Received 28 May 2020; revised 12 July 2020; accepted 27 July 2020. Date of publication 6 August 2020; date of current version 24 August 2020.

Digital Object Identifier 10.1109/OJCOMS.2020.3013947

Routing Efficiency of Infrastructureless Networks: A Comparative Analysis

SULEYMAN KONDAKCI (Member, IEEE)

Kalabak Mah. 3057 Sk. 17, 35430 Urla–Izmir, Turkey

CORRESPONDING AUTHOR: S. KONDAKCI (e-mail: suley.kondakci@gmail.com)

ABSTRACT This paper presents a novel approach to analyse the traffic efficiency of infrastructureless networks. Wireless communication is rapidly moving towards more infrastructureless routing schemes, where the devices encounter frequent topology changes and communicate with each other anonymously and sporadically. Nodes do not have the opportunity to choose an optimum routing path, they do rather obtain a valid routing path on the fly during the data exchange by cooperating with the nearest neighbours. Packets emanated from an infrastructureless node are disseminated through each node found in the vicinity. This produces a vast number of packet duplicates within the entire communication environment, causing the problem of increased congestion, increased packet delay, more packet drops, and undesirable energy consumption compared to the infrastructured networks. Data exchange and packet routing operations in large networks are highly stochastic. Thus, the performance analysis of routing operations is modeled as a stochastic queueing system, where the input and output traffics and throughput operations of routing systems are precisely formulated in the form of stochastic processes. A hypothetical routing algorithm is also defined here as a reference model in order to perform a comparative analysis of the infrastructureless routing scheme.

INDEX TERMS Ad-hoc routing, collaborative routing, energy sustainability, infrastructureless networks, routing optimization.

I. INTRODUCTION

THE WIDESPREAD use of Internet of Things (IoT) and mobile ad-hoc networks is rapidly growing with immense complexity, where routing is performed independent of any communication infrastructure. Obviously, it is a stochastic communication system with a significant complexity, where a vast number of heterogeneous devices exchange data through complex meshes in an ad-hoc manner.

It is hard to find an appropriate work dealing with the efficiency analysis of infrastructureless networks. Even though challenges are great, we are deeply motivated to initiate a research investigating whether the traffic efficiency of infrastructureless networks can be accurately modelled, if so, can we apply the model to performing numerical analysis of the network traffic in a reasonable manner?

It is difficult to assess the efficiency of a system without referring to a reference model for guiding the assessment process. The throughput analysis of infrastructureless networks presented in this paper will use a reference model, which is also introduced here. Hence, we propose a hypothetical routing scheme, acronymed as SOCRA (sink-oriented collaborative routing algorithm), as a reference model. Since our major objective is to focus on the throughput efficiency of the infrastructureless routing schemes, we omit dealing with the details of the SOCRA algorithm, and give only a brief overview of it here.

Mobile ad–hoc network (MANET) is an earlier wireless network architecture consisting of a set of mobile devices that are formed on a dynamic basis with a routing structure, which is built at random and operates in a multi hop manner, [1]. Due to dynamically (ad–hoc) growth of network nodes, developing an efficient routing algorithm became a prominent research field. Hence, several routing protocols have been developed to address the challenges found in MANET routing algorithms, some which are Dynamic Source Routing (DSR), [2], and Ad–hoc on-demand Distance Vector (AODV), [3]. Other references to these and some other related protocols are given in the Related Work subsection.

An infrastructureless (e.g., mobile ad-hoc networks, sensor networks, IoTs, etc.) network is a collection of mostly wireless nodes having regular identities (i.e., IP and physical addresses) and no specifically defined routing structure, where each node behaves as a gateway to its adjacent nodes. Data packets with unknown destinations are handed off to the nearest reachable devices almost with unlimited constraints. Latency in packet forwarding and energy consumption are the most prevailing problems under the focus of many researchers and engineers developing technologies for the infrastructureless communication. The higher the latency in exchanging the data, the more the energy required by the network elements. The situation is worsened for ubiquitous network elements communicating under harsh circumstances. Besides, path lifetime of most mobile nodes (e.g., velocity-aware nodes) is critical when a source node tries to setup a routing topology, [4], in the meantime the current topology undergoes a rapid change.

Frequent updates of routing paths is a major concern for most ad–hoc routing protocols. New routing paths are built whenever the current routing paths become invalid. Besides, each data exchange request at a source node causes a new route setup even though an operational path already exists. In order to setup a routing path the source node disseminates a route request packet across the entire network, [5]. Any node receiving the route request packet responds with a route reply packet to the source node. The route reply packets received from multiple nodes at the source node are then used to recompute a shortest path between the source and the destination hosts. Since the route request packets are flooded throughout the entire network a vast number of packet duplicates will be exchanged among the active nodes. Furthermore, current shortest path may vanish (due to topology change) before the reply packets reach the source node. Hence, the newly computed shortest path to the destination node may become the longest path leading to additional deficiencies, e.g., packet delays and losses.

Infrastructureless communication basically relies on the eligibility of nearest neighbours for quickly discovering effective routing paths to destinations. However, due to geographical and/or RF range constraints, hosts need multi-hop forwarding for quickly finding a shortest path to handoff the packets further. Unfortunately, an obvious drawback with the infrastructureless flooding scheme is the fact that frequently and unpredictably changing of the network topology leads to tremendous amount of packet delays and packet drops, hence undesirably more energy consumption.

Since the nodes of infrastructureless networks flood all unknown packets to all adjacent channels within the communication range, the adjacent nodes receive duplicates of flooded data packets leading to increased traffic congestion, more energy consumption, frequent power interruptions, and path losses. These problems are encountered more frequently when the number of nodes and the geographical distances among the nodes increase unduly. Our simulations prove also this fact, where the simulation results show that the flooding–based ad–hoc routing protocols perform poorly compared to that of the sink-oriented collaborative routing (SOCRA).

It is hard to numerically determine efficiency of infrastructureless routing schemes used with the mobile ad-hoc devices. However, we can analyze the routing efficiency by a comparative approach using a predefined reference routing scheme. Since the network activities occur randomly, we apply stochastic modeling techniques to the throughput analysis with the support of computer simulations. We do avoid the manipulation of mathematical proofs of already proven theories in order to keep the content concise and reader–friendly. Accordingly, the reader may refer to [6] and [7] for the details of stochastic processes and queueing theory.

A. CHALLENGES AND MOTIVATIONS

Considering many critical cases, infrastructureless communication might be highly inefficient and unreliable as well. Therefore, it is important to select the right communication technology to match the requirements of the application area. The task of reliable data exchange under critical circumstances is highly complicated, which primarily requires (i) mobile connectivity for reaching physically hard-to-access locations, (ii) interoperability among heterogeneous devices, (iii) secure data transportation, (vi) minimum energy consumption, (v) sustainable routing in large networks, and (vi) collaborative routing among distant autonomous networks.

Dynamic topology changes and the rapid growth of the Internet cause a serious impact on the routing efficiency of end systems. Reference [8] considers the problem of the sustainability of the Internet growth mostly dominated by the distributed management of autonomous networks owned by various business, which makes the Internet a set of self-organized systems taking the control over the routing infrastructure of the Internet. Thus, the steady increase in size and dynamics of the Internet leads to rapidly growing routing complexity, which causes concerns about the Internet routing architecture that may have serious sustainability problems in the next decade. Hence, [8] suggests a greedy forwarding routing algorithm supported by a method that maps the coordinates of the Internet elements into a hyperbolic space. Another recent work considering the evaluation of network navigability is presented in [9], which mainly focuses on the greedy routing assessment of complex networks using the node geometrical coordinates.

It is a fact that a wide spectrum of mobile wireless devices and applications are connected together via public networks in an ad–hoc manner. Through this conglomerate structure users require secure and reliable data exchange, while some operations demand an efficient coordination of critical communication tasks such as public protection, disaster relief, and remote data gathering and reconnaissance tasks for critical operations (e.g., ambulatory services, disaster control systems, and anti-terror operations). These operations bear many challenges, for instance, coordination of ambulatory operations provided by an emergency/medical center within some harsh geographical regions. As a typical case, suppose that a medical center has some sensors and devices that monitor and transmit vital patient data to the emergency center, while allowing medical personnel to remotely guide the patients for self-care. The communication system has now a critical situation, where delay in data transmission and power loss of the devices cannot be tolerated during the remote patient care. A number of critical rescue operations that are remotely coordinated/controlled using wireless devices are some other cases that rely on an effective infrastructureless communication.

Cyber-physical systems often make use of mobile wireless devices for collecting and monitoring data, diagnosing, maintaining, and controlling of remote entities. Powering of mobile devices in the long run is a prominent challenge. To enable sustainable powering, it is often desirable to harness unmanned aerial vehicles (UAVs) to work as mobile charge units for powering up the devices (e.g., under disaster management). For example, an UAV–assisted intelligent transportation system is detailed in [10]. In order to achieve a reliable and delay–free communication under critical circumstances we need to have efficient routing algorithms rather than the ones used in the infrastructureless communication. One way to solve such a problem is to configure an UAV to operate as an intermediate gateway for devices that are located in hard-to-access (critical) locations.

Signal strength degradation, jamming, extensive bandwidth usage, interference, and power leakage are some other challenges with wireless sensors and mobile IoT devices. Physical obstacles, intensive mobility, and geographically scattering of devices may lead to substantial signal degradation. This will, in turn, cause excessive packet delays and duplicates, and consequently higher network congestion and frequent packet rejections. Therefore, it is necessary to asses the routing capability of these devices in order to determine eventual throughput problems. Solving the problems may increase the channel throughput, reduce the bandwidth usage and minimize the power consumption of the devices.

One can quickly perceive the fact that complexity and heterogeneity of the devices used today bear grate challenges for designing efficient policies, algorithms, and techniques in order to incorporate the necessary mechanisms that can enable the systems to interoperate efficiently and securely with each other, as well as with external communication facilities.

B. OUTLINE OF THE PAPER

In the following, Section II gives a brief overview over the related topics. The SOCRA approach is introduced in Section III. Section IV considers the theoretical model of the efficiency analysis, which consists of the analysis of network packet traversal and the related mathematical descriptions. Throughput analyses will be considered under two sections, single channel and multiple channels, Section V and Section VI, respectively. Simulation and results are discussed in Section VII, and the conclusion of the paper is given in Section VIII.

II. RELATED WORK

Infrastructureless networking is still in its premature phase, where mostly the ad-hoc communication techniques have been used to establish task-based networking incorporating wireless devices and sensor hubs. For example, Device–to-Device (D2D), Vehicle-to-Vehicle (V2V), Machine–to–Machine (M2M), Mobile Ad–hoc Networks (MANET), and Vehicular Ad–hoc Network (VANET) are some of the task–based systems. A routing protocol for Multi-hop Device-to-Device (MD2D) communications is presented in [11]. A routing algorithm for bus-based routing techniques in Urban Vehicular Networks is presented in [12]. Another system for fog-based distributed routing for V2V Communication in Urban Vehicular Ad-hoc Networks is discussed in [13]. Due to the rapid growth of IoT and wireless sensor network (WSN) deployments, inter-domain routing with infrastructureless communication has became more complicated. The work given in [14] discusses some approaches dealing with the evolution of inter-domain routing approaches. The AODV routing protocol has been mentioned as one of the leading MANET protocols, [15], [16], [17], and [18]. A survey on LTE, [19], and its supported applications with the focus on strengths and weaknesses of LTE in use with VANETs is presented in [20]. Energy efficiency of wireless sensor networks (WSNs), [21], is still being considered as one of the prevailing research field. Related to this, [22] proposes an algorithm called Energy and Traffic Aware Routing (TEAR) algorithm, where based on the simulation results, the authors claim that TEAR performs better than the cluster-based routing algorithms used in similar topologies. Furthermore, performance evaluation of routing protocols for ad-hoc WSNs is discussed in [23]. The survey in [24] goes through energy-efficient routing protocols in WSNs. Yet another survey on wireless sensor networks with mobile sinks is given by [25].

Rapidly growing IoT installations and ad-hoc sensor network environments with a large number of heterogeneous nodes may have a substantial impact on the communication speed. Low-power wireless personal-area network (6LoWPAN) devices communicate ubiquitously in an ad-hoc manner. Reference [26] proposes an energy optimization algorithm using artificial bee colony algorithm for MANETs. A modified version of the existing dynamic source routing protocol (DSR) for MANETs is presented by [2]. An energy consumption model for delay-aware communications in 5G wireless networks is presented in [27]. Furthermore, [28] discusses joint scheduling and routing issues by focusing on power control for centralized WSNs. In order to improve the capacity and performance, a protocol for multi–channel networks with infrastructure support is proposed in [29].

As already mentioned, mobile ad-hoc devices use a built-in routing algorithm that exchange routing information via neighbouring devices. The reader my refer to the following literature in order to gain an overview on the existing routing systems, some of them are ad-hoc On-demand Distance Vector (AODV), On-Demand Multicast Routing Protocol (ODMRP), Location-Aided Routing (LAR), Optimized Link-State Routing protocol (OLSR), Dynamic Source Routing (DSR), zone routing protocol (ZRP), Gossip based ad-hoc Routing (GOSSIP), and Dynamic Manet on Demand (DYMO) protocol. For instance, a survey on the routing protocols used in ad–hoc networks is presented by [30], whereas [31] considers another survey on geographical routing in wireless ad–hoc networks. RouT, yet another routing protocol based on topologies for heterogeneous WSNs is presented in [32]. An extensive survey on routing protocols of WSNs is considered by [33]. The design of an adaptive broadcast protocol for VANETs is presented in [34]. Reference [35] consideres performance analysis of AODV and DSR protocols in some detail. There exist some approaches for estimating route distances in WSNs, e.g., [36], [37], and [38]. A geographical routing protocol based on the road perception in VANETs is presented in [39]. A topology–aware routing protocol specially designed for unmanned aerial vehicle networks with the focus on low-latency and high-mobility applications is proposed in [40].

There is a tight interdependence between the transmission signal strength and the power consumption. Related to this, an approach is proposed by [41] that adjusts signal strength according to predetermined distances to neighbor nodes. It is also a fact that noise and interference are some of the everlasting problems in the wireless communication. In this regard, [42] considers challenges and research directions dealing with interference alignment for WNs communicating over multi–hops. A variety of ant colony optimization algorithms has become quite popular for dealing with problems related to energy-efficiency in multi–path routing structures, [43]. To improve the spacial throughput of self-organizing small-cell networks [44] proposes an approach based on the coalitional game model. A load–balanced algorithm for the collection and distribution of data with minimal traffic among static WS nodes is suggested in [45].

III. BRIEF DESCRIPTION OF SOCRA

The hypothetical SOCRA scheme is an algorithm incorporating mobile end systems and gateways that embody an adaptive collaborative routing infrastructure. It is a self–organizing wireless ad-hoc network supported by a quasi–infrastructure to improve the network throughput. Fig. 1 illustrates a hypothetical topology of three different collaborative domains (D1, D2, and D3). A task domain is a collaborative network configured to perform a specific set of communication tasks. A communication system for ambulatory services of a hospital, a network designed for critical rescue operations performed under harsh circumstances (e.g., forest fire) are typical task domains. As shown in Fig. 1, the geographical sizes of the domains are denoted by width

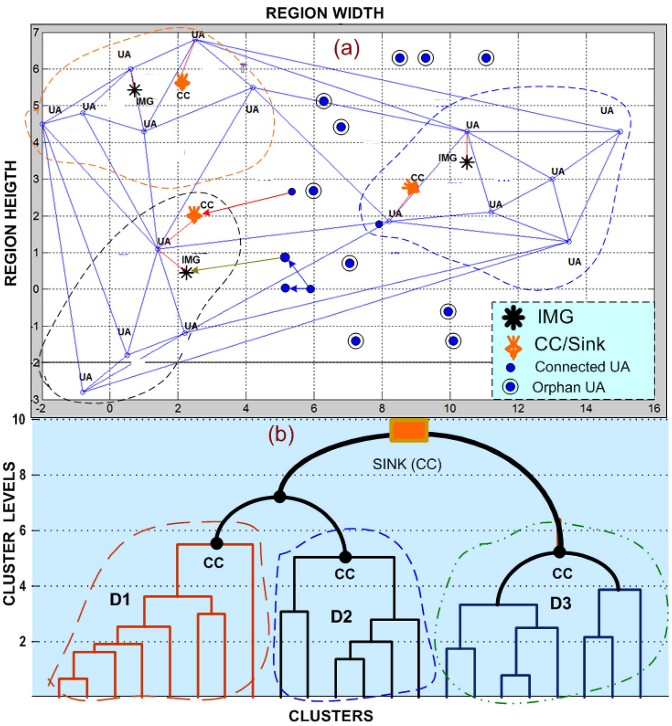

FIGURE 1. (a) Three different task domains D1, D2, and D3 (dashed areas) configured as collaborative networks, where each domain element is illustrated by a symbol related to its role in the network. Encircled dots represent UAs that are idle. Hypothetically designed geographic coordinates vary between −10 and +10 generic units of distance. (b) shows the hierarchical organization of the domains highlighting the different cluster levels.

(X) and height (Y) coordinates as the dimensions in X times Y units of distances. Each domain consists of control centres (CC), intermediate gateways (IMGs), and user agents (UAs), where IMGs are authorized domain routers and authenticators, and UAs are the nodes used for collecting and transmitting data for different purposes. A CC is an ultimate destination (sink) for the transmission of payload and control data that manage the respective domain. That is, a CC is the control center of a domain or a set of domains, however a sink is either an ultimate destination or a CC node, if the CC is the ultimate destination of a communication end. For example, when a CC constructs a domain it broadcast a delegation packet containing a list of IMGs being delegated as the domain/cluster heads. In this case, the CC is one end communicating with the IMGs, where it is a sink during that communication period.

The encircled nodes (dots) represent UAs that are tied to a collaborative structure (colony), whereas an orphan node (encircled dot) is a homeless UA that is not currently roamed within any cluster. End systems called user agents (UAs) and intermediate gateways (IMGs) cooperate towards a sink (destination) for increased transport efficiency. An agent node can be an IoT device, a VANET node, a sensor node, or generally any device that can remotely collect and broadcast data to a target (sink) node. An IMG is a beacon node with some additional capabilities, e.g., building collaborative

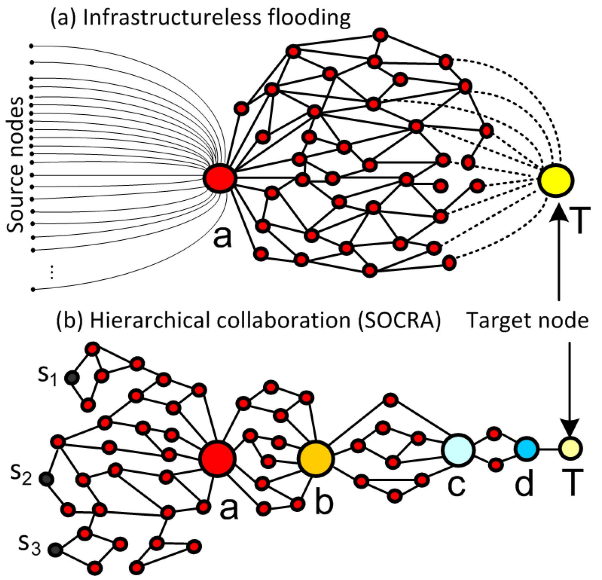

FIGURE 2. Illustration of packet propagation through flooding and SOCRA topologies for equal number of incoming packets. Source nodes of SOCRA S_1,S_2, and S_3 pass the same traffic towards node a, whereas the infrastructureless node a receives all incoming packets.

clusters, managing/delegating authentication of the collaborating nodes and serving as a cluster head for dynamically organizing routing clusters to colonize the nodes. Locations of IMGs are determined before and after localization operations, so that UAs will quickly find cluster heads to optimize their relative positions. SOCRA enables adaptive hierarchical colonization by computing the shortest delay to all neighbour nodes and by locating the nearest IMGs among the authenticated head candidates.

Unlike the ad-hoc network routing, SOCRA prevents a node from flooding its packets to all outgoing links and to exterior domains. Hence, it reduces the number of packet duplicates, delays, packet losses, and energy consumption. Fig. 2 shows a simplified illustration of the topological difference between the SOCRA and infrastructureless domains, where some packets are destined to the target node **T**. It is a fact that in an infrastructureless network a node can be arbitrarily visited by a vast number of connection requests for forwarding data packets from different sources. On the contrary, SOCRA has a selective request policy, which significantly reduces the number of requests for packet forwarding. As indicated in Fig. 2 (a), the number of incoming packets to node **a**, due to the specification of the infrastructureless routing policy, is extremely large compared to node **a** of the SOCRA routing topology (Fig. 2 (b)), because a SOCRA node forwards all incoming packets with unknown destinations to its cluster head (i.e., the nearest IMG). Here, the node **a** behaves as the nearest IMG. Likewise, nodes **b**, **c**, and **d** can operate as IMGs for some selected clusters, but not for the entire domain. However, if these nodes were found in an infrastructureless topology, then they would have to behave like an IMG for the entire network.

End systems, i.e., UAs, do only collect and forward data within their own domain (or hierarchy) or within the collaborating domains. As with the infrastructureless networks, transmitting data via randomly scattered anonymous nodes takes much longer time. To overcome this, SOCRA has an adaptive selection of channels that can quickly deliver data to the destination with significantly mitigated packet delays and losses.

IV. MODELLING THE PACKET TRAVERSAL

The number of input packets (traffic density) and per packet processing time at a node affect both the energy consumption and the communication efficiency throughout the intermediate nodes in a network. For the evaluation purpose, two types of routing schemes are considered here, flooding as of the infrastructureless networks and the SOCRA routing scheme. Simple stochastic models from the queuing theory are applied to the efficiency analysis consisting of the traffic analysis for input and output and the overall throughput of a given route path. The, most common delays in a communication network are due to (i) packet propagation time increased with the number of flooding nodes and mobility of nodes, (ii) packet processing (segregation and input buffering) at a node, and (iii) packet retransmission (output buffering time plus forwarding of duplicated packets), (v) physical circumstances of the transmission media/environment (noise, multipath signal degradation, jamming, and so on).

A network traffic consists of data packets from multiple stations operating in a random access manner, where each station exchanges a large number of packets behaving as a stochastic process. Both the number of packets and the size of each packet are also random. Thus, input system of a node can be modelled as a Poisson process, [46], whereas the segregation and forwarding of packets at a node can be best expressed by Bernoulli trials, which follows the Binomial distribution. Packet arrival rate of a Poisson process is generally high, while the packet delivery rate to the respective node is low, [6] and [7]. Only a small portion of the packets are delivered, even though a large number of packets arriving at a node at a time, remaining packets are forwarded to their destinations. Thus, for example, equation (19) expresses the packet arrival for single channels, whereas (21) expresses the packet arrival for multiple channel, and (20) expresses the packet delivery or packet drop at a node.

Input to Nodes: Incoming packets from a single channel to infrastructureless (ad-hoc) nodes and to SOCRA nodes are identically modeled, and follow the Poisson distribution. Hence, the probability of finding exactly m packets at an intermediate node coming form a single channel is given by

$$P\{X_{<1,m>}(t) = m\} = \frac{(\lambda t)^m}{m!} e^{-\lambda t}, \; m = 0, 1, 2, \ldots, \quad (1)$$

where λ denotes the packet arrival rate in t time units. However, probability that packets coming from n multiple independent sources each with different arrival rate (λ_i) is computed by applying the additive property of the Poisson process, i.e.,

$$P\{X_{<n,m>}(t) = m\} = \frac{(\Lambda_m t)^m}{m!} e^{-\Lambda_n t}, \tag{2}$$

where $\Lambda_n = \lambda_1 + \lambda_1 +, \ldots, +\lambda_n$ gives the sum of all incoming channel rates, and $\Lambda_m = \lambda_2 + \lambda_1 +, \ldots, +\lambda_m$ denotes the sum of the rates for m packets. The arrival rate (or traffic density) defined by λ_i can be either constant and equal for all incoming channels or random for each channel. In the latter case, the rate is a random function $\xi(\lambda)$ defined as a random process, which can be empirically determined and justified to match the characteristics of the respective channel.

Output From SOCRA Nodes: SOCRA applies a selective flooding to incoming packet streams, where packets with unknown destinations are forwarded to the nearest active cluster. Hence, the number of outgoing packets is relatively small and the probability of the total number of forwarded packets is adequately modelled as a Bernoulli process described by the binomial distribution,

$$P_m(t) = \binom{N}{m} p(t)^m [1 - p(t)]^{N-m}, \quad m = 0, 1, \ldots, N. \tag{3}$$

where N denotes the number of incoming packets per input channel, m denotes the number of selected packets for forwarding, and p is the probability of successfully segregating the forwarded packets. $1 - p(t)$ denotes the probability of dropped/delivered packets, either due to delivering of the packet to this node or due to packet ageing (excessive delay or TTL value expires). For n input channels each with success probability p_i, we then have

$$X(t) = \sum_{i=1}^{n} \binom{N}{m} p_i(t)^m [1 - p_i(t)]^{N-m}, \quad m = 0, 1, \ldots, N. \tag{4}$$

as the total number of packet to forward further. This can be computed by convolving the binomial variables obtained for each input channel. Assuming the distribution of the sum of two discrete random variables is the convolution of their values, then, given

$$X_1 \sim Binom(n, p_1), \quad X_2 \sim Binom(n, p_2),$$

and their sum $Z = X_1 + X_2$ is the convolution of the distributions X_1 and X_2 given by

$$P\{Z = z\} = \sum_{k=1}^{\infty} P\{X_1 = k\} P\{X_2 = z - k\}. \tag{5}$$

Hence, by running the convolution iteratively over the remaining distributions $[X_3, X_4, \ldots, X_n]$, we obtain the overall distribution for n incoming lines. Let

$$Z_1 = X_1 * X_2 \text{ be the first convolution,}$$

then the overall convolution of the remaining variables can be iteratively computed as

$$Z = \biguplus_{k=1}^{n} Z_k * X_{k-1}; \quad Z_k \text{ being the current result.} \tag{6}$$

However, if all incoming channels have an equal probability p, then the combined distribution can be easily obtained by

$$Z \sim Binom\left(\sum_{k=0}^{n} k, p\right). \tag{7}$$

Although it is extremely hard to experiment, bitwise throughput analysis can give more accurate results. However, in most cases, the analysis will end up as a typical queuing process with a small number of parameters such that we have here. Now then, a simpler experiment can be sufficient to quantitatively substantiate the efficiency of SOCRA throughput. Based on the queuing theory, mean time delay per packet is $1/(\mu - \lambda)$, given that incoming packet rate λ and outgoing packet rate μ at a node are independent and exponentially distributed parameters (M/M/1 queuing system, [6]).

Output From Infrastructureless Nodes: A node in a Mesh (mobile ad-hoc) network behaves as a source of a branching process, i.e., it just bursts out every incoming packet to all outgoing channels. In most cases, the nodes are spread over large geographical areas, where each subarea contains several levels of branches generating immense network traffic. Therefore, we assume a branching process based on a modified Borel–Tanner distribution with parameters X_0, and $\xi(\alpha)$ will suffice to model the probability that the total number of packets received at a flooded node is k and given by

$$P\{k \mid X_0, \xi(\alpha)\} = \frac{X_0 (k\xi(\alpha))^{(k-X_0)}}{k(k - X_0)} e^{-k\xi(\alpha)}. \tag{8}$$

where, X_0 denotes the initial number of the flooding nodes and $\xi(\alpha)$ holds the mean traffic density, which is a random process distributed according to the case at hand.

Since it is likely that some packets are dropped and/or sunk at the flooded node, it is appropriate to model $\xi(\alpha)$ to match the Weibull distribution,

$$\xi(\alpha) = \left(\frac{\beta}{\alpha}\right)\left(\frac{t}{\alpha}\right)^{(\beta-1)} e^{-\zeta}; \quad \zeta = \left(\frac{t}{\alpha}\right)^{\beta}, t > 0. \tag{9}$$

When its shape parameter goes below 1 ($\beta < 1$, i.e., higher packet drops) then the branching process will tend to extinct with a certain rate over some specific time. Increasing the value of scale parameter α while keeping β constant has the effect of prolonging the extinction time. That is, increased scale parameter gives higher input packet rate.

A branching process is a stochastic process whose state is a set of counts, where each count in the set independently generates some random number of offspring, which recursively reproduce descendants until an extinction state takes place. That is, let

$$P\{\xi_i = i\} = p_i, \quad \sum_{i=0}^{\infty} p_i = 1, \quad i = 0, 1, 2, \ldots,$$

meaning that p_i is the probability of generating ξ_i branches (offspring) at a stage. Hence, ξ_0 denotes the initial number of descendants, ξ_1 denotes the number of descendants of ξ_0, ξ_2 denotes the number of descendants of ξ_1, and so on. We have two assumptions here, (1) all individuals reproduce independent of each other, (2) sizes of different offspring are independently distributed stochastic variables. Let us assume some generations of different sizes denoted by $\{X_0, X_1, X_2, \ldots, X_n\}$, where the population size of the first generation is denoted by X_0 and the size of nth generation is denoted by X_n. Assuming the initial population size $X_0 = 1$, then we can readily obtain the population size of any generation, e.g., nth generation, by

$$X_n = \sum_{i=1}^{X_{n-1}} \xi_i. \tag{10}$$

A common probability generation function, $G(s)$, is the core of the branching process, which generates independent and identically distributed random variables, i.e., ξ_i $(i = 1, 2, \ldots, k)$. In the general form, the probability generation function of a single event is

$$G_1(s) = G(s) = \sum_{i=0}^{\infty} p_i s^i, \tag{11}$$

and for the nth generation

$$G_n(s) = \sum_{i=0}^{\infty} P\{X_n = i\} p_i s^i, \quad n = 0, 1, 2, \ldots \tag{12}$$

The sum of the random variables, $\xi_1 + \xi_2 + \cdots \xi_k$, given in Eq. (10) has the probability generation function obtained by recurring over all generations as

$$\begin{aligned} G_{n+1}(s) &= G_n(G_{n+1}(s)); \text{ recurrence relation,} \\ G_{n+1}(s) &= \sum_{k=0}^{\infty} P\{X_n = k\} \sum_{i=0}^{\infty} P\{\xi_1 + \xi_2 + \cdots\} s^i. \end{aligned} \tag{13}$$

Assuming the offspring has the Poisson distribution $\xi(\lambda) \sim Poisson(\lambda), \lambda > 1$, then the function for the branching growth can be expressed as

$$\begin{aligned} G_n(s) &= \sum_{i=0}^{\infty} P\{X_n = i\} p_i s^i \\ &= \sum_{i=0}^{\infty} \frac{s^i \lambda^i}{i!} e^{-\lambda} = e^{\lambda(s-1)}, \quad \lambda \geq 1. \end{aligned} \tag{14}$$

Expected size of the n^{th} generation, $\mathbf{E}(X_n)$ conditioned on $\mathbf{E}(X_{n-1})$, is simply computed by iterating through the sums of the all intermediate generations up to X_{n-1}, i.e.,

$$\begin{aligned} \mathbf{E}(X_n) &= \mathbf{E}(\mathbf{E}(X_n \mid X_{n-1})), \\ \mathbf{E}(X_n) &= \sum_{k=0}^{\infty} P\{X_{n-1} = k\} \mathbf{E}(X_n \mid X_{n-1}) \\ &= \sum_{k=0}^{\infty} \lambda k P\{X_{n-1} = k\}. \end{aligned} \tag{15}$$

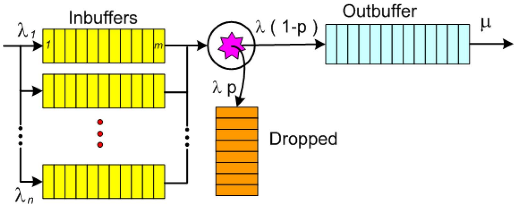

FIGURE 3. Packet traversal from multiple channels through a single node.

Hence, recurring proceeds over all sums of the expected values as

$$\mathbf{E}(X_n) = \sum_{k=0}^{\infty} \lambda k P\{X_{n-1} = k\} = \lambda \mathbf{E}(X_{n-1}). \tag{16}$$

Thus, incorporating Eq. (8) into (16), we obtain the total expected number of packets by

$$\mathbf{E}(X_n) = \sum_{k=0}^{n} \frac{X_0 (k\xi(\alpha))^{(k-X_0)}}{k(k-X_0)} e^{-k\xi(\alpha)}. \tag{17}$$

Consequently, each individual packet will extend to many branches of flooding. Given that the initial number of individuals $X_0 \geq 1$ and a constant flooding rate λ we obtain the expected size of the n^{th} generation as

$$\alpha^n \mathbf{E}(X_0) = \alpha^n, \quad n = 1, 2, \ldots \tag{18}$$

It is assumed that the branching process goes extinct with probability 1 when $\lambda \leq 1$, which means that the last generated packet has been either dropped or delivered to its destination.

V. SINGLE CHANNEL THROUGHPUT

A node is modeled as a single server with infinite buffer sizes both for incoming packets and for outgoing packets. The model consists of two tandem processes, one for handling the arrival process and one for the departure (forwarding) of packets, Fig. 3.

A router (or a forwarding node) works as an $M/M/1$ single queue system, i.e., it has exponentially distributed packet arrival (λ) and departure (μ) rates, with Markovian behaviour, [47]. Here, M stands for exponential distribution and 1 stands for single machine (system). As shown in Fig. 3, outgoing packets are segregated and buffered with probability $1 - p$. Some of the incoming packets are dropped and/or delivered (sunk) locally with probability p, which can be described as a Bernoulli process, see Eq. (4). Dropping of a packet means that a packet is rejected either due to excessive delay or the packet is being delivered (sunk) to its ultimate destination.

A Poisson sequence model can appropriately depict the stream of packets arriving from a large number of sources. For the incoming packet sequence, the number of incoming packets from channel i at time t is Poisson distributed with rate λ_i. Thus, the probability that there are m packets arrived through an arbitrary channel at time t is given by

$$P\{m, t\} = \frac{(\lambda_i t)^m}{m!} e^{-\lambda_i t}. \quad (19)$$

The mean interarrival rate for the incoming packets is exponentially distributed as $\lambda_i e^{-\lambda_i t}$. Now, suppose we have k packets remaining in the inbuffer after extracting the locally delivered and dropped packets. This can be modeled as a Bernoulli process, i.e., the probability that $k(0, \ldots, m)$ packets are still in the inbuffer coming from a channel at time t is obtained by

$$P\{k, m \mid t\} = \binom{m}{k} p^k (1-p)^{m-k}, \quad (20)$$

where, p denotes the probability of a packet found in the inbuffer at time t. The Poisson process describing the multiple channel packet arrival, given by Eq. (2), will then become

$$P\{k, m \mid t\} = \frac{(\Lambda_m t)^m}{m!} e^{-\Lambda_n t} \binom{m}{k} p^k (1-p)^{m-k}. \quad (21)$$

This process introduces a time delay $\mathbf{T_{in}}$ for each incoming channel, which consists of time slots for segregating and outbuffering of the packets for forwarding later, i.e., waiting time plus process time for outbuffering a single packet. Thus, we obtain the total time usage for collecting N packets from n incoming channels as

$$\mathbf{T_{in}} = \sum_{j=1}^{n} \sum_{i=1}^{N} \theta_i (S_i + B_i) \lambda_j e^{-\lambda_j}. \quad (22)$$

Where, S_i denotes the segregation time of packet i, B_i denotes the buffering time of segregated packets to forward, $\lambda_j e^{-\lambda_j}$ gives the mean interarrival rate of packets for channel j, and θ_i denotes the size of packet i given as the random number of bytes. One can also by Little's law, [6], observe that the average number of packets found in the outbuffer for transmitting via the outgoing channels (adjacent nodes) is obtained by

$$\mathbf{N_t} = \lambda(1-p)\mathbf{T_{in}}.$$

Forwarding of packets from the outbuffer will also introduce a time delay, $\mathbf{T_{out}}$, given by

$$\mathbf{T_{out}} = \sum_{i=1}^{\mathbf{N_t}} (W_i + T_i \theta_i), \quad (23)$$

where, W_i denotes the average time packet i being held in the outbuffer and T_i stands for the average transmission time of packet i while forwarding it from the output buffer. Thus, the overall transmission time through a single channel is computed by

$$\mathbf{T} = \mathbf{T_{in}} + \mathbf{T_{out}}. \quad (24)$$

A. STEADY–STATE ANALYSIS

It is assumed that in a long run, the router is in the steady–state, where $p(m) = \lim_{t\to\infty} P\{M(t) = m\}$ is the steady–state probability of having m packets in the system coming from a single source. The utilization factor of a stable system in its steady–state is denoted by $\rho = \lambda/\mu < 1$. It has been also proven that in a steady–state, $\mathbf{L} = \lambda\mathbf{W}$ gives the number of requests in a system, which has λ as the expected arrival rate of requests and $\mathbf{W}$ as the expected time spend by each request in the system. Replacing $\mathbf{L}$ with $\mathbf{N}$ and incorporating Eq. (22), we have the expected number of packets in the routing system given by

$$\mathbf{N} = \lambda\mathbf{T_{in}}.$$

Based on the PASTA (Poisson Arrivals See Time Averages) of queueing theory, [7], the probability of having n packets in the system is given by

$$p_n = \rho^n p_0, \quad p_0 = \left(\sum_{n=0}^{\infty} \rho^n\right)^{-1} = 1-\rho, \sum_{n=0} p_n = 1.$$

Therefore, considering λ as the incoming packet rate and μ as the input buffer processing rate, the expected number of packets in the input buffer will be obtained by

$$\mathbf{N} = \sum_{n=0}^{\infty} n p_n = \sum_{n=0}^{\infty} n\left(1 - \frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)^n, \quad n = 0, 1, 2, \ldots \quad (25)$$

Thus, solving Eq. (25) for steady–state with $\rho = \lambda/\mu$, we obtain a practical solution to the expected number of packets and mean delay (so called sojourn time) per packet as

$$\mathbf{N} = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu - \lambda},$$
$$\mathbf{T} = \frac{\mathbf{N}}{\lambda} = \frac{\rho}{1-\rho}\left(\frac{1}{\lambda}\right) = \frac{1}{\mu-\lambda}, \quad \rho < 1. \quad (26)$$

The expected number of packets before transmitting (queue length found in the outbuffer) is given by

$$\mathbf{N_q} = \mathbf{N} - (1-p_0) = \mathbf{N} - [1-(1-\rho)] = \frac{\lambda^2}{\mu(\mu-\lambda)} = \frac{\rho^2}{1-\rho}. \quad (27)$$

One can also compute this by Little's Theorem using (26) as

$$\mathbf{N_q} = \lambda\mathbf{T} = \frac{\rho^2}{1-\rho}. \quad (28)$$

Based on the experiments combined with simulation results, the throughput of the SOCRA scheme for a constant input rate was increased by a factor of β compared to that of the ad-hoc flooding scheme. Hence, the SOCRA scheme has an efficiency factor (complement of utilization) different than that of the ad-hoc networks (i.e., $1-\rho$), given by,

$$\Theta = 1 - \frac{\lambda}{\beta\mu} \le 1, \quad \beta = 1, 2, \ldots \quad (29)$$

It can be noted that $\frac{\lambda}{\beta\mu}$ also represents the utilization of a multiple–server system. Hence, the expected length of the output queue of SOCRA is

$$\mathbf{N_q} = \sum_{n=\beta}^{\infty}(n-\beta)p_n = \frac{\rho^{\beta}\rho p_0}{\beta!(1-\rho)^2}. \tag{30}$$

The expected utilization factor of the infrastructureless routing scheme is $\rho = \lambda/\mu$. Incorporating the dropped packets at a node, the expected number of packets from channel i found in the respective input buffer (see, Fig. 3) under the steady–state, is given by

$$\mathbf{N_{in}} = \frac{\rho_i}{1-\rho_i} = \frac{\lambda_i}{\mu_i - \lambda_i} - \lambda_i p_i. \tag{31}$$

Here, $\lambda_i p_i$ denotes the expected number of dropped and/or delivered packets for the respective channel (i.e., channel i). λ_i and μ_i stand for input and output buffering rates and ρ_i denotes the traffic utilization of channel i. From Eq. (25) we can obtain the probability that there are at least m packets in a single system by

$$P\{\mathbf{N} \geq m\} = \sum_{i=m}^{\infty}(1-\rho)\rho^i = \rho^m. \tag{32}$$

Reader may refer to [48] for a typical stochastic queueing example applied to a scenario of mass e-mail propagation.

VI. MULTIPLE CHANNEL THROUGHPUT

Assume we have $\mathbf{N_{in}}$ as the average total number of packets segregated from n incoming channels in the outbuffer, then the overall number of packets $\mathbf{N}$ and the total delay time $\mathbf{T}$ for all incoming channels will evolve as

$$\mathbf{N} = \sum_{k=1}^{n}\mathbf{N_{in}}, \quad \mathbf{T} = \sum_{k=1}^{n}\mathbf{T_m} = \frac{\mathbf{N}}{\sum_{i=1}^{n}\mu_i} = \frac{\mathbf{N}}{\mathbf{E}\mu}. \tag{33}$$

respectively. One can quickly observe from Eq (33) that the average throughput is

$$\Upsilon = \frac{\mathbf{N}}{\mathbf{T}} = \mathbf{E}\mu. \tag{34}$$

However, the delay through the hierarchically clustered nodes, Fig. 2 (b), will gradually decrease while the packets propagate through the cluster heads, where the time delay of each cluster depends on the number of the forwarding nodes in the cluster within a chain of domains. Briefly, we can estimate the intermediate delays for the cluster heads a, b, c, d, and T by

$$T = \sum_{i=1}^{S}\frac{(I_i - D_i)}{\mu_i - \lambda_i} + \sum_{k=1}^{m}\frac{O_k}{\mu_k - \lambda_k}. \tag{35}$$

Here, I_i and O_k stand for the number of packets in the input and output packet streams, respectively. Hence, the first term of the summation represents the process time for dropped and locally delivered packets (D_i), whereas the second term represents the delay for forwarding the remaining packets to the adjacent nodes (here mostly the cluster heads of the SOCRA structure). λ_k denotes the delivery rate of packets to outbuffer and μ_k denotes the transmission rate from outbuffer to outgoing channels.

Every incoming packet is prone to fading and multipath delay problems, which can be modeled as a Rayleigh distribution in order to add the fading effect. The multipath delay phenomenon (so called delay spread) causes intersymbol interference, [49], which in some cases leads to large amount of packet drops. In the simulation, we used root mean square (RMS) delay to denote the overall multipath delay of a given packet. The power of the received signal is modified by multipath fading and shadowing effects, which were also incorporated into the simulation process. When starting up a node or when the node is awakened from the idle state, it measures the average link lifetime with the adjacent nodes. If the predefined duration of the lifetime of a node is expired then the node will be deleted from the routing table. In order to obtain the link lifetime and average delays to neighbor nodes, the active node measures the delay by the following algorithm.

$$D_A = \frac{1}{N}\sum_{i=1}^{N}\sum_{j}^{B}\frac{b_{<i,j>}}{j}, \tag{36}$$

where, N denotes the number of neighbors, B number of packet bursts, and $b_{<i,j>}$ denotes the delay of the burst packet i to the neighbor station j measured as the round trip time (TTL) of packet i. Table 1 shows per node and per branch delays of a tree of multiple branches of an infrastructureless setup each branch containing 8 nodes. Some delay factors are due to the beaconing process, neighbour discovery process, and the determination process for the node lifetime.

We can, by simulation, emphasize the effect of the exponential growth of the number of branches. We limit the number of nodes in each branch while keeping the number of branches as large as possible. Since the number of branches grows recursively to infinity, it is highly possible to obtain unrealistic results. That is, packets emanated from an infrastructureless node will proliferate in the form of a recursive branching process propagating through a random branch of nodes. Since we intend to illustrate the increase in packet delay changing with number of branches (generations) of nodes, it would suffice keeping the number of nodes in each branch at a manageable level. Otherwise, unrealistic results would be obtained, e.g., more packet loss, and very large simulation time (several minutes or hours). For example, each branch of 8 nodes recursively producing another branch of 8 nodes, and more sub-branches iteratively can produce infinitely number of node. As can be noted in Table 1, each node has a small amount of delay at the first generation, while the delay values accumulate (increase exponentially) throughout the next generations.

VII. SIMULATION AND RESULTS

Delay characteristics of a given network can be empirically obtained. However, setting up a network with many (e.g.,

TABLE 1. Simulation results showing per node and per branch delay values (in seconds) of packets traversed through 30 branches each with 8 nodes. A node disseminates a packet to a sub–branch of new 8 nodes.

Branch	node 1	node 2	node 3	node 4	node 5	node 6	node 7	node 8
1	0.9566	0.8866	0.8978	0.9420	0.9813	0.9799	0.9809	1.0487
2	1.3140	1.3047	1.3234	1.3746	1.4577	1.4127	1.4028	1.4420
3	1.5829	1.6162	1.6445	1.7564	1.6537	1.7100	1.7839	1.7259
...	...	...	...	...	...	...	...	...
16	7.0341	7.5769	7.1473	7.9903	8.3530	8.1303	7.9527	8.7653
17	7.5451	7.6183	7.8155	7.8074	8.2964	8.3058	8.7171	9.0428
18	7.1910	7.6415	7.8967	7.7661	8.4218	8.6544	8.8271	9.4355
...	...	...	...	...	...	...	...	...
22	9.1292	8.9783	0.2748	10.6556	10.5194	10.8433	12.6632	13.4731
23	9.6634	9.6920	9.5673	10.5283	11.2848	11.9872	11.7693	13.1254
24	9.2377	9.6128	10.1389	10.6889	10.8195	11.3391	12.1400	13.3478
...	...	...	...	...	...	...	...	...
28	12.1400	13.3478	13.6711	16.0290	18.5920	19.6238	24.1810	35.7512
29	11.7693	13.1254	14.5953	15.5365	16.8460	20.1248	24.3144	34.8973
30	13.4731	12.6632	13.9471	15.4682	17.0762	19.9540	24.6699	36.1589
...	...	...	...	...	...	...	...	...

100) wireless nodes in a laboratory environment is often practically infeasible. Therefore, we have collected some data from a small set of nodes, 8 access points and 24 client nodes communicating with the wireless access points, and used the data to justify the simulation results. A large number of packets have been observed from all clients in the experiment and flooded to the sample routers, and respective delays were measured. For the SOCRA topology, 3 clusters each with 8 client nodes have been experimented. Results obtained from the experiments are highly close to that of shown in Fig. 5 (d). Python/scapy packet generator, TCPDUMP packet capture, and Nmap/nping have been used for the experiments. Furthermore, in order to substantiate the theoretical behaviour of the packet queueing a MATLAB simulation has been developed for infrastructureless flooding and SOCRA scheme with varying hierarchy of clusters of nodes. Due to reduced number of hops in SOCRA, clustering scheme has shown a performance of approximately four times higher than that of the infrastructureless flooding. However, on-demand clustering and the authentication of the SOCRA nodes introduced additional delays, which are constant and generally insignificant.

Designing a precise simulation of wireless ad-hoc networks can be extremely complicated. It is, therefore, required to keep the number of parameters as concise as possible in order to avoid the extreme complexity. Thus, in order to highlight this issue more, it is convenient to summarize the major parameters used in the simulation. Unless otherwise specified, we use generic units for all parameter values. For example, we prefer using generic "time units", since simulations are generally performed independent of system specifications. Hence, saying 5 ms does not make much sense for a Cray computer, since it runs much faster in the nano-second time scale compared to a system with limited resources. A wireless network packet has a maximum packet size of 2312 bytes. Thus, a random number within this range has been chosen for packet sizes. The simulation parameters shown in Table 2 are self explanatory. As expressed by Eq. (19), (20), and (21), input packets were generated by a Poisson process. SOCRA clusters and infrastructureless topology were also constructed using the Poisson process. The CPU time, as described by Eq. (22) uses random values for segregation time (S_i), for buffering time (B_i), and for the packet length (θ). Per path (from source to destination) packet drops as well as the per node packet drops are conditioned on per node delay given by CPU time plus a platform dependent random value $random(Gamma)$. We modeled platform dependent system capability as a Gamma-distributed parameter.

TABLE 2. List of the parameters used in the simulation.

Parameter	Description
$m = Poisson(\lambda)$	Number of packets
$\theta = random(normal) * 2312$	Packet size
$C_t = random(normal)$	Cluster building time
$A_t = random(normal)$	Authentication time
$P_t = random(normal)$	Path optimization time
$CPU_v = \sum_{i=1}^{n} \theta * (S_i + B_i)$	Per node CPU time
$CPU_p = \sum_{j=1}^{N} CPU_v$	Per path CPU time
$R_v = COND(D_v)$	Per node drop
$R_p = \sum_{i=1}^{N} R_v$	Per path drop
$D_v = random(Gamma) + CPU_v$	Per node delay
$D_p = \sum_{i=1}^{n} D_v$	Per path delay
$B_r = random(exponential)$	branching rate
$PLT = TTL + CPU_v$	packet life time
$E_v = \theta * CPU_v * n$	Per node energy usage
$E_p \sum_{i=1}^{N} E_v$	Per path energy usage
C_v = NULL	Per node congestion
C_p = NULL	Per path congestion

The network setup used under the simulation considered equal number of nodes and the same communication scenario both for the infrastructureless and SOCRA schemes. Fig. 4 shows delays at different branches, which engender through several levels of offspring (branch). We can see that the propagation delay for the infrastructureless branches is higher than that of the SOCRA scheme. This is due to the fact that SOCRA has fewer number of hops in each branches

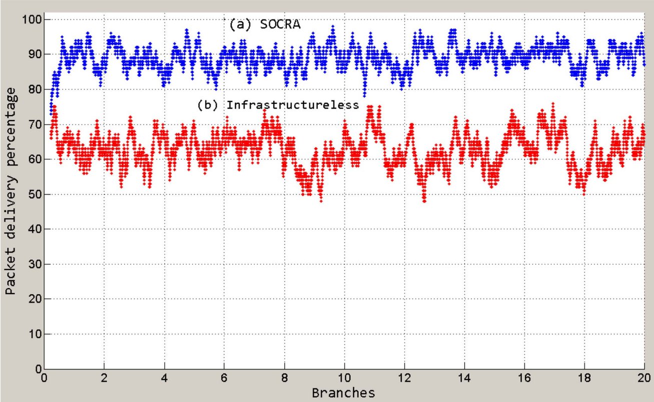

FIGURE 4. Packet delivery percentages for mobile nodes through randomly selected paths of 20 different branches. Average delivery rate for a SOCRA node is 95% and 67% for an infrastructureless node.

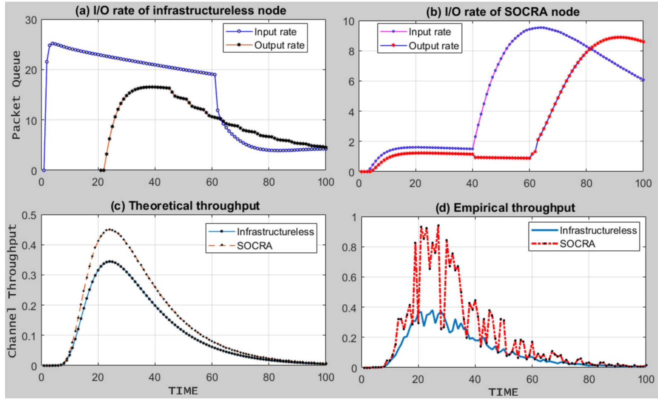

FIGURE 5. Illustration of buffer processing delays and network throughput: (a) load at an infrastructureless node, (b) load at a SOCRA node, where loads are shown on the y-axis as a multiple of 100.000 each and the delay values are shown on the x-axis as generic time units. (c) expected theoretical throughput factor, and (d) empirical throughput factor, normalized to 1.

compared to the infrastructureless branches, which disseminates packets destined to a given target through all reachable nodes in each branch.

We have created 1000 randomly scattered nodes arranged in 20 branches with equal number of nodes, where each node having the same capabilities (i.e., equal CPU usage and packet processing rate) both for the SOCRA and the Infrastructureless nodes, partly shown in Fig. 6. For the test, we have chosen a target in branch 20 (furthest offspring). Here, we tried to emphasize the percentage of packets delivered to the destination. Due to the selective route decision of the SOCRA protocol, the throughput of the infrastructureless path is less than a SOCRA path, and also the overall time delay of the infrastructureless path is accordingly greater than that of the SOCRA path. The results are plotted in Fig. 4. Cumulative time delays are shown in Fig. 7.

Based on the simulation results, we observe that the throughput of a flooded (infrastructureless) node is almost as low as half of the SOCRA node, Fig. 5. This is clearly due to the intensive CPU usage for processing incoming and outgoing packets through the Infrastructureless nodes, which consequently leads to larger packet delays and higher loads throughout the intermediate nodes as well. On the contrary, the SOCRA nodes show small delays between the incoming and outgoing packet streams. Hence, both the input and output buffers of the flooded nodes grow generally more than double of the SOCRA node.

As can be noted from the above analysis, buffering and forwarding of the incoming packets introduce time delays affecting the overall throughput of the traffic passing via a node. Fig. 5 (a) and (b) show the buffer load factors and delays of the two schemes used in the simulation, where the process utilization (load) at the ad-hoc router infrastructureless node is higher than that of the SOCRA routing. That is, due to the selective routing, a SOCRA node delivers higher packet throughput, where the transmission speed picks instantly during some intervals, Fig. 5 (d).

It is obvious from the simulation results shown in Fig. 5 (a), the flooding node starts flooding off the output queue with high latency. This is due to the overwhelming input traffic, which makes the flooding node to work intensively. Thus, regarding the entire network, the cumulative delay across the routing path becomes extremely high, consequently more packet drops and severe energy consumption are inevitable at the intermediate nodes. Fig. 5 (c) and (d) show the theoretical and empirical packet flow rate of the two routing schemes used in the simulation.

Algorithm 1 shows the pseudo-code for the throughput simulation. A sample routing topology with varying number of branches (10, 20, 40, and 100) has been configured under two different setups, flooding and SOCRA. Furthermore, 10, 20, 60, 100, 200, and 1000 nodes have ben created under each branch. A computer with 64-bit Windows 7 OS using Intel Core i7-3630QM CPU of 2.4GHz clock frequency has been used for the simulation. It has been observed that the simulation with 100 branches and 100 nodes under each branch (i.e., 10.000 nodes in total) could not be successfully performed, instead we ought to reduce the number of branches to a more realistic value, e.g., 20. Most routing algorithms (e.g., AODV) designed for infrastructureless networks cannot handle large networks with hundreds or thousands of nodes. In most cases the routing tables, when limit exceeds, start overflowing, [50]. Therefore, it is more realistic to keep the total number of nodes below 1000 or even less. This, however dos not matter for SOCRA, since it has a scalable selective routing scheme that dedicates routing tasks to a small set of nodes (IMGs) among all other nodes.

Simulation results are justified by applying the law of large numbers to the characteristic functions of the throughput expressions presented in the previous section. Since the product of characteristic functions of independent random variables is equal to their sum, it can be used to justify the solution related to the problems involving independent random variables such as the ones we used here in Binomial, Poisson, and branching processes.

Algorithm 1: Throughput Simulation

```
Algorithm singleHost(packetCount, outChannels)
for currentPacket ← 1 to packetCount do
    pSize ← Rand(normalRND() ∗ 2312) ;
    currentDelay ← Rand(gammaRND() ∗ pSize) ;
    if currentDelay ≥ dropDelay then
        Increment(perHostDrop) ;
        Drop(currentPacket) ;
    else
        Forward(currentPacket, outchannels) ;
        Increment(perHostPass) ;
    end if
    updateDelay(perHostDelay, currentDelay) ;
end for
return (perHostDelay, perHostDrop, perHostPass) ;

Algorithm buildSOCRA(maxClusters, r, inChannels)
// Generate cluster heads according to radius r;
N ← Rand(poissonRND(maxClusters ∗ (1 + 2 ∗ r))) ;
// Generate cluster coordinates according to radius r;
C ← Rand(N,2) ∗(1 + 2 ∗ r) − r ;
i ← 0 ;
while i ≤ maxNodes do
    sPower ← Rand(normalRND()) ; //Signal strength
    nodeLocation ← 4 ∗ r∗Rand(1,2) −r;
    Node ← PowerToEuclid(sPower, nodeLocation, C) ;
    // Compute signal power and convert to Euclidean distance:
    if Coordinates (Node ≤ r) then
        C[i] ← Node; //Node is within the coverage of this cluster,
        // assign it to this cluster;
    end if
    i ← i + 1;
    singleHost (C[i]);
end while
function
doCompute(isSOCRA, inChannels, packetCount, outChannels)
for i ← 1 to inChannels do
    if isSOCRA = true then
        for cluster ← 1 to N do
            singleHost(packetCount, outChannels);
        end for
    else
        for neighbor ← 1 to Length(outChannels) do
            Flood(packetCount, neighbor);
        end for
    end if
end for
return;
```

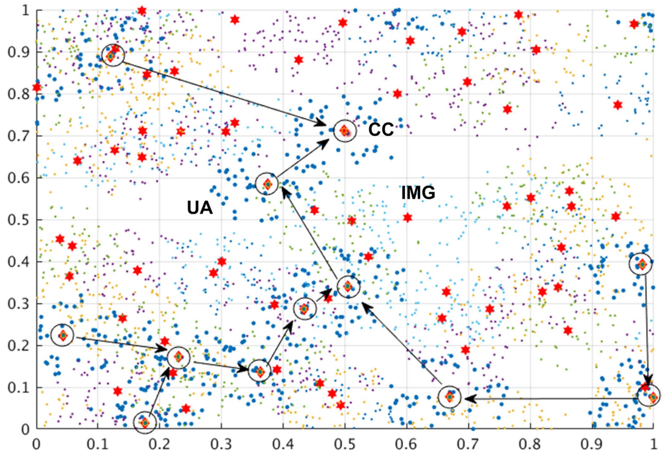

FIGURE 6. Distribution of nodes in a unity coordinate system. Here, the elements of the topology are used for both the SOCRA and infrastructureless modes. The arrows, represent a typical set of authenticated intermediate routing elements of SOCRA working for a given path to a destination (CC). Infrastructureless nodes do only identify and set a nearest path during the beaconing process. Therefore, we do not show any active routing path for the infrastructureless nodes.

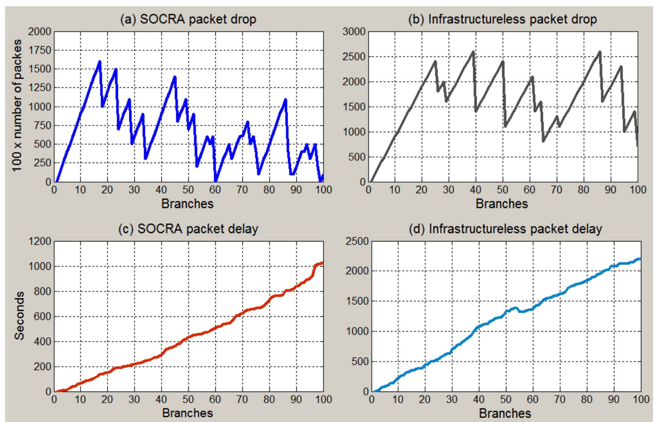

FIGURE 7. Per branch packet drops and cumulative delay for the simulation of topology shown in Fig. 6. Here, (a) and (b) show the packet drops per branch, and (c) and (d) show the cumulative packet delays for the overall branches. Dropped packet delay is not included in the total delay.

It can be clearly deduced from Fig. 2 that the packet traffic emanating from multiple source nodes will be flooded through the nodes a to T (the target node). Therefore, the overall delay for these nodes will gradually increase leading to higher packet loss. We can observe from Fig. 5 and the upper part (above the dashed line) of Fig. 4 that the delay accumulate mostly via each branch while forwarding the packets to the next branch. The propagation delay through the hierarchically clustered nodes (i.e., cluster heads) of SOCRA is much lower, see the lower part of Fig. 4. Here the delay of each cluster depends on the size of the active hops in the respective cluster. In other words, we have the intermediate delays for the cluster heads estimated by

$$T_x = \binom{N}{k}\left(\frac{1}{\mu - \lambda}\right), \qquad (37)$$

where k packets out of N incoming packets are being forwarded with each packet having a delay factor of $\frac{1}{\mu-\lambda}$, Eq. (26).

It was shown that increasing the number of nodes up to 10.000 can be managed in the SOCRA topology, while delay and packet drops become unmanageable with the infrastructureless topology, Fig. 6.

It is also interesting to observe packet drop rates of individual branches and the cumulative time delays from source to destination when packets propagate through all branches. To do so, a simulation with 100 branches each having random number of nodes and packets was performed. As shown in Fig. 7 (a) and (b), packet drop rates increase linearly during the first branches for both schemes. This is, naturally, due to the initial route discovery process that is required by any routing scheme. When a packet dropped its current processing time spent is not included in the computation of the cumulative delay. This means that packets are initially broadcast to a large number of nodes, thereafter, the number of forwarded packets changes randomly depending on the path length of the individual branch. As shown in Fig. 7 (a), the packet drop rates of the SOCRA branches gradually decrease towards the target node. This is due to the selective route decision of the SOCRA protocol. On the contrary, each node of the Infrastructureless network in any branch floods all incoming packets to all outgoing channels. It can be deduced from the simulation results that having huge number of nodes in each branch for the infrastructureless mode can give unacceptably high rate of packet delays.

VIII. CONCLUSION

This paper has considered throughput analysis of ad-hoc routing for infrastructureless networks in a comparative manner. Assessment of the traffic efficiency has been adequately modelled as a stochastic queueing system. The model has been used to simulate various scenarios of both the ad-hoc and SOCRA routing schemes. To enable the analysis a reference model (sink-oriented collaborative routing algorithm) has been applied for guiding the comparative assessment process used here.

The rapid growth of the Internet containing a vast number of ad–hoc devices emerging dynamically each day makes the Internet routing unsustainable. Researchers are intensively investigating to improve the efficiency of greedy routing that currently dominate the holistic space of the Internet routing techniques. The application of efficient routing algorithms has an important implication on dynamically grown networks with higher mobility.

Time and energy efficiency for infrastructureless routing has an important role for critical operations. For example, data transmission from hard-to-access locations, real-time monitoring of various events, e.g., monitoring of power plants, critical sensor networks, tracking of criminal activities, network security reconnaissance, and anti-terror operations in crowded areas require time– and energy–efficient operations without confronting unresolvable obstacles and severe risks.

We have also introduced some features of the hypothetical routing algorithm (SOCRA), which aims at developing collaborative networks comprised of hierarchically clustered nodes used for critical data exchange operations. Based on the analysis and simulations considered here, such a scheme can perform reliable data transmission across physically hard-to-access environments faster than the store-and-flood routing protocols used in infrastructureless networks. A routing algorithm with similar capabilities to that of SOCRA is believed to be essential for constructing collaborative network topologies that can provide higher mobility, improved power sustainability, and faster data exchange for infrastructureless devices.

REFERENCES

[1] G. Aggelou, *Mobile Ad Hoc Networks: From Wireless LANs to 4G Networks*, 1st ed. New York, NY, USA: McGraw-Hill Prof., 2005.

[2] S. Shankar, G. Varaprasad, and S. H. Narayanagowda, "Implementing a new power aware routing algorithm based on existing dynamic source routing protocol for mobile ad hoc networks," *IET Netw.*, vol. 3, no. 2, pp. 137–142, Jun. 2014.

[3] S. Alaparthi, S. R. Parvataneni, C. S. Vaishnavi, P. Sathvika, M. Chandrika, and P. Sharanya, "Dynamic source routing protocol—A comparative analysis with AODV and dymo in ZigBee based wireless personal area network," in *Proc. 6th Int. Conf. Signal Process. Integr. Netw. (SPIN)*, Noida, India, 2019, pp. 1042–1046.

[4] Z. Ma *et al.*, "VaSe-MRP: Velocity-aware and stability-estimation-based multi-path routing protocol in flying ad hoc network," *Int. J. Distrib. Sens. Netw.*, vol. 15, no. 11, 2019, Art. no. 1550147719883128.

[5] H. Choi and J. Lee, "Local flooding-based on-demand routing protocol for mobile ad hoc networks," *IEEE Access*, vol. 7, pp. 85937–85948, 2019.

[6] D. Gross, J. F. Shortle, J. M. Thompson, and C. M. Harris, *Fundamentals of Queueing Theory, 4th Edition*, 4th ed. New York, NY, USA: Wiley, 2013.

[7] W. J. Stewart, *Probability, Markov Chains, Queues, and Simulation*, 1st ed. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[8] M. Boguñá, F. Papadopoulos, and D. Krioukov, "Sustaining the Internet with hyperbolic mapping," *Nat. Commun.*, vol. 1, no. 1, p. 62, 2010.

[9] A. Muscoloni and C. V. Cannistraci, "Navigability evaluation of complex networks by greedy routing efficiency," *Proc. Nat. Acad. Sci.*, vol. 116, no. 5, pp. 1468–1469, 2019. [Online]. Available: https://www.pnas.org/content/116/5/1468

[10] O. S. Oubbati, N. Chaib, A. Lakas, P. Lorenz, and A. Rachedi, "UAV-assisted supporting services connectivity in urban VANETs," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3944–3951, Apr. 2019.

[11] M. Abdollahi, M. Abolhasan, N. Shariati, J. Lipman, A. Jamalipour, and W. Ni, "A routing protocol for SDN-based multi-hop D2D communications," in *Proc. 16th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, Jan 2019, pp. 1–4.

[12] N. Chaib, O. S. Oubbati, M. L. Bensaad, A. Lakas, P. Lorenz, and A. Jamalipour, "BRT: Bus-based routing technique in urban vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, early access, Sep. 12, 2019, doi: 10.1109/TITS.2019.2938871.

[13] G. Sun, Y. Zhang, H. Yu, X. Du, and M. Guizani, "Intersection fog-based distributed routing for V2V communication in urban vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2409–2426, Jun. 2020.

[14] W. Silva and D. Sadok, "A survey on efforts to evolve the control plane of inter-domain routing," *Information (Switzerland)*, vol. 9, no. 5, p. 125, 2018.

[15] N. A. M. Saudi, M. A. Arshad, A. G. Buja, A. F. A. Fadzil, and R. M. Saidi, "Mobile ad-hoc network (MANET) routing protocols: A performance assessment," in *Proc. 3rd Int. Conf. Comput. Math. Stat. (iCMS2017)*, 2019, pp. 53–59.

[16] G. Sallam and A. Mahmoud, "Performance evaluation of OLSR and AODV in VANET cloud computing using fading model with sumo and NS3," in *Proc. Int. Conf. Cloud Comput. (ICCC)*, Riyadh, Saudi Arabia, Apr. 2015, pp. 1–5.

[17] Y.-B. Ko and N. H. Vaidya, "Flooding-based geocasting protocols for mobile ad hoc networks," *Mobile Netw. Appl.*, vol. 7, no. 6, pp. 471–480, Dec. 2002. [Online]. Available: https://doi.org/10.1023/A:1020712802004

[18] I. C. Avramopoulos and H. Kobayashi, "Guaranteed delivery flooding protocols for mobile ad hoc networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (IEEE Cat. No.04TH8733)*, vol. 3. Atlanta, GA, USA, Mar. 2004, pp. 1672–1678.

[19] D. Rupprecht, K. Kohls, T. Holz, and C. Pöpper, "Breaking lte on layer two," in *Proc. IEEE Symp. Security Privacy (SP)*, San Francisco, CA, USA, 2019, pp. 1121–1136.

[20] G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro, "LTE for vehicular networking: A survey," *IEEE Commun. Mag.*, vol. 51, no. 5, pp. 148–157, May 2013.

[21] W. Dargie and C. Poellabauer, *Fundamentals of Wireless Sensor Networks: Theory and Practice*. Hoboken, NJ, USA: Wiley, Jan. 2011.

[22] D. Sharma and A. P. Bhondekar, "Traffic and energy aware routing for heterogeneous wireless sensor networks," *IEEE Commun. Lett.*, vol. 22, no. 8, pp. 1608–1611, Aug. 2018.

[23] A. Boukerche, "Performance evaluation of routing protocols for ad hoc wireless networks," *Mobile Netw. Appl.*, vol. 9, no. 4, pp. 333–342, Aug. 2004.

[24] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 551–591, 2nd Quart., 2013.

[25] S. Yu, B. Zhang, C. Li, and H. T. Mouftah, "Routing protocols for wireless sensor networks with mobile sinks: A survey," *IEEE Commun. Mag.*, vol. 52, no. 7, pp. 150–157, Jul. 2014.

[26] M. Tareq, R. Alsaqour, M. Abdelhaq, and M. Uddin, "Mobile ad hoc network energy cost algorithm based on artificial bee colony," *Wireless Commun. Mobile Comput.*, vol. 2017, pp. 1–14, Aug. 2017.

[27] M. Sinaie, A. Zappone, E. A. Jorswieck, and P. Azmi, "A novel power consumption model for effective energy efficiency in wireless networks," *IEEE Wireless Commun. Lett.*, vol. 5, no. 2, pp. 152–155, Apr. 2016.

[28] C. Buratti and R. Verdone, "Joint scheduling and routing with power control for centralized wireless sensor networks," *Wireless Netw.*, vol. 24, no. 5, pp. 1699–1714, 2018.

[29] H. Dai, R. C. Wong, and H. Wang, "On capacity and delay of multichannel wireless networks with infrastructure support," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 1589–1604, Feb. 2017.

[30] A. Boukerche, B. Turgut, N. Aydin, M. Z. Ahmad, L. Bölöni, and D. Turgut, "Routing protocols in ad hoc networks: A survey," *Comput. Netw.*, vol. 55, no. 13, pp. 3032–3080, 2011.

[31] F. Cadger, K. Curran, J. Santos, and S. Moffett, "A survey of geographical routing in wireless ad-hoc networks," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 621–653, 2nd Quart., 2013.

[32] D. L. Guidoni, F. S. H. Souza, J. Ueyama, and L. A. Villas, "RouT: A routing protocol based on topologies for heterogeneous wireless sensor networks," *IEEE Latin Amer. Trans.*, vol. 12, no. 4, pp. 812–817, Jun. 2014.

[33] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Netw.*, vol. 3, no. 3, pp. 325–349, 2005.

[34] R. Oliveira, C. Montez, A. Boukerche, and M. S. Wangham, "Reliable data dissemination protocol for VANET traffic safety applications," *Ad Hoc Netw.*, vol. 63, pp. 30–44, Aug. 2017.

[35] A. Biswas, B. Saha, and S. Guha, "Performance analysis of AODV and DSR routing protocols for ad-hoc networks," in *Advanced Computing, Networking and Security* (Lecture Notes in Computer Science), vol. 7135, P. S. Thilagam, A. R. Pais, K. Chandrasekaran, and N. Balakrishnan, Eds. Berlin, Germany: Springer, 2012, pp. 297–305.

[36] Z. Guan, B. Zhang, Y. Zhang, S. Zhang, and F. Wang, "Delaunay triangulation based localization scheme," in *Proc. 29th Chin. Control Decis. Conf. (CCDC)*, Chongqing, China, May 2017, pp. 2627–2631.

[37] W. Li, T. Li, and Z. Ge, "A delaunay triangulation based method for optimizing backbone wireless mesh networks," in *Proc. Int. Conf. Comput. Sci. Services Syst. (CSSS)*, Nanjing, China, Jun. 2011, pp. 959–962.

[38] Y. Qu and Q. Tian, "Multi-UAV cooperative positioning based on delaunay triangulation," in *Proc. Int. Conf. Comput. Aspects Soc. Netw.*, Taiyuan, China, Sep. 2010, pp. 401–404.

[39] K. N. Qureshi, A. H. Abdullah, and J. Lloret, "Road perception based geographical routing protocol for vehicular ad hoc networks," *Int. J. Distrib. Sens. Netw.*, vol. 12, no. 2, 2016, Art. no. 2617480.

[40] A. A. Ateya, A. Muthanna, I. Gudkova, Y. Gaidamaka, and A. D. Algarni, "Latency and energy-efficient multi-hop routing protocol for unmanned aerial vehicle networks," *Int. J. Distrib. Sens. Netw.*, vol. 15, no. 8, 2019, Art. no. 1550147719866392.

[41] J. Marchang, R. Douglas, B. Ghita, D. Lancaster, and B. Sanders, "Dynamic neighbour aware power-controlled MAC for multi-hop adhoc networks," *Ad Hoc Netw.*, vols. 75–76, pp. 119–134, Jun. 2018.

[42] H. Zeng, F. Tian, Y. T. Hou, W. Lou, and S. F. Midkiff, "Interference alignment for multihop wireless networks: Challenges and research directions," *IEEE Netw.*, vol. 30, no. 2, pp. 74–80, Mar./Apr. 2016.

[43] M. Tong, Y. Chen, F. Chen, X. Wu, and G. Shou, "An energy-efficient multipath routing algorithm based on ant colony optimization for wireless sensor networks," *Int. J. Distrib. Sens. Netw.*, vol. 11, no. 6, 2015, Art. no. 642189.

[44] C. Yao, L. Zhu, Y. Jia, and L. Wang, "Demand-aware traffic cooperation for self-organizing cognitive small-cell networks," *Int. J. Distrib. Sens. Netw.*, vol. 15, no. 1, 2019, Art. no. 1550147718817289.

[45] P. Chatterjee, S. C. Ghosh, and N. Das, "Load balanced coverage with graded node deployment in wireless sensor networks," *IEEE Trans. Multi-Scale Comput. Syst.*, vol. 3, no. 2, pp. 100–112, Apr.–Jun. 2017.

[46] N. Abramson, "Development of the ALOHANET," *IEEE Trans. Inf. Theory*, vol. 31, no. 2, pp. 119–123, Mar. 1985. [Online]. Available: https://doi.org/10.1109/TIT.1985.1057021

[47] K. S. Trivedi, *Probability & Statistics With Reliability Queuing and Computer Science Applications 2nd Edition*. New York, NY, USA: Wiley, 2001.

[48] S. Kondakci, "Analysis of information security reliability: A tutorial," *Rel. Eng. Syst. Safety*, vol. 133, pp. 275–299, Jan. 2015.

[49] H. Ma, N. Shehadeh, and J. Vanelli, "Effects of intersymbol interference on a Rayleigh fast-fading channel," *IEEE Trans. Commun.*, vol. 28, no. 1, pp. 128–131, Jan. 1980.

[50] USENIX Network Working Group and M. Musuvathi. (Aug. 2002). *Description of AODV Protocol*. [Online]. Available: https://www.usenix.org/legacy/publications/

SULEYMAN KONDAKCI (Member, IEEE) received the EE Engineering degree from the University of Gazi, Turkey, and the Candidatus Magisterii (B.S.), Candidatus Scientiarum (M.S.), and Ph.D. degrees in computer science from the University of Oslo, Norway.

He was a Professor with the Faculty of Engineering and Computer Sciences, IEU. He has been involved in a number of international projects prior to his academic career, where he has made several contributions to the NATO Information Security Framework. He is the Founder of Information Security Test and Evaluation Laboratory, Scientific and Technological Research Council of Turkey (TUBITAK-UAKEA). His former research area and publications were mostly focused on the theoretical network security and risk assessment. His current interests are stochastic system modeling, ubiquitous computing, and security analyses of critical systems and infrastructures.