

Neighbor Discovery for Ultraviolet Ad Hoc Networks

Yiyang Li, Leijie Wang, Zhengyuan Xu, and Srikanth V. Krishnamurthy

Abstract—The solar blind ultraviolet (UV) scattering channel makes non-line-of-sight UV communications very attractive for military applications, particularly for communication on-the-move with low probability of detection and low probability of interception. Despite significant research effort on the UV physical layer, work on protocol design at the upper layers is quite limited. We consider a mobile ad hoc UV network, with each node equipped with a transceiver capable of transmitting in multiple directions and performing omni-directional receptions. Full-duplexing is enabled. We develop efficient neighbor discovery protocols by accounting for the unique UV physical (PHY) layer characteristics, namely varying channel qualities along different scattering directions. In addition to a list of neighbor nodes' identities, our protocols also construct and maintain a table which contains a ranked list of node pointing directions between each pair of nodes in terms of channel qualities. Our approach does not need support from the global positioning system (GPS) or temporal synchronization across nodes like many radio frequency (RF) protocols. Specifically, two algorithms are proposed with and without the need for direction synchronization. We further improve the latter by more efficiently utilizing neighbor feedback. We perform extensive simulations to evaluate our algorithms.

Index Terms—Neighbor discovery, ultraviolet, scattering, ad hoc network.

I. INTRODUCTION

Given the proliferation of wireless deployments in the ISM RF bands, unlicensed UV communications (carrier wavelength below 400nm) have been recently considered as a viable alternative to their RF counterparts [1]. While intense solar radiation exhibits significant energy in the infrared and the visible light spectrum, its contributions in the deep UV band (with wavelength 200-280nm) are absorbed by the ozone layer. Thus, a terrestrial UV communication network (operating in this band) is immune to solar noise. In addition, atmospheric scattering helps two communication nodes, possibly on-the-move, easily connect through a non-line-of-sight (NLOS) link [2],[3], without restrictive pointing and acquisition. This distinguishes UV from a traditional laser communication system. The atmosphere attenuates UV signal more significantly beyond a few kilometers, and this makes a UV network in some

sense, a *secure* local area communication network. In other words, due to their unique power decay profile, UV signals are resistant to interception at long distances. UV signals are also inherently resistant to RF jamming. Finally, the NLOS operability makes it difficult for an adverse detector to find the direction of the signal source. All these properties make UV communications attractive for military operations in rural/open and metropolitan/urban outdoor environments and for inter-soldier or inter-vehicle communications. UV transceivers can be miniaturized using commercially available light emitting diodes (LEDs) and photodetectors.

For most military settings, the use of a centralized infrastructure is infeasible. Rapid reconfigurations and re-deployments are more the norm than the exception. Thus, ad hoc network deployments are typical in these contexts. Although there are some preliminary studies on medium access control with UV [4], the network initialization process has not been considered before. This process is crucial for bootstrapping a network prior to operations. Since in typical cases, nodes are randomly deployed, each node will have to determine the existence as well as the locations of the surrounding nodes in order for information sharing. Thus, a distributed neighbor discovery algorithm is desirable for establishing connectivity and for self-organizing the network. Development of such algorithms is the focus of this work.

Each node in the network is assumed to be capable of directional transmissions and omni-directional receptions, given the unique properties of the UV channel (discussed later). At the termination of neighbor discovery, a node in the network will have sufficient knowledge about each of its neighbors.

In order to design efficient neighbor discovery algorithms, an understanding of the UV PHY layer is critical. Towards this, we perform extensive channel measurement experiments in UV-C band. From these measurements, we make the following observations: (1) NLOS links are likely to exist with various beam pointing configurations and, (2) a UV transceiver can work in a full-duplex mode with certain restrictions. Using these measurements we develop an empirical path loss model for NLOS communications to capture the extent of signal attenuation.

Based on the propagation properties derived above, we design our neighbor discovery mechanisms. First, we consider a well-planned deployment, wherein nodes are assigned a fixed set of synchronized directions in which they can point their beams. In other words, the sense of direction is global; when a node receives a message from a neighbor, it can immediately determine the direction in which that neighbor transmitted its packet. We refer to this as neighbor discovery with direction

Manuscript received November 15, 2010; revised April 29, 2011. This work was supported in part by the MRPI program of the University of California, Army Research Office Grant W911NF-09-1-0293, and National Science Foundation NETS grant 0626912.

Y. Li and L. Wang are with the Department of Electrical Engineering, University of California, Riverside, CA 92521 USA (e-mail: {yiyli, lwang}@ee.ucr.edu).

Z. Xu is with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, P.R. China (e-mail: xuzzy@tsinghua.edu.cn).

S. Krishnamurthy is with the Department of Computer Science and Engineering, University of California, Riverside, CA 92521 USA (e-mail: krish@cs.ucr.edu).

synchronization. Second, we propose a more general neighbor discovery scheme, wherein nodes cannot determine a neighbor's direction by simply receiving the latter's transmission. The approach incorporates a novel handshake mechanism that is tightly dependent on the UV propagation characteristics.

More specifically, we make the following contributions:

- **Experimentation on a UV test-bed:** We report NLOS path loss results obtained from extensive experiments on a UV test-bed. On the basis of these results, we provide insights into the physical aspects that impact the neighbor discovery process. First, we find that due to scattering, there may exist multiple communication links (in different configurations) between each communicating pair. Second, full-duplex communications are feasible in some specific configurations; if appropriately invoked it can in essence increase the capacity of the link, thus accelerate the network initialization process. Our neighbor discovery algorithms described later do effectively exploit NLOS communication links and full-duplex possibilities.
- **Neighbor discovery with direction synchronization:** With direction synchronization, as discussed earlier, a receiver can immediately determine the direction using which a transmitter sent its packets. Based on this, it can infer the correct direction in which it should send a response back to the transmitter. Since we do not assume any multiuser detection technique, it is impossible to distinguish the desired signal from interference or background noise. We adopt a credit collection method to solve this problem. We design a mechanism for neighbor discovery given those features and analyze the process via extensive simulations.
- **Neighbor discovery in general scenarios:** In most cases where the time to set up a network is limited or the cost is prohibitive, direction synchronization may be infeasible. We design a second neighbor discovery algorithm for applicability in a network without restrictions on direction synchronization. Our approach includes a handshaking mechanism that is tightly integrated with UV properties. In a nutshell, each node sends a request containing information about itself and receives feedback from its neighbors. It then counts *collected credits* for each direction and for each neighbor. It uses these credits to create a table in which, the directions of communication with each neighbor are ranked.

Organization: The rest of the paper is structured as follows. In Section II, we discuss existing neighbor discovery protocols designed for RF ad hoc networks. We present the PHY model and characteristics of UV outdoor communications in Section III. In Section IV, we develop our algorithm for the direction synchronization scenario and evaluate it using OPNET-based simulations. In Section V, we propose our neighbor discovery mechanisms for the case of general deployments and evaluate their performance. We conclude our work in Section VI.

II. RELATED WORK

There have been a number of efforts on neighbor discovery for RF wireless networks assuming either directional or omni-

directional antennas. Vasudevan et al. classified neighbor discovery algorithms into two categories: direct-discovery and gossip-based [5]. They analyzed these two possibilities in synchronous and asynchronous scenarios and determined the frequency with which each node should send control packets to maximize the discovery probability within a certain time period. They found that the transmission probability is related to the number of neighbors and transmission beamwidth. The gossip-based algorithms outperform the direct-discovery algorithms. Moreover, the performance of the gossip-based algorithm is not sensitive to the density of the nodes. A handshake-based neighbor discovery algorithm was proposed with a time-division-multiple-access (TDMA) based media-access-control (MAC) protocol in 3D space, in [6]. With an assumption that each node was equipped with GPS or inertial navigation system (INS), the authors suggested that the transmission power could be reduced to first find neighbors nearby and then increased gradually such that the probability of detection was increased. Luo et al. analyzed neighbor discovery in a code-division-multiple-access (CDMA)-like system and assumed each node is aware of its neighbors' names and signatures [7]. It is shown good performance is achieved by applying multiuser detection algorithms; however, the assumptions are unrealistic for an ad hoc network. In [8], methods to save energy in neighbor discovery were researched and a flood-like procedure to achieve neighbor discovery was also suggested. Jakllari et al. proposed an integrated neighbor discovery and MAC protocol for ad hoc networks equipped with directional antennas [9]. The algorithm effectively utilizes directional antennas and accounts for the mobility of the nodes while performing neighbor discovery and maintenance. Vasudevan et al. comprehensively analyzed ALOHA-like neighbor discovery algorithms and proposed a collision detection mechanism to improve the performance in [10]. The algorithm enables each node to know when to terminate the neighbor discovery process without any a priori knowledge of neighbors.

Note that the above algorithms are not directly suitable for a UV ad hoc network. The unique UV scattering channel is fundamentally different from a RF channel. The scenarios are different with both directional and omni-directional antennas. The flexibility in both directional and omni-directional transmissions by a UV source (such as a UV LED) provides degrees of freedom in system design and operation that are not possible with RF. It is possible for a node to know not only the existence of the neighbors but also information about their locations without assistance of GPS or additional devices, contrary to most RF approaches.

III. THE UV PHY LAYER

We discuss two physical characteristics of a UV channel that impact neighbor discovery: (1) the existence of NLOS communication paths, and (2) the ability to establish full-duplex communications. Our inferences are based on our extensive experiments under transceiver non-coplanar geometry, i.e., the transmission beam axis and the receiving field-of-view (FOV) axis are not on the same plane. Subsequently, we derive an empirical non-coplanar path loss model for UV

communications, to facilitate protocol design with arbitrary network geometries.

Experimental setup: In our experiments, we employ LEDs as transmitters, a solar blind filter, and a photon detector as a receiver. Each transceiver employs energy-based modulation (e.g., pulse position modulation or on-off keying) and an energy detector such as a photon bucket. Fig. 1 shows the transceiver architecture. The dots on the side facets represent the LED transmitters (a transmitter could consist of multiple simultaneously emitting LEDs) and the dot on the top facet represents the omni-directional photon detecting receiver [4]. With this configuration, the node¹ is a “directionally transmitting” and an “omni-directionally receiving” device. This configuration provides significant benefits in size, cost, and full-duplex communications on links (discussed later). In some cases, it also favors easy installation and deployment (e.g. on the helmet of a soldier or on the top of a vehicle). All following experimental results and the design and evaluation of our neighbor discovery mechanisms are established with this configuration. Note that a practical implementation is not restricted to this shape or the number of facets illustrated in the figure. The number of directions or facets is determined by the transmitting beam angle and is typically chosen so as to maximize spatial coverage and minimize beam overlap simultaneously. For example, if the beam angle of a LED on a side facet of a device is 18° , then this device can have up to 20 side facets. In other words, the number of directions is a system parameter that can be chosen according to source specifications.

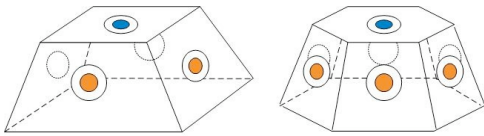


Fig. 1. A UV transceiver with 4 (left) and 6 (right) transmission directions. Dots on the side facets are the directional LED transmitters; each dot on the top facet is the omni-directional photon receiver.

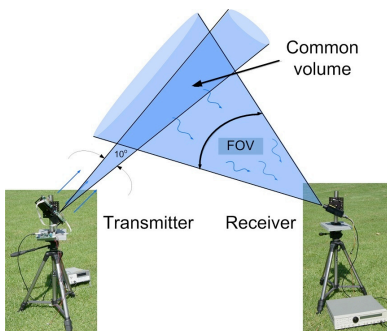


Fig. 2. Common (overlap) volume between transmission beam and receiving FOV to form a NLOS link.

NLOS communication paths: Figure 2 shows a typical NLOS UV communication link. The UV communication link performance adheres to the commonly used signal-to-

interference-plus-noise-ratio (SINR) model. In order to compute the SINR, it is necessary to characterize the atmospheric attenuation that a UV signal experiences. From Reilly’s common-volume single scattering theory [11], if there is an overlap volume between the transmission beam and the receiver’s FOV, the transmitted signal could be potentially detected. Thus, the transmitter and the receiver can successfully communicate not only via the LOS path but also via NLOS paths. NLOS links allow a node to have the choice of more than one configuration to communicate with a neighbor node. The different NLOS links refer to the different direction choices in this work under the assumption that other factors remain fixed. However, the increase in communication opportunities also leads to increased interference in a transmitter’s neighborhood which could hinder the network initialization.

To date, there is no simple path loss model for non-coplanar geometries that captures the multiple scattering effect. In a network setting, there is no guarantee of coplanarity of the beam axis and the FOV axis. The coplanar path loss models presented in [2], [3] require the transmission beam axis and the receiver FOV axis to lie in the same plane. The more recently proposed non-coplanar path loss model in [12] captures the effect of partial alignment. That model is however unable to account for multiple scattering which occurs in practice. In the following, we describe our empirical non-coplanar path loss model developed on the basis of experiments. Part of the results were reported in our recent work [4].

Estimating path loss: We experimentally characterize the impact of non-coplanar geometries on path loss by varying the offset angles of the beam axis and the FOV axis. The transmitter and receiver baseline separation is at most $50m$. The parameter settings are listed in Table I. The values of the filter transmission and photomultiplier tube (PMT) detection efficiencies are hardware parameters that affect the received energy. The LED power refers to that of a single LED; we used two LEDs (bound together) in all of our experiments. Given the LED power and actual received energy per counting interval, we were able to obtain the path loss as simply their ratio. The background noise and device dark noise were negligible during the experiment since all the experiments were conducted at night. The beam angle defines the full beamwidth of transmissions [3]. The relationship between the transmitter beam angle, the receiver FOV and the pointing angle are depicted in Fig. 3.

TABLE I
EXPERIMENTAL SETTINGS

The filter transmission efficiency	0.1
The PMT detection efficiency	0.2
LED power	0.2mW
Wavelength	259nm
Noise	16 photons/s
Mismatch	0.05
Beam angle	15°
Field of view (FOV)	30°
Transmitting pointing angle	10°
Receiving pointing angle	90°

Off-axis angle and pointing angle: The off-axis angle φ is depicted in Fig. 4 and derived in Eq. (1). It specifies the

¹Without loss of generality, we also refer to a node as a transceiver.

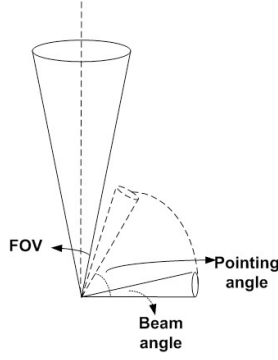


Fig. 3. Beam angle, receiver FOV and pointing angle.

horizontal deviation from the coplanar geometry of two nodes. The vertical deviation from the coplanar axis is referred to as the *pointing angle*. The transmitting angle $\alpha = \frac{\pi}{K} + n\frac{2\pi}{K}$ ($n = 0, 1, 2, \dots, K-1$); K is the total number of directions ($K = 4$ in Fig. 4). With fixed transmitter and receiver coordinates, angle β is determined (see Fig. 4) as

$$\begin{aligned} \gamma &= (\pi - \alpha + \beta) \bmod 2\pi \\ \varphi &= \begin{cases} \gamma & 0 < \gamma < \pi \\ 2\pi - \gamma & \pi < \gamma < 2\pi \end{cases} \end{aligned} \quad (1)$$

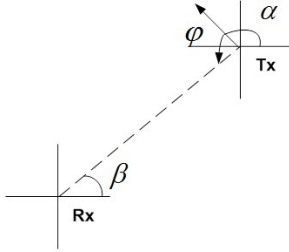


Fig. 4. An illustration of off-axis angle φ from atop.

UV non-coplanar path loss model: Our experimental results (see Fig. 5) suggest that the coplanar path loss model [2] can be extended to account for off-axis angles. We collected photon counts 20 samples every second, at 2° off-axis angle increments until the signal was overwhelmed by noise. Using the process of curve-fitting on the experimental raw data, we

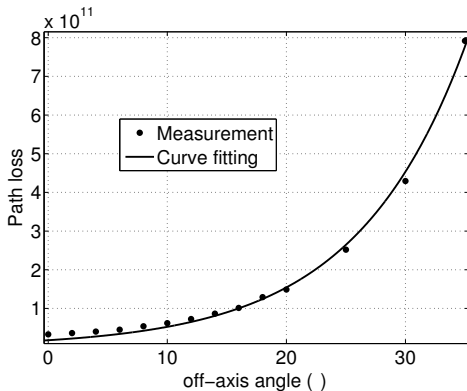


Fig. 5. Exponential curve fitting (pointing angle = 15° , distance = $15m$).

observed that the path loss increases almost exponentially with increasing off-axis angles and can be described by the following model:

$$L_{non-coplanar} = P_t/P_r = \xi r^a \exp(b\varphi), \quad (2)$$

where L denotes the path loss, P_t is the transmission power, P_r is the received power at the detector, r is the horizontal separation distance of the transmitter and receiver, ξ is the path loss factor, and a is the path loss exponent. ξ and a are functions of the transmitter and receiver pointing angles [2]. The off-axis angle φ is given by Eq. (1). The exponent factor b is not very sensitive to varying distance r but increases as the transmitting pointing angle decreases [4]. Unless specified otherwise in our later discussions, we adopt the parameters obtained from experiments with a transmitter pointing angle equal to 10° ; these parameters are as follows, $\xi = 5 \times 10^9$, $a = 0.4$, $b = 8.7$, $\varphi \in [0, \pi]$.

Full-duplex communications: We have assumed that a node consists of co-located transmitter and receiver. According to Reilly's common volume theory, simultaneous transmission and reception at the same node will not interfere each other as long as there is no intersection between its own transmission beam and its receiver's FOV. Thus, full duplex communications are theoretically possible in many cases. As an example, with a 30° FOV facing upwards and a 30° beam pointing along the horizontal axis, full-duplex communications are achievable. In fact, this is possible as long as the pointing angle is less than 60° (see Fig. 3).

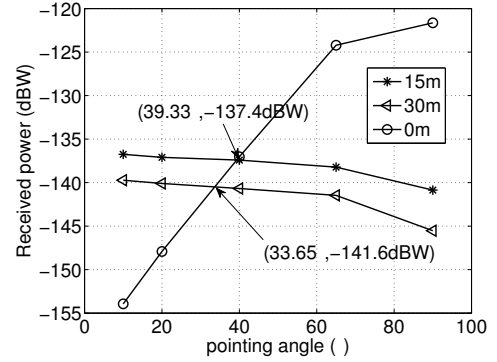


Fig. 6. Illustration of the feasibility of full-duplex.

We performed experiments to test the feasibility of full-duplex communications. The received signal powers at a local receiver for different pointing angles and distances (0m, 15m, 30m) of a remote transmitter are plotted in Fig. 6. The curve corresponding to 0m represents the self interference due to transmission of co-located transmitter. Even with a small pointing angle (with reference to that curve), there is a non-trivial interference level due to multiple scattering (although Reilly's signal scattering theory predicts no contribution of interference from a node's own transmitter). Each of the two cross points in the figure indicates that the signal power is equal to the interference power which hinders reliable information decoding. We thus expect a lower pointing angle to ensure weaker self-interference. We generally find that small pointing angles result in lower interference power to

the co-located receiver, and thus are better suited for full-duplex communications. A larger pointing angle possibly results in overlap volume of its own beam and FOV, and causes increased interference levels. Full-duplex communications are prohibited in such regimes.

In summary, small pointing angles can help achieve full-duplex communications. However at these angles, signals are more likely to be blocked by obstacles. Our neighbor discovery mechanisms are based on an assumption that there are no obstacles and a fixed pointing angle of 10° is used until the discovery process ends.

IV. NEIGHBOR DISCOVERY WITH DIRECTION SYNCHRONIZATION

Since NLOS links are feasible for UV communications, we can expect that given several transmission directions from a remote node to the node of interest, more than one direction can be utilized for communication between this pair of nodes. Although the link loss varies significantly with pointing direction, all those links can function well if the background noise and multiuser interference are negligible. In practice, some small angles might not be feasible due to blockage by buildings, trees, vehicles, even people. However, some other directions resulting in links of degraded quality may still be usable. The neighbor discovery process should make all such link information accessible to neighbors. Each node must be made aware of the different NLOS links that it can use to its neighbors and the ranking of these links in terms of channel quality. It suffices if each node is made aware of the existence of its neighbors in a network with omni-directional antennas (as most prior efforts on neighbor discovery demonstrate). However, if directional antennas are utilized, a node needs to know the direction in which to send information to its neighbors. It is mentioned in [5] that GPS might be necessary or an angle-of-arrival (AOA) estimation technique must be applied to solve this problem. For many reasons including security, GPS might be unavailable in military contexts and low cost is preferable. We choose to develop an algorithm without a requirement for GPS. On the other hand, based on the energy detection and channel scattering characteristics of a UV communication system, AOA estimation is hard to implement. Thus, the issue of how to send information has to be solved by the neighbor discovery algorithm itself.

The neighbor discovery algorithm described in this section is specifically designed for a scenario with *direction synchronization*. Direction synchronization assumes that all the nodes in the network have the same number of fixed directions. The direction labeled “zero” in the direction table of each node should always reflect the same direction. With this assumption, once a receiver node, B receives a message sent by a transmitter node A , containing information with regards to the direction of node A ’s transmission, it can easily infer the best direction to reply to A , based on geometric symmetry (detailed discussion later). The reciprocal property of the channel is attributed to a homogeneous environment; the transmission pointing angles, the beam divergence angle and the FOV are set to be the same across all nodes. In

this case, no handshake mechanism is necessary for direction synchronization. If the restriction of direction synchronization is relaxed, a feedback mechanism should be incorporated. We shall discuss such cases in the next section.

To rank each direction, intuitively, a receiver can record the received energy from a specific transmitter, from each direction. However, since we do not assume any multiuser detection technique, it is impossible to distinguish the desired signal from interference or background noise. We adopt a credit collection method to solve this problem.

A. Algorithm description

The number and names of neighbors are unknown to the present node. The only information that the node has is the format of the transmitted message for the purposes of decoding. Only one type of message is transmitted in the network, i.e., the request packet which contains information of *node identification (ID)* and *direction ID*. Here, we require that each node has unique identification, which could be MAC address or any sequential code for node differentiation.

Every node sends out a request packet of duration of τ and receives all the time (due to the full-duplex capability). The interval between two successive transmissions follows exponential distribution with parameter λ . The node chooses the sending direction randomly.

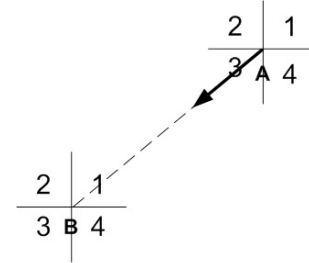


Fig. 7. Illustration of direction synchronization.

Once a receiver node receives a request from another node, it extracts the information of source ID and the associated direction ID. Because all the nodes are assumed to be direction-synchronized, the receiver can figure out the direction in which it can respond to the transmitter. As shown in Fig.7, if node A transmits a request to node B using direction “3” and node B successfully receives A ’s request, node B could conclude that its direction labeled “1” could be used to communicate with node A . Correspondingly, a reception from direction “2” suggests responses in direction “4”, and vice versa. Since whether or not a node transmits is a random event, the total interference level the receiver experiences is a random variable (as investigated in [4]). Thus, the successful reception of packets in a given direction follows a random process. Multiple transmissions of the same packet may lead to different outcomes; it may or may not be received depending on the interference levels from other nearby transmitters.

We collect the successfully received neighbor requests (ND_request contains the transmitter’s ID and the chosen transmitting direction ID), and create a *credit table* in which the directions of the successful requests are recorded. Each

direction is then ranked as follows. A direction to a neighbor is assigned a credit upon correct detection of a packet from that direction. Each node transmits in a randomly chosen direction; the choice of direction is uniformly distributed. Packets sent from a *better* direction are deemed to be received with a higher possibility. Thus, for such directions the credits collected will be higher. Each node will then establish a ranked *connection table* for every neighbor after a sufficiently large number of transmissions are received, based on *credit table*. The more the packets are exchanged, the more reliable is the table. The steps of the process are formalized with the pseudocode below:

```

K = total no. of directions;
Clear credit_table;
Transmitter:
if timer is up then
    ND_request.source_ID=self_id;
    ND_request.direc_ID=uniformly_choose(K);
    send ND_request;
    set timer;
end if
Receiver:
if receive ND_request then
    neighbor_ID=ND_request.source_ID;
    prev_direc=ND_request.direc_ID;
    current_direc=[prev_direc+ceil(K/2)] mod K;
    credit_table.node[neighbor_ID].direction[current_direc]++;
end if

```

B. Simulation results

We study the neighbor discovery performance with direction synchronization, via simulations using OPNET version 16.0 [13]. The simulation settings are listed in Table II. The chosen transmission power corresponds to the typical short UV transmission range (approx 100m). We adopt the channel attenuation results from Section III to characterize signal propagation. The SIR threshold is set to 1dB, indicating that the signal is decodable if the signal power is slightly larger than the total interference power.

TABLE II
SIMULATION SETTINGS

Transmission power	4 mW
# of directions	6
SIR threshold	1 dB
Collision model	Physical (accumulative) model
Data rate	1 Mbps
Traffic pattern	10000 pkt/sec
Packet size	32 bits/pkt
Network size	100m by 100m
# of nodes	10

In Fig. 8, we consider two criteria for neighbor discovery. First, we impose a somewhat less stringent requirement (loose criterion) wherein a node seeks to establish only the top one third of all the directions to a neighbor; the ranking among these directions is inconsequential. For example, assume a node can perform transmissions in 6 directions, then we only focus on the top 2 directions between that node and any other node; the relative ordering of the two directions in terms of

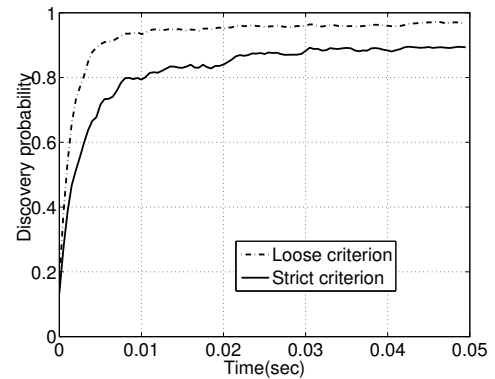


Fig. 8. The discovery probability with loose and strict criteria.

performance is not considered. We consider a direction to be correctly found if it is accurately classified as being or not being within the top one third of the possible directions.

Next, we consider a strict requirement (strict criterion) wherein, the relative ordering of the directions is necessary. In other words, the direction should be correctly classified in terms of its rank. In the solid curve, we consider the two top directions, now correctly ordered. In other words, a direction is considered to be found correctly if and only if it is exactly at the right position in the *connection table*.

It is observed that with the loose criterion, reaching a discovery probability of 90 % takes less than 0.005s, while the solid curve reaching the same probability takes about 0.05s. When there is a strict temporal constraint on the neighbor discovery process, the algorithms can still work well without providing poor directions (the top one third directions can be found) for communications among neighboring nodes.

We observe that while we can discover about 80 % of the directions with high accuracy relatively quickly, ordering among the top directions takes much longer (as evident from the figure). This is because, it is hard to distinguish between the top few directions; this is in some sense to advantageous since it becomes less important as to which direction is ranked higher.

V. NEIGHBOR DISCOVERY IN A GENERAL SCENARIO

In Section IV we considered a scenario in which all the nodes in the network were synchronized with regard to their sense of direction. Next, we relax the synchronization constraints to consider a more general case. In military applications, the network deployment typically requires to be accomplished quickly. For example, the nodes might just be thrown from an aircraft on to a sensor field. Thus, direction synchronization is difficult to obtain. Even if direction synchronization is ensured in the initial deployment, it is likely to be lost later due to the dynamics of the environment. Thus we seek to design an algorithm without the assumption of direction synchronization. With our approach, the receiver resorts to a handshaking process to build what we call a *credit table* (discussed later). The different directions are then ranked based on the *credit table*. As one might expect, the time needed for neighbor discovery grows as compared to the case wherein nodes are synchronized with respect to directions.

A. Algorithm description

There are two types of packets: (1) the request packet which is identical to the one described in Section IV; (2) the feedback packet. Like with the direction synchronization scenarios, a node receives all the time and sends a request packet (see clarification below) after a waiting time which is exponentially distributed with parameter λ . After each successful reception of a request, a feedback packet is expected to be sent back to the sender of the request. The feedback packet contains *the ID of the node that sent the request, the node's own ID, and an identifier of the direction (direction ID) specified in the received request packet*. When it is time to send, each node first checks if any feedback was generated but unsent. If that is the case, then the node chooses to send the feedback and then delete it from the queue of unsent feedback packets. The queuing policy is first-in-first-out (FIFO). If no feedback requests are in the queue, the node sends a request packet (as previously discussed). Note here that, a node does not exactly know the direction using which a request packet sender can be reached; it can only either randomly choose a direction to send or omni-directionally transmit the feedback. Recalling the structural design of the node in Section III, an omni-directional transmission can be realized as long as all the LEDs on the node are powered up and send the same packet simultaneously. If the feedback packet is only sent in one direction (chosen randomly), the likelihood that the original sender receives the feedback is small. However the downside of sending feedback omni-directionally is that the interference increases. With our approach, we choose to send feedback packet omni-directionally. The overall transmission power of the LEDs on a facet (one direction) is equivalent to the power level with which the request packet is sent. To solve the interference issue, when the feedback packet is being sent, the LEDs could scale down their transmission power levels appropriately. More details and comparisons are deliberated on in a following simulation subsection.

We wish to emphasize here that the node chooses to send feedback whenever such packets are available to send; in other words feedback packets are prioritized. This is because our objective is to finish an existing handshaking process prior to starting new handshaking processes. Only when the feedback is successfully received by the original sender of the request, the handshaking process is considered completed. In some extreme cases, all the transmission opportunities will be occupied by feedback packets for a long time thus, precluding a node from sending its own request packets. When this situation occurs, the node under discussion may keep replying to the same node repetitively, which causes a wastage of available resources. More importantly, the node under consideration is deprived of opportunities to find its neighbors. Therefore, we count the number of successive feedback packets sent and set a predefined threshold beyond which, priority is given to request packets. After sending a request packet, priority is returned to the feedback packets and the counter is re-initialized. We also use a finite buffer for storing queued feedback packets; if this buffer is filled, additional feedback packets are discarded. This allows us to control the maximum number of feedback

messages that a node has to respond to.

The steps of our algorithm are succinctly captured by the pseudocode below:

```

K = total no. of directions;
Clear credit_table;
Transmitter:
if timer is up then
  if feedback.head=0 || send feedback 3 times consecutively
  then
    ND_request.source_ID=self_id;
    ND_request.direc_ID=uniformly_choose(K);
    send ND_request;
    set timer;
  else
    ND_feedback.source_ID=self_id;
    ND_feedback.dest_ID=feedback.head→neighbor_ID;
    ND_feedback.prev_direc=feedback.head→prev_direc;
    ND_feedback.direc_ID=Omni-direction;
    send ND_feedback;
    set timer;
  end if
end if
Receiver:
if receive ND_request then
  neighbor_ID=ND_request.source_ID;
  prev_direc=ND_request.direc_ID;
else
  if receive ND_feedback then
    neighbor_ID=ND_feedback.source_ID;
    prev_direc=ND_feedback.direc_ID;
    credit_table.node[neighbor_ID].direction[prev_direc]++;
  end if
end if

```

B. Simulation results

TABLE III
SIMULATION SETTINGS

Request packet size	32 bits/pkt
Feedback packet size	80 bits/pkt

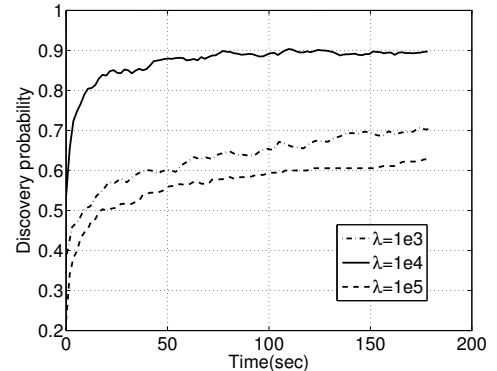


Fig. 9. The discovery probability in *Algorithm 1* with different λ .

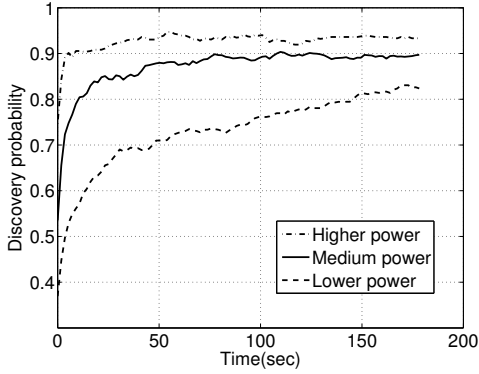


Fig. 10. The discovery probability in *Algorithm I* with different feedback power.

We once again perform extensive simulations to evaluate our approach for the general scenarios. The OPNET simulation settings are listed in Tables II and III. The two kinds of packets exchanged in the communication process are labeled ND_request and ND_feedback (which contains self ID, the request's host-node ID and its chosen transmitting direction ID), respectively. The time interval for which a node should wait prior to sending packets should be appropriately set to optimize the performance. Shorter waiting intervals cause higher packet volume and consequently, higher interference levels; longer waiting intervals trade temporal efficiency for lower interference levels. We seek to find the value that provides the best trade-off between the two. This value depends on the number of nodes and number of directions used for transmissions by a node. In Fig. 9, the cases with different λ s (using which the waiting time is characterized) are compared. We see that $\lambda = 10^4$ provides the best performance; it is the sweet spot and yields the fastest discovery speed. We observe that when $\lambda = 10^3$ or $\lambda = 10^5$, the discovery is slower.

As mentioned in Section V-A, omni-directional feedback packets increase the interference levels. One method suggested was to lower the power levels at each LED for such feedback packets; however, if the power is reduced to a large extent, the signal quality is compromised. We compare the performance of our algorithm with different power settings for sending the feedback packets. In our first setting, the transmission power of each of the LEDs is equal to the transmission power used for request packets. This case corresponds to the medium power case in Fig. 10. In our second setting we make the total transmission power of all facets (together) to be equal to the transmission power of request packets. This setting is represented by the lower power case in Fig. 10. Finally in the third setting, our higher power case corresponds to a case where transmission power on each facet is six times of transmission power of request packets. It appears that the higher power case gives the best performance. This is because the algorithm heavily relies on the number of received feedback packets. As discussed earlier, the *credit table* tries to capture the number of received request packets. However, the collisions of feedback packets reduce the accuracy of the table entries. Increasing the successful reception rate of feedback packets by using higher transmission powers seemingly assists the discovery

probability to approach 1. Motivated by this observation, we next propose an improvement to our algorithm.

C. Improved version of our approach

1) *Algorithm II*: We refer to our original algorithm as *Algorithm I* and the new improved version we propose below is named as *Algorithm II*. As more credits are collected, the fraction of discovered directions grows to 1 gradually. To accelerate this process, we have to collect credits as fast as possible. The number of credits a node collects is equal to the number of feedback packets it successfully receives. This is in turn determined by the number of feedback packets sent to the node and the successful reception rate for these feedback packets. The number of feedback packets sent by a neighbor is less than or equal to the number of request received from the node by that neighbor. In the handshaking protocol described above, we used feedback packets to notify the node how many request packets that it sent have been received by a neighbor. However, due to the loss of feedback packets due to collisions, the number of received request packets by a neighbor is underestimated by the originating node. A solution is proposed here to make credit collection more efficient: when a node (say *B*) sends a feedback packet to node *A* as a response to the reception from one of *A*'s directions (say direction 1), it indicates number of request packets that it has received so far from *A*, sent in direction 1. If *A* successfully decodes the information in the feedback packet, it can update its *credit table* correspondingly. As discussed before, the number of feedback packets being received from a neighbor is always smaller than the number of request packets that were successfully received by that neighbor. The credits collected with the new protocol will provide much more accurate information and allows the credit table entries to grow faster with increased accuracy than with *Algorithm I*. This decreases the importance of the reception of feedback packets towards accurately updating the credit table. Thus, we can now change the omni-directional feedback transmissions to directional transmissions to decrease interference levels. The direction could be chosen randomly.

The steps of the improved algorithms are described by the pseudocode below:

```

K = total no. of directions;
Clear credit_table and req_cnt;
Transmitter:
if timer is up then
  if feedback.head=0|| send feedback 3 times consecutively
  then
    ND_request.source_ID=self_id;
    ND_request.direc_ID=uniformly_choose(K);
    send ND_request;
    set timer;
  else
    ND_feedback.source_ID=self_id;
    ND_feedback.dest_ID=feedback.head→neighbor_ID;
    ND_feedback.prev_direc=feedback.head→prev_direc;

//Algorithm II:

```



```

ND_feedback.direc_ID=uniformly_choose(K);
//Algorithm III:
ND_feedback.direc_ID=Max(credit_table[ND_feedback.
    dest_ID].direc[ND_feedback.prev_direc]);

ND_feedback.request_count=req_cnt[ND_feedback.
    dest_ID][ND_feedback.prev_direc];
send ND_feedback;
set timer;
end if
end if
Receiver:
if receive ND_request then
neighbor_ID=ND_request.source_ID;
prev_direc=ND_request.direc_ID;
req_cnt[neighbor_ID][prev_direc]++;
else
if receive ND_feedback then
neighbor_ID=ND_feedback.source_ID;
prev_direc=ND_feedback.direc_ID;
cnt=ND_feedback.request_count;
credit_table.node[neighbor_ID].direction[prev_direc]=cnt;
end if
end if

```

Compared with ND_feedback packet used in Algorithm I, the ND_feedback packet used here has been added in one entry called request_count which records the accumulative counts of received requests from that combination of node and direction. Figure 11 shows the variations in the performance of Algorithm II with different λ . Again, due to the phenomenon discussed earlier with our initial approach in Fig. 9, the case of $\lambda = 10^4$ provides the best performance. In Fig. 12, the neighbor discovery performances with different transmission powers employed for feedback packets, are compared. The medium power is six times the lower power and the higher power is six times the medium power, i.e., the configuration is the same as that used to generate Fig. 10. Although the higher power case still shows the best performance, the performance difference between the different powers is not as high as with Algorithm II. This is because Algorithm II does not heavily rely on the number of received feedback packets.

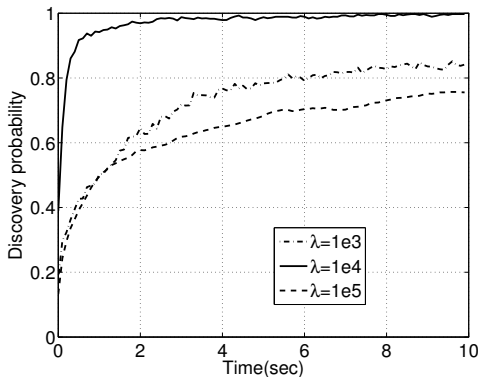


Fig. 11. The discovery probability in Algorithm II with different λ .

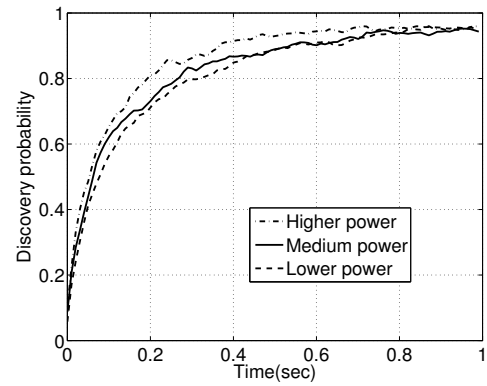


Fig. 12. The discovery probability in Algorithm II with different feedback power.

2) Algorithm III: One can easily envision an extension wherein feedback packets can be sent based on the information already available in the *credit table*. If node B needs to respond to node A 's request, it checks its *credit table* to determine the current *best* direction to A and transmits in the corresponding direction.

Our simulations however suggest that this method does not perform as well as Algorithm II. In Figure 13 we depict the time taken for neighbor discovery with the three algorithms that we have considered (we set $\lambda = 10^4$ in these simulation experiments). With Algorithm III we find that feedback packets experience an increase in collisions from transmissions from other neighbors since in many cases they end up choosing the same direction (the best direction to a common requesting node); this especially hurts neighbors that are further away than those that are closer to the sender. As an example, consider a case where node A sends a request packet in direction 1. Two neighbors, B and C receive the packet and reply to A with their own feedback packets. Although B and C will set timers to wait for an exponentially distributed period, the sending times of the two feedback packets may be very close if $1/\lambda$ is not large compared to τ . When both nodes transmit to A in similar directions, a collision occurs. If one of these nodes (say B) is much closer to A than C , with high probability, B 's feedback goes through while C 's does not. We like to emphasize that the above effect is a direct artifact of both B and C choosing the same (the best) direction to respond to A . Although the performance of Algorithm III is affected due to this (slower time for neighbor discovery), the long term performance is still as good as that of Algorithm II.

VI. CONCLUSION

In this paper, we propose protocols for neighbor discovery in an ultraviolet (UV) ad hoc network. Due to the unique propagation properties of UV signal, each node may transmit or receive signals through NLOS links in different directions. A node should know the exact direction in which to transmit to its neighbors for best performance. Towards achieving this, our approach collects "credits" to rank all possible directions. We begin with basic algorithms with and without direction synchronization and then propose various techniques that can

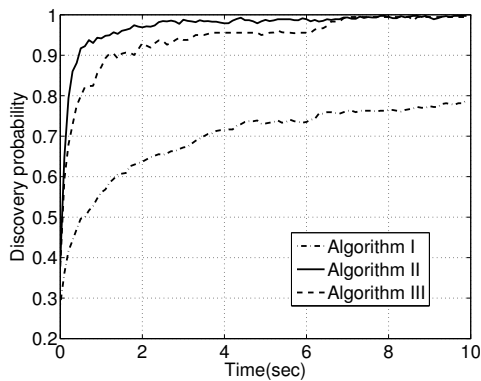


Fig. 13. Comparison of *Algorithm I*, *Algorithm II* and *Algorithm III*.

significantly enhance the performance of the neighbor discovery process. We perform extensive simulations to not only quantify the performance of our algorithms, but also showcase the effects of various parameters such as the transmission power in use, and the random back-off times that node follow prior to sending neighbor discovery request or feedback packets.

REFERENCES

- [1] Z. Xu and B. M. Sadler, "Ultraviolet communications: potential and state-of-the-art," *IEEE Commun. Magazine*, vol. 46, no. 5, pp. 67-73, May 2008.
- [2] G. Chen, Z. Xu, H. Ding, and B. M. Sadler, "Path loss modeling and performance trade-off study for short-range none-line-of-sight ultraviolet communication," *Optics Express*, vol. 17, no. 5, pp. 3929-3940, March 2009.
- [3] Z. Xu, H. Ding, B. M. Sadler, and G. Chen, "Analytical performance study of solar blind non-line-of-sight ultraviolet short-range communication links," *Optics Letters*, vol. 33, no. 16, pp. 1860-1862, August 2008.
- [4] Y. Li, J. Ning, Z. Xu, S. V. Krishnamurthy, and G. Chen, "UVOC-MAC: A MAC protocol for outdoor ultraviolet networks," *18th IEEE International Conference on Network Protocols (ICNP 2010)*, Kyoto, Japan, October 2010.
- [5] S. Vasudevan, J. Kurose, and D. Towsley, "On neighbor discovery in wireless networks with directional antennas," *IEEE INFOCOM*, Miami, USA, Mar. 2005.
- [6] G. Pei, M. Albuquerque, J. Kim, D. Nast, and P. Norris, "A neighbor discovery protocol for directional antenna networks," *MILCOM*, Atlantic city, USA, Oct. 2005.
- [7] J. Luo and D. Guo, "Neighbor discovery in wireless ad hoc networks based on group testing," *46th Annual Alerton Conf.*, IL, USA, Sept. 2008.
- [8] M. J. McGlynn and S. A. Borbash, "Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks," *Mobihoc*, Long beach, USA, Oct. 2001.
- [9] G. Jakllari, W. Luo, and S. V. Krishnamurthy, "An integrated neighbor discovery and MAC protocol for ad hoc networks using directional antennas," *IEEE Trans. on Wireless Communications*, vol. 6, no. 3, pp. 1-12, March 2007.
- [10] S. Vasudevan, D. Towsley, R. Khalili, and D. Goeckel, "Neighbor discovery in wireless networks and the coupon collector's problem," *Mobicom*, Chicago, USA, Sept. 2010.
- [11] D. M. Reilly, "Atmospheric optical communications in the middle ultraviolet," *M.S. Thesis*, MIT, Cambridge, MA, 1976.
- [12] L. Wang, Z. Xu, and B. M. Sadler, "Non-line-of-sight ultraviolet link loss in noncoplanar geometry," *Optics Letters*, vol. 35, no. 8, pp. 1263-1265, April 2010.
- [13] *OPNET User's Manual*, <http://www.opnet.com>.



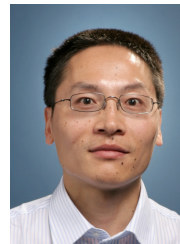
communication systems, networking, and next-generation wireless systems.

Yiyang Li received her B.S. degree in Electronic Engineering from Southeast University, Nanjing, China, in 2005, and M.S. degree in Electrical Engineering from China Academy of Telecommunication Technology in 2008. She is currently pursuing her Ph.D. degree in Electrical Engineering from University of California, Riverside, CA. She was an algorithm engineer in Datang Mobile telecommunications in China with a main focus on TD-SCDMA and LTE-TDD channel estimation. Her current research interests include optical wireless



wireless communications, sensing and networking, and wireless connectivity test solutions.

Leijie Wang received the B.S. degree in Electrical Engineering from Peking University, Beijing, China, in 2005, M.S. degree in Electrical Engineering from China Academy of Telecommunication Technology, Beijing, China, in 2008, and Ph.D. degree in Electrical Engineering from University of California, Riverside, CA, USA, in 2011. He was a Software Engineer in Beijing Xinwei Telecom Technology Inc. in China from 2006 to 2008. He joined LitePoint Corporation in USA as a System Application Engineer from 2011. His research interests lie in optical



funded by the University of California Office of the President. In 2010, he was selected by "Thousand Talents Program" of the Chinese government and appointed as a professor in the Department of Electronic Engineering, Tsinghua University. His research interests lie in wireless communications, networking, optical wireless communication, geolocation, intelligent transportation systems, and signal processing.

Zhengyuan Xu (S'97-M'99-SM'02) received the B.S. and M.S. degrees in electronic engineering from Tsinghua University, Beijing, China, in 1989 and 1991, respectively, and the Ph.D. degree in electrical engineering from Stevens Institute of Technology, Hoboken, NJ, in 1999. Then he joined the Department of Electrical Engineering, University of California, Riverside, as an Assistant Professor, and was promoted to Professor with tenure later on. He was Founding Director of the Multi-campus Center for Ubiquitous Communication by Light (UC-Light)

Dr. Xu has served as an associate editor and guest editor for various IEEE journals in communications, vehicle technology, or signal processing. He has served as a chair, session chair, technical program chair, and technical program committee member for numerous international conferences. He was also an elected member of the IEEE Signal Processing Society's Technical Committee on Signal Processing for Communications for six years.



MC2R from 2007 to 2009. He is a senior member of the IEEE.

Srikanth V. Krishnamurthy received his Ph.D degree in electrical and computer engineering from the University of California at San Diego in 1997. From 1998 to 2000, he was a Research Staff Scientist at the Information Sciences Laboratory, HRL Laboratories, LLC, Malibu, CA. Currently, he is a professor of Computer Science at the University of California, Riverside. His research interests are primarily in wireless networks and security. Dr. Krishnamurthy is the recipient of the NSF CAREER Award from ANI in 2003. He was the editor-in-chief for ACM