receiving the packet will forward it, while the others will discard the packet. The selection and prioritization of candidates, referred to as *candidate selection algorithm*, has a great impact on OR performance. In this paper we propose and study two new candidate selection algorithms based on the geographic position of nodes. This information is used by the candidate selection algorithms in order to maximize the *distance progress* towards the destination. We compare our proposals with other well-known candidate selection algorithms proposed in the literature through mathematical analysis and simulation. We show that candidate selection algorithms based on distance progress achieve almost the same performance as the optimum algorithms proposed in the literature, while the computational cost is dramatically reduced.**

I. Introduction

Opportunistic Routing (OR) [4], also referred to as cooperative forwarding [14] or any-path routing [11], has been proposed to increase the performance of Wireless Mesh Networks (WMNs) by taking advantage of its broadcast nature. In OR, in contrast to traditional routing, instead of preselecting a single specific node to be the next-hop as a forwarder for a given destination, an ordered set of nodes (referred to as candidates) are selected as the potential next-hop forwarders. Thus, the source can use multiple potential paths to deliver the packets to the destination. After the packet has been transmitted, the candidates that successfully receive it will coordinate among themselves to determine which one would actually forward it, while the others will simply discard the packet.

One of the main issues in OR is selection and priority assignment to candidates. All nodes in the network must run an algorithm for selecting and sorting the set of neighboring nodes that can better help in the forwarding process to a given destination. We shall refer to this as *candidate selection algorithm, CSA*. The aim of CSAs is minimizing the expected number of transmissions from the source to the destination. Numerous routing protocols building on the idea of OR have been proposed [3, 22, 19, 12, 16]. Apart from other implementation aspects, these solutions specify how to select and prioritize the candidates.

Choosing and sorting appropriately the candidates is essential to maximize the gain of OR over traditional routing. For this reason, CSA is the topic that has been investigated the most in OR. Some proposals, such as ExOR [3], are simple to implement and fast, but their result are far from optimal. On the other extreme, some proposals are able to achieve an optimal selection of candidates (e.g., MTS [16]). However, the candidate selection for a node depends on the candidates chosen by its neighbors, and so on until the destination. Therefore, optimum candidate selection requires a perfect knowledge of the whole network topology. Additionally, the computational cost of evaluating the mathematical formulas used in optimum CSAs increases very rapidly with the number of nodes in the network [9].

In this paper, we propose and study two CSAs that leverage geographic information with the aim of striking the right balance between performance and cost. More specifically, location information of neighbor nodes and the destination is used to estimate the expected *D*istance Progress (DP) towards the destination at every transmission shot. Clearly, maximizing the DP is equivalent to minimizing the expected number of transmissions. Moreover, CSAs based on DPs can lead to algorithms that are much faster and require less information that traditional topology-based CSAs.

Following this idea we propose two DP-based CSAs. The first one, that we call *Distance Progress Based Opportunistic Routing* (DPOR) relies on link delivery probabilities between the forwarder and its neighbors, and on the geographic position of the latter. DPOR uses this information to estimate the DPs, and select the candidates. The second one, that we call *Candidate selection based on Maximum Progress Distances* (CMPD), the candidate selection of a node is based on what would be the optimal positions of its candidates, and the actual positions of its neighbors.

We compare DPOR and CMPD with other CSAs in terms of the expected number of transmissions needed to send a packet from the source to the destination, and the execution time of each algorithm using numerical tool. Furthermore, the performance of our proposals and other CSAs has been investigated through simulation. In our comparison we use MTS [16], which is optimal but computational costly, as a benchmark for performance, and ExOR [4] as benchmark for computational cost. The results of proposed CSAs yield a very good relative performance, while their cost is comparable or even lower than that of ExOR.

The remainder of this paper is organized as follows. Section II surveys the related work. In Section III, we introduce a new metric (EDP) and describe the first of our CSAs proposals based on this metric, DPOR. Section IV describes the second CSA proposed in this paper, CMPD. In Section V, we explain the methodology of our experiments, and present and discuss numerical and simulation results comparing the CSAs proposed in this paper with other relevant CSAs proposed in the literature. Finally, some concluding remarks are made in Section VI.

II. Related work

Biswas and Morris proposed ExOR [4], one of the first and most referenced OR protocols. The selection of candidates in ExOR is based on the metric called *Expected Transmission Count*, (ETX) [10], which is computed assuming unipath routing. Thus, using ETX does not seem an appropriate metric for OR. In [24], Zhong et al. proposed a new metric —*expected any-path transmission*, (EAX)— that generalizes ETX to an OR framework. MORE [7] is a MAC independent protocol that uses both the idea of OR and network coding. It avoids duplicate transmissions by randomly mixing packets before forwarding. In [13], a distributed algorithm for computing minimum cost opportunistic routes is presented. The authors also alert about the risk of using too many relay candidates. In [16] the key problem of how to optimally select the forwarder list is addressed, and an optimal algorithm that minimizes the expected total number of transmissions is developed. Different OR candidate selection algorithms are compared in [8] in terms of the expected number of transmissions from source to the destination and the execution time to find the candidates sets.

There are some papers which propose analytic models to study the performance of OR. Baccelli et al. [2] used simulations to show that OR protocols significantly improve the performance of multihop wireless networks compared to the shortest path routing algorithms, and elaborated a mathematical framework to prove some of the observations obtained by the simulations. In [18] an algebraic approach is applied to study the interaction of OR routing algorithms and routing metrics. In [6, 9] a Markov model to assess the improvement that may be achieved using opportunistic routing has been proposed. At the same time, Li and Zhang published an analytic framework to estimate the transmission costs of packet forwarding in wireless networks [17]. In [5] the authors derived the equations that yield the distances of the candidates in OR such that the per transmission progress towards the destination is maximized. There, a lower bound to the expected number of transmissions needed to send a packet using OR is also derived.

Geographic Random Forwarding (GeRaF) [25] is a geographical forwarding protocol which selects a candidates set and prioritizes them using location information. Only those neighboring nodes closer to the destination than the sender can be included in the candidates set. The priority of selected candidates is based on their geo-distances to the destination. The candidates set selection and prioritization can easily be implemented via an RTS-CTS dialog at the MAC layer, which also ensures that a single forwarder is chosen. GOR [23] is used in geographic routing scenarios and adopts timer-based coordination with local candidate order. Authors showed that giving higher priority to the nodes closer to the destination does not always yield the optimal throughput. They proposed a local metric named *Expected One-hop Throughput* (EOT) to characterize the local behavior of GOR in terms of bit-meter advancement per second. Based on EOT, which considers the coordination overhead, they proposed a candidate selection scheme. S.Yang et. al. [21] used the idea of opportunistic routing in the position-based protocols and proposed a protocol called position based opportunistic routing, POR. They fixed the maximum number of candidates in each node to 5. When a candidate receives a packet, it checks its position in the candidates set and waits for some time slots to forward the packet; if a transmission of this packet is heard during the waiting time the packet will be discarded.

Our proposal in this work is differentiated from those in other works in the sense that our CSAs depend on the local information of the neighbors. The proposed CSAs only need the geographic position of neighboring nodes and, in one of them (DPOR), the link delivery probability to reach the neighbors. Therefore, these two CSAs can be considered as two fast CSA which need less information to obtain the candidate set while their performance is close to the optimum CSA proposed in [16].

III. Distance Progress Based Opportunistic Routing (DPOR)

In this section, we define a new metric to estimate the expected distance progress achieved in a transition of packet, as a function of the set of candidates. Then, based on this metric we propose a candidate selection and prioritization algorithm to maximize the expected distance progress towards of a packet transmission.

Let $\mathcal{N}$ be the set of nodes in the network, and denote by s the source node and by d the destination node. We assumed that: (i) all nodes $v \in \mathcal{N}$ know the position coordinates of their neighbors ($N(v)$), (ii) each node v knows the link delivery probability between v and its neighbors ($p_{v,i}, i \in N(v)$), and (iii) all nodes know the position of the destination. This assumptions could be easily implemented, e.g. by using a location registration and lookup service which maps node addresses to locations as in [21, 15].

A. Expected distance progress

Let $D_{i,d}$ be the geographic distance between node i and destination d. The *Distance Progress* of a data packet sent by source s towards destination d using next-hop c_i is given by: $DP_{c_i}^{s,d} = D_{s,d} - D_{c_i,d}$. We define the *Expected Distance Progress* (EDP) from node s to the destination d using candidates set $C^{s,d} = \{c_1, c_2, \cdots, c_n\}$ (with c_1 being the highest priority, and c_n the least one) as:

$$\mathrm{EDP}(s,d,C^{s,d}) = \sum_{i=1}^{n}(D_{s,d}-D_{c_i,d})\times p_{s,c_i}\prod_{j=1}^{i-1}(1-p_{s,c_j}) = \sum_{i=1}^{n} DP_{c_i}^{s,d}\times p_{s,c_i}\prod_{j=1}^{i-1}(1-p_{s,c_j}) \quad (1)$$

Where $p_{i,j}$ is the delivery probability of the link between node i and j. Note that upon a packet transmission, the higher the EDP, the higher is the expected approach of the packet to the destination.

Intuitively, increasing the number of candidates would result in a larger EDP. Additionally, the maximum EDP for a given candidates set of $C^{s,d}$ can only be achieved by assigning the priority to each node based on their distances to the destination. That is, the node closest to the destination among the candidates receiving the packet should try to forward it first; if it did not receive the packet, the second closest node should try, and so on.

B. *EDP candidate selection*

In this section we propose a candidate selection algorithm that we call *Distance Progress Based Opportunistic Routing*, DPOR, which tries to maximize the EDP. Algorithm 1 shows the pseudo-code of DPOR. It is worth mentioning that DPOR considers not only the closeness of candidates to the destination, but also the link delivery probability between the forwarder and the candidates. Basically, DPOR selects the candidates trying to balance the closeness to the destination, and link delivery probability between the forwarder and the candidates.

Algorithm 1 works as follows: assume that a generic node s wants to choose its candidates set for a specific destination d. First, node s finds its neighbors which are closer to the destination than itself. We shall refer to this set as $N(s)$. A neighbor j of s is included in $N(s)$ only if $D_{j,d} < D_{s,d}$. Then, node s selects, among its neighbors, the candidate that increases the most the EDP toward the destination (line 4); this candidate is added to the candidates set $C^{s,d}$ and removed from the neighbors set (lines 7 and 8). This process is repeated until there is not any other suitable node to be included in the candidates set of s, or the number of candidates in $C^{s,d}$ reaches the maximum number of candidates (n). Note that in each iteration $\mathrm{EDP}(s,d,C^{s,d})$ is calculated ordering the candidates $C^{s,d} = \{c_1, c_2, ..., c_n\}$ by their distance to the destination, i.e, $D_{c_1,d} < D_{c_2,d} < ... < D_{c_n,d}$. We remark that in DPOR, each node i selects its candidates set independently from other nodes' candidates set, and knowing only the position of its neighbors and the delivery probability towards them.

IV. Candidate selection algorithm based on MPD

In this section we first summarize the algorithm proposed in [5], which derives the optimum position of the candidates. We then propose a new CSA based on this information.

Algorithm 1: Candidate.selection.DPOR(s, d, n).

```
1  $m_p \leftarrow -1$
2  $N(s) = \{n = neighbor(s) | D_{n,d} < D_{s,d}\}$
3  while $|C^{s,d}| < n$ do
4      $cand \leftarrow \arg\max_{c \in N(s)} \mathrm{EDP}(s, d, C^{s,d} \cup c)$
5      $m_c \leftarrow \mathrm{EDP}(C^{s,d} \cup cand, s, d)$
6      if $m_c > m_p$ then
7          $C^{s,d} \leftarrow C^{s,d} \cup cand$
8          $N(s) \leftarrow N(s) \setminus cand$
9          $m_p \leftarrow m_c$
10     else
11         break
12     end
13 end
```

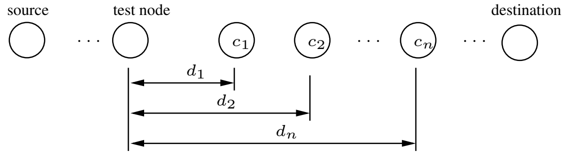

Fig. 1. Test node and its candidates.

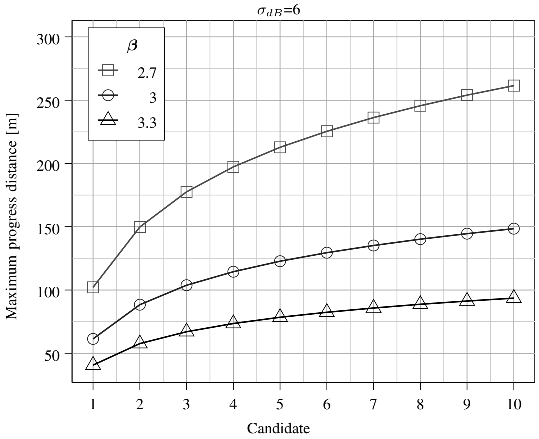

Fig. 2. Maximum progress distances for the candidates.

A. *Maximum Progress Distances*

The idea of the algorithm is computing the position of the candidates that maximize the progress of transmitted packets towards the destination. The components of the model are: the maximum number of candidates per node, n, and the formula for the delivery probability at a distance d, $p(d)$, which we suppose to be the same for all the nodes. Assume that the destination is far from a generic test node whose candidates we are looking for. Clearly, the optimum candidates will be located over the segment between the test node and the destination (see Figure 1).

Let $\{c_1,\ c_2, \cdots c_n\}$ be the ordered set of candidates of the generic test node (c_n is the highest priority, and c_1 the lowest one), and d_i the distance from the test node to the candidate c_i (see Figure 1). We assume that a coordination protocol exist among the candidates, such that the highest priority candidate receiving the packet will forward the packet (if it is not the destination), while the other nodes will simply discard it. Assume that $p(d_i)$ is the delivery probability from the test node to the candidate c_i, and let Δ_n be the random variable equal to the distance reached after one transmission shot. Clearly,

$$\mathrm{E}[\Delta_n] = d_n\, p(d_n) + d_{n-1}\, p(d_{n-1})\,(1 - p(d_n)) + \cdots + d_1\, p(d_1) \prod_{i=2}^{n} (1 - p(d_i)) \qquad (2)$$

That is, the packet will progress a distance d_n if the most priority candidate n receives it, or a distance d_i ($i = 1, \cdots n-1$) if candidate i receives it, and no higher priority candidates receive the packet. A key observation is that Equation (2) can be rewritten recursively as:

$$\mathrm{E}[\Delta_n] = d_n\, p(d_n) + (1 - p(d_n))\, \mathrm{E}[\Delta_{n-1}] = \mathrm{E}[\Delta_{n-1}] + (d_n - \mathrm{E}[\Delta_{n-1}])\, p(d_n). \qquad (3)$$

We are interested in finding the value $d_n \in (d_{n-1}, \infty)$ that maximizes Equation (3). In [5], it is derived that these values can be computed using the set of equations:

$$p(d_i) + (d_i - \mathrm{E}[\Delta_{i-1}])\, p'(d_i) = 0, \quad d_i \in (d_{i-1}, \infty),\ i = 1, \cdots, n \qquad (4)$$

where $\mathrm{E}[\Delta_0] = 0$ and $d_0 = 0$. Note that using (4) we can compute d_1 by solving $p(d_1) + p'(d_1)\, d_1 = 0$. Then, substituting in (3) we have $\mathrm{E}[\Delta_1] = d_1\, p(d_1)$, which can be used to compute d_2 using (4), and so on until d_n. We shall refer to these distances as the *Maximum Progress Distances*, MPD. In the sequel we shall refer to them as $d_1, \cdots, d_n$, and denote the expected number of transmissions given by Equation (3) using these distances as $\mathrm{E}[\Delta^*_n]$. Note also that a consequence of Equation (4) is that the maximum progress distances for the already existing candidates in the set do not change if we decide to add a new candidate to the candidate set.

Figure 2 shows the maximum progress distances for different number of candidates. This figure has been obtained assuming that $p(d)$ is given by the shadowing propagation model used to obtain the numerical results. This model will be summarized in section V-A. Figure 2 shows three curves, which correspond to three values of the loss exponent of the propagation model: $\beta = 2.7$, $\beta = 3$ and $\beta = 3.3$. Note that the larger is β, the lower is the transmission range of the nodes, and thus, the shorter are the distance of the candidates.

B. MPD candidate selection

In this section we proposed a new candidate selection algorithm based on MPD that we call *Candidate selection based on MPD*, CMPD. It tries to select the candidates that are located near the positions given by the MPD.

Let $\mathcal{N}$ be the set of nodes in the network, and denote by s the source node and by d the destination node. We assume that all nodes $v \in \mathcal{N}$ know the position coordinates of their neighbors ($N(v)$) and the destination d. This assumptions could be easily implemented, e.g. by using a location registration and lookup service which maps node addresses to locations as in [21, 15]. We have used $D_{x,y}$ to refer the geographic distance between two nodes x and y.

Algorithm 2: Candidate.selection.MPD(v, d, n).

```
   Data:
   $D_{x,y}$: Geographic distance between nodes $x$ and $y$
   $\hat{c}_i$: $i$th candidate which is located at the optimum position.
   $d_i$: Geographic distance between $v$ and $\hat{c}_i$.
1  if $D_{v,d} < d_n$ then
2      for $i$=1 to$n$ do
3          $d_i \leftarrow d_i * D_{v,d}/d_n$;
4      end
5  end
6  $N(v) = \{n = neighbor(v) | D_{n,d} < D_{v,d}\}$
7  $i \leftarrow 1$
8  $C^{v,d} \leftarrow \emptyset$
9  while $|C^{v,d}| < n$ & $N(v) \neq \emptyset$ do
10     $cand \leftarrow \arg\min_{c \in N(v)} D_{c,d_i}$
11     $C^{v,d} \leftarrow C^{v,d} \cup cand$
12     $N(v) \leftarrow N(v) \setminus cand$
13     $i \leftarrow i + 1$
14 end
15 Order $C^{v,d}$ according to $D_{c_i,d}, c_i \in C^{v,d}$
```

Algorithm 2 shows the pseudo-code of CMPD for a node v to select its candidates set to reach the destination d. The parameter n in Algorithm 2 is the maximum number of candidates in each node. Let $\hat{c}_i$ be a virtual candidate of v that lies on the straight line between v and the destination d at distance d_i. The value of d_i is given by the previous results obtained in Section IV-A (see Figure 2). Note that, the obtained value for d_i, $i \in \{1, 2, \ldots, n\}$, is valid when the destination is far away from the forwarder (i.e., $D_{v,d} > d_n$. Therefore, when the distance between source and the destination is shorter than d_n we shrink the MPD distances (d_i, $i \in \{1, 2, \ldots, n\}$) such that $d_n = D_{v,d}$ (see lines 1– 5). The corresponding candidate c_i is chosen as the node in $N(v)$ which is the one closest to $\hat{c}_i$, (i.e, $c_i = \arg\min_{c \in N(v)} D_{c,d_i}$). Note that c_i should be closer than v to the destination ($D_{c_i,d} < D_{v,d}$). Finally, the candidates set is order according to the closeness of each candidate to the destination. The candidates which is nearer to the destination will have higher priority.

V. Performance Evaluation

To evaluate the performance of the CSAs proposed in this paper (DPOR and CMPD) we have compared them with two other CSAs that have been proposed in the literature: ExOR [4, 3] and MTS [16]. The results are obtained using both an analytic procedure (Section V-B) and computer simulation (Section V-C). Analytic results have been obtained as follows: once the position of the nodes is decided, the delivery probabilities are computed using a shadowing propagation model (described in Section V-A). Then, the CSA is run to assign the candidates. Finally, the expected number of transmissions is computed analytically, as explained in [9]. The analytic results have been obtained using R [20].

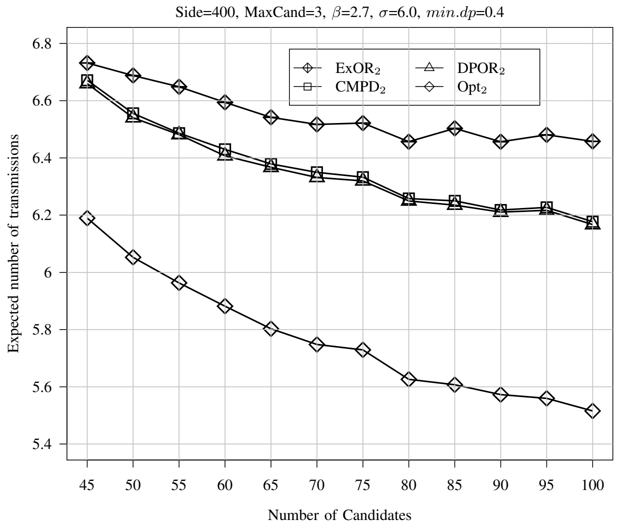

Fig. 3. Expected number of transmissions varying the number of nodes for a maximum number of candidates $n = 2$.

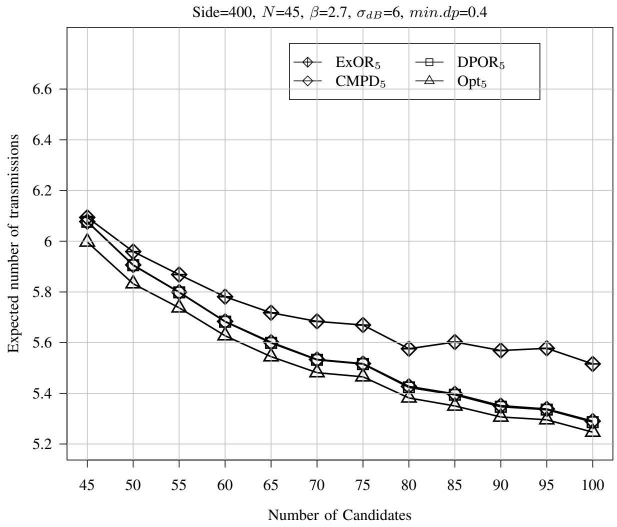

Fig. 4. Expected number of transmissions for a maximum number of candidates $n = 5$.

The simulation results have been obtained using *network simulator* (ns-2) [1]. In this case, the nodes have been placed at the same positions as in the analytical model, and the simulation has been run using the same shadowing propagation model. However, in the simulation a realistic 802.11 MAC is used. Additionally, node coordination is also considered.

As mentioned above, we have selected two other well-known CSAs for our comparisons: ExOR [4, 3] as a simple and fast candidate selection algorithm, and MTS [16], which finds the optimum candidate sets in the sense that the expected number of transmissions from source to the destination is minimized. In the following we briefly describe these two algorithms (see [9] for more details).

Extremely Opportunistic Routing (ExOR) [4, 3] is the most well-known OR protocol. The basic idea of ExOR is running the *shortest path first* (SPF) with the Expected Transmission count (ETX) [10] of each link as the weight of link to find the shortest path. The first node after the source is selected as candidate. Then, ExOR removes the link between the current node and the selected candidate, and re-runs the SPF algorithm with the new topology. The node that is searching for candidates continues the above process until it finds all necessary candidates. ExOR uses the ETX of each candidate to reach the destination as the metric for the prioritization of the selected candidates. The candidate with lower ETX to the destination will be assigned a higher priority.

Minimum Transmission Selection (MTS) [16]: Yanhua Li et al. [16] proposed *Expected Any-path Transmissions* (EAX) as a new metric for OR which calculates the expected number of transmissions from source to the destination when OR is used. Based on this new metric, an optimum candidate selection algorithm was proposed. Recall that, for a given destination n_d, the optimum candidates set for a node n_s is a set of nodes that, when used as the candidates set of n_s, minimize the expected total number of transmissions from n_s to n_d. The general idea of MTS is to start by looking for the candidate sets of the nodes closer to the destination and then proceed backwards to the source. It adds the destination as the candidate for all of the nodes in the neighborhood of the destination. Clearly, the nodes which are close to the destination have fewer EAX than those which are a further distance away. The algorithm finds the node v with the least EAX to the destination. Then the neighbors of v add v and its candidates to their candidate set as the new candidates and update their EAX value. The process of finding a node with the lowest EAX and adding it and its candidates set to the neighboring node will be continued until the source finds its candidates set. The EAX of each selected candidate is used to assign the priority to each candidate.

A. Propagation Model

The prediction for received power in the two-ray and free space propagation models is a deterministic function. On the other hand, due to fading effects, the received signal strength at a certain distance is a random variable. In order to model the delivery probabilities we will assume that the channel impairments are characterized by a shadowing propagation model, which is a more general model in wireless networks. In contrast to the deterministic models where each existing link is perfect, shadowing model consists of deterministic path loss and large scale fading. Packets are correctly delivered if the received power at a distance d, $P_r(d)$, is greater than or equal to a reception threshold `RXThresh`. The probability of this event is given by:

$$p(d) = Prob(P_r(d)|_{dB} \geq 10\,\log_{10}(\texttt{RXThresh})) = \\ Q\left(\frac{1}{\sigma_{dB}} 10\,\log_{10}\left(\frac{\texttt{RXThresh}\; L\,(4\,\pi)^2\,d^{\beta}}{P_t\,G_t\,G_r\,\lambda^2}\right)\right) \quad (5)$$

where $Q(z) = \frac{1}{\sqrt{2\,\pi}}\int_z^{\infty} \mathrm{e}^{-y^2/2}dy$. Here G_t and G_r are the transmission and reception antenna gains respectively, L is a system loss, λ is the signal wavelength (c/f, with $c = 3 \times 10^8$ m/s), β is a path loss exponent, and σ_{dB} is the standard deviation of the zero mean Gaussian random variable that models the fading.

In our numerical experiments we have set the model parameters to the default values used by ns-2 [1], given in Table I. Table II shows typical values for β and σ_{dB}.

TABLE I
DEFAULT NS VALUES FOR THE SHADOWING PROPAGATION MODEL.

Parameter	Value
P_t	0.28183815 Watt
`RXThresh`	3.652×10^{-10} Watt
G_t, G_r, L	1
f	914 MHz

TABLE II
TYPICAL VALUES FOR β AND σ_{dB}.

Environment		β	σ_{dB}
Outdoor	Free space	2	4 $\sim$12
	urban	2.7 $\sim$ 5	
Office	Line-of-sight	1.6 $\sim$ 1.8	7 $\sim$ 9.6
	Obstructed	4 $\sim$ 6	

B. Analytic Results

In order to compare different algorithms, and since we want to focus on the effect of candidate selection, we have assumed that the nodes continue transmitting the packet until at least one candidate receives it. Furthermore, we have assumed that candidate coordination is done perfectly. That is, the highest priority candidate receiving the packet will forward the packet and the other candidates will discard it.

We consider scenarios with different number of nodes ($45 \leq N \leq 100$) randomly placed in a square with sides equal to 400 m, except the source and the destination which are placed at the diagonal end points. Each point in the plots is an average of 100 runs with different random node positions. The delivery probabilities have been assigned with the shadowing model with $\beta = 2.7$ and $\sigma_{db} = 6.0$.

In the candidate selection algorithm, we have assumed that a link between any two nodes exists only if the delivery probability between them is at least $min.dp = 0.4$. We have compared the algorithms for different maximum number of candidates: $n = 2, 3, \ldots, 5$. In the following figures, we shall use the notation ExOR$_n$ to refer to ExOR with maximum number of candidates n, and similarly for the other algorithms under study.

Using numerical results we compare the performance of each algorithm in terms of the expected number of transmissions needed to send a packet from the source to the destination and the execution time which is needed to find the candidates sets in each algorithm.

1) Expected number of transmissions: Figure 3 shows the expected number of transmissions of the different algorithms when the maximum number of candidates is $n = 2$. The curves have been obtained varying the number of nodes, but maintaining the distance $D = 400\sqrt{2}$ m between the source and the destination, thus, increasing the density of the network. In all figures we use the notation Opt$_n$ to refer to the optimum candidate selection algorithm (MTS) selecting a maximum number of candidates n.

As a first observation in Figure 3, we see that increasing the number of nodes causes a decrease in the expected number of transmissions in all algorithms. Figure 3 also shows that ExOR has largest expected number of transmissions, and the optimum algorithm (MTS) the lowest. The curves for CMPD and DPOR lie in between those of ExOR and the the optimum algorithm, and they are close to each other. Recall that CMPD chooses the closest nodes to the virtual nodes located at the optimal positions, and DPOR selects the candidates that yield the lowest EDP.

Obviously, increasing the number of candidates for each node decreases the expected number of transmissions. Figure 4 shows the expected number of transmissions of each algorithm with the maximum number is set to $n = 5$. The expected number of transmissions for DPOR and CMPD is very close to the optimum algorithm, while ExOR still has a higher expected number of transmissions than the others, especially when the number of nodes in the network increases.

In another experiment, we set the number of nodes to $N = 45$ and 100 and vary the maximum number of candidates to $n = 2, 3, \ldots, 5$. From this point forward, we refer to the scenarios with 45 and 100 nodes as the low and high density network, respectively. The results of the expected number of transmissions for the low and high density networks varying the number of candidates are shown in Figures 5 and 6, respectively. As we can see, increasing the number of candidates results in a reduction in the expected number of transmissions; this occurs for all the CSAs and for both the low and high density networks. It is also observed that the advantage of the optimal CSA over the other CSAs shrinks when the maximum number of candidates, n, is increased. In particular, when $n = 5$ and the network density is high (Figure 6) the advantage of the optimal algorithm over the ones proposed in this paper is negligible. This is due to the fact that in a dense network there is a large number of possible choices for the candidate sets. Therefore, CMPD and DPOR will select the nodes as the candidates which are similar to the candidates selected by the optimum algorithm.

2) Execution Time: In this section we evaluate the computational cost of the algorithms under study by measuring the *execution time* to compute all the necessary candidates sets to send packets from the source towards the destination. The algorithms were run on a PC with 2 processors Intel Xeon Quad-Core 2.13 GHz and 24 GB of memory.

Figure 7 shows the the execution times in a logarithmic scale. We have selected a maximum number of candidates $n = 3$ as a sample case for our study. As expected, the optimal algorithm is by far the slowest one. For instance, when the number of nodes in the network is equal to 100, the optimum algorithm needs about 680 seconds. Obviously, with more than 3 candidates per node ($n > 3$) the execution time will be much larger. At the other end of the scale, CMPD is not only the fastest CSA, but also the one with the lowest increase rate of the execution time when the number of nodes in the network grows: the curve for CMPD is relatively flat compared with the other three curves. The execution time, and also its growth rate, of ExOR and DPOR lie in between those of CMPD and MTS. It is worth mentioning here that, although we took

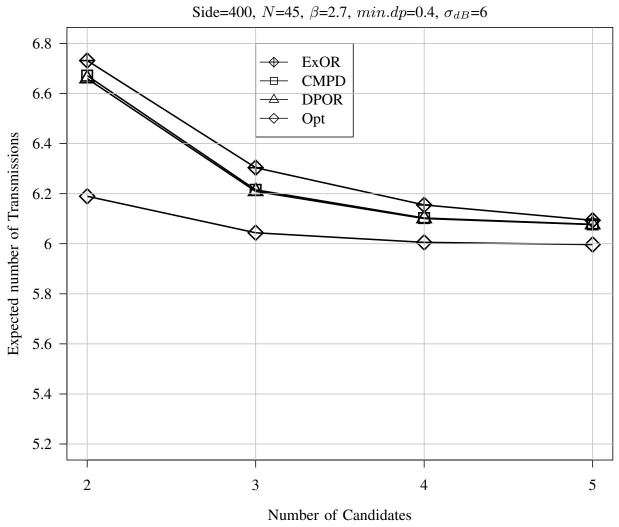

Fig. 5. Expected number of transmissions varying the maximum number of candidates for $N = 45$ nodes.

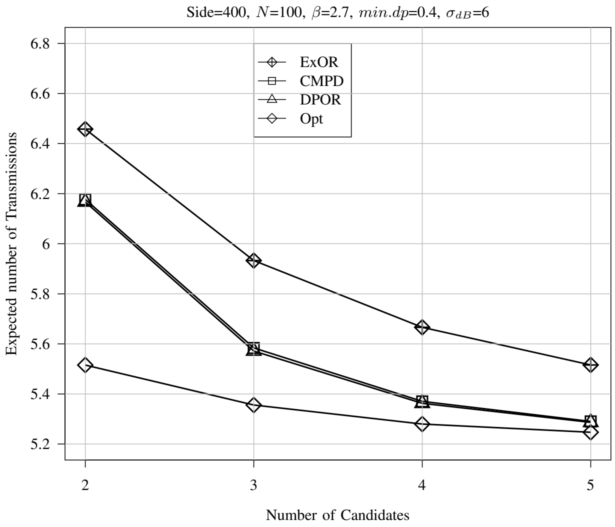

Fig. 6. Expected number of transmissions varying the maximum number of candidates for $N = 100$ nodes.

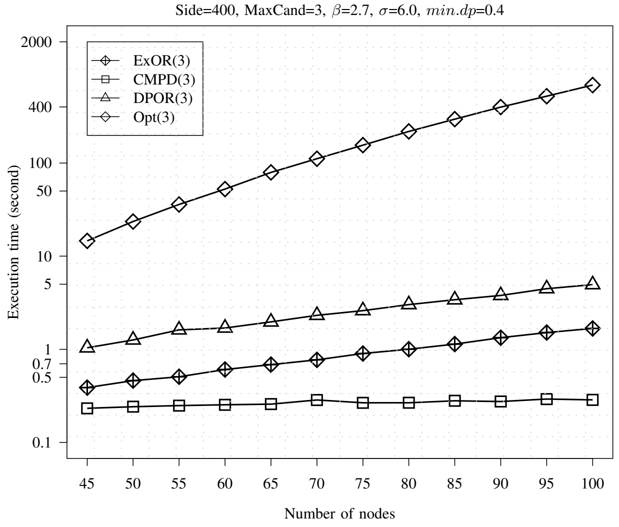

Fig. 7. Execution time of different algorithms for a maximum number of candidates $n = 3$.

ExOR as the reference for a simple CSA, CMPD outperforms it not only in the expected number of transmissions but also on execution speed. Furthermore, ExOR needs to know the whole network topology to find the candidates sets, while CMPD just needs local information (position of the neighboring nodes and delivery probability to them) and the position of the destination.

C. Candidate coordination during simulation

As mentioned in Section I, each OR protocol has two parts: candidate selection and candidate coordination. Since the main goal in this paper is to compare the performance of different candidate selection algorithms, we have implemented a perfect coordination between candidates in ns-2 such that the highest priority candidate that has received the packet will forward the packet while the lower priority ones will discard the packet. The *timer-based* approach is used as the coordination method for all protocols under study. In this method, each candidate c_i has to wait for a time T_{c_i} before transmitting. The higher the priority of the candidate, the shorter the waiting time. We have used $T_{c_i} = (i-1) \cdot T_{Default}$, where $T_{Default}$ is a predefined time that in our simulations has been set to $50\ ms$. Therefore, the highest priority candidate (c_1) will not wait, the second candidate (c_2) will wait for $1 \cdot T_{Default} = 50\ ms$, c_3 will wait for $2 \cdot T_{Default} = 100\ ms$, and so on.

We used the simulations to asses and compare the CSAs under study in terms of: expected number of transmissions, hop-count, and end-to-end delay.

D. Expected Number of Transmissions

Figure 8 shows the expected number of transmissions of the different protocols under study when the maximum number of candidates is equal to 3 ($n = 3$) and varying the number of nodes in the network. For the sake of comparison, we have included the numerical results obtained in Section V-B1. The labels $\texttt{prot-name}_n^{Num}$ and $\texttt{prot-name}_n^{Sim}$ refers to the numerical results obtained analytically and by simulation, respectively, with a maximum number of candidates equal to n. We observe that, except in the case of the optimal protocol for the other three cases the analytic and simulation results almost coincide. In the case of the optimal protocol the simulation results are worse than the analytic one, and the difference between them increases as the number of nodes in the network grows. Indeed, the curve for the simulation results of the optimal protocol approaches the curves of DPOR and CMPD when the network size increases. Thus, when the assumptions of the analytic model are relaxed, the advantage of the optimal protocol over DPOR or CMPD diminishes, and it vanishes in a dense network. On the other hand the advantage of DPOR and CMPD over ExOR is maintained in the simulation results.

E. Hop-count

Figure 9 depicts the results of different protocols in terms of average number of hop-count of received packets to the

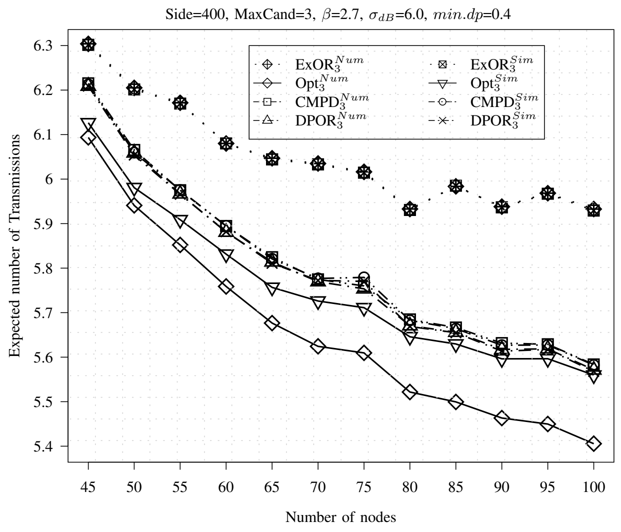

Fig. 8. Expected number of transmissions varying the number of nodes for a maximum number of candidates $n = 3$.

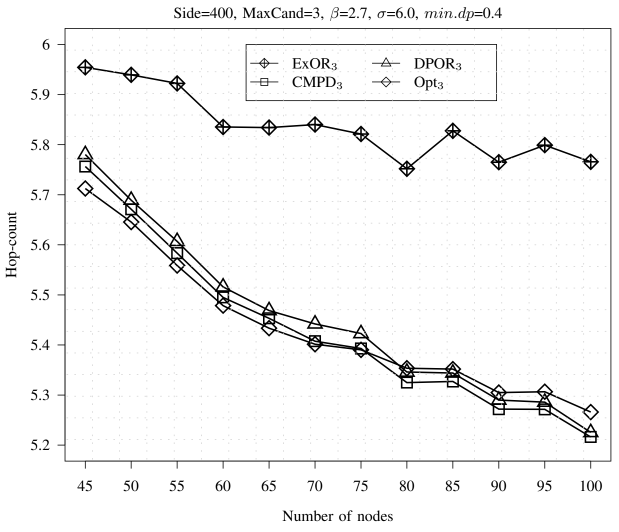

Fig. 9. Hop-count varying the number of nodes for a maximum number of candidates $n = 3$.

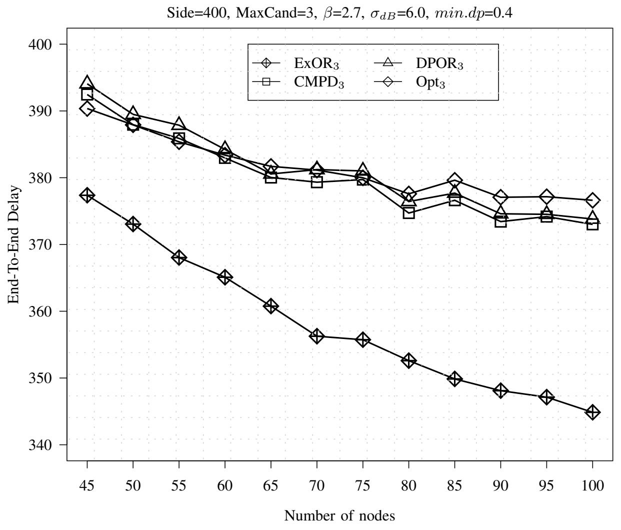

Fig. 10. End-to-End delay of OR protocols for a maximum number of candidates $n = 3$.

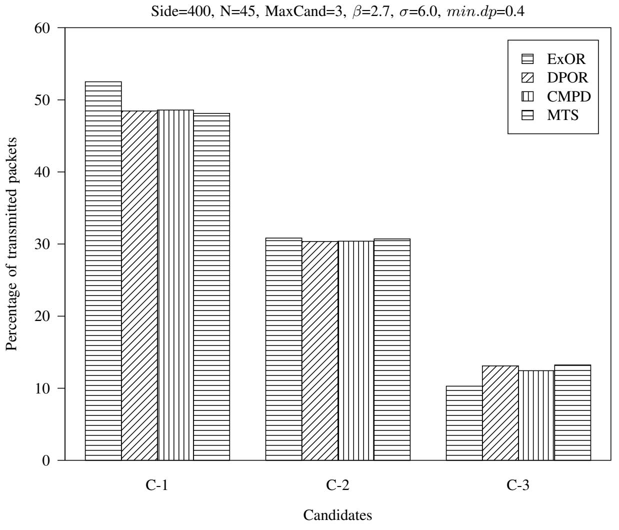

Fig. 11. Percentage of transmissions by each candidate of OR protocol for N=45 and a maximum number of candidates $n = 3$.

destination. As we can see the average hop-count of DPOR, CMPD and MTS is very close to each other, while ExOR has a higher average hop-count. Moreover, the difference between ExOR and the other protocols grows when the number of nodes in the network increases. This comes from the fact that DPOR and CMPD select the candidates based on the position of nodes while ExOR just considers the ETX of each link to the neighboring nodes for its decision in selecting the candidate sets. Clearly, having more nodes in the network causes all algorithms select better candidates which are closer to the destination and the packets can reach the destination with fewer hops.

F. End-to-End delay

In addition to the expected number of transmissions and average hop-count of each OR protocol, we have obtained the results of end-to-end delay of the different protocols under study. This is a very important performance metric that, to the best of our knowledge, has not been evaluated before for OR protocols.

Figure 10 shows the end-to-end delay of received packet to the destination varying the number of nodes while the maximum number of candidates is set to $n = 3$. It is clear that increasing the number of nodes in the network gives the chance to the OR protocols to select better candidates and, therefore, the end-to-end delay is reduced. The results for all protocols except ExOR is almost the same. As we can see the end-to-end delay of ExOR is better than the other protocols. Considering the obtained results for the hop-count we would have expected that the end-to-end delay of the other protocols was better than that of ExOR, since they have a lower hop-count. Actually these apparently contradictory results come from the candidate coordination phase in OR, and the packet retransmissions triggered by a timer because none of the candidates has received the packet correctly. Now we examine each of these two factors separately.

Recall that, the lower priority candidates in OR will forward a received packet if none of the higher priority ones has forwarded it. Therefore, each candidate except the highest priority one has to wait for some time before it can proceed to forward the packet. Consider now Figure 11. This figure shows the percentages of packets which are sent through the first $(C-1)$, second $(C-2)$ and the third candidate $(C-3)$ when nodes are equal to 45 (N=45) and the number of candidates is set to 3 $(n = 3)$. As we can see, about 53% of packets in ExOR transmitted through the first candidate while the other protocols send only about 48% of packets through their first candidate, and the remaining packets are forwarded using the second and third candidates. Therefore, because of the timer-based approach, the second and third candidates have to wait for some time before forwarding the packet. Since in CMPD, DPOR and MTS there are more packets which are transmitted through the second and, especially, the third candidates, the end-to-end delay of them will be higher than in ExOR.

As mentioned above, the delay introduced by packet re-transmissions also affects the end-to-end delay results. In OR protocols, if none of the candidates of a forwarder receives the packet, the packet will be re-transmitted. The forwarder node realizes that none of its candidates has received correctly a packet because it does not overhear any of them to forward the packet onto the next hop. Therefore, the duration of the re-transmission timer has to be lower bounded by the time the lowest priority candidate has to wait before transmitting (i.e., $(n-1) \times T_{Default}$ where n is the maximum number of candidates). Thus, it is clear that retransmissions could degrade end-to-end delay more severely than the timer-based coordination mechanism. Figure 12 depicts the number re-transmissions relative to the number of transmissions for the each of the protocols under study. As revealed by the graph the number of re-transmissions in ExOR is significantly lower than in the other three protocols.

Therefore, in ExOR, the lower number of re-transmissions, and the higher proportion of packets forwarded by the first candidate, can outweigh the effect of a higher hop-count on the end-to-end delay. Indeed, the results in Figures 10–12 tell us this is what is occurring in the studied scenarios.

VI. Conclusion

In this paper we have proposed and analyzed two candidate selection algorithms (CSAs) for Opportunistic Routing based on *distance progress*. Such CSAs take into consideration the geographic position of the nodes, and use it with the aim of maximizing the *distance progress* towards the destination. In contrast, most CSAs proposed in the literature are *topology based*. This type of CSAs use the whole network topology and link delivery probabilities, with the aim of minimizing the expected number of transmissions. Using distance progress in CSAs can have important benefits over topology-based CSAs: first, candidate selection can be done without a knowledge of the overall network topology, thus, less information is required; and second, the computational complexity of the CSA can be much lower, and thus, run faster than topology-based CSAs.

The first CSA we have proposed, that we call *Distance Progress Based Opportunistic Routing* (DPOR) estimates the distance progress for candidate selection. The second one, that we call *Candidate selection based on Maximum Progress Distances* (CMPD), relies on a previous knowledge of the optimal positions of the candidates. We use the result of a previous work where we showed how to compute candidates optimal positions, given the radio propagation model.

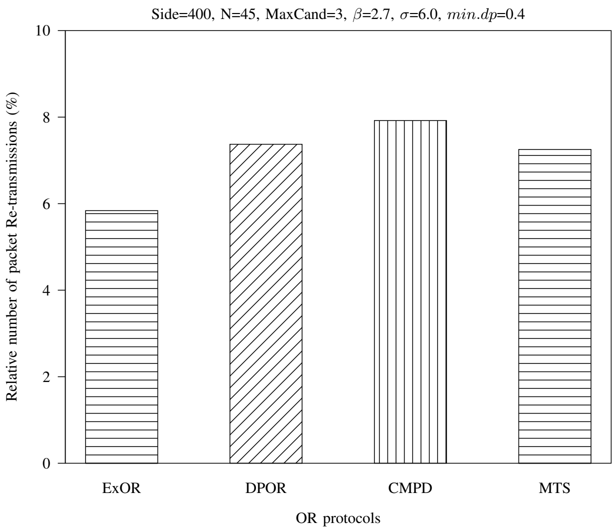

Fig. 12. Relative number of re-transmissions to the number of transmissions for OR protocols under study.

We have investigated the performance of DPOR and CMPD compared with two well known CSAs proposed in the literature: ExOR and MTS. ExOR runs a simple CSA which consists of iteratively running the Shortest Path algorithm. MTS is an optimum algorithm that seeks for candidates that minimize the expected number of transmissions. Numerical results have been obtained by analytic methods and by simulation. The analytic results assume perfect coordination, and compute the expected number of transmissions given by the sets of candidates yielded by the CSA under comparison.

The obtained numerical results confirm that the expected number of transmissions using ExOR can be significantly higher than with the other CSAs. This is specially true in dense networks, which demonstrates that the higher the number of nodes to be chosen as candidates, the higher the impact of the CSA on performance is. Regarding DPOR and CMPD, both have similar behavior, which is very close to the optimum CSA (MTS), specially in dense networks.

In order to measure the computational complexity of the CSAs under comparison, we have measured the execution time necessary for the algorithms to select the candidates. We have obtained that using a modern PC, MTS needs more than 10 minutes to compute the sets of 3 candidates for all nodes to a single destination. The other CSAs are much faster, specially CMPD, which requires only around 0.2 seconds, and, more importantly, it is almost independent of the number of nodes of the network.

Regarding the simulation results we have observed that, except with MTS, the expected number of transmissions almost coincide with analytic results. With MTS, simulation results are worse. This shows that when the ideal assumptions of the analytic model are relaxed, the advantage of the optimal protocol over the other CSAs diminishes. Simulation results also show that ExOR requires the lowest end-to-end delay. This is because with the simple timer-based coordination mechanism, the lower the candidate priority, the higher the transmission delay is. With ExOR, higher priority candidates are chosen with higher delivery probability than the other CSAs, thus, they transmit a higher percentage of packets. This fact shows that, even if the number of transmissions is higher with ExOR than with the other CSAs, the impact of the coordination mechanism can make the overall delays to be lower.

References

[1] The Network Simulator ns-2. `http://www.isi.edu/nsnam/ns`

[2] F. Baccelli, B. Blaszczyszyn, and P. Muhlethaler. On the performance of time-space opportunistic routing in multihop mobile ad hoc networks. In *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks and Workshops. 6th International Symposium on*, pages 307–316, 2008.

[3] S. Biswas and R. Morris. ExOR: opportunistic multi-hop routing for wireless networks. *ACM SIGCOMM Computer Communication Review*, 35(4), 133–144, 2005.

[4] Sanjit Biswas and Robert Morris. Opportunistic routing in multi-hop wireless networks. *ACM SIGCOMM Computer Communication Review*, 34(1), 69–74, 2004.

[5] Llorenç Cerdà-Alabern, Amir Darehshoorzadeh, and Vicent Pla. Optimum node placement in wireless opportunistic routing networks. *Ad Hoc Networks*, 11(8), 2273–2287, 2013.

[6] Llorenç Cerdà-Alabern, Vicent Pla, and Amir Darehshoorzadeh. On the performance modeling of opportunistic routing. In *MobiOpp'10: Proceedings of the Second International Workshop on Mobile Opportunistic Networking*, pages 15–21, 2010.

[7] Szymon Chachulski, Michael Jennings, Sachin Katti, and Dina Katabi. Trading structure for randomness in wireless opportunistic routing. In *SIGCOMM '07: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communications*, pages 169–180. ACM, 2007.

[8] Amir Darehshoorzadeh and Llorenç Cerdà-Alabern. Candidate selection algorithms in opportunistic routing. In *PM2HW2N '10: Proceedings of the 5th ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*, pages 48–54. ACM, 2010.

[9] Amir Darehshoorzadeh, Llorenç Cerdà-Alabern, and Vicent Pla. Modeling and comparison of candidate selection algorithms in opportunistic routing. *Computer Networks*, 55(13), 2886 – 2898, 2011.

[10] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris. A high-throughput path metric for multi-hop wireless routing. *Wireless Networks*, 11(4), 419–434, 2005.

[11] H. Dubois-Ferriéandre, M. Grossglauser, and M. Vetterli. Least-cost opportunistic routing. In *Allerton Conference on Communication, Control, and Computing*, 2007.

[12] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli. Least-cost opportunistic routing. In *Proceedings of Allerton Conference on Communication, Control, and Computing*, 2007.

[13] H. Dubois-Ferriére, M. Grossglauser, and M. Vetterli. Valuable detours: Least-cost anypath routing. *Networking, IEEE/ACM Transactions on*, 19(2), 333–346, 2011.

[14] D. Fuste-Vilella, J. Garcia-Vidal, and J.D. Morillo-Pozo. Cooperative forwarding in IEEE 802.11-based MANETs. In *Wireless Days, 1st IFIP*, pages 1–5, 2008.

[15] Brad Karp and H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, MobiCom '00, pages 243–254. ACM, 2000.

[16] Yanhua Li, Wei Chen, and Zhi-Li Zhang. Optimal forwarder list selection in opportunistic routing. In *Mobile Adhoc and Sensor Systems. IEEE 6th International Conference on*, pages 670 –675, 2009.

[17] Yanhua Li and Zhi-Li Zhang. Random walks on digraphs: a theoretical framework for estimating transmission costs in wireless routing. In *29th conference on Information communications*, pages 2775–2783, 2010.

[18] Mingming Lu and Jie Wu. Opportunistic routing algebra and its applications. In *28th conference on Information communications*, pages 2374–2382, 2009.

[19] E. Rozner, J. Seshadri, Y. Mehta, and L. Qiu. Simple opportunistic routing protocol for wireless mesh networks. In *2nd IEEE Workshop on Wireless Mesh Networks, WiMesh*, pages 48–54, 2006.

[20] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008.

[21] Shengbo Yang, Feng Zhong, Chai Kiat Yeo, Bu Sung Lee, and Jeff Boleng. Position based opportunistic routing for robust data delivery in manets. In *Proceedings of the 28th IEEE conference on Global telecommunications*, GLOBECOM'09, pages 1325–1330, 2009.

[22] Y. Yuan, H. Yang, S.H.Y. Wong, S. Lu, and W. Arbaugh. ROMER: Resilient opportunistic mesh routing for wireless mesh networks. In *IEEE Workshop on Wireless Mesh Networks (WiMesh)*, 2005.

[23] Kai Zeng, Wenjing Lou, Jie Yang, and Donald R. Brown, III. On throughput efficiency of geographic opportunistic routing in multihop wireless networks. *Mob. Netw. Appl.*, 12, 347–357, 2007.

[24] Zifei Zhong, Junling Wang, Srihari Nelakuditi, and Guor-Huar Lu. On selection of candidates for opportunistic anypath forwarding. *SIGMOBILE Mob. Comput. Commun. Rev.*, 10(4), 1–2, 2006.

[25] M. Zorzi and R.R. Rao. Geographic random forwarding (GeRaF) for ad hoc and sensor networks: multihop performance. *Mobile Computing, IEEE Transactions on*, 2(4), 337–348, 2003.