On the Price of Security in Large-Scale Wireless Ad Hoc Networks

Chi Zhang, *Student Member, IEEE*, Yang Song, *Student Member, IEEE*, Yuguang Fang, *Fellow, IEEE, Member, ACM*, and Yanchao Zhang, *Member, IEEE*

***Abstract*—Security always comes with a price in terms of performance degradation, which should be carefully quantified. This is especially the case for wireless ad hoc networks (WANETs), which offer communications over a shared wireless channel without any preexisting infrastructure. Forming end-to-end secure paths in such WANETs is more challenging than in conventional networks due to the lack of central authorities, and its impact on network performance is largely untouched in the literature. In this paper, based on a general random network model, the asymptotic behaviors of secure throughput and delay with the common transmission range r_n and the probability p_f of neighboring nodes having a primary security association are quantified when the network size n is sufficiently large. The costs and benefits of secure-link-augmentation operations on the secure throughput and delay are also analyzed. In general, security has a cost: Since we require all the communications operate on secure links, there is a degradation in the network performance when $p_f < 1$. However, one important exception is that when p_f is $\Omega(1/\log n)$, the secure throughput remains at the Gupta and Kumar bound of $\Theta(1/\sqrt{n \log n})$ packets/time slot, wherein no security requirements are enforced on WANETs. This implies that even when the p_f goes to zero as the network size becomes arbitrarily large, it is still possible to build throughput-order-optimal secure WANETs, which is of practical interest since p_f is very small in many practical large-scale WANETs.**

***Index Terms*—Ad hoc networks, network performance, network security, wireless networks.**

I. Introduction

THE GROWTH of modern communication networks, such as the Internet and wireless cellular systems, over the last decade has surpassed many expectations. Indeed, going back in time to the origins of these networks, it would have been hard to imagine the importance and scale to which these networks have developed. Now, projecting into the future, we strongly believe that this trend will continue, if not accelerate. Hence, the communication devices and protocols of today must be capable of operating with the same efficiency in the very large-scale networks of the future. This highlights the need for asymptotic analysis on a network and its corresponding protocol design, which characterizes the asymptotic behaviors of network performance as its size n grows. This is especially the case for wireless ad hoc networks (WANETs), which offer communications over a shared wireless channel without any preexisting infrastructure, since more effort needs to be made to harmonize the behavior of different participants and manage distributed network resources to support end-to-end (e2e) communication demands compared to the network with infrastructure. Obviously, this kind of unavoidable coordination overhead, which may be tolerable in small-scale networks, is possible to become a dominant factor in large-scale networks and should be quantitatively analyzed.

Since both throughput and delay are important network performance metrics, significant effort in the last few years has been devoted to understanding the scaling laws on throughput and delay and their relationship in WANETs. In their seminal work, Gupta and Kumar [1] show that the per-flow throughput capacity for static WANETs scales as $\Theta(1/\sqrt{n \log n})$ (refer to Appendix A for the standard asymptotic notation used throughout this paper) under the assumption that nodes with common transmission range are randomly distributed. Note that this work [1] implicitly uses a fluid model for establishing throughput scaling. Later work by Kulkarni and Viswanath [2] consolidates the result in [1] with an explicit constant-packet-size model. Following the same methodology, the corresponding delay of $\Theta(\sqrt{n/\log n})$ and the complete throughput-delay tradeoffs of static WANETs are first obtained in [3]. Recently, with the percolation theory, Franceschetti *et al.* [4] show that the per-flow throughput of $\Theta(1/\sqrt{n})$ is achievable if each node can adjust its transmission range through power control. The throughput-delay tradeoffs under different mobility models are also studied in the literature (e.g., [3] and [5]–[12]).

A drawback common to all the above results is the neglect of security requirements, which are receiving growing attention in recent years because many large-scale WANETs are expected to be deployed in hostile scenarios such as military and homeland security operations [13]. It is known that security always comes with a price, as securing communications against the adversary typically consumes more network resources in terms of bandwidth and/or hardware capacities. This price may be tolerable in small-scale WANETs, but it may dominate the consumption of

Manuscript received January 06, 2009; revised November 24, 2009 and March 08, 2010; accepted May 05, 2010; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor D. Agrawal. Date of publication February 04, 2011; date of current version April 15, 2011. This work was supported in part by the U.S. National Science Foundation under Grants CNS-0916391, CNS-0721744, and CNS-0716450. The work of Y. Fang was supported in part by the National Science Foundation of China under Grant 61003300, the Fundamental Research Funds for the Central Universities under Grant JY10000901021, and the China 111 Project under Grant B08038 with Xidian University, Xi'an, China. The work of Y. Zhang was supported by the U.S. National Science Foundation under Grants CNS-0716302 and CNS-0844972 (the CAREER Award).

C. Zhang and Y. Song are with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA (e-mail: zhangchi@ufl.edu; yangsong@ufl.edu).

Y. Fang is with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32611 USA, and also with the National Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: fang@ece.ufl.edu).

Y. Zhang is with the School of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe, AZ 85287-7206 USA (e-mail: yczhang@asu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNET.2011.2106162

scarce network resources in large-scale WANETs. This situation makes the investigation of throughput-delay tradeoffs with security requirements in large-scale WANETs an important open challenge.

Although security requirements in WANETs are application-dependent, in this paper we focus on the most critical and fundamental one that reflects the distinct nature of WANETs and enables analytical tractability; that is, *we require that wireless communications should operate on secure links whenever necessary*. A WANET can be informally visualized as a group of wireless communication devices/nodes held by users coming together spontaneously to form a network for a common purpose (e.g., emergency response). Some keying materials for primary security associations (SAs), which we will formally define later (cf. Section II-A), are already preconfigured in communication devices based on the trust relationships among the persons involved. The problem is how to exploit those primary SAs to provide secure communications for arbitrary node pairs when needed. Neighbor authentication or securing the physical link, which provides hop-by-hop security, is the first step for providing e2e secure communications in all kinds of networks. This is especially crucial for WANETs since every node needs to act as a router to forward packets for others. If the node cannot authenticate its neighbors,[1] how can it trust them to handle its packet correctly? Obviously, neighboring nodes with primary SAs can authenticate each other directly with preconfigured keying materials, and the physical links between them can be secured accordingly. Since the number and the distribution of primary SAs are determined by the embedded social network (e.g., trust relationship) of users, a node may not have primary SAs with any of its neighbors. In fact, the probability that a node shares a primary SA with any other node, i.e., p_f, will be very small in practice when the node population n in the WANET increases. In this case, if the physical link still needs to be secured, *secure link augmentation* (SLA) operations[2] are required and performed with the help of physically connected common friends of two end-nodes of this physical link.

When $p_f < 1$, there is in general network performance degradation because we require all communications operate on secure links, and some network resources, i.e., the unsecured links, cannot be utilized compared to WANETs without secure requirements. Although we can obtain more derived secure links with SLA, network resources consumed by SLA is another kind of security cost that should be taken into consideration. Therefore, it is natural to ask the following: What is the price of security (performance degradation) we have to pay in WANETs? Can we design a protocol to achieve the optimal secure network performance or minimize the price of security? In this paper, we answer these questions with rigorous analysis based on reasonable assumptions on WANETs. We formally characterize the tradeoffs between key predistribution related to p_f and secure network performance. Our results show that the minimal price of security with SLA is strictly smaller than that without SLA, which theoretically necessitates SLA operations in WANETs with security requirements. We also design two schemes to achieve the minimal price of security with or without SLA, respectively. Furthermore, these schemes provide several important insights on protocol design for secure communications in WANETs as follows. 1) It is unnecessary and even harmful to think that in order to achieve the minimal price of security, we have to obtain as many derived secure links as possible. In fact, the physical links that need to be secured with SLA are few and should be carefully selected. 2) Our schemes show that it is possible to construct the secure backbone and select the physical links that need to be secured in a totally distributed fashion with negligible communication overhead, and this "secure infrastructure" is unrelated to source–destination (S–D) pairs and can be reused again and again. 3) Although, in general, secure network performance degrades with p_f, with or without SLA, one important exception we find is that when p_f is $\Omega(1/\log n)$, the secure throughput remains at the Gupta and Kumar bound of $\Theta(1/\sqrt{n \log n})$ packets/time slot, wherein no security requirements are enforced on WANETs. This implies that even when p_f goes to zero as the network size becomes arbitrarily large, it is still possible to build throughput-order-optimal secure WANETs, which is of practical interest since p_f is very small in many practical large-scale WANETs.

II. Background and Related Work

The impact of security requirements on the performance of WANETs is largely untouched in the literature with only a few exceptions [14], [15]. In this section, we first review some keying schemes and secure operations related to the fulfillment of our security requirements and then present recent results on secure connectivity and throughput, respectively. We compare these results with ours obtained in this paper and point out our own contributions.

A. On Predistribution of Keying Materials/SAs

When we say two nodes have a primary SA, we mean that two nodes trust each other in the sense that either a symmetric key is shared between them or they know each other's authentic public keys. We further assume that SAs are always symmetric because trust relationship is symmetric in nature [14], [16]. The concept of SA here is closely related to the topic of predistributed key establishment and management in the security protocol design [17]–[20]. In what follows, we summarize some representative schemes proposed in the literature and demonstrate that the parameter p_f is a good abstraction of trust relationships among network nodes regardless of the implementation details of keying schemes.

1) Eschenauer and Gligor's Key Pool Scheme [17]: Before the nodes are deployed, an offline trust authority (TA) will provide a large key pool of size P. Each node randomly picks k different keys from this key pool. Therefore, two neighboring nodes have a primary SA if they share at least one common key in the key pool with probability p_f, which is given as

$$p_f = 1 - \frac{\binom{P-k}{k}}{\binom{P}{k}} = 1 - \frac{((P-k)!)^2}{(P-2k)!\,P!} \tag{1}$$

where the second equality holds for $P > 2k$.

[1] We say two nodes are *friends* when they have a primary SA. We say two nodes are *neighbors* if their Euclidean distance is no greater than the transmission range r_n. There is a *physical link* between any two neighboring nodes, and this link can be secured with the primary SA if the neighboring nodes are also friends. We call this kind of secure link a *primary secure link*. A link can also be secured with the help of other authenticated neighbors; we call this kind of secure link a *derived secure link*.

[2] SLA here means the procedure of securing a physical link between two neighboring nodes that are not friends. A detailed description of SLA operations is given in Scheme 2 in Section V-A.

2) Chan et al.'s Random-Pairwise Key Scheme [18]: Each node identity (ID) is paired with f other randomly selected distinct node IDs, and a pairwise key is pregenerated for each pair of nodes. The key is stored in both nodes' key ring along with the ID of the other node that knows the key. Therefore, the probability p_f of two selected nodes sharing a primary SA is directly given by

$$p_f = f/n. \tag{2}$$

3) Hubaux et al.'s Self-Organized Public-Key Scheme [19], [20]: Like in PGP [21], each node's public and private keys are created by the node itself. Unlike in PGP, where certificates of the public keys are mainly stored in centralized certificate repositories, certificates in this scheme are stored and distributed by the nodes in a fully self-organized manner. For simplicity, we assume that each node has f friends, and it has already stored the certificates of its friends' public keys. Therefore, the probability p_f that a node can directly authenticate one of its neighbor is also given by (2).

4) Multiple Trust Authority Scheme [19], [20]: In this scheme, there are P secure domains. In each secure domain, there exists one offline trust authority, which creates public–private key pairs for each node belonging to its domain. Each node belongs to k randomly selected domains. Therefore, two nodes have a primary SA if they belong to the same domain, with probability p_f, which is given in (1).

Note that the processes of neighbor authentication and pairwise key establishment based on primary SAs and the ways to secure more physical links with SLA have been presented in a unified approach in our previous work (cf. [14, Section II-B]), which is omitted here due to space constraints. Here, we just emphasize that we assume a homogeneous trust model, i.e., each pair of nodes has a primary SA with the same probability p_f, and p_f's are pairwise-independent for each pair of nodes.

To sum up, we have the following observations from the schemes discussed above. First, we need to keep p_f as small as possible. A larger p_f requires more memory space for storing keying materials in each node, and when that node is compromised, the revealed keys will have a larger impact on network security. Therefore, with the same network performance, we are interested in the scheme with the minimal p_f. Second, all these schemes assume homogeneous and independent trust relationships among network nodes, i.e., every node pair has the same probability p_f of sharing a primary SA, which happens independently of other node pairs. Although these two properties are not necessarily valid in all practical situations (cf. [14, Section II-A2]), we adhere to these assumptions throughout the paper for the analytical tractability.

B. *On Secure Connectivity*

Secure connectivity here refers to the requirement that there should exist a secure path connecting arbitrary node pairs,[3] which indicates the availability of secure communications. Primary results analyzing secure connectivity have been presented in [17] based on an approximation model for sensor networks. More precise analyses are given in [22] and [23]. These results suffer from two main drawbacks. First, they assume that in order to achieve the secure connectivity, the network should at least be connected with primary secure links, which is not necessarily the case. Second, they only study the case under certain requirement on p_f for a given $r_n = \Theta(\sqrt{\log n/n})$, the common transmission range.

In our previous work [14], we overcome these limitations by giving a thorough study on r_n–p_f tradeoffs with the secure connectivity constraint under the assumptions that n nodes are randomly distributed in a unit area and that primary SAs are predistributed as described. Our main results are as follows.

- The network is securely connected without SLA or with one-hop SLA when $p_f \cdot n \cdot \pi r_n^2 = \Omega(\log n)$.
- The network is securely connected with k_n-hop SLA when $p_f \cdot n \cdot \pi r_n^2 \geq c$, where $c = \Theta(1)$ and $k_n = O(\log n)$.
- It is impossible for the network to be securely connected when $p_f \cdot n \cdot \pi r_n^2 < c$ for the *routing-security dependency loop* problem, where $c = \Theta(1)$.

We want to keep r_n and p_f as small as possible. However, the above results show that we cannot minimize both to maintain secure connectivity. There are tradeoffs between r_n and p_f under the secure-connectivity constraint, but we can achieve a better r_n–p_f tradeoff when multihop SLA is utilized.

The problem left here is that secure connectivity alone is not a good performance metric for secure WANETs. From the above results, if we only consider secure connectivity, then for the situation of $p_f < 1$, we can trivially achieve the secure connectivity by taking a larger r_n. However, this will lead to a dramatic reduction in achievable throughput (cf. Section VI). Therefore, when we say there exists a secure path for an S–D pair, we must also ask what the secure throughput can be supported. Otherwise, the existence of secure paths is meaningless because the throughput that can be supported may be very low. In this paper, we continue our investigation on characterizing those network performance metrics for secure WANETs.

C. *On Secure Throughput*

Recently, Bhandari and Vaidya [15] suggest that their techniques developed for studying the capacity of multichannel WANETs with random (c, f) assignment [24] can be utilized to analyze the secure throughput with the key pool scheme [17]. For multichannel WANETs with random (c, f) assignment, there are c channels of equal bandwidth available. Each node can only work on a subset of f channels, which is preassigned from c channels randomly. It can be mapped into secure WANETs with each node randomly selecting f keys from a key pool of size c. The ability of two neighboring nodes switching on a common channel can be viewed as having a common key to secure their physical link. Based on this idea, they obtain the secure throughput of $\Theta\left(\sqrt{\frac{p_f}{n \log n}}\right)$ for $p_f = \Omega(1/\log n)$. To the best of our knowledge, this is the only work contributing to this topic. Our work is done concurrently with and independently of the work in [15] and differentiates itself from [15] as follows.

First of all, it is worth noting that there are some fundamental differences between multichannel WANETs and secure WANETs. If two neighboring nodes in a multichannel WANET

[3] A *secure path* consists of consecutive *secure links*. Of course, this requirement is reasonable only when the S–D pair is in the same trust domain and there exists at least one physical path connecting the S–D pair. Therefore, it is necessary to have $p_f = \Omega(\log n/n)$ and $r_n = \Omega(\sqrt{\log n/n})$ (cf. [14, Section II-C]). In what follows, we always assume that it is the case.

do not share a common assigned channel, there exists no physical link between them. In contrast, if two neighboring nodes in a secure WANET do not share a common key, it only means that they cannot establish a primary secure link. The physical link still exists and can be secured and utilized if they can find a connected common friend to help them. Also note that if two concurrent transmission pairs in a multichannel WANET use different channels, they will not interfere with each other. By comparison, even if two concurrent transmission pairs in a secure WANET use different keys to secure their transmissions, it is still possible for them to interfere with each other. By taking these differences into consideration, we show that when SLA is utilized, a secure throughput of $\Theta(1/\sqrt{n \cdot \log n})$ is achievable, which is much higher than the result in [15] for $p_f = \Omega(1/\log n)$.

Second, we adopt different models more suitable for secure WANETs. Following previous works, e.g., [1], [6]–[9], and [11], the results in [15] and [24] are implicitly based on the fluid model, in which the packets are allowed to be arbitrarily small as $n \to \infty$. In contrast, we follow [2], [3], and [10] and explicitly assume the constant-packet-size model, where the packet size remains constant, i.e., does not scale down with n. Although the analysis of the constant-packet-size model is much harder than that of the fluid model [10], we still prefer the former since in reality the packet size does not change when more nodes are added to the network. Furthermore, for a WANET with secure requirements, each packet includes a message authentication code of at least constant size for cryptographic operations. This security overhead on the packet level can be ignored asymptotically only with the constant-packet-size model. The adoption of the constant-packet-size model also facilitates our analysis on packet delays [2], [3], [10].

Finally, we utilize different techniques to derive more general results. We demonstrate how to take advantage of the considerable similarity between our problem and existing work on parallel computing and leverage the results on faulty arrays to obtain scaling laws on secure throughput and delay. Our results actually apply to all possible p_f's when $p_f = \Omega(\log n/n)$.

III. System Assumptions and Main Results

A. Random Network Model of WANETs

1) Node Distribution: We are mainly interested in static WANETs or networks with slow mobility, in which the round-trip time (RTT) of a packet between any S–D pair is much smaller than the timescale of network topology changes. We do not consider the WANETs with rapid topology changes because in this case the overhead of maintaining end-to-end paths will dominate wireless transmissions, while in this paper we focus on the overhead introduced by security requirements and its impact on data transmissions.

We model the node positions as a random point process as follows. Let $\{X_1, X_2, \ldots\}$ be independent and uniformly distributed random points on a bounded region A in the plane. Given a positive integer n, the point process $\{X_1, X_2, \ldots, X_n\}$ is referred to as the *uniform n-point process* on A and denoted by $\mathcal{X}_n$. Given a positive number $\lambda = \frac{n}{|A|}$, let $Po(\lambda)$ be a Poisson random variable with parameter λ, independent of $\{X_1, X_2, \ldots\}$. Then, it can be shown that the point process $\{X_1, X_2, \ldots, X_{Po(\lambda)}\}$ is a *Poisson point process* with mean λ on A, [25, p. 18, Section 1.7] and is denoted by $\mathcal{P}_\lambda$. It is assumed throughout this paper that for any n or λ, the random point processes $\mathcal{X}_n$ and $\mathcal{P}_\lambda$ are coupled in this manner. $\mathcal{V}_n$ is shorthand either for $\mathcal{X}_n$ or $\mathcal{P}_\lambda$. Recall that $\mathcal{P}_\lambda$ is characterized by the following the *spatial independence property*: If $A_1, A_2, \ldots, A_m$ are arbitrarily disjoint regions of A, then the numbers of points in $\mathcal{P}_\lambda$ on $A_1, A_2, \ldots, A_m$ are mutually independent Poisson random variables with mean $\lambda|A_1|, \lambda|A_2|, \ldots, \lambda|A_m|$, respectively. Because of this extreme independence property, it is often easier to work with $\mathcal{P}_\lambda$ rather than $\mathcal{X}_n$. Therefore, we shall often start by proving limit theorems about $\mathcal{P}_\lambda$ as $\lambda \to \infty$ and then deduce results for $\mathcal{X}_n$ from these. The rationale behind this de-Poissonization technique (cf. [25, p. 37, Section 2.5]) is that given that there are exactly k points of $\mathcal{P}_\lambda$ in a region A, these k points are independently and uniformly distributed in A. Thus, $\mathcal{X}_n$ can be well approximated by $\mathcal{P}_\lambda$ as n or λ tends to infinity. Note that the results obtained in this paper apply to both $\mathcal{X}_n$ and $\mathcal{P}_\lambda$ (i.e.,$\mathcal{V}_n$).

We further assume that A is a torus[4] with a unit area and take $\lambda = n$ as $|A| = 1$, which corresponds to the *dense network model* [1], [3] because the area is fixed and the density of nodes increases with the network size n. Another possible model that can be used to study the asymptotic behavior of large-scale WANETs is to keep the node density λ as a constant and let the area of A increase linearly with n, which corresponds to the *extended network model* [26], [27]. In this paper, we concentrate on the dense network model just for fair comparisons, as most known results about WANETs without secure requirements are based on this model [1]–[3], [10]. We note, however, that our results can also be applied to the extended network model by utilizing the scaling technique introduced in [26, p. 28, Section 2.2].

2) Interference Models: We adopt the following two widely used models [1] to describe the necessary and sufficient condition for the successful reception of a transmission over one hop. In what follows, we assume that time is slotted for packetized transmissions and that only $O(1)$ packets can be transmitted per time slot, i.e., our analysis is explicitly based on the constant-packet-size model. A transmitter sends data at a constant rate of W packets/time slot for a successful transmission, and zero for an unsuccessful transmission, where $W = O(1)$.

3) Protocol Model: We assume that all nodes use a common range r_n for their transmissions, and a transmission from node i to node j is successful if and only if $d_{ij} \le r_n$ and $d_{kj} \ge (1+\Delta)r_n$ for any other simultaneous transmitter, say node k. Here, d_{ij} is the distance between nodes i and j, and Δ is a positive constant independent of n.

4) Physical Model: We assume that all nodes use a common power P_n for their transmissions, a transmission from node i to node j is successful if and only if for a concurrent transmitter set $\mathcal{S}$, we have the signal-to-interference-plus-noise ratio (SINR) at receiver j, denoted as SINR_{ij}, satisfying

$$\mathrm{SINR}_{ij} = \frac{P_n \cdot G_{ij}}{N_0 + \sum_{k \in \mathcal{S} \setminus \{i\}} P_n \cdot G_{kj}} \ge \beta.$$

[4] We assume the torus to avoid border effects, which otherwise complicates the analysis. We note, however, that the results in this paper hold for square, disk, or any other shapes of practical interest.

Here, β is the SINR threshold, N_0 represents the ambient noise, and G_{ij} denotes the link gain on link $i \to j$. We use $G_{ij} = d_{ij}^{-\alpha}$ for simplicity, where $\alpha > 2$ is the path-loss exponent.

We mainly focus on the protocol model in this paper for a cleaner presentation of the key ideas. We also show that the same results on secure WANETs can be obtained under the physical model in Appendix C.

5) Traffic Pattern: Similar to previous works [2], [3], [9], [10], we consider the uniform-permutation traffic pattern, i.e., there are n flows/sessions and each node is a source node for only one unicast session and a destination node for another unicast session. Suppose that the source node $i \in \{1, \dots, n\}$ has data intended for destination node $d(i)$, and then $d(1), d(2), \dots, d(n)$ is a random permutation of $1, 2, \dots, n$, where $d(i) \neq i$ for all i.

B. Network Performance Metrics

1) (Secure) Throughput: A per-flow throughput τ is said to be feasible/achievable if every node can send at least at a rate of τ packets/time slot to its chosen destination. We denote by $T(n)$, the maximum feasible throughput as the *throughput capacity* for the network. When security requirements are enforced, we define *secure throughput* as the maximum throughput that can be supported on secure paths for all S–D pairs. Note that when SLA is utilized, the traffic overhead for constructing secure paths will be excluded from the total traffic on secure paths, so secure throughput is only measured as the data rate achieved on the application layer.

2) (Secure) Delay: The delay of a packet is the time it takes the packet to reach the destination after it leaves the source. We do not take the queueing delay at the source into account since our interest is in the network delay. We are interested in the expectation of the average packet delay over all S–D pairs and all random network configurations, which is denoted as $D(n)$ throughout the paper. Note that for secure WANETs, the secure delay is measured only on secure paths. If SLA is utilized, the time required to construct the secure path for the packet going through this path will be calculated as a part of secure delay of that packet.

3) Price for Security: The loss on the secure throughput or the increase on the secure delay compared to WANETs without secure requirements will be defined as the price for security.

C. Main Results of Our Work

The goal of this paper is to study the impact of r_n and p_f on the secure throughput and delay of random networks defined in Section III-A. The following results hold with high probability (w.h.p.)[5] when the network size $n \to \infty$. Here, we only consider the situation when $p_f = \Omega(\log n/n)$ and $r_n = \Omega(\sqrt{\log n/n})$ (cf. Footnote 3).

Theorem 1: When $p_f = \Omega(\log n/n)$, the secure throughput without SLA is $T(n) = \Theta\left(\sqrt{\frac{p_f}{n \cdot \log n}}\right)$ packets/time slot (segment A–B in Fig. 1), and the secure delay is $D(n) = \Theta\left(\sqrt{\frac{n \cdot p_f}{\log n}}\right)$.

Theorem 2:

1) When $p_f = \Omega(\log n/n)$ and also $p_f = O(1/\log n)$, the secure throughput with SLA is $T(n) = \Theta(\sqrt{p_f/n})$ packets/time slot (segment C–D in Fig. 1), and the corresponding delay is $D(n) = \Theta(\sqrt{n/p_f})$.
2) When $p_f = \Omega(1/\log n)$, the secure throughput with SLA is $T(n) = \Theta(1/\sqrt{n \cdot \log n})$ packets/time slot (segment D–E in Fig. 1), and the corresponding delay is $D(n) = \Theta(\sqrt{n/\log n})$.

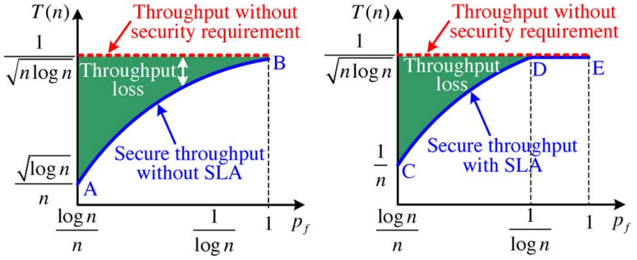


Fig. 1. Impact of security requirements on throughput scaling in random networks. The shaded area represents throughput loss due to secure requirements. The scales of the axes are in terms of the orders in n.

Comparing Theorem 1 to Theorem 2, we can conclude that SLA is necessary because, in general, it can increase the achievable throughput as a factor of $\Theta(\sqrt{\log n})$. However, it does not mean that we should try to secure all physical links. Remember that SLA also incurs extra communication overhead, and the scheme we design to achieve the throughput in Theorem 2 shows that we need to carefully choose links to be secured with the help of friends in order to guarantee that the benefits from SLA always exceed its costs (cf. Section V).

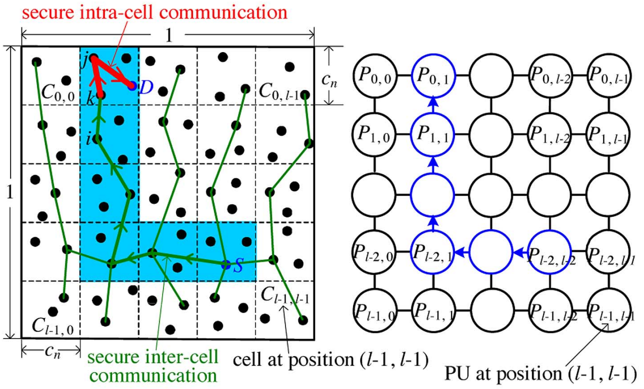
In order to calculate the price of security, we recall the results on network performance of WANETs without security requirements [2], [3], [10], which can be summarized as the following theorem.

Theorem 3: The throughput capacity of WANETs without security requirements is $T(n) = \Theta(1/\sqrt{n \cdot \log n})$ packets/time slot (dashed lines in Fig. 1), and the corresponding delay is $D(n) = \Theta(\sqrt{n/\log n})$.[6]

From Theorem 3, we can find that the price of security mainly exhibits in the loss of throughput. Fig. 1 gives an illustration of the comparison on throughput capacity with or without security requirements. It is worth noting that when p_f is $\Omega(1/\log n)$, security comes with no price in an asymptotic sense, i.e., secure network performance remains on the same order compared to the networks without security requirements (cf. Theorem 2-2). We believe that this result is quite important because it provides valuable insight on the desirable operating points that balance security and efficiency concerns. We need to minimize p_f in order to reduce the memory size for keying materials and mitigate the impact of nodes being compromised. Our result implies that even when p_f goes to zero as the network size becomes arbitrarily large, as long as $p_f = \Omega(1/\log n)$, it is still possible to secure a large-scale WANET with negligible overhead. Our results also show that security requirements in general will not

[5] Here, w.h.p. refers to a probability at least $1 - \epsilon(n)$, for a function $\epsilon(n)$ going to 0 with $n \to \infty$.

[6] The throughput capacity of $\Theta(1/\sqrt{n \cdot \log n})$ was first proved by Gupta and Kumar in [1], but their analysis is based on the fluid model. The same result was obtained by Kulkarni and Viswanath [2] through the constant packet size model. El Gamal *et al.* [3], [10] further improved this result by giving bounds on $D(n)$. Note that recently Franceschetti *et al.* [4] showed that the $\Theta(1/\sqrt{n})$ throughput capacity is achievable if we relax the assumption that all nodes use the same r_n. Here, we still use $\Theta(1/\sqrt{n \cdot \log n})$ bound on throughput because our trust model is a homogeneous one, and for a fair comparison, we also assume the random network model is homogeneous, i.e., all nodes have the same r_n.


secure intra-cell communication
1
j
D
k
i
$C_{0,0}$
$C_{0,l\text{-}1}$
c_n
1
S
$C_{l\text{-}1,l\text{-}1}$
$C_{l\text{-}1,0}$
c_n
secure inter-cell communication
cell at position (l-1, l-1)
$P_{0,0}$
$P_{0,1}$
$P_{0,l\text{-}2}$
$P_{0,l\text{-}1}$
$P_{1,0}$
$P_{1,1}$
$P_{1,l\text{-}2}$
$P_{1,l\text{-}1}$
$P_{l\text{-}2,0}$
$P_{l\text{-}2,1}$
$P_{l\text{-}2,l\text{-}2}$
$P_{l\text{-}2,l\text{-}1}$
$P_{l\text{-}1,0}$
$P_{l\text{-}1,1}$
$P_{l\text{-}1,l\text{-}2}$
$P_{l\text{-}1,l\text{-}1}$
PU at position (l-1, l-1)

(a) (b)

Fig. 2. Secure communication scheme without SLA. (a) Dividing unit torus. (b) Corresponding $l \times l$ array.

increase the e2e delay. This can be intuitively explained as follows: In order to keep the secure connectivity, which is the primary requirement for secure services, the negative effect of a smaller p_f should be compensated by a larger r_n, which will effectively decrease the number of hops a packet needs to travel and thus the e2e delay.

IV. Network Performance Without SLA

We now present a parameterized secure communication scheme without SLA and analyze its performance. Our theoretical results in Theorem 4 confirm that the bounds given in Theorem 1 are achievable and tight.

A. Scheme Description

1) Scheme 1: The Secure Communication Scheme Without SLA:

1) *Torus Partition*: Divide the unit torus into a set of regular cells, each of side length $c_n = \sqrt{\frac{c_1 \cdot \log n}{n \cdot p_f}}$, where c_1 is a constant [see Fig. 2(a) for an illustration].
2) *Setting Transmission Range*: Set $r_n = \sqrt{5}\, c_n$, which guarantees that each node can directly communicate with any node in the same cell or in the immediate vertical and horizontal neighboring cells.
3) *Routing*: Packets are delivered from the source to the destination in two phases. First, they are forwarded along the cells in the row that contains the source cell until they reach the column that contains the destination cell. In the second phase, packets are forwarded along the cells in the same column to their destination. The L-shaped curve connecting the source and destination as described above is called S–D routes [shaded area in Fig. 2(a)].
4) *Cell Scheduling*: A cellular time-division multi-access (TDMA) transmission scheme is used, in which each cell becomes active, i.e., its nodes can transmit successfully to nodes in the same cell or in neighboring cells, at regularly scheduled time slots (cf. Proposition 1).
5) *Packet Transmission Scheduling*: Each packet will have a timestamp t_b denoting the timeslot the packet was transmitted by the source. When a cell becomes active, it will select the packet with the smallest t_b in the cell to transmit. If there are ties, choose the packet from the S–D pair i that maximizes $(t_b + i) \bmod n$. Note that only one packet is transmitted per time slot per cell.[7] Our packet transmission scheduling scheme will treat the packets from different sessions equally and prefer the oldest packet in each session.
6) *Secure Inter/Intracell Transmission*: All the packets will be transmitted on primary secure links. When a node needs to transmit a packet to its neighboring cell, it always transmits the packet to one of its friends in that cell (secure intercell transmission). Otherwise, it will drop the packet. When a node needs to transmit a packet to the node in the same cell and they are not friends, the node will find one of their common friends in the same cell as the relay node (secure intracell transmission). If it cannot find one, it will drop the packet.

Here, we give some primary results on the scheme described. We first recall the following result on the property of cell scheduling in Step 4, which is widely known now [2].

Proposition 1: Under the protocol model, there exists an interference-free schedule such that each cell becomes active regularly once in K^2 time slots without interfering with any other simultaneously transmitting cell. Here, K depends only on Δ and is independent of n.

Next, we show that the probability that the scheduled packet is dropped in Step 6 in Scheme 1 approaches to zero as $n \rightarrow \infty$. This claim is true due to the following lemma.

Lemma 1: In Scheme 1, we can always find a constant c_1 such that we have the following.

1) Each cell contains $\Theta(\log n/p_f)$ nodes w.h.p.
2) Given an arbitrary node i, each cell contains $\Theta(\log n)$ friends of i w.h.p.
3) Given two arbitrary nodes i and j in the same cell, either they are friends or they have at least one common friend in that cell w.h.p.

Proof: See Appendix B. ■

Lemma 1 shows that, for any source node S [see Fig. 2(a) for an example], it can always find a friend in each neighboring cells w.h.p. If multiple friends are available in a cell, S randomly chooses one and defines this friend as its secure relay in this cell. With the routing rule in Step 3, this friend-finding procedure (i.e., each secure relay find its own friends in the following neighboring cells) is continued until there is a secure backbone [regular solid lines in Fig. 2(a)] spanning all cells. Based on Lemma 1-2, every node can construct its own secure backbone[8] as described. Therefore, each packet can follow this secure backbone until it reaches the secure backbone node k in the same cell as the destination node D through secure intercell transmissions. If k is a friend of D, it can directly transmit the packet to D. Otherwise, it needs to find a common friend j to relay the packet, which is guaranteed to happen w.h.p. according to Lemma 1-3. Therefore, there are at most two secure

[7] Each S–D pair is identified by the source node's ID.

[8] Note that the source node does not need to construct this secure backbone beforehand. It will emerge gradually with the data flow and extend to a new cell when the source node's packet goes through that cell. Here, we just show that there exists such a secure backbone for each node to facilitate the analysis in Section IV-B.

intracell transmissions [bold solid lines in Fig. 2(a)] for each packet w.h.p.

B. Performance Analysis of Scheme 1

Note that Scheme 1 only operates on primary secure links. These links will be established after direct neighbor authentication operations, which only needs to local broadcasts. Obviously, the overhead and the delay incurred here is negligible compared to multihop data communications (cf. [14, Section II-B]). Therefore, in what follows, we only concentrate on the throughput and delay in the data delivery phase.

Our analysis on Scheme 1 mainly relies on some well-known results about two-dimensional (2-D) arrays [28], [29], which have been extensively studied in the parallel and distributed computing research community. Therefore, we first review some related definitions and results.

A *2-D $l \times l$ array* consists of $L = l^2$ processors or processing units (PUs) arranged in a 2-D $l \times l$ grid. Each PU is connected to its four neighbors via point-to-point wired communication links. In the multiple-instruction–multiple-data (MIMD) mode, the PUs perform routing in a series of synchronous time slots. During each time slot, a PU may send one packet to its neighbors along each of the (up to four) links incident on it. A PU may also receive one packet along each of its incident links during a time slot. The PUs can be indicated by their coordinates within the array; the PU at position (i, j), $0 \leq i, j \leq l$, is denoted $P_{i,j}$. Here, position (0, 0) lies in the upper left corner [see Fig. 2(b) for an illustration]. A torus is an array with so-called wraparound links, which connect $P_{i,0}$ with $P_{i,l-1}$ and $P_{0,j}$ with $P_{l-1,j}$. Throughout this paper, all results about an array can be extended to the corresponding torus, so we do not distinguish tori from arrays hereafter and simply call them arrays[9] for simplicity. An *h–h routing problem* on 2-D arrays refers to the scenarios that each PU is the source and destination of exactly h packets.

Lemma 2: ([28] and [29]) h–h routing on $l \times l$ arrays can be performed deterministically in $h \cdot l/2 + O(h^{5/6} \cdot l^{2/3})$ time slots, with average packet delay $\Theta(l)$.

We now point out the correspondence between Scheme 1 and the optimal communication scheme for the 2-D array. Let

$$l = \left\lceil \frac{1}{c_n} \right\rceil = \Theta\left(\sqrt{\frac{n \cdot p_f}{\log n}}\right). \tag{3}$$

Cell $C_{i,j}$ in Fig. 2(a) corresponds to PU $P_{i,j}$ in Fig. 2(b) for $0 \leq i, j \leq l$. Without loss of generality, we assume that each source node has only one packet in our WANET model, so there are $\Theta(\log n/p_f)$ packets generated in each cell based on Lemma 1-1. By letting

$$h = \Theta(\log n/p_f) \tag{4}$$

we have formed a correspondence in the traffic pattern between our WANET model and the array, i.e., we associate the h packets generated in a PU with the packets of the nodes contained in the corresponding cell. Routing and scheduling algorithms used by the array to achieve the performance given in Lemma 2 are the same as the schemes we described in Steps 3 and 5, respectively. In fact, Scheme 1 simulates the optimal communication scheme for the array by requiring the scheduled node in cell $C_{i,j}$ to perform the communication operation performed by PU $P_{i,j}$ of the array.

Next, we discuss the difference between our WANET model and the array. Note that each PU can transmit and receive up to four packets in each time slot, while in our cell scheduling scheme in Step 4, each cell will be scheduled to be active once in K^2 time slots (cf. Proposition 1). Therefore, compared to the array, the scheme performed in WANET will have a slowdown by no more than a factor of $4K^2$.

Therefore, we make a correspondence between the performance of secure intercell communications with that of communications between neighboring PUs. Based on Lemma 2, we can conclude that the total number of time slots needed to deliver n packets (one for each source node) to their destination nodes' cells is equal to

$$\Theta\left(4K^2 \frac{hl}{2}\right) \overset{K=\Theta(1)}{=} \Theta(hl) \overset{(3),(4)}{=} \Theta\left(\sqrt{\frac{n \cdot \log n}{p_f}}\right). \tag{5}$$

We have already shown in Section IV-A that each packet only needs at most two secure intracell transmissions, corresponding to $2K^2$ time slots. Therefore, the total number of time slots needed to deliver n packets to their destination nodes, denoted as $\varphi(n)$, can still be expressed in (5). Since only one packet has been delivered for each node, we obtain per-node throughput as $\tau = 1/\varphi(n)$ packets/time slot. Since we assume the constant-packet-size model and that one packet can be transmitted in each time slot, we have $T(n) = 1/\varphi(n)$ packets/time slot. From Lemma 2 and equation (3), we can directly obtain $D(n) = \Theta(\sqrt{n \cdot p_f/\log n})$ time slots.

Based on the above analysis, we have in fact given a constructive lower bound on $T(n)$ and upper bound on $D(n)$ without SLA as follows.

Theorem 4: When $p_f = \Omega(\log n/n)$, the secure throughput without SLA is $T(n) = \Omega\left(\sqrt{\frac{p_f}{n \cdot \log n}}\right)$ packets/time slot, and the corresponding delay is $D(n) = O\left(\sqrt{\frac{n \cdot p_f}{\log n}}\right)$ time slots.

In particular, when $p_f = 1$ or without any security requirement, we can obtain Gupta and Kumar's result [1]. El Gamal *et al.* [3], [10] reprove their result under the constant-packet-size model with complicated analysis on a discrete-time queueing network. Here, we follow Kulkarni and Viswanath's methodology [2] to avoid these complicated queueing analyses by exploiting the similarity between the cell-based network model and the array. Therefore, our proof is desirable in its simplicity. Moreover, it provides some necessary background for understanding our more complicated scheme designed with SLA.

V. Network Performance With SLA

In this section, we analyze achievable secure network performance when SLA is allowed. We first present the following schemes to achieve the performance bounds in Theorem 2.

A. Scheme Description

As a prelude to describing the scheme, we review the SLA operations defined in our previous work [14]. When $p_f \cdot n \cdot \pi r_n^2 \geq$

[9] Also notice that, throughout this paper, we only consider 2-D arrays within the MIMD mode.

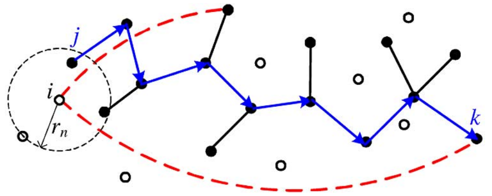


Fig. 3. Multihop SLA operations. Here, solid lines and dashed lines represent primary secure links and primary SAs, respectively. Solid and open points represent nodes on the giant cluster (secure backbone) or not, respectively. Node i is isolated and needs to be connected with the secure backbone with the help of its friend, e.g., node k.

c_2 for some constant c_2, the network consisting of nodes and primary secure links (modeled as *primary secure graph*) is in the percolated phase, i.e., most nodes are connected by a secure backbone (also called the *giant cluster* in percolation theory) with primary secure links. There are still $p_i \cdot n$ nodes disconnected from the giant cluster, where $0 < p_i < 1$ is a constant only depending on c_2. We call all these nodes *isolated nodes*, though their degrees in the physical graph may be larger than 1. Take node i in Fig. 3 as an example. When $r_n = \Theta(1/\sqrt{np_f})$ and $r_n = \Omega(1/\sqrt{n \log n})$ (or more precisely, when r_n is set as in Step 2 of Scheme 3′), even if node i is isolated, w.h.p. there exists at least one node in its transmission range, e.g., node j, belonging to the secure backbone. Also note that when $p_f = \Omega(\log n/n)$, w.h.p. node i has at least one friend, e.g., node k, in the secure backbone (cf. Lemma 3 for a justification of these two statements). Therefore, with the help from node j and k, we can perform multihop SLA as the following.

1) Scheme 2: The Multihop SLA Scheme:

1) Isolated node i first sends a secure connection request (SEC-REQ) message to one of its neighboring nodes j in the secure backbone.
2) Node j will forward this SEC-REQ message to one of its neighbors in the secure backbone, as long as the latter never receives this message.
3) When the receiver, say, node k, receives the SEC-REQ message, it will check whether node i is its friend. If it is not the case, node k will forward the SEC-REQ message as described in Step 2. Otherwise, it will send back a secure connection approval (SEC-APV) message to the sender. This process will continue until node j receives the SEC-APV message.
4) Nodes i and j mutually authenticate each other and secure the physical link $i \leftrightarrow j$.

One important property of Scheme 2 we obtained in [14] is that node k is $O(\log n)$ hops away from node j. Put it in another way, in order to find a friend of node i in the secure backbone, we need to visit $O(\log n)$ nodes w.h.p.

Also note that one prerequisite of Scheme 2 is that every node should know whether it is an isolated node or a node in the secure backbone. This necessitates a secure network partition detection algorithm performed in each node to decide its role in the primary secure graph. Our previous research [14] shows that in the percolated phase, isolated nodes only form clusters with size $O(1)$ even when $n \to \infty$. Therefore, each node can send a probe message, which will be forwarded only through primary secure links. If the probe message can only go through $O(1)$ hops, w.h.p. the node is isolated. The associated overhead for the secure network partition detection is on the same order of direct neighbor authentication, as they both require communications within $O(1)$ hops, which can be ignored as compared to the network-wide multihop communications.

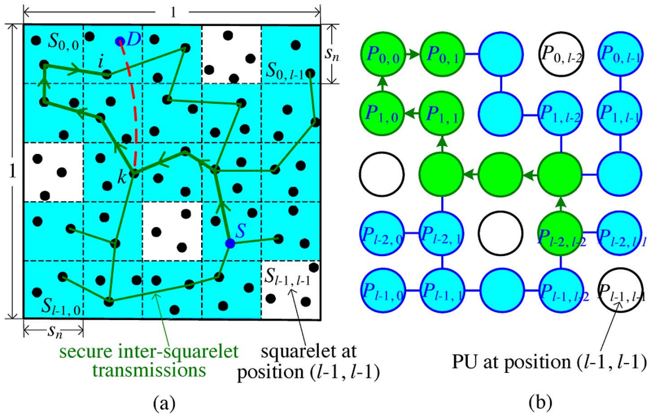


Fig. 4. Secure communication Scheme 3′ with SLA. (a) Dividing unit torus. Solid lines and dashed lines represent primary secure links and primary SAs, respectively. The shaded area represents the squarelets that can be covered by the secure backbone of source node S. (b) Corresponding $l \times l$ array. Solid points and open points represent PUs on and off the giant cluster, respectively.

Next, we present a primary secure communication scheme for S–D pairs on the secure backbone, and our task is to deliver a packet from the source node to the squarelet in which the destination node dwells.

2) Scheme 3′: Secure Communications on the Secure Backbone:

1) *Torus Partition*: Divide the unit torus into a set of regular squarelets, each of side length
$$s_n = \begin{cases} \sqrt{\frac{c_3 \log n}{n}}, & \text{if } p_f = \Omega\left(\frac{1}{\log n}\right) \\ \sqrt{\frac{c_3}{np_f}}, & \text{otherwise} \end{cases} \tag{6}$$
where c_3 is a constant. Note that a cell used in Scheme 1 is much larger than a squarelet in general. In fact, a cell contains $\Theta(\log n)$ squarelets when $p_f = o(1/\log n)$ [see Fig. 4(a) for an illustration].
2) *Setting Transmission Range*: Set $r_n = \sqrt{5}\, s_n$, which guarantees that nodes in neighboring squarelets can communicate directly. Also notice that this r_n guarantees that the primary secure graph is in the percolated phase w.h.p. (this is a direct consequence of Theorem 2 in our previous work [14]).
3) *Squarelet and Packet Transmission Scheduling*: The squarelet scheduling and the packet transmission scheduling in each active squarelet are the same as the cell scheduling in Step 4 of Scheme 1, and the packet scheduling in each active cell described in Step 5 of Scheme 1, respectively.
4) *Secure Intersquarelet Transmission*: All the packets will be transmitted on primary secure links crossing neighboring squarelets. In other words, when a node needs to transmit a packet to its neighboring squarelet, it always transmits the packet to one of its friends in that squarelet. As what we have done in Scheme 1, we can establish a correspondence between our squarelet system and the $l \times l$ array by setting $l = \lceil 1/s_n \rceil$. See Fig. 4 for an illustration. Here, two neighboring PUs will have a link

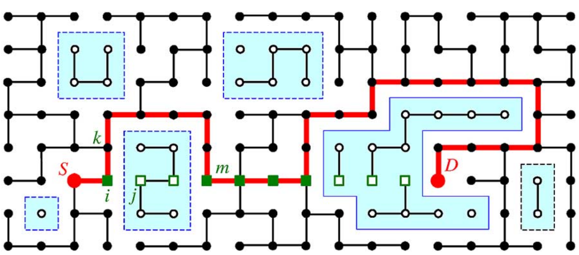


Fig. 5. Routing scheme on the percolated grid. Here, solid lines represent secure links between neighboring PUs/squarelets. Solid and open points represent PUs on and off the giant cluster (secure backbone), respectively. The shaded area represents the finite clusters of PUs disconnected from the secure backbone, and dashed lines represent borders of these clusters.

in the array if the packet holder in the corresponding squarelet can find a friend in another squarelet. Therefore, the array we obtained in Fig. 4(b) is a faulty array (which will be defined more precisely soon) with link failures, where a failure indicates there is no friend in a neighboring squarelet.

5) *Routing*: Since here we only consider the intersquarelet communications, routing between squarelets is equivalent to that operating on PUs. Therefore, we use the faulty array as an example for a cleaner presentation. Note that the corresponding array is also percolated, and we can guarantee that there exists a path connecting the S–D PUs if the corresponding S–D node pairs are on the secure backbone. See Fig. 5 for an illustration. We first fix one shortest path of length k connecting the S–D PUs in the array without faulty links, which consists of square nodes in Fig. 5. Our routing algorithm attempts to follow this shortest path until it encounters a failure link, e.g., at node i. At this point, we simply "circumnavigate" the cluster of isolated nodes (the shaded area) that blocks the path until either the destination PU is reached or the algorithm is back onto the original shortest path to it (e.g., reach node m). Since the average size of the cluster of isolated nodes is a constant w.h.p., the path length of the route (bold lines in Fig. 5) found by our scheme is $O(k)$ on average [30].

We next summarize some basic results about torus partition in Scheme 3′ in the following lemma.

Lemma 3: In Scheme 3′, we can always find a constant c_3 such that we have the following.

1) Each squarelet contains $\Theta\left(ns_n^2\right)$ or $\Omega(\log n)$ nodes w.h.p.
2) Each squarelet w.h.p. contains $\Theta\left((1-p_i)ns_n^2\right)$ and $\Theta\left(p_i ns_n^2\right)$ nodes on and off the secure backbone, respectively, where $0 < p_i < 1$ is a constant.
3) Given an arbitrary node i, each squarelet contains at least one friend of i with probability p_l, independently of each other, where p_l is a constant.
4) Given an arbitrary node i, there exists at least one node in node i's transmission range that belongs to the secure backbone, and node i has at least one friend in the secure backbone w.h.p.

Proof: See Appendix B. ■

Based on these discussions, we now give the complete description of our scheme supporting secure communications between all S–D pairs as the following:

3) *Scheme 3: The Secure Communication Scheme With SLA:*

- *Phase 1—Primary secure link establishment and secure network partition detection*: After this phase, each node finds its neighboring friends and knows whether it is on the secure backbone.
- *Phase 2—Connecting isolated nodes to the secure backbone*: After this phase, each isolated node will connect to a node on the secure backbone with a derived secure link. We further require that each isolated node only connects to a secure backbone node in the same squarelet. For each squarelet, this phase consists of the following three steps.
  1) We first select a secure backbone node for each isolated node in the same squarelet. Given a squarelet, denote NI and NS as the node set of isolated nodes and secure backbone nodes in that squarelet, respectively. For $u \in \mathrm{NI}$, $\psi(u) = v$ means that we select node $v \in \mathrm{NS}$ for node u. Then, we choose v in such a way that for all $v \in \mathrm{NS}$, we have $|\{\omega \in \mathrm{NI} : \psi(\omega) = v\}| \leq \lceil |\mathrm{NI}|/|\mathrm{NS}| \rceil$.
  2) See Fig. 3 for an example. When node $j \in \mathrm{NS}$ is selected for node $i \in \mathrm{NI}$, we call node j as the deputy of node i. We will run Scheme 2 for node i. Note that Scheme 2 has two multihop communications on the secure backbone. One is from node j to node k, and the other is from node k to node j. These two communications can both be implemented with Scheme 3′.
  3) Isolated nodes transmit the packets generated by themselves to their deputies, respectively.
- *Phase 3—Secure-backbone communication*: After Phase 2, all the packets generated by sources are redistributed on secure-backbone nodes only, and then we can utilize Scheme 3′ to deliver these packets from the source nodes or deputies to their corresponding destination squarelets [solid lines from S to i in Fig. 4(a)]. More precisely, if the destination node is also on the secure backbone, we define the squarelet in which it dwells as the destination squarelet. Otherwise, we define one of the closest squarelets to the destination node, which is also covered by the secure backbone, as the destination squarelet. Note that the destination squarelet is always covered by the transmission range of the destination node, which is guaranteed by our torus partition in Scheme 3′.
- *Phase 4—Last-hop delivery*: See Fig. 4(a) for an example. If node i is a friend of the destination node D, then i can directly transmit the packet to D. Otherwise, we need to secure link $i \to D$. This can be done by utilizing Scheme 2 again: We find a friend of D, say node k, and then secure the link $i \to D$ with the help of k. As described in Step 2 of Phase 2, we need to use Scheme 3′ twice to fulfill this operation.

B. *Performance Analysis of Scheme 3*

We first analyze the performance of Scheme 3′. Our analysis mainly relies on the following results on faulty arrays. A q*-faulty array* refers to the array in which each link may fail independently with some probability bounded above by a fixed value q.

Lemma 4: There exists a scheme for a q-faulty $l \times l$ array to solve the 1–1 routing problem in $\Theta(l)$ time slots with probability $1 - 1/l$ when q is small enough. Note that for faulty arrays, we are required to route packets on live links, and we only need to route packets among all PUs connected by live paths.

Remark: Mathies proves Lemma 4 in [30] for $q < 0.5$. It is trivial to extend Lemma 4 to the h–h routing problem: in the same way, we can perform the h–h routing on a q-faulty $l \times l$ array within $\Theta(h \cdot l)$ time slots with the average packet delay of $\Theta(l)$ w.h.p. Compared to Lemma 2, the result here shows that when $q < 0.5$, the running time for a q-faulty array is almost the same as if there were no faults in the array links (up to constant factors), if a routing scheme similar to the one we described in Step 5 in Scheme $3'$ is adopted. It is trivial to find the correspondence between our Scheme $3'$ and the $(1-p_l)$-faulty array. Therefore, the above results can be leveraged to analyze Scheme $3'$ when $p_l > 0.5$, which can be easily achieved by tuning the parameter c_2 mentioned in Section V-A. Following the same argument given in Section IV-B, we obtain the following result.

Corollary 1: When each node on the secure backbone has at most $O(1)$ packets, Scheme $3'$ can deliver all these packets within $\Theta(n \cdot s_n)$ time slots with the average packet delay of $\Theta(1/s_n)$ w.h.p.

Proof: This can be directly obtained from Lemmas 3 and 4. ■

We now analyze the performance of Scheme 3. Note that Phase 1 and Steps 1 and 3 in Phase 2 only need local broadcasts, which will be dominated by other phases involving Scheme $3'$. We thus ignore them in our asymptotic analysis.

Step 1 in Phase 2 guarantees that every secure-backbone node will act as the deputy for $\Theta(|\mathrm{NI}|/|\mathrm{NS}|)$ isolated nodes. From Lemma 3-2, we know that it is equal to $\Theta(1)$. Therefore, every secure-backbone node only needs to handle $\Theta(1)$ SEC-REQ or SEC-APV messages. Then, the performance of Scheme $3'$ used in Step 2 of Phase 2 can be bounded as in Corollary 1. For the same reason, the network performance in Phase 4 is also bounded as in Corollary 1. From Steps 1 and 3 of Phase 2, we can guarantee that each secure-backbone node only holds $\Theta(1)$ packets at the beginning of Phase 3, assuming that each source node only generates one packet. Therefore, we can apply Corollary 1 again to Phase 3. To sum up, the performance of Scheme 3 is on the same order of that of Scheme $3'$, which is characterized by Corollary 1. By substituting (6) into Corollary 1 and following the argument given in Section IV-B, we can obtain the bounds on the secure throughput and delay.

Based on the above analysis, we have in fact obtained a constructive lower bound on $T(n)$ and upper bound on $D(n)$ with SLA as follows.

Theorem 5:

1) When $p_f = \Omega(\log n/n)$ and also $p_f = O(1/\log n)$, the secure throughput with SLA is $T(n) = \Omega(\sqrt{p_f/n})$ packets/time slot, and the corresponding delay is $D(n) = O(\sqrt{n/p_f})$ time slots.
2) When $p_f = \Omega(1/\log n)$, the secure throughput with SLA is $T(n) = \Omega(1/\sqrt{n \cdot \log n})$ packets/time slot, and the corresponding delay is $D(n) = O(\sqrt{n/\log n})$ time slots.

VI. Optimality of Our Schemes

In this section, we present upper bounds on the secure throughput with or without SLA. The corresponding lower bounds on the e2e delay will also be obtained. Since the upper bounds derived here match the constructive lower bounds obtained in Sections IV and V, we complete the proof of Theorems 1 and 2 of this paper under the protocol model. The results in this section also show that the schemes we designed in Sections IV and V are optimal at least in the order sense. Note that we defer the proofs of Theorems 1 and 2 under the physical model to Appendix C.

A. Upper Bounds on Secure Throughputs

The secure throughput of random networks defined in Section III-A is limited by the following three constraints. The maximum feasible throughput satisfying all these constraints is an upper bound on the secure throughput. While there may be other constraints under secure throughput as well, the constraints we consider here are sufficient to provide tight bounds, as the upper bounds obtained here match the constructive lower bounds provided in Sections IV and V.

1) Physical-Connectivity Constraint: We first need to make sure that the network is physically connected, which constrains r_n as $r_n = \Omega(\sqrt{\log n/n})$ [25], [31].

2) Secure-Connectivity Constraint: The throughput of secure WANETs is constrained by the need to ensure that the network is securely connected, so that every S–D pair can communicate through at least one secure path. Our previous work [14] quantifies this constraint as follows (cf. Section II-B):

- $r_n = \Omega\left(\sqrt{\frac{\log n}{n \cdot p_f}}\right)$ without SLA;
- $r_n = \Omega(1/\sqrt{n \cdot p_f})$ with SLA.

3) Interference Constraint: The secure throughput is also constrained by interference. Since the wireless channel is a shared medium, under the protocol model, two nodes simultaneously receiving a packet from different transmitters must be separated by enough distance. This implies a constraint on the maximum number of simultaneous transmissions in torus A. We characterize this constraint with the following lemma.

Lemma 5: The interference constraint requires that $T(n) \leq \frac{c_3}{n \cdot r_n}$, where c_3 is a constant.

Proof: We first consider the case when $p_f = 1$. Let $\bar{L}$ be the expected distance between S–D pairs within the unit-area torus, and then $\bar{L} = \Theta(1)$ w.h.p. (cf. [2, Claim 3.1 (3)]). Thus, on average each packet needs to traverse at least $\Theta(\frac{\bar{L}}{r_n})$ hops to reach the destination. Since each node generates packets at rate $T(n)$, this means that the packets per time slot being transmitted by the whole network are at least $nT(n)\frac{\bar{L}}{r_n}$. Under the protocol model, each transmission "consumes" area, i.e., disks of radius $\frac{\Delta}{2}r_n$ around every transmitter should be disjoint [1]. Since the area "consumed" is bounded above by the total area $|A| = 1$, the maximum number of feasible simultaneous transmissions is no more than $\frac{4}{\pi\Delta^2 r_n^2}$. Hence, we have the constraint

$$nT(n)\frac{\bar{L}}{r_n} \leq W\frac{4}{\pi\Delta^2 r_n^2} \Rightarrow T(n) \leq \frac{c_1}{n \cdot r_n}.$$

The throughput of network when $p_f = 1$ is at least as large as the throughput of the network when $p_f < 1$ (this is trivially true by not using unsecured physical links), so $\frac{c_1}{n \cdot r_n}$ is also an upper bound for $T(n)$ when $p_f \leq 1$. ■

By combining the above constraints, we obtain the following theorem on the upper bounds on the secure throughput.

Theorem 6:

1) When $p_f = \Omega(\log n/n)$, the secure throughput without SLA is $T(n) = O\left(\sqrt{\frac{p_f}{n \cdot \log n}}\right)$ packets/time slot.

2) When $p_f = \Omega(\log n/n)$ and also $p_f = O(1/\log n)$, the secure throughput with SLA is $T(n) = O(\sqrt{p_f/n})$ packets/time slot.
3) When $p_f = O(1)$ and also $p_f = \Omega(1/\log n)$, the secure throughput with SLA is $T(n) = O(1/\sqrt{n \cdot \log n})$ packets/time slot.

B. Lower Bounds on Secure Delays

Lower bounds on secure delays can be analyzed in a similar fashion. The only thing we need to do is to replace the interference constraint with the following path-length constraint.

1) Path-Length Constraint: Since only a single packet can be transmitted per cell per time slot, the e2e delay is lower bounded by the number of hops on the path. Let $\overline{L}$ be the expected distance between S–D pairs. We then have $D(n) \geq \frac{\overline{L}}{r_n}$. If we require that the packet is always transmitted through the secure path, $D(n)$ is even larger, therefore $D(n) = \Omega(1/r_n)$.

By combining the above constraint with physical and secure connectivity constraints, we obtain the following theorem for the lower bounds on the secure delay.

Theorem 7:

1) When $p_f = \Omega(\log n/n)$, the secure delay without SLA is $D(n) = \Omega\left(\sqrt{\frac{n \cdot p_f}{\log n}}\right)$ time slots.
2) When $p_f = \Omega(\log n/n)$ and also $p_f = O(1/\log n)$, the secure delay with SLA is $D(n) = \Omega(\sqrt{n/p_f})$ time slots.
3) When $p_f = O(1)$ and also $p_f = \Omega(1/\log n)$, the secure delay with SLA is $D(n) = \Omega(\sqrt{n/\log n})$ time slots.

VII. Concluding Remarks

In this paper, based on a general random network model, the asymptotic behaviors of secure throughput and delay with the common transmission range r_n and the probability p_f of neighboring nodes having a primary security association are quantified when the network size n is sufficiently large. The costs and benefits of secure link augmentation operations on the secure network performance are also analyzed.

Appendix A
Asymptotic Notation

We use the following standard notation throughout the paper. For two nonnegative functions $f(\cdot)$ and $g(\cdot)$, the following apply.

1) $f(n) = O(g(n))$ means that there exists a constant c and an integer N such that $f(n) \leq c \cdot g(n)$ for $n > N$ (i.e., asymptotic upper bound).
2) $f(n) = o(g(n))$ means that $\lim_{n\to\infty} f(n)/g(n) = 0$ (i.e., asymptotic insignificance).
3) $f(n) = \Omega(g(n))$ means that there exists a constant c and an integer N such that $f(n) \geq c \cdot g(n)$ for $n > N$ (i.e., asymptotic lower bound).
4) $f(n) = \omega(g(n))$ means that $\lim_{n\to\infty} f(n)/g(n) = \infty$ (i.e., asymptotic dominance).
5) $f(n) = \Theta(g(n))$ means that $f(n) = O(g(n))$ and $g(n) = O(f(n))$ (i.e., asymptotic tight bound).

Appendix B
Some Results About Torus Partitions in Schemes 1 and 3′

As a prelude, we first establish the following Chernoff bound [32] for a Poisson random variable X of parameter λ.

Lemma 6: Let X be a Poisson random variable of parameter λ. We have

$$\mathbf{Pr}[X \geq a] \leq \frac{e^{-\lambda}(e\lambda)^a}{a^a}, \qquad \text{for } a > \lambda \tag{7}$$

and

$$\mathbf{Pr}[X \leq a] \leq \frac{e^{-\lambda}(e\lambda)^a}{a^a}, \qquad \text{for } a < \lambda. \tag{8}$$

For $0 < \delta < 1$, Chernoff bounds given in (7) and (8) can be combined and simplified to

$$\mathbf{Pr}[|X - \lambda| \geq \delta\lambda] < 2\, e^{-\delta^2\lambda/2}. \tag{9}$$

Proof: Note that for any random variable $X \geq 0$, and constants $a, t \geq 0$, we have $X \geq a$ if and only if $e^{tX} \geq e^{ta}$. Thus, by Markov's inequality, we have

$$\mathbf{Pr}[X \geq a] \leq \frac{\mathbf{E}[e^{tX}]}{e^{ta}}.$$

For a Poisson random variable X, we have

$$\begin{aligned}\mathbf{E}[e^{tX}] &= \sum_{k\in\mathbb{N}} \frac{e^{tk}e^{-\lambda}\lambda^k}{k!}\\ &= e^{-\lambda}\sum_{k\in\mathbb{N}}\frac{(\lambda e^t)^k}{k!}\\ &= e^{-\lambda}e^{\lambda e^t} = e^{\lambda(e^t-1)}.\end{aligned}$$

Therefore, we have $\mathbf{Pr}[X \geq a] \leq e^{\lambda(e^t-1)}e^{-ta} = e^{\lambda(e^t-1)-ta}$. For $a > \lambda$, we choose $t = \log(a/\lambda) > 0$ and obtain (7). Following a similar approach, we can obtain (8) for $a < \lambda$ by choosing $t = \log(a/\lambda) < 0$.

By substituting $a = (1+\delta)\lambda$ into (7), we obtain

$$\mathbf{Pr}[X \geq (1+\delta)\lambda] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\lambda} < e^{-\delta^2\lambda/4}. \tag{10}$$

By substituting $a = (1-\delta)\lambda$ into (8), we obtain

$$\mathbf{Pr}[X \leq (1-\delta)\lambda] \leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\lambda} < e^{-\delta^2\lambda/2}. \tag{11}$$

Therefore, we can obtain (9) by combining (10) and (11). ■

Then, we prove Lemma 1 for Scheme 1 in Section IV-A and Lemma 3 for Scheme 3′ in Section V-A. We first show that these two lemmas hold when node positions follow the Poisson point process, i.e., $\mathcal{P}_n$.

Proof of Lemma 1 with $\mathcal{P}_n$:

1) Based on the description of Scheme 1 in Section IV-A, we know that there are $m = \left\lceil \frac{1}{c_n^2} \right\rceil = \frac{np_f}{c_1 \log n}$ cells, and the number of nodes in each cell is a Poisson random variable X with parameter $\lambda = nc_n^2 = c_1 \log n/p_f$, where c_1 is a constant and $p_f = \Omega(\log n/n)$. For $0 < \delta < 1$, let A_n be the event that there is at least one cell with more

than $(1+\delta)\lambda$ or less than $(1-\delta)\lambda$ nodes. By the union bound and (9) in Lemma 6, we have

$$\mathbf{Pr}[A_n] \le m\mathbf{Pr}[|X-\lambda| \ge \delta\lambda] < \frac{2np_f}{c_1 \log n}\left(\frac{1}{n}\right)^{\frac{c_1\delta^2}{2p_f}} \to 0$$

as n tends to infinity for any $c_1 \ge 4/\delta^2$. Therefore, each cell contains $\Theta(\lambda) = \Theta(\log n/p_f)$ nodes w.h.p.

2) Given an arbitrary node i in a particular cell, the number of i's friends in that cell is a Poisson random variable Y of parameter $\lambda' = p_f\lambda = c_1 \log n$. For $0 < \delta < 1$, by the Chernoff bound in (9), we have

$$\mathbf{Pr}[|Y-\lambda'| \ge \delta\lambda'] < 2\left(\frac{1}{n}\right)^{c_1\delta^2/2} < 2\left(\frac{1}{n}\right)^2$$

for any $c_1 \ge 4/\delta^2$. Applying union bound over all $m \le n$ cells in the network, the probability that this happens in any cell is at most $2/n$, which tends to zero as n tends to infinity. Therefore, each cell contains $\Theta(\lambda') = \Theta(\log n)$ friends of node i w.h.p.

3) Consider an arbitrary cell. All nodes in this cell and primary secure links between these nodes form a subgraph, which can be modeled as an Erdös–Rényi random graph [33], [34]. From the above proof, we know that the number of nodes in each cell is $\Theta(\log n/p_f)$ and that the average node degree in this subgraph is $\Theta(\log n)$, which is larger than the logarithm of the number of nodes in the cell, given that $p_f = \Omega(\log n/n)$. Therefore, by the properties of the Erdös–Rényi random graph [33], [34], this subgraph is connected, i.e., there exists a secure path connecting arbitrary node pairs in the cell. Because all nodes in the cell are in the transmission range of each other, to find this secure path only needs one-hop local communications, which can be ignored compared to the multihop data communications. ■

Proof of Lemma 3 With $\mathcal{P}_n$:

1) Based on the description of Scheme 3′ in Section V-A, we know that when $p_f = o(1/\log n)$, there are $m = 1/s_n^2 = \frac{np_f}{c_3}$ squarelets, and the number of nodes in each squarelet is a Poisson random variable X of parameter $\lambda = ns_n^2 = c_3/p_f$, where c_3 is a constant and $p_f = \Omega(\log n/n)$ and $p_f = o(1/\log n)$. For $0 < \delta < 1$, let A_n be the event that there is at least one squarelet with more than $(1+\delta)\lambda$ or less than $(1-\delta)\lambda$ nodes. By the union bound and (9) in Lemma 6, we have

$$\mathbf{Pr}[A_n] \le m\mathbf{Pr}[|X-\lambda| \ge \delta\lambda] < \frac{2np_f}{c_3}\left(\frac{1}{n}\right)^{\frac{c_3\delta^2}{2}} \to 0$$

as n tends to infinity for any $c_3 \ge 4/\delta^2$. Therefore, each squarelet contains $\Theta(\lambda) = \Theta\left(ns_n^2\right)$ nodes w.h.p.

When $p_f = \Omega(1/\log n)$, the proof of Lemma 3-1 is straightforward and is omitted here due to space constraints.

2) Suppose there are Z nodes in the network, where Z is a Poisson random variable of parameter n. From our previous work [14], we know that $(1-p_i)\cdot|Z|$ nodes (called backbone nodes) are connected by a secure backbone (also called the *giant cluster* in the percolation literature) with primary secure links. There are still $p_i \cdot |Z|$ nodes (called isolated nodes) disconnected from the giant cluster, where $0 < p_i < 1$ is a constant only depending on parameter c_3 in Scheme 3′. From the randomness of the construction of the network model, we know that all backbone nodes or isolated nodes are also uniformly distributed in the unit torus. Therefore, following the similar argument in the proof of Lemma 3-1, we can prove that each squarelet w.h.p. contains $\Theta\left((1-p_i)ns_n^2\right)$ backbone nodes and $\Theta\left(p_i ns_n^2\right)$ isolated nodes.

3) Because the number of nodes in each squarelet is a Poisson random variable independent of that in any other squarelet, and p_f's between different node pairs are also independent, the event that a squarelet contains at least one friend of a given node i is independent of that in any other squarelet. Next, we show that this event happens with probability p_l, which is lower-bounded by a constant. Recall that in Scheme 3′, when $p_f = \Omega(1/\log n)$, the number of nodes in each squarelet, i.e., $|X|$, is lower-bounded by $(1-\delta)c_3 \log n$. We thus have

$$\begin{aligned} p_l &= 1-(1-p_f)^{|X|} \\ &> 1-\left(1-\frac{c_4}{\log n}\right)^{(1-\delta)c_3\log n} > 1-e^{-(1-\delta)c_3c_4} \end{aligned}$$

where δ, c_3, and c_4 are all constants. When $p_f = o(1/\log n)$, the number of nodes in each squarelet, i.e., $|X|$, is lower-bounded by $(1-\delta)c_3/p_f$. We thus have

$$\begin{aligned} p_l &= 1-(1-p_f)^{|X|} \\ &> 1-(1-p_f)^{(1-\delta)c_3/p_f} > 1-e^{-(1-\delta)c_3} \end{aligned}$$

where δ and c_3 are all constants.

4) First, from (6) in Scheme 3′, we directly arrive at the conclusion that each squarelet contains at least one node on the secure backbone w.h.p. Since $r_n = \sqrt{5}s_n$, we know that in node i's transmission range, there exists at least one squarelet. Therefore, there exists at least one node in node i's transmission range that belongs to the secure backbone w.h.p. Second, recall that in Scheme 3′ we can guarantee that the primary secure graph is in the percolated phase w.h.p.. We also have proven in our previous work (cf. [14, Theorem 2]) that when the primary secure graph is in the percolated phase, each node belongs to the secure backbone with a probability S, where S is a constant. When $p_f = \Omega(\log n/n)$, there are at least $\Theta(\log n)$ friends of node i in the whole network, and each friend belongs to the secure backbone with the probability S. From the Chernoff bound, it is easy to show that at least one of these $\Theta(\log n)$ friends belongs to the secure backbone. ■

Note that $\mathcal{X}_n$ can be well approximated by $\mathcal{P}_n$ as n tends to infinity. Therefore, by the de-Poissonization technique introduced in [25, p. 37, Section 2.5], we can prove that Lemmas 1 and 3 also hold when nodes follow a uniform point process, i.e., $\mathcal{X}_n$, for n tending to infinity. Due to space constraints, we omit this routine proof here.

Appendix C
Secure Network Performance Under the Physical Model

Here, we show that the same results on secure WANETs as in Theorems 1 and 2 can be obtained under the physical model.

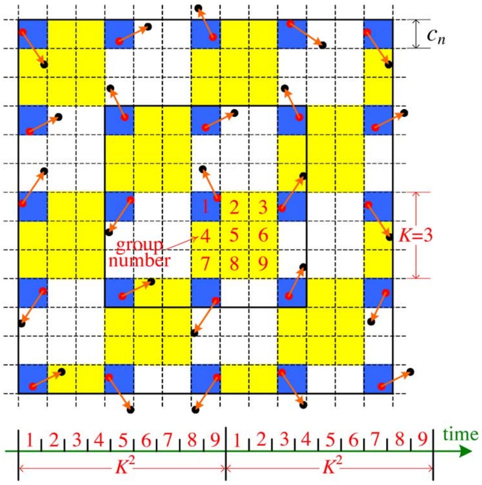


Fig. 6. Cell scheduling scheme. Here is an illustration of the cells being divided into K^2 groups for the case of $K = 3$, i.e., nine groups. All the dark cells that are in group 1 transmit in the same time slot. In the next time slot, all the cells in group 2 transmit, and so on.

We first show that the constructive lower bounds provided in Sections IV and V will not be changed under the physical model. Note that the protocol model only relates to the cell scheduling part of schemes proposed in Sections IV and V. Therefore, if we can show that the same property of the cell scheduling as described in Proposition 1 still holds for the physical model, we are done. In what follows, we prove this claim based on the assumption that $\alpha > 2$.

Proof of Proposition 1 Under the Physical Model: We use the same cell scheduling scheme as in Proposition 1 under the protocol model (see Fig. 6 for an illustration). The received power of the desired signal is lower-bounded by

$$P_n \cdot G_{ij} = P_n \cdot d_{ij}^{-\alpha} \geq P_n \cdot (\sqrt{5}c_n)^{-\alpha}$$

where c_n is the side length of each cell.

We then bound the interference, i.e., I. Consider a particular cell C. If one node from this cell is transmitting, all other simultaneous transmissions may occur in cells belonging to the same set of cells that are as a vertical and horizontal distance of exactly some multiples of a particular integer K. Actually, the interfering cells are placed along the perimeter of concentric squares, whose center is C, and each square contains $2Ki(i = 1, 2, \ldots, L)$ interfering cells as depicted in Fig. 6, where L is the number of such concentric squares. For example, the first concentric square contains eight interfering cells, whereas the second concentric square contains 16 interfering cells, for the particular case where $K = 4$. Each node in the intended cell C transmits data packets to nodes in the four neighboring cells. Then, the distance between these nodes (the possible receivers in the four adjacent cells) and the interfering ones is at least $(K - 2)c_n i(i = 1, 2, \ldots, L)$. As we are considering a lower bound, we take the worst case. Then, the number of concentric squares (irrespective of the position of the intended cell, because the worst case is when the intended cell is at one corner of the area) is at most $L \leq \lceil \frac{1}{2Kc_n} \rceil$. We proceed to upper-bound the interference at the receiver j as

$$\begin{aligned}
I &= \sum_{k \in \mathcal{S} \setminus \{j\}} P_n d_{kj}^{-\alpha} \qquad \text{(Recall that } \mathcal{S} \text{ is the concurrent transmitter set)}\\
&\leq \sum_{i=1}^{L} \frac{P_n \cdot 2Ki}{[(K-2)c_n i]^\alpha}\\
&= \frac{2P_n K}{[(K-2)c_n]^\alpha} \sum_{i=1}^{L} i^{1-\alpha}\\
&\leq \frac{2P_n K}{[(K-2)c_n]^\alpha} \left[1 + \int_1^L x^{1-\alpha} dx\right]\\
&= \frac{2P_n K}{[(K-2)c_n]^\alpha} \left[1 + \frac{1}{2-\alpha}(L^{2-\alpha} - 1)\right]\\
&= \frac{2P_n K}{[(K-2)c_n]^\alpha} \left(\frac{\alpha-1}{\alpha-2}\right) + \frac{2P_n K}{[(K-2)c_n]^\alpha} \left(\frac{L^{2-\alpha}}{2-\alpha}\right)\\
&\leq c_5 \frac{P_n K}{[(K-2)c_n]^\alpha} \qquad \text{(Recall that } \alpha > 2\text{)}
\end{aligned}$$

where c_5 is a positive constant. Therefore, based on the physical model (cf. Section III-A2), we have

$$\text{SINR}_{ij} \geq \frac{P_n \cdot (\sqrt{5}c_n)^{-\alpha}}{N_0 + c_5 \frac{P_n K}{[(K-2)c_n]^\alpha}} = \frac{c_6 P_n}{c_7 c_n^\alpha N_0 + c_8 P_n} \tag{12}$$

where c_6, c_7, and c_8 are constants. Recall that $c_n \leq 1$. Therefore, SINR_{ij} in (12) can be lower-bounded by some constant β, which guarantees the successful reception of packets at node j. Thus, we complete the proof that Proposition 1 also holds under the physical model. ■

Next, we show that the upper bound on the secure throughput and the lower bound on the e2e delay provided in Section VI will not be changed under the physical model. Note that the interference model only affects the interference constraint. Therefore, if we can show that the physical model yields the same interference constraint, we are done. The following lemma on the existence of a correspondence between physical and protocol models on simultaneous transmission sets guarantees that it is indeed the case.

Lemma 7: Let $\Delta(\beta) = \left(48(\frac{2^{\alpha-2}}{\alpha-2})\beta\right)^{1/\alpha}$. Suppose that for $\Delta > \Delta(\beta)$, the protocol model allows simultaneous transmissions for a transmitter–receiver (T–R) pair in a set $\mathcal{S}$. Then, there exists a power assignment $\{P_i, 1 \leq i \leq n\}$ allowing the same T–S pair set $\mathcal{S}$ under the physical model with threshold β.

Proof: Refer to the proof of Theorem 4.1 in [35, p. 174]. ■

REFERENCES

[1] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[2] S. Kulkarni and P. Viswanath, "A deterministic approach to throughput scaling in wireless networks," *IEEE Trans. Inf. Theory*, vol. 50, no. 6, pp. 1041–1049, Jun. 2004.

[3] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, "Throughput-delay trade-off in wireless networks," in *Proc. IEEE INFOCOM* , Hong Kong, China, Mar. 2004, vol. 1, pp. 464–475.

[4] M. Franceschetti, O. Dousse, D. Tse, and P. Thiran, "Closing the gap in the capacity of wireless networks via percolation theory," *IEEE Trans. Inf. Theory*, vol. 53, no. 3, pp. 1009–1018, Mar. 2007.

[5] N. Bansal and Z. Liu, "Capacity, delay and mobility in wireless ad hoc networks," in *Proc. IEEE INFOCOM*, San Francisco, CA, Mar. 2003, vol. 2, pp. 1553–1563.

[6] X. Lin and N. B. Shroff, “The fundamental capacity-delay tradeoff in large mobile ad hoc networks,” presented at the 3rd Annu. Mediterr. Ad Hoc Netw. Workshop, Bodrum, Turkey, Jun. 2004.

[7] X. Lin, G. Sharma, R. Mazumdar, and N. Shroff, “Degenerate delay-capacity trade-offs in ad hoc networks with brownian mobility,” *IEEE/ACM Trans. Netw.*, vol. 52, no. 3, pp. 2777–2784, Jun. 2006.

[8] S. Toumpis and A. Goldsmith, “Large wireless networks under fading, mobility, and delay constraints,” in *Proc. IEEE INFOCOM*, Hong Kong, China, Mar. 2004, vol. 1, pp. 609–619.

[9] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, “Optimal throughput-delay scaling in wireless networks—Part I: The fluid model,” *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2568–2592, Jun. 2006.

[10] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, “Optimal throughput-delay scaling in wireless networks—Part II: Constant-size packets,” *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5111–5116, Nov. 2006.

[11] G. Sharma, R. R. Mazumdar, and N. B. Shroff, “Delay and capacity trade-offs in mobile ad hoc networks: A global perspective,” in *Proc. IEEE INFOCOM*, Barcelona, Spain, Apr. 2006, pp. 1–12.

[12] M. Neely and E. Modiano, “Capacity and delay tradeoffs for ad-hoc mobile networks,” *IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 1917–1937, Jun. 2005.

[13] Y. Zhang, W. Liu, W. Lou, and Y. Fang, “Securing mobile ad hoc networks with certificateless public keys,” *IEEE Trans. Depend. Secure Comput.*, vol. 3, no. 4, pp. 386–399, Oct.–Dec. 2006.

[14] C. Zhang, Y. Song, and Y. Fang, “Modeling secure connectivity of self-organized wireless ad hoc networks,” in *Proc. INFOCOM*, Phoenix, AZ, Apr. 2008, pp. 251–255.

[15] V. Bhandari and N. Vaidya, “Secure capacity of multi-hop wireless networks with random key pre-distribution,” in *Proc. 2nd IEEE Workshop Mission-Critical Netw.*, Phoenix, AZ, Apr. 2008, pp. 1–6.

[16] L. Eschenauer, V. Gligor, and J. Baras, “On trust establishment in mobile ad-hoc networks,” in *Proc. Security Protocols Workshop*, Cambridge, U.K., Apr. 2002, pp. 47–66.

[17] L. Eschenauer and V. Gligor, “A key-management scheme for distributed sensor networks,” in *Proc. ACM CCS*, Washingtion, DC, Nov. 2002, pp. 41–47.

[18] H. Chan, A. Perrig, and D. Song, “Random key predistribution schemes for sensor networks,” in *Proc. IEEE S&P*, Berkeley, CA, May 2003, pp. 197–213.

[19] J. Hubaux, L. Buttyan, and S. Capkun, “The quest for security in mobile ad hoc networks,” in *Proc. ACM MobiHoc*, Long Beach, CA, Oct. 2001, pp. 146–155.

[20] S. Capkun, L. Buttyan, and J. Hubaux, “Self-organized public-key management for mobile ad hoc networks,” *IEEE Trans. Mobile Comput.*, vol. 2, no. 1, pp. 52–64, Jan. 2003.

[21] P. Zimmerman, *The Official PGP User’s Guide*. Cambridge, MA: MIT Press, 1995.

[22] W. Du, J. Deng, Y. S. Han, and P. K. Varshney, “A pairwise key predistribution scheme for wireless sensor networks,” in *Proc. ACM CCS*, Washingtion, DC, Oct. 2003, pp. 42–51.

[23] D. Huang, M. Mehta, A. van de Liefvoort, and D. Medhi, “Modeling pairwise key establishment for random key predistribution in large-scale sensor networks,” *IEEE/ACM Trans. Netw.*, vol. 15, no. 5, pp. 1204–1215, Oct. 2007.

[24] V. Bhandari and N. Vaidya, “Capacity of multi-channel wireless networks with random (c, f) assignment,” in *Proc. ACM MobiHoc*, Montreal, QC, Canada, Sep. 2007, pp. 229–238.

[25] M. Penrose, *Random Geometric Graphs*. Oxford, U.K.: Oxford Univ. Press, 2003.

[26] R. Meester and R. Roy, *Continuum Percolation*. Cambridge, U.K.: Cambridge Univ. Press, 1996.

[27] O. Dousse, M. Franceschetti, and P. Thiran, “Information theoretic bounds on the throughput scaling of wireless relay networks,” in *Proc. IEEE INFOCOM*, Miami, FL, Mar. 2005, vol. 4, pp. 2670–2678.

[28] M. Kunde, “Block gossiping on grids and tori: Deterministic sorting and routing match the bisection bound,” in *Proc. 1st Eur. Symp. Algor.*, Honnef, Germany, Sep. 1993, pp. 272–283.

[29] M. Kaufmann, J. F. Sibeyn, and T. Suel, “Derandomizing algorithms for routing and sorting on meshes,” in *Proc. 5th ACM–SIAM SODA*, Arlington, VA, Jan. 1994, pp. 669–679.

[30] T. R. Mathies, “Percolation theory and computing with faulty arrays of processors,” in *Proc. 3rd ACM–SIAM SODA*, Orlando, FL, Jan. 1992, pp. 100–103.

[31] P. Gupta and P. R. Kumar, “Critical power for asymptotic connectivity in wireless networks,” in *Stochastic Analysis, Control, Optimization and Applications: A Volume in Honor of W.H. Fleming*. Boston, MA: Birkhauser, 1998.

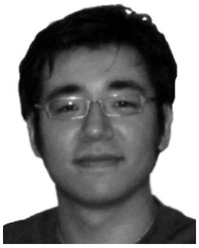
[32] T. Hagerup and C. Rüb, “A guided tour of Chernoff bounds,” *Inf. Process. Lett.*, vol. 33, no. 6, pp. 305–308, Feb. 1990.

[33] B. Bollobás, *Random Graphs*. Orlando, FL: Academic, 1985.

[34] S. Janson, T. Luczak, and A. Rucinski, *Random Graphs*. New York: Wiley, 2000.

[35] F. Xue and P. R. Kumar, *Scaling Laws for Ad Hoc Wireless Networks: An Information Theoretic Approach*. Delft, The Netherlands: NOW, 2006.

Chi Zhang (S’06) received the B.E. and M.E. degrees in electrical engineering from Huazhong University of Science and Technology, Wuhan, China, in 1999 and 2002, respectively, and is currently pursuing the Ph.D. degree in electrical and computer engineering at the University of Florida, Gainesville.



Yang Song (S’06) received the B.E. degree in electrical engineering from Dalian University of Technology, Dalian, China, in 2004, the M.E. degree in electrical engineering from the University of Hawaii at Manoa, Honolulu, in 2006, and is currently pursuing the Ph.D. degree in electrical and computer engineering at the University of Florida, Gainesville.

Yuguang Fang (S’92–M’97–SM’99–F’08) received the Ph.D. degree in systems engineering from Case Western Reserve University, Cleveland, OH, in 1994, and the Ph.D. degree in electrical engineering from Boston University, Boston, MA, in 1997.

He was an Assistant Professor with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, from July 1998 to May 2000. He joined the Department of Electrical and Computer Engineering, University of Florida, Gainesville, in May 2000 as an Assistant Professor, then received an early promotion to an Associate Professor with tenure in August 2003 and to a Full Professor in August 2005.

Dr. Fang is currently serving as the Editor-in-Chief of IEEE *Wireless Communications* and serves or has served on several editorial boards of technical journals, including the IEEE TRANSACTIONS ON MOBILE COMPUTING and IEEE TRANSACTIONS ON COMMUNICATIONS. He received the National Science Foundation CAREER Award and the Office of Naval Research Young Investigator Award.

Yanchao Zhang (S’03–M’06) received the B.E. degree in computer science and technology from Nanjing University of Posts and Telecommmunications, Nanjing, China, in 1999, the M.E. degree in computer science and technology from Beijing University of Posts and Telecommunications, Beijing, China, in 2002, and the Ph.D. in electrical and computer engineering from the University of Florida, Gainesville, in 2006.

He joined Arizona State University, Tempe, in June 2010 as an Associate Professor with the School of Electrical, Computer, and Energy Engineering. He was an Assistant Professor of electrical and computer engineering with the New Jersey Institute of Technology, Newark, from 2006 to 2010.

Dr. Zhang is an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, a Feature Editor of IEEE *Wireless Communications*, and a Guest Editor of the IEEE *Wireless Communications* Special Issue on Security and Privacy in 2010. He received the National Science Foundation CAREER Award in 2009.