Overcoming Adversaries in Sensor Networks: A Survey of Theoretical Models and Algorithmic Approaches for Tolerating Malicious Interference

Maxwell Young and Raouf Boutaba

***Abstract*—Interference is an unavoidable property of the wireless communication medium and, in sensor networks, such interference is exacerbated due to the energy-starved nature of the network devices themselves. In the presence of antagonistic interference, reliable communication in sensor networks becomes an extremely challenging problem that, in recent years, has attracted significant attention from the research community.**

This survey presents the current state of affairs in the formulation of theoretical models for adversarial interference in sensor networks and the different algorithmic remedies developed by the research community. There is a particular focus on jamming adversaries and Byzantine faults as these capture a wide range of benign faults as well as malicious attacks. The models in the literature are examined and contrasted with the aim of discerning the underlying assumptions that dictate analytical bounds with regards to feasibility and a number of performance metrics such as communication complexity, latency, and energy efficiency. Limitations are also highlighted with a focus on how various results impact real world applications and, conversely, how the current sensor network technology informs newer models. Finally, directions for future research are discussed.

***Index Terms*—Sensor networks, security, adversary, Byzantine fault, jamming, reliable broadcast, algorithm/protocol design and analysis, fault tolerance.**

I. Introduction

IN ADDITION to traditional network security challenges, the shared communication medium of sensor networks renders them vulnerable to a variety of malicious attacks [1]. A determined attacker may engage in any number of activities aimed at disrupting communication such as rerouting, message injection, wormhole attacks, black hole attacks, and others (see [2], [3] and references therein). This challenging state of affairs is further compounded by the strict energy constraints placed on the network devices themselves which are typically battery powered. Consequently, many of the standard cryptographic techniques for thwarting such attacks in the wired domain cannot be employed in sensor networks. Moreover, cryptography is ineffective against an attacker that simply jams the communication medium in order to disrupt all communications within range.

Manuscript received 29 November 2010; revised 18 March 2011.

The authors are with the David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada, N2L 3G1 (e-mail: {m22young,rboutaba}@cs.uwaterloo.ca. R. Boutaba is also affiliated with the Division of IT Convergence Engineering, POSTECH, Republic of South Korea.

Digital Object Identifier 10.1109/SURV.2011.041311.00156

Communication itself comes at a hefty price. Wireless network cards generally offer states such as *off, sleep, receive* and *transmit*. While the sleep state requires negligible power, the cost of the transmit and receive states are roughly equivalent. For example, the transmit and receive costs of the Telos motes are 38mW and 35mW, respectively, while the sleep state cost is 15μW [4]. Therefore, the cost of the transmit/receive state exceeds that of the sleep state by a factor greater than 2000. Similar relationships hold for the MICAz and MICA2 motes [5], [6]. For many potential applications, sensor networks will be deployed in hard-to-reach areas or along dangerous terrain (see [7], [8] and references therein), and once a device has exhausted its energy supply, it is permanently inactive. Therefore, maximizing the lifetime of a device is a critical goal.

Unfortunately, this goal is often at odds with the redundancy needed to overcome communication interference caused by an attacker. The research community is aware of this dilemma and, in recent years, it has been the focus of a number of efforts. In this survey paper, we take a close look at many results which focus on abstract models of adversarial interference and whose aim is typically to delineate fundamental boundaries on what is feasible algorithmically under such models. We elaborate on the scope of this work below.

A. The Scope of this Survey

There exist several survey papers on wireless sensor networks [8]–[10] and, more specifically, on security [1]–[3], [11]–[14]. This existing literature presents a valuable categorization of attacks and practical security considerations. Framing results from the domain of sensor networks is difficult, not only due to the voluminous amount of work, but also because of the differing network models that have been adopted by the research community as surveyed in [15], [16] and briefly highlighted in [17]. This issue is treated by both Raman and Chebrolu [18] and Chockler *et al*. [19] who provide an important discussion on the incongruity between the domains of theory and practice. In the context of adversarial attacks, this schism is particularly evident and we believe that a number of results belonging to the former domain are conspicuously absent from existing survey works.

Here, we address this omission by surveying the literature on adversarial fault tolerance with an eye towards results with a theoretical flavor. Specifically, we examine models of adversarial interference where at least one of the following

is present: (1) jamming attacks (also known as denial-of-service attacks), (2) Byzantine faults and adversarially chosen crash failures, and (3) spoofing or message injection attacks. Our overall aim is three-fold. First, we report on the adversarial models present in the literature with a focus on structuring them according to several underlying features and assumptions. This allows for a sensible comparison of models that are conceptually similar and delineate trade-offs between models that differ. Second, given this framework, we present an overview of the upper and lower bound results that accompany these models with respect to adversarial fault tolerance, message and bit complexity, communication latency and energy efficiency. In this sense, our focus is on models that admit provable bounds, in contrast to techniques aimed at mitigating adversarial behavior in practice. Third, as the area of adversarial fault tolerance in sensor networks is relatively new and still the subject of intense interest, we illustrate the progression of results and highlight directions for future work.

Admittedly, the application of our criteria for examining a work is not always clear and the boundaries of our investigation may blur at times. For completeness, this survey incorporates a large number of results outside our focus that are dedicated to various aspects of adversarial fault tolerance. This includes work spanning purely experimental work on the real-world impact of malicious behavior to the taxonomies of adversarial attacks to the practicalities of currently implemented fault-tolerant techniques. We also note that there exists a substantial body of work regarding the application of game theory to this area (see the survey by Manshaei *et al*. [20]); however, here we focus on worst-case adversarial settings where the participants may not act rationally. While these other results are not central to our survey, they are by no means tangential to the area of adversarial fault tolerance in sensor networks, and our work serves a dual purpose as a pointer to in-depth resources on these topics. The bulk of this survey is dedicated to a core subset of papers that we believe constitutes a cohesive body of work and meets the criteria described above, but has not yet received the comprehensive summary it deserves.

A Roadmap: The remainder of this survey is comprised of three main sections. Section II provides the preliminaries for establishing the context in which we examine the core papers of this survey. Here, we address several important aspects of the design space for sensor network models such as assumptions regarding synchronization, collision detection, cryptography, and the different types of adversaries. In the process, we touch on a number of empirical studies and results from applied research fields in the area of mitigating adversarial interference. We also briefly describe several fundamental communication problems that arise in a number of the works we survey and enumerate the important metrics used to compare and contrast related results.

In Section III, we focus on adversaries that aim to disrupt network functionality by jamming the communication channels. Due to the shared medium, such attacks are easily launched and require no special hardware to execute. Consequently, this type of behavior represents a simple denial-of-service (DoS) attack that threatens the availability of sensor networks. We survey papers dedicated to examining the limits of such attacks and a number of possible algorithmic solutions. This is an area which has witnessed a flurry of work over the past several years and, consequently, a number of different models have been proposed and the question of which design aspects are most appropriate remains unresolved. In light of this, one of our aims is to convey the progression of the problems tackled by the research community. We highlight both the virtues and shortcomings of these results and offer a discussion of potential areas for future research.

In Section IV, our attention shifts to Byzantine adversaries and, specifically, the problem of Byzantine agreement in multi-hop sensor networks. In this context, the problem has sometimes been referred to by the lengthy designations of *reliable broadcast tolerating Byzantine adversarial behavior* or *reliable broadcast tolerating Byzantine faults*. Throughout, we will assume the presence of a Byzantine adversary and, as is typical in many works, employ the simpler title of *reliable broadcast*. This is a problem that has received significant interest by the research community over the past half-decade and we note that the results covered in this section are not fully disjoint from those in Section III as jamming-tolerant reliable broadcast is addressed. Again, we attempt to convey the evolution of results and there is a discussion of the different models surveyed as well as an exploration of avenues of future research.

II. Preliminaries

In this survey, we examine several results that, at times, may appear to be disparate. However, underlying all of these results are several common parameters whose attributes define key aspects of the particular model being surveyed. In this section, we discuss these parameters and the common choices made by researchers in laying a realistic framework for their analysis. Later, as we review the literature, we refer back to these parameters in order to make sensible comparisons between results.

A. Communication and Medium Access Control

Devices are assumed to possess a transceiver that operates on a single-channel or multiple-channels; however, typically, transmitting and receiving are assumed not to occur simultaneously. This is in line with current technology such as the MICA2, MICAz, MICA2dot [21] and Telos motes [4], [22]. The models in this survey assume a time division multiple access (TDMA)-like medium access control (MAC) protocol. For example, the well-known LEACH [23] protocol for sensor networks is TDMA-based. Of course, the literature on MAC protocols is vast and we refer the reader to the survey by Kumar *et al*. [24] for an overview of the area. The discretization of time appears to be a common aspect of the theoretical models present in the literature since it facilitates an analysis of various performance metrics. Therefore, such discretization is a common feature of the models we examine in this survey.

In our discussion of multi-hop networks, the notion of a *global* broadcast schedule appears in several works. It seems likely that this is not essential so long as we are willing to suffer an increase in latency and energy expenditure

assuming nodes can hold multiple schedules that are locally synchronized; such is the case for S-MAC [25]. However, the exact implications of such an approach is an open question. In any event, global scheduling (and the resulting advantages in terms of energy efficiency) has been clearly demonstrated by experimental work in [26].

B. Collision Detection

Despite the assumption of a time-slotted network where nodes are assigned non-conflicting slots, deliberate interference may still occur due to adversarial behavior. A message collision occurs when two or more transmissions interfere with each other. In this case, devices that are listening to the medium are typically assumed to either receive none of the messages being transmitted, or receive a message chosen by an adversary. Of course, in practice, when two simultaneous transmissions occur, the contents of the message are not necessarily lost. Indeed, both signals are received and, in some cases, the stronger signal can be discerned; this is known as the *capture-effect* [27]. However, it is often impossible to reliably separate between multiple transmissions due to fading effects.

We point out that in adversarial settings, the utility of collision detection differs from its typical uses in backoff-based protocols for transmitting frames over a wireless channel. Instead, the ability to detect collisions allows correct devices to determine whether an adversary is attempting to thwart communication. As we will see, in many cases this is critical in establishing the successful termination of a communication protocol. The correct modeling of collision detection is an issue of some debate. In practice, channel activity is determined by performing *clear channel assessment* (CCA) [28]. In particular, the detection of channel activity is performed via the radio chip using the *received signal strength indicator* (RSSI) [29]. If the RSSI value is below the clear channel threshold, then the channel is assumed to be clear. In theory, as we will see, there are several approaches to collision detection that are typically adopted. The collision detection model employed is a design choice that significantly impacts what is feasible under a particular adversarial model. Consequently, this aspect is important to consider when placing a result in context with other results in the area. There are several papers that address the challenges of providing an appropriate mathematical model of wireless interference and collision detection; in particular, the reader is referred to work by Kuhn *et al*. [30] and Chockler *et al*. [31].

C. Cryptographic Authentication

Another key dimension along which we may compare models is whether the results rely on any cryptographic assumptions. Standard cryptographic tools are often too costly for the sensor network domain. Moreover, the distribution of shared secret keys poses another host of problems. Consequently, several works have explored the pessimistic scenario where neither message nor sender authentication is possible. On the other hand, several recent results show how *light-weight* cryptographic authentication can be implemented in sensor networks [32]–[35]. These proposed schemes provide high security while being energy efficient. In tandem, the research community has examined how to establish efficient and secure key-distribution schemes; for more on this topic, we refer the reader to the survey by Xiao *et al*. [36]. Therefore, in addition to models that do not assume authentication, we will also see models that leverage the benefits of cryptographic authentication to achieve more efficient operations. For an in-depth examination of the role of cryptography in sensor networks, we refer the reader to the work of Walters *et al*. [1] and the survey by Wang *et al*. [13].

D. Adversaries

In all of the works considered here, there exists an adversary who attempts to disrupt the functionality of the network. The type of malicious behavior differs between models and we delay discussion of certain details to more appropriate sections. However, there are a number of commonalities which we discuss here. There are several ways of interpreting the presence of an adversary. Certainly, this behavior captures the worst-case disruption of transmissions due to non-malicious failures such as software errors or accidental deviations from a global broadcasting schedule. Furthermore, such an adversary also models challenging attacks on the network and these are discussed in this section.

Jamming Adversaries: A jamming adversary embodies one or more devices that attempt to interfere with communications by intentionally causing message collisions or simply flooding the channel with useless information. Various jamming strategies are featured prominently in the literature such as *constant, random, adaptive*, and *reactive jamming* [1], [37], and we use these characterizations in categorizing the results we survey.

Constant jamming is the simplest strategy where the adversary perpetually disrupts the communication medium. Of course, little can be done against an adversary with an unbounded energy supply since such an adversary may broadcast in perpetuity. However, such continuous disruption is easily identifiable and subject to various defensive techniques (see [38], [39] and references therein). Furthermore, constant jamming requires an abundant energy supply and devices, including those that have suffered adversarial faults, are typically battery powered; therefore, a constant jammer will rapidly deplete its energy supply. In cases where the adversary is (1) equipped with a finite budget for causing interference or (2) cannot jam all channels in a multi-channel scenario, we will see results where a constant jammer can be overcome.

Random jamming is a less aggressive and energy-efficient strategy that can allow the adversary to remain concealed; however, by its nature, this type of jamming may not be particularly accurate, especially if the adversary wishes to target specific communications. The history of the network might allow for more effective jamming attacks. For instance, devices might periodically activate in order to disseminate data; at this point, an informed adversary can jam the medium. Such an adversary, one who exploits the history of the network in order to increase the efficacy of its attacks, is called adaptive. Finally, a reactive jamming adversary is one who may detect a transmission and then quickly jam the medium. This is a particularly challenging strategy since it does not necessarily require the adversary to jam aggressively as specific transmissions can be targeted.

Finally, we note that wireless jamming has received significant attention in the context of more applied research [37], [39]–[47]. However, these works fall outside our scope and, instead, we refer the interested reader to the surveys of Wood and Stankovic [38], Karlof and Wagner [2], Wang *et al*. [13] and Pelechrinis *et al*. [48] for an overview of these results.

Byzantine Adversaries: When a network device suffers a Byzantine fault it is assumed to be controlled by an adversary who uses that device to disrupt the network. In sensor networks, a number of attack models exist, but a common theme is the use of message collisions or message corruption by the adversary to thwart the fundamental task of communication. For instance, Byzantine devices may deviate from a broadcast schedule intentionally, acting to jam transmissions as discussed above. Such devices may also attempt to impersonate correct devices; this is generally referred to as *spoofing*. Without some form of authentication, ownership of a transmission over the wireless medium may be claimed by anyone. A similar disruptive technique involves *message injection* where a Byzantine device can change portions of a legitimate transmission. Again, without authentication, this type of interference can go undetected. Unlike the situation for jamming attacks, here the lightweight cryptographic techniques discussed above can help; however, as we will see, authentication is not always accomplished via cryptographic means.

In Section IV, we will observe two main Byzantine fault scenarios. The first involves faults that occur uniformly at random and independently with some constant probability; we call this the *probabilistic scenario*. The second constrains, for each node, the number of neighbors that can suffer a Byzantine fault; we refer to this as the *locally-bounded scenario*. Later on, when we compare and contrast the results from this area, we will categorize the surveyed works using these two scenarios.

Finally, it is important to mention the closely related *Sybil attack* where an adversary assumes control of a large number of identities in order to influence the network [49]. However, in contrast to the Byzantine attacks, Sybil attacks rely on the fact that identities may be cheaply acquired. For instance, while a Byzantine adversary may actually control abundant physical resources such as many sensor network devices, a Sybil adversary might control only a single device and then spoof multiple identities in the network. In this manner, a Sybil adversary may even constitute a majority of identities in the network thus making defense a challenging problem. Surprisingly, despite such a dire attack scenario, there have been several proposals for mitigating Sybil attacks in sensor networks [50]–[53]. For both a taxonomy of attacks and a survey of the results in this area, we refer the reader to the work of Newsome *et al*. [54], and we do not examine Sybil adversaries further.

E. Fundamental Networking Problems

Given an attack model, papers in Section III will typically establish a building block routine for communication between two devices. The utility of this building block is then often demonstrated by generalizing it to networking problems involving n devices. In this way, upper bounds for several fundamental problems are derived. For completeness, we briefly summarize several such problems in order to refer to them later.

Reliable Broadcast: We discuss this problem first as, in contrast to the remaining problems listed here, it is the main subject in Section IV. The problem of reliable broadcast is one of agreement on a particular value. However, here the key difference is the existence of a special *dealer* d (also often termed the *source*) who issues a value v and wishes to have all correct nodes commit to v. More formally, reliable broadcast is achieved if the following properties hold: (1) *Correctness*- no two correct nodes commit to different values, (2) *Completeness* - every node eventually commits to a value and (3) if d is correct, then all correct nodes commit to v. We will come back to this in much greater detail in Section IV.

Consensus: Informally, consensus is the problem of n nodes coming to an agreement on a value v held initially by some node in the presence of faults. The problem has many applications in the domain of fault-tolerant distributed computing such as state machine replication and distributed control systems. Formally, the consensus problem begins with a group G of n nodes each proposing a value; consensus is achieved if three properties hold: (1) *Agreement* - no two correct nodes in G commit to different values, (2) *Validity* - if a node commits to a value v, then v was an initial value of some node in G, and (3) *Termination* - all nodes in G eventually commit to some value. An important restriction on consensus is one where the message $m \in \{0, 1\}$; this is known as *binary consensus*. Note that the ability to perform reliable broadcast allows for binary consensus. By having each node act as the dealer, the initial value of each node can be propagated to all other nodes, after which each node commits to the majority value.

Leader Election: The problem of leader election involves electing a single leader to manage distributed tasks. Formally, prior to executing an election algorithm, each node is in an undecided state. Subsequent to completion of the algorithm a single node is in a leader state while all other nodes are in non-leader states. The election process is typically distributed and it is not known *a priori* whether a node will become the leader.

Gossiping: The gossiping problem generally refers to the dissemination of a value whereby each node passes received information onto its immediate neighbors. There are several formulations of the what exactly constitutes a gossip protocol; however, typically, each node begins with its own value which must be distributed to every other node in the network.

F. Metrics of Interest

The papers that form the core of this survey all establish theoretical bounds on a number of important performance metrics. Throughout this survey, we will highlight the following:

- *Tolerance:* This refers to the maximum number of faults that can be present in the model while still admitting a feasible solution. For example, in Section IV, we will see a grid model where reliable broadcast remains possible

under the locally-bounded scenario so long as the number of faulty neighbors for any node is less than $(r/2)(2r+1)$ where r is the transmission radius.
- *Communication Complexity:* This refers to the number of messages or the number of bits that each node must send under a given protocol and it will always be clear from the context which metric is being used.
- *Latency:* This is a measure of the amount of time required to achieve a certain goal, such as communication between two parties. The models we examine are time-slotted as discussed in Section II-A and typically latency is given as the number of time slots or the number of communication rounds; again, it will be clear from the context which is being used.
- *Energy Efficiency:* This is measured by the amount of time a node must spend either in the transmit or receive state. In some sense, this metric subsumes communication complexity as the amount of time in the transmit state is proportional to the message or bit complexity of a protocol. However, in contrast to the communication complexity, which holds implications for the bandwidth usage in the network, energy-efficiency has bearing on the functional lifetime of the network.

III. Models of Adversarial Jamming

In this first section, we examine several works that model a variety of adversarial jamming scenarios. For each model, we highlight the various decisions made with regards to the parameters of the network discussed in the previous section, as well as the type of jamming adversary. Typically, we provide some in-depth discussion of the main results; however, our aim is to convey the general 'flavor' of the work and the interested reader may then decide to delve into the details of a particular work. Given the nature of the surveyed results, it is impossible to do away with all analysis but we keep the amount of mathematical notation to a minimum.

Gilbert et al. [55]

Consider two players, Alice and Bob, who wish to communicate in the presence of an adversary who can interfere with a bounded, but unknown, number of communications. What is the communication delay introduced by such an adversary? How efficiently can the adversary use its limited budget? These are the questions that motivate the work by Gilbert *et al.* [55].

In a single-hop setting where nodes can detect collisions, the authors derive bounds on (1) the *jamming gain*, which is defined as the amount of energy used to prevent communication relative to the amount of energy used by continuous jamming [44], and (2) *disruption-free complexity*, which measures how long the adversary may disrupt a protocol *without* broadcasting. The adversary is assumed to be reactive and has the ability to spoof both the sender and the receiver; there is no message or identity authentication. A finite budget corresponds to the extent to which the adversary can cause disruption without being discovered. Once discovered, faulty nodes are subject to a variety of remedies [38].

Communication Between Two Parties: The authors first demonstrate lower bounds for communication between Alice and Bob. The adversary is allotted a budget of β; that is, β messages may be spoofed or jammed. The two players Alice and Bob each start off with an initial message m_A and m_B that they convey to the other player, respectively. All possible messages are represented by the set V and it is assumed that V is known to both players and the adversary. The authors show that the adversary may delay communication for at least $2\beta + (\lg |V|)/2$ rounds. This implies a jamming gain of at least 2 for *all* protocols. Furthermore, the disruption-free complexity is at least $\Theta(\lg |V|)$. Therefore, a larger set of possible messages V allows the adversary to obtain a higher disruption-free complexity.

For the upper bounds, the authors focus on the scenario where Alice wishes to convey a message m to Bob. Communication proceeds by alternating between *data rounds* and *veto rounds*. In a data round i, a single bit $b_i \in \{0, 1\}$ of m is transmitted by Alice. If $b_i = 1$, then Alice transmits b_i; otherwise, if $b_i = 0$, then Alice remains silent. Bob is listening in each data round. A veto round is used to confirm whether the transmission of b_i has been successful. For example, the adversary might transmit a 1 when in fact the bit to be sent was a 0 and Alice remained silent. Alice can detect this deception and notify Bob by broadcasting in the veto round. Alternatively, the adversary might broadcast in a veto round hoping to convince Bob that Alice is vetoing the previous transmission. In this case, Alice repeats the procedure for sending b_i since, even though she knows the veto is false, Bob does not. This alternation between data and veto rounds continues until a veto round is silent. At this point, since the adversary cannot forge silence on the channel, both Alice and Bob are convinced that b_i has been successfully communicated. The entire procedure continues for each of the remaining bits of m.

This protocol allows for successful communication of m within $2\beta + \max\{2\,\Delta \cdot |V|^{1/\Delta}, 4\lg |V|\}$ rounds. Therefore, for $\Delta = \Omega(\lg |V|)$, this result yields an upper bound on the jamming gain of 2 and the disruption-free complexity of $\max\{2\Delta \cdot |V|^{1/\Delta}, 4\lg |V|\}$.

Underlying this initial result is the assumption that Alice and Bob initiate the protocol in the same round; that is, synchronization is assumed. Without this assumption, additional challenges arise since the adversary may attempt to thwart attempts at synchronizing communication between the two players. In [56], the authors address this difficult situation and a bound of $2\beta + O(\log |V|)$ rounds is demonstrated.

Extension to Multiple Parties

The results of the scenario involving Alice and Bob generalize to a number of fundamental problems involving n players in a single-hop network. For the problem of reliable broadcast, the dealer takes on the role of Alice sending a message while all other players adopt the role of Bob. The number of rounds till completion is again $2\beta + O(\log |V|)$. This result extends to the binary consensus problem where there are t crashes in the manner described above in Section II-E by having each player perform reliable broadcast. The resulting number

of communication rounds is $2\beta + \Theta(t)$. Finally, the authors describe how the Alice and Bob scenario allows n nodes to elect a leader within $2\beta\frac{c+1}{c}+2c\lg n+2$ communication rounds where $c \geq 1$ is an input parameter to the algorithm.

Awerbuch et al. [57]

Many of the models surveyed in this paper have a parameter that is associated with limitations on the jamming power of the adversary. For instance, we just witnessed in [55] that the value β represented a finite budget which might be interpreted as the duration for which an adversary can behave badly without being detected. In their model, Awerbuch *et al.* [57] constrain the adversary in a different fashion. For any $T \in \mathbb{N}$ and any window of time $w \geq T$, a (T, λ)-bounded adversary is defined as one which can jam at most λw time steps in that window. This type of adversary models so-called *bursty* jamming.

The particular adversary considered here is $(T, 1-\epsilon)$-bounded where $\epsilon > 0$ is an arbtrary constant that is *unknown* to the nodes. This lack of knowledge regarding the jamming duration of the adversary is in line with many other works we survey. However, nodes are assumed to possess very loose estimates of T and the number of nodes n. The adversary is assumed to be adaptive, but non-reactive. That is, the adversary is assumed to know the entire history of the network and may use this information to plan its attacks; however, in any time slot, the adversary makes its decision about whether to jam without any knowledge of actions by the nodes. While there are no Byzantine faults (or any fail-stop faults) in the system, the adversary is assumed to have knowledge of any communication protocol used by the nodes. Collision detection is available to the nodes; however, it is not possible to discern whether the collision is due to malicious interference or due to legitimate transmissions by the nodes.

A Robust MAC Protocol: Here, the MAC protocol coordinates access by the nodes to a single shared channel in a time-slotted network. The authors focus on devising a MAC protocol for a single-hop network that utilizes at least a c-fraction of the unjammed time steps for communication amongst the nodes; such a protocol is called *c-competitive*.

Consider the following simple protocol which is used by the authors to illustrate the motivation behind their work. Let node v send a message in each time step with probability p_v with $p_v \leq \hat{p}$ for a small constant $0 < \hat{p} < 1$ and let $p = \sum_v p_v$. Let an *idle time slot* be a time slot where the channel is unused (and unjammed) and let a *single-message time slot* be a time slot where exactly one node uses the channel to broadcast a message. Let q_0 and q_1 be the probability of an idle time slot and a single-message time slot, respectively. The authors show that $q_0 \cdot p \leq q_1 \leq \frac{q_0}{1-p} \cdot p$. The implication of this result is that if the number of idle time slots is roughly equal to the number of single-message time slots, then $p \approx 1$. Therefore, if a node v observes many more idle time slots than single-message time slots, it can infer that it should increase its probability of sending p_v. Conversely, if v observes many single-message time slots, then it should decrease p_v. For a suitably small $\gamma > 0$, node v's protocol can be formulated as: (1) if an idle time slot is witnessed, then $p_v \leftarrow (1+\gamma)p_v$, (2) if a single-message slot is witnessed, then $p_v \leftarrow p_v/(1+\gamma)$; or (3) otherwise, do nothing (i.e. jammed slots are ignored). A point of concern is that this simple protocol encounters trouble if p is close to 1 as it becomes unlikely that either an idle or single-message time slot will occur. A mechanism is required that allows each node v to reduce p_v and avoid this saturation of the medium.

The key idea for overcoming this challenge is the use of a threshold value T_v. If a single-message time slot is not observed within T_v time slots, then p_v is decreased. However, if T_v is too small, then p may decrease too rapidly resulting in underutilization of the channel. Consider the following rules. If a single-message time slot is witnessed, node v increments T_v. Else, if no single-message time slot is witnessed within T_v time slots, then T_v is decremented. Letting $\gamma = O(1/(\log T + \log\log n))$, the authors prove that these simple rules give rise to a MAC protocol that is constant competitive with high probability if executed for $\Omega(\frac{1}{\epsilon}\log N \max\{T, \frac{1}{\epsilon}(\log^3 N)(\log T + \log\log n)^2\})$ time slots where the constant $\epsilon = \Omega(1/\log^3 N)$ and $N = \max\{T, n\}$.

Communication Complexity and Leader Election: The authors evaluate the number of times a node needs to transmit a message. It turns out that this quantity is rather small at $O(\log^3 N/\gamma^2)$. However, it appears that the amount of listening time is significant as each node v must listen to the channel in order to detect idle or single-message time slots in order to modify p_v. Detecting an idle channel can be done through CCA methods and would likely consume only a negligible amount of energy. However, if the node detects activity on the channel, the transceiver must be activated in order to determine whether the message is successfully being sent (a single-message slot) or a collision has occured. At this point, the node must incur the substantial receive state cost. Therefore, over the execution of the protocol, the listening costs might be substantial.

Finally, the authors illustrate how their MAC protocol can be applied to solve the leader election problem in the presence of a $(T, 1-\epsilon)$-bounded adversary. In particular, they give a leader election protocol that, with high probability, elects a leader within $O(\frac{1}{\epsilon}\log N \max\{T, \frac{1}{\epsilon\gamma^2}\log^3 N\})$ time slots.

Richa et al. [58]

In the immediately previous work by Awerbuch *et al.* [57], we saw that it is possible to obtain good throughput against a jamming adversary in a single-hop network. In this subsequent work, Richa *et al.* [58] look at achieving a similar result for multi-hop networks. Again, the jamming adversary is $(T, 1-\epsilon)$-bounded. The authors also define a k-uniform adversary as follows. Consider partitioning nodes into k sets. A k-uniform adversary is one whose jamming capabilities are such that in each of the k sets, either all or none of the nodes are jammed. For example, a 1-uniform adversary jams either all or none of the nodes in the network, while a n-uniform adversary can decide whether or not to jam each node individually. The same network assumptions are made regarding synchronization and the ability to detect activity on the channel. However, a key difference is that the network under consideration is now a *unit disk graph* (UDG). Under this model, two nodes $u, v \in V$ can communicate if they are within a normalized distance of 1 from each other.

Richa *et al.* provide a symmetric MAC protocol for multi-hop networks called JADE with the following important property. Assume that JADE is run for $\Omega(T \log n/\epsilon + (\log^4 n/(\gamma\epsilon)^2))$ time steps and, again, $\gamma = O(1/(\log T + \log\log n))$. Then JADE is constant competitive with high probability in n so long as (1) the adversary is 1-uniform and the UDG is connected or (2) every node has at least $2/\epsilon$ neighbors. The authors also demonstrate that if (1) the UDG is not connected or (2) the adversary is 2-uniform and there exist nodes with $o(1/\epsilon)$ neighhors, then there are situations where JADE is constant competitive with high probability.

The JADE protocol is somewhat more complicated than its single-hop counterpart in [57] but is still relatively simple to state. Each node v holds a probability value p_v, a threshold value T_v and a counter value c_v. For a positive fixed parameter $\hat{p} \leq 1/24$, each node initiates the protocol with $T_v = 1$, $c_v = 1$ and $p_v = \hat{p}$. In each round, each node v transmits with probability p_v. If v does not transmit, then it acts according to the following cases: (1) if v detects that the channel is idle, then $p_v \leftarrow \min\{(1+\gamma)p_v, \hat{p}\}$ and (2) if v receives a message, then $p_v \leftarrow (1+\gamma)^{-1}p_v$ and $T_v \leftarrow \max\{T_v - 1, 1\}$. At this point, v increments its counter c_v and, if $c_v > T_v$, then v sets $c_v \leftarrow 1$. Finally, if there is no round in the last T_v rounds where (1) v sensed a successful message transmission or (2) detected that the channel was idle, then $p_v \leftarrow (1+\gamma)^{-1}p_v$ and $T_v \leftarrow \min\{T_v + 1, 2^{1/(4\gamma)}\}$.

While much of the JADE protocol is motivated by the same reasoning in the previous section describing [57], the analysis is significantly different. The authors also provide some preliminary simulation results to illustrate the effectiveness of their protocol. Under both uniform and normal distributions for node placement with $n = 500$, JADE demonstrates a normalized throughput of roughly 0.3 and 0.25, respectively. Moreover, in the case of the uniform distribution, the convergence to the constant throughput value is shown to be rapid, achieving the peak throughput within roughly 200 rounds.

A couple final comments are in order. First, as with the protocol of [57], JADE requires nodes to spend a substantial amount of time listening to the channel in order to detect either successful transmissions or idle time slots. An interesting question is whether there exists a constant-competitive multi-hop MAC protocol that is more energy efficient. Second, the focus here is on the UDG model, and it is worth considering whether JADE might be modified for other topologies; indeed, the authors conjecture that such an extension is possible.

Gilbert et al. [59]

To this point, we have reviewed material that dealt with the problem of communication in single-channel networks. However, sensor network devices may employ more than one channel. For example, the transceivers on the MicaZ and Telos motes offer 16 different channels in the 2.4 GHz band [60]. Therefore, it is important to explore fault tolerance in a multi-channel setting.

Gilbert *et al.* [59] address a gossiping problem where nodes are communicating in the presence of an adversary who can simultaneously interfere with t of C available channels in a single hop network. Informally, each node i has an initial value v_i that it wishes to distribute to all other nodes and the goal is for each node to learn as many values as possible. Formally, each node p_i is initialized with value v_i for $i = 1, ..., n$. Communication is synchronized and in each round, each node selects a single channel $x \in \{1, ..., C\}$ and either listens for a value or transmits a value on channel x. If a single node transmits on x, then any nodes that are listening to channel x will receive the transmission; otherwise, nothing is received and there is no collision detection. Moreover, in the case of multiple nodes sending on channel x, such listening nodes are unaware of any collisions.

The goal is for each node to learn at least $(n - t) - 1$ other values. Note that learning more than $(n - t) - 1$ other values may be impossible since the adversary can prevent any exchange of values by perpetually targeting a set P' of t nodes and interfering with the appropriate t channels corresponding to P'. Therefore, the formal goal is to achieve $(n-t)$-to-$(n-t)$ information exchange: at least $n - t$ nodes learn at least $n - t$ values.

Selectors and Multi-Selectors: The authors begin their work by generalizing the notion of *selectors*. Selectors were first proposed by Komlos and Greenberg [61] and have found subsequent application in the area of fault-tolerant communication. Throughout, the number of channels used in the protocols of [59] is $c = (5t+1)^2$. Assume that $n \geq c \geq k \geq 1$. Gilbert *et al.* define two new combinatorial structures. First, a (n, c, k)-*multi-selector* is a sequence of functions $M_1, ..., M_m$ from $P \rightarrow [1, c]$ with the following property. For each subset $S \subseteq P$ of size k, there exists an $\ell \in [1, m]$ such that M_ℓ maps each element in S to a unique value in $[1, c]$.

Second, a *generalized* (n, c, k, t)-*multi-selector* is a sequence of functions $M_1, ..., M_m$ from $P \rightarrow [1, c]$ such that for every subset $S \subseteq P$ of size r, for every subset $S' \subseteq S$ of size k, there exists some $\ell \in \{1, ..., m\}$ such that (1) M_ℓ maps each element in S' to a unique value $\{1, ..., c\}$ and (2) M_ℓ maps each element in $S - S'$ to 0. Therefore, each element of S' is "selected" while all other elements in $S - S'$ are avoided. A probabilistic argument is given to demonstrate that multi-selectors exist and have size polynomial in k. A concrete construction is also given for a (n, c, k)-multi-selector of size $O(k^6 \log^5 n)$.

Information Exchange: Why are multi-selectors important? Consider a set of nodes, P_r, that wish to receive messages from some set of transmitting nodes, P_t. The set P_t is created by an earlier aggregation phase whereby at most $2t$ values are not known to all members of P_t. Assume that P_t is divided into subsets each of size c where (1) the members of the subset know the values of all the other members and (2) in each round, these values are transmitted on a unique channel (i.e. not used by another subset). If the nodes in P_r select their channels according to a $(n, c, t+1)$-multi-selector then, for any set of size $t + 1$ listening nodes, there is a round where they are all listening on different channels. The adversary can only disrupt t of these channels and there are c sets of transmitting nodes in P_t. Therefore, at most ct nodes in P_r do not receive a value from all c sets in P_t.

The algorithm then proceeds by selecting a set P'_r of $c(ct + 1)$ nodes in P_r. By the above argument, there exists a subset $P \subset P'_r$ of size c such that all nodes in P know all values. By

transmitting the values in the same fashion as above, at most t nodes do not know all values.

The formation of a set P_t, where all but $2t$ values are known to the members of P_t, and the subsequent dissemination of values, occurs over two phases where each phase consists of multiple rounds. Therefore, after these two phases, at most $4t$ values have not been received by at least $n-t$ nodes. A final special phase is executed that uses a $(n, c, 5t)$-multi-selector in order to transmit values in a manner similar to the above. At the conclusion of the special epoch, at most t values are unknown to at most $n-t$ nodes which is the best that can be achieved. Finally, the latency is measured in terms of the number of rounds which is $O(n/t^2 + t^5 \log^2 n)$.

The Number of Channels and a Lower Bound: The authors consider values of C where $t+1 \leq C \in \Theta(t^2)$. Of particular interest is the challenging case where $C = t+1$. Here, the authors provide a modification to their original protocol that utilizes a $(n, C, C, 2t+1)$-generalized-multi-selector. An upper bound on the latency of $O(n(C+1)^{3t} \log(n/t))$ is derived and a lower bound of $\Omega(2^{t+1}/\sqrt{t+1})$ is also demonstrated. Therefore, in this case, it is impossible to avoid a latency that is exponential in t.

Dolev et al. [62]

Recall from Section II-E, that the classical gossiping problem addresses the all-to-all exchange of values between nodes. As observed in some of the previous surveyed works, such a full all-to-all exchange may be impossible due to interference by an adversary. This is true of the information exchange problem we just examined by Gilbert *et al*. [59] and it is also true of the authenticated pair-wise message exchange setting examined by Dolev *et al*. [63] that we survey next. In this work [62], Dolev *et al*. define the more general (ϵ, δ)-gossip where the goal is for at least $(1-\epsilon)n$ values to be known to at least $(1-\delta)n$ nodes; for example, $(0,0)$-gossip is the classical all-to-all gossiping problem. In this work, the focus is on achieving $(\epsilon, t/n)$-gossip which is referred to throughout simply as ϵ-gossip.

There are assumed to be n nodes each with access to c channels and communication occurs over synchronous rounds in a single-hop network. There exists an adversary that can corrupt $t < c$ channels per round and there is no collision detection. A key departure from previous models is that the authors focus on *deterministic oblivious* gossip algorithms. Such an algorithm is formally defined as a sequence $\mathcal{A} = \langle A_1, ..., A_r \rangle$ where each $A_j : [1..n] \rightarrow \{\texttt{trans}, \texttt{recv}, \perp\} \times [1..c]$ is a function that describes the behavior or each process in round i. That is, $A_j(i) = \langle \texttt{trans}, k \rangle$ denotes that in round j, process i transmits on channel k; the meaning is similar for receiving (`recv`) or inaction ($\perp$). Therefore, we see that under deterministic oblivious algorithms, the actions of transmitting and receiving are scheduled and fixed in advance; this is a significant deviation from many of the previous works we have surveyed so far.

Lower and Upper Bounds for $t = 1$: The authors focus on proving lower and upper bounds on the number of rounds required to achieve ϵ-gossip and the case for $t = 1$ is presented in detail. In order to derive lower bounds, the (n, ϵ)-Clique Destruction Game is introduced which provides a graph abstraction for many aspects of achieving ϵ-gossip in the presence of a malicious adversary. For a complete graph $G = (V, E)$ where $|V| = n$, a solution to the (n, ϵ)-*Clique Destruction Game* is a set of edges $S \subseteq E$ such that $G' = (V, E - S)$ contains no clique of size greater than ϵn. An edge between nodes i and j signifies that these two nodes broadcast in the same round. Therefore, a clique of size at most ϵn implies that the adversary can only disrupt gossip between ϵn nodes. The authors rely on a result by Turán [64] that states: for a graph $G = (V, E)$ that contains no clique of size at least $k+1$, then $|E| \leq (1 - 1/k)(n^2/2)$. From this, the authors can show that if G contains no cliques with size exceeding ϵn, then the number of edges removed must be $\Theta(n/\epsilon)$. By employing this result, the authors derive a lower bound of (roughly) $\Omega\left(\frac{(1-\epsilon)n}{\epsilon c^2}\right)$ on the number of rounds required to achieve ϵ-gossip with an oblivious deterministic algorithm.

In terms of an upper bound, the authors provide an algorithm that consists of two main phases. The first phase handles the collection of values by a subset of nodes. The nodes are divided into two sets: a set of $2c$ listeners and a set of at least $4c$ transmitters. Each channel is a pair of listeners while the transmitters are divided into roughly ϵn sets B_i each of size roughly $1/\epsilon$. For each B_i, the transmitters can be scheduled such that the adversary can stop at most one member of B_i from sending its value to some pair of listers. This collection phases runs in $O\left(\frac{n}{\epsilon c^2}\right)$ rounds.

The second phase involves the dissemination of values and bears some similarities with the work done in [59]. The nodes are divided into c sets and each set is assigned to a channel. The pair of listeners assigned to each channel in the value collection phase now transmit their information to the set of transmitters assigned to the corresponding channel. This procedure occurs in parallel over $c/2$ channels and thus the values are merged over $O(\log n)$ rounds. At this point, at most a single node in a set has received those values which were known to the listeners on their channel; these sets are called *knowledgeable*. These knowledgeable sets are then combined resulting in the nodes knowing $(1-\epsilon)n$ values. The total number of rounds required for this phase is $O(\log^2 n)$. Therefore, over the entire algorithm, the total number of rounds until ϵ-gossip is achieved is $O\left(\frac{(1-\epsilon)n}{\epsilon c^2} + \log^2 n\right) = O\left(\frac{(1-\epsilon)n}{\epsilon c^2}\right)$ which is asymptotically tight with the lower bound.

Extensions

The more general multi-channel case where $t < c < n$ is analyzed in a similar fashion and yields a lower bound of $\Theta(n^{t+1}/c^{t+1})$ and an upper bound of $O\left(\frac{ne^{t+1}}{c\epsilon^t} + c(t+1)^t \log^{t+1} n\right)$. A key point is that instead of a pair of listeners assigned to each channel, $t+1$ listeners are assigned. In this sense, it seems necessary to have knowledge of t, or at least an upper bound on t. The authors discuss the situation where up to t nodes may suffer Byzantine faults. This more challenging case is handled by having $(2t+1)(t+1)$ nodes, instead of simply $(t+1)$ nodes, assigned as listeners to each channel in the protocol. By running the dissemination and combination components of the protocol $2t+1$ times, this additional redundancy allows for each correct node to majority

filter on incoming transmissions and only accept a value if it is received in at least $t+1$ of $(2t+1)$ executions. In this case, the number of rounds required to achieve ϵ-gossip is increased by a factor of $\Theta(t)$.

Dolev et al. [63]

As we have seen, communication in sensor networks is subject to several threats. However, a number of challenges can be overcome if devices possess shared secrets such as shared secret keys. While cryptography cannot prevent jamming attacks, it can provide message and identity authentication. Consequently, the problems posed by spoofing and message injection attacks can be overcome. A crucial question is: how can shared secrets be established? This is the primary issue addressed by the work of Dolev *et al.* [63].

To this end, the authors propose the problem of Authenticated Message Exchange (AME) in which pairs of devices wish to exchange information in an authenticated manner. A single-hop multi-channel network model is adopted and communication proceeds in rounds. Each node has access to C channels where $t < C$ channels are subject to interference by the adversary (spoofing, message injection, jamming) in any given time slot; there is no collision detection. However, the adversary is not reactive; random choices made by correct nodes are known to the adversary only after the end of a round. The goal is to develop a protocol for solving AME; this is called an AME protocol. As input, an AME protocol takes in a set of ordered pairs E from a set of vertices (nodes) V where, for each ordered pair (u, v), the node u wishes to send a message $m_{u,v}$ to node v. An AME protocol must guarantee three properties with high probability: (1) *authenticity* - each correct node can differentiate between messages from other correct nodes and messages spoofed by the adversary, (2) *sender awareness* - each correct node is aware of whether their messages are successfully received, (3) *d-disruptability* - for a set of edges $E' \subseteq E$ where communication fails, the minimum vertex cover of a *disruption graph* $G = (V, E')$ consists of no more than d vertices.

The Fast-AME Protocol: Regarding the property of authenticity, so long as $t + 1$ pairwise communications occur concurrently, then at least one transmission is successful and, if channels are scheduled deterministically, then spoofing is prevented since the adversary's transmission will simply result in a collision. In terms of disruptability, note that, for a d-disruptable protocol, there exists a set of no more than d nodes such that if these nodes are not considered, all other nodes will succeed in sending their respective messages. Therefore, this property is important; however, in order to achieve d-disruptability and sender awareness, further insights are necessary.

Note that no deterministic protocol can yield d-disruptability for $d < t$ since the adversary may easily target a set of d nodes. In fact, the authors also prove that randomization cannot help and, therefore, t is a lower bound on the disruptability. It follows that the adversary is always guaranteed to be able to prevent t nodes from successfully communicating. The aspect of d-disruptability in the AME problem can be viewed as a directed graph problem where vertices correspond to devices and edges correspond to pairwise communication between these devices. Given a directed graph $G(V, E)$, the goal is to produce a new graph that has a minimum vertex cover of size at most t where the modifications to G occur via a game played between two parties: a player and the referee. This game is called the (G, t)-*starred-edge removal game* and we refer the reader to the original paper for details. The final result of this game is the disruption graph G', with its reduced set of the edges $E' \subseteq E$ and a minimum vertex cover of at most t. The authors provide a greedy removal strategy for the (G, t)-starred-edge removal game that terminates in $O(|E|)$ rounds.

The fast-AME protocol (f-AME) proceeds in phases where, in each phase, the protocol simulates a step of the greedy removal strategy for the (G, t)-starred-edge removal game. However, nodes must agree both on the proposed set P and the choices made by the referee at each step. To this end, the authors propose a *feedback* protocol that, with high probability in n, allows nodes to agree on this information and terminates in $O(t^2 \log n)$ rounds of communication. Using the feedback protocol, f-AME achieves its goals by alternating phases of the (G, t)-starred-edge removal game and the feedback protocol, eventually terminating in $O(|E|t^2 \log n)$ rounds of communication.

The authors explore other cases where $C \geq t + 1$, $C \geq 2t$ and $C \geq 2t^2$; as expected, the communication and time complexity of f-AME decreases. The communication complexity of the original feedback protocol can be quite large and the authors demonstrate how to reduce this communication complexity. Finally, the issue of establishing a shared secret key is demonstrated. A set of $t + 1$ leader nodes are selected and f-AME is used to establish a shared secret key between each leader node and each non-leader node. A leader node that succeeds in establishing a key with $n-t$ other nodes is called *complete*. Each complete leader v selects a special *leader key* K_v and, using the secret key previously exchanged, transmits K_v using a pseudo-random channel hopping technique. Eventually, all the nodes agree on a leader key to be used as the shared group key using an agreement strategy. The total number of communication rounds required for establishing such a shared key is $\Theta(nt^3 \log n)$.

Meier et al. [65]

The ability to transmit over more than one channel is critical to a number of methods aimed at mitigating jamming attacks. For instance, channel-hopping defenses have been shown to be effective in permitting communication in the presence of jamming adversaries [41], [42]. However, a prerequisite for any multi-hop network communication is often the discovery of the local topology. That is, a node requires some knowledge regarding the existence of its neighbors in order to facilitate successful routing. In this work, Meier *et al.* focus on the problem of *node discovery* in the presence of a jamming adversary. Formally, two nodes v_1 and v_2 discover each other if the following criteria hold: (1) v_1 and v_2 are on the same channel c, (2) v_1 is in the listening state and v_2 sends its contact information on channel c (or vice versa) and (3) the channel c is not jammed. The adversary has the ability to

jam t out of m channels that would otherwise minimize the number of time slots prior to v_1 and v_2 from discovering each other; the authors label the expected amount of time prior to discovery as the *discovery time*. The reactive capability of the adversary to select its jammed channels in this fashion makes the problem challenging.

A key metric in evaluating the performance of the discovery algorithms is *competitiveness* which is defined as follows. Let REF be an optimal randomized algorithm which has knowledge of t. Let T^t_{REF} denote the expected time until v_1 and v_2 discover each other using REF. Let ALG be a discovery algorithm and T^t_{ALG} denote the corresponding discovery time for v_1 and v_2. Then the competitive ratio ρ is: $\max_{0\leq t\leq m-1} T^t_{ALG}/T^t_{REF}$. Therefore, an efficient discovery algorithm has a small corresponding ρ value.

Main Results: The authors begin by considering the case where t is *known* to v_1 and v_2. Clearly, for $t = 0$, communication can occur over a single channel agreed upon in advance. For $1 \leq t \leq m/2$, the authors prove that having each node choose one of the first $2t$ channels uniformly at random is optimal and yields a discovery time of $8t$. Finally, for $m/2 + 1 \leq t \leq m - 1$, the optimal strategy is to select each of the m channels with probability $1/m$ which gives a discovery time of $2m^2/(m - t)$. Therefore, the behavior of the optimal algorithm REF is established. The stateless and independent nature of this randomized algorithm is clearly well-suited to sensor networks.

The next step is to consider a particular type of algorithm when t is unknown. Let $\hat{t}$ denote the set of channels $c_1, ..., c_{2\hat{t}}$ which are referred to as a *class* of channels. The authors examine algorithms that: (1) select each class of channels i uniformly at random, and then (2) select uniformly at random a particular channel on which to transmit. This general approach to a discovery algorithm leads to the notion of ALG_k which selects randomly from the classes $\hat{t} = m^{1/k}, ..., m^{i/k}, ..., m$. The authors demonstrate that for $k = \lfloor \log m \rfloor$, the competitive ratio is $O(\log^2 m)$. It is then shown that the optimal competitive ratio in this scenario can be formulated as an optimization problem and yields an algorithm OPT with $\rho = \Theta(\log^2 m)$ which illustrates that $ALG_{\log m}$ is indeed asymptotically optimal.

Extensions and Experimental Results: As an intermediate step, the authors next consider the case when we have partial knowledge of t in the sense that the distribution for the number of jammed channels is known *a priori*. The analysis of this case yields a non-linear optimization formulation that can only be solved numerically. A multi-player setting is also considered. Here, the objective is to minimize the expected discovery time for two nodes in the presence of other nodes that are similarly attempting to discover each other. It is shown that for n nodes, when t is known, if $\Omega(n + t)$ channels are available, then the asymptotically optimal expected discovery time is $\Theta(n + t)$. Finally, the authors experiment with their algorithms. For a random jamming adversary, they observe improved performance of $ALG_{\log m}$ and OPT over the algorithms that simply select channels uniformly at random (UNI) when the number of jammed channels is not excessive (roughly 100 out of the total 128 channels). The authors also experiment with a situation where interference occurs due to non-malicious radio interference. To this end, they experiment with the upper-end Bluetooth channels which are subject to interference from a microwave oven. Experiments with a Bluetooth device discovery algorithm illustrate that OPT achieves a substantial reduction in discovery time; up to $1/3$ of the Bluetooth algorithm when interference is present.

King et al. [66]

Several of the results we have surveyed consider the energy expenditures of the adversary and the correct players in isolation. However, if energy is a scarce resource for both the adversary and correct nodes, then it seems reasonable to consider a notion of relative cost. In [66], the authors pursue this idea by looking at the expected cost of a correct node relative to the cost incurred by the adversary, where cost is measured in the number of time slots spent in either the transmit or receive state. The overall goal is to design a communication algorithm that yields a relative cost that is in favor of the players; such an algorithm is called favorable.

A Simple Scenario: Similar to that considered in [55] the authors first examine a simple two-player scenario in a single-channel time-slotted network where the players, Alice and Bob, can detect collisions. Alice (the sender) wishes to transmit a message m to Bob (the receiver) and there is a jamming adversary who aims to prevent communication. The jamming adversary is allowed to follow any jamming strategy it wishes (i.e. constant, random, adaptive, reactive) and incurs a cost of T by either (1) jamming the channel or (2) forging collisions whereby a correct player will detect that a message collision has occurred.

A significant deviation from the work in [55] is that cryptographic authentication is assumed; recall the discussion in Section II-C. Therefore, the adversary is unable to spoof either the sender or receiver. In contrast to other work, another important difference is that communication must be guaranteed with probability 1 instead of with high probability. As we have seen in [57] and [63], high probability guarantees are provided in terms of n. When n is large, then the probability of failure can easily be made quite small. However, in sensor networks, n will likely correspond to the number of nodes within a broadcast radius which can be small relative to the total number of nodes N. Consider the situation where the probability of failure is at most $O(n^{-c})$, but the total number of nodes is exponential in n (i.e. $N = \Theta(2^n)$). Then for a message that transits $\Omega(N)$ hops, the probability of failure along the path of traversal is substantial.

The protocol specified by the authors achieves communication in the following general way. The protocol proceeds in rounds each consisting of a pair of phases where the number of time slots per phase increases geometrically per round. In phase 1, Alice transmits m randomly while Bob listens randomly. If Bob receives m, he terminates the protocol. Otherwise, in phase 2, Bob sends a request to Alice for m to be retransmitted in the next round. Alice listens to a sampling of slots in phase 2 and, if a retransmission request is received *or* a collision is detected, she proceeds to the next round and begins transmitting again in phase 1; otherwise, Alice terminates the protocol.

The adversary can prevent termination of the players in more than one fashion. In order to prevent Bob from terminating, the adversary can jam the transmission of m. However, the analysis shows that this is difficult to do without the adversary incurring far more cost than either correct player. On the other hand, the adversary can prevent Alice from terminating by simply forging collisions which Alice will interpret as having originated from Bob. This can be achieved by having two Byzantine devices broadcast simultaneously or replaying noise from a previous message collision over the channel. Despite these challenges, the authors prove that their protocol requires the players to incur an expected cost of $O(T^{\varphi-1}+1)$ where $\varphi = \frac{1+\sqrt{5}}{2} \approx 1.618$ is the golden ratio. Since the adversary expends T, the players spend asymptotically less in expectation; that is, the protocol is favorable. Finally, the authors show that their results hold in the presence of a reactive adversary, so long as there is sufficient communication traffic in the network.

Extensions to Multiple Players and Reliable Broadcast: The favorable communication algorithm can be extended to a multi-player scenario where there is a single sender and n receivers, some of which may be controlled by the adversary. These n nodes represent neighbors within the broadcast neighborhood of the sender. The Byzantine nodes behave arbitrarily, either jamming or issuing spurious retransmission requests. Favorability is shown in a similar fashion to that of the two-player scenario. The authors are then able to extend the multi-player scenario to the problem of reliable broadcast for sensor networks. The main result here is that players achieve an expected cost that is asymptotically superior to previous results where an adversary seeks to thwart reliable broadcast.

Summary of Results and Discussion

Table I summarizes key aspects of the models and the main results surveyed in Section III. In these works, we examined several models of jamming adversaries and the algorithmic approaches for addressing various communication problems. Through this examination, some common themes can be discerned.

To begin, all such models seem to agree on the use of a synchronized and discrete notion of time. Transmissions occur in fixed length time slots and all deivces must typically agree on the beginning and ending of such slots. However, in the context of real-world deployments, it is far from clear whether this property will hold true. Gilbert *et al*. [55] go some distance in addressing the problem where nodes may activate without a notion of a common start time. On the other hand, clock synchronization schemes have been proposed, even in adversarial settings [67], [68], and so this provides some justification for a synchronization assumption.

The ability of nodes to detect collisions appears to have a clear effect on the feasibility of certain tasks. For instance, in a single-channel network, the *unforgeability of silence* assumption introduced by Gilbert *et al*. [55], and used again by King *et al*. [66], seems necessary in order to achieve communication between two parties. However, in multi-channel networks, so long as $t < C$, it seems that several fundamental communication tasks are feasible without the need for collision detection. It is also important to note that the issue of whether or not to assume collision detecion is not a simple binary feature of these models. For example, in the work by Dolev *et al*. [63], collision detection is not assumed; however, when a collision does occur, it is assumed that all transmissions are lost and the receiver hears nothing. This is in contrast to the more challenging model where the adversary can decide which message is delivered upon a collision, which might be the case if we wish to account for the capture effect. Moreover, other notions regarding the accuracy of collision detectors have been examined, along with their impact on the consensus problem, by Chockler *et al*. [31].

With regards to the type of adversary modeled, we can argue that the reactive adversary aligns well with reality. For starters, RSSI requries negligible energy and can be executed several times within a typically-sized time slot. Therefore, it is reasonable to assume a determined malcontent can detect activity in real-time and then decide to jam. Indeed, reactive adversaries pose a serious threat as their effectiveness has been clearly demonstrated [37], [47], [69], [70] and typically many models make this worst case assumption on the power of the adversary. Furthermore, although it is often not explicitly stated, most results that tolerate a reactive adversary are also able to tolerate an adaptive adversary. Given this state of affairs, it seems that future models should strive to obtain results in this context.

In discussing potential future work, it is important to keep in mind that the study of adversarial interference in sensor networks is fast-moving; exciting and surprising results are sure to emerge. However, from the surveyed works above, we can recognize several aspects that deserve attention. For starters, a logical step that seems absent in many of the surveyed works is an extension to multi-hop networks. Technology permitting, sensor networks of considerable size may be deployed at which point a complete-graph will no longer be an accurate model of communication. The work of Richa *et al*. [58] demonstrates that, while a multi-hop extension may not require a radically new protocol, the corresponding analysis can be challenging and require new techniques.

Many protocols we surveyed have tended to neglect receive-state costs and, instead, have focused on message complexity. Are there ways to make existing protocols more energy efficient? Conversely, are there limits (lower bounds) on the energy efficiency of protocols for certain fundamental problems such as reliable broadcast, consensus, leader election and various types of gossiping? On the other hand, with regards to energy efficiency, it is important to keep in mind that this concern is motivated by *current* technology. For example, there has been a significant amount of work done on sensor network devices that can harvest solar energy in order to replenish their respective energy supplies [71], [72]. To date, theoretical models and the algorithmic implications of such technology have not been fully explored.

The adversarial models we have surveyed here treat interference in a fairly simplistic fashion where the success or failure of communication depends solely on the actions of the adversary. The range of interference is also simplified in that these models consider communication between a pair of nodes in isolation. Of course, such an approach is employed

TABLE I
A COMPARISON OF THE DIFFERENT ADVERSARIAL JAMMING MODELS AND SUMMARY OF MAIN RESULTS SURVEYED IN SECTION III.

Paper	Adversary Type	Spoofing	Collision Detection	Main Result & Advantages/Disadvantages
Gilbert *et al*. [55]	Constant, Random, Adaptive, Reactive	Yes	Yes	Communication in the face of all adversaries when spoofing is allowed and adversarial budget is unknown. Extensions to reliable broadcast, consensus and leader election. Applies to single-hop, single-channel networks and communication protocol is not energy efficient as it requires nodes to perform significant listening.
Awerbuch *et al*. [57]	Adaptive	No	Yes	Constant throughput is possible against a $(T, 1-\epsilon)$-bounded adversaries. Extension to leader election. Applies to single-hop, single-channel networks and cannot handle reactive adversaries. While the communication complexity of the protocol is relatively small, the amount of listening required is significant.
Richa *et al*. [58]	Adaptive	No	Yes	Constant throughput is possible against $(T, 1-\epsilon)$-bounded adversaries even in multi-hop, single-channel networks. Cannot handle reactive adversaries, current results rely on the UDG model, and the amount of listening required by the protocol is significant.
Gilbert *et al*. [59]	Random, Adaptive, Reactive	No	No	Each node may learn at least $(n-t)$ values held by nodes in a multi-channel network where the adversary jams $t < C$ channels. Applies to single-hop networks. Exponential (in t) round complexity when $t = C-1$ is shown to be unavoidable in this model.
Dolev *et al*. [62]	Random, Adaptive, Reactive	No	No	Demonstrates an ϵ-gossip protocol in a multi-channel network. Protocol can be modified to tolerate Byzantine players. Analysis restricted to deterministic oblivious algorithms and single-hop networks. The value of t, or an upper bound on t, seems necessary for the protocol to function.
Dolev *et al*. [63]	Random, Adaptive, Reactive	Yes	No	Establish a shared secret after nodes have been deployed in a multi-channel network when the adversary can jam $t < C$ channels. Applies to single-hop networks and the round complexity for establishing a shared key grows as $O(nt^3 \log n)$ and so may rapidly increase with the power of the adversary.
Meier *et al*. [65]	Random, Adaptive, Reactive	No	No	Addresses the problem of node discovery in a multi-channel network when $t < C$ channels can be jammed and t is unknown. Experimental results show promising behavior. Analysis uses the notion of a competitive ratio, but overall communication and listening costs are not reported.
King *et al*. [66]	Constant, Random, Adaptive, Reactive	No	Yes	Provides a favorable protocol in a single-channel, single-hop network. Extension to favorable reliable broadcast in multi-hop networks. Assumes the existence of cryptographic authentication.

in order to ease the analysis; however, it also raises questions about the accuracy of the results. There is a growing literature on the various analytical models of wireless interference and we refer the reader to the survey by Cardieri [73]. For example, the popular signal-to-interference-plus-noise (SINR) model incorporates a number of factors that effect the message reception and this model tends to align well with reality. Consequently, it would be of interest to derive results on adversarial interference under such a model.

Finally, to the best of our knowledge, with the exception of the work by Meier [65], the protocols we have surveyed have not been implemented and tested either with a simulator or with a testbed. Such an endeavor is likely to be arduous and might be viewed as pedestrian. However, on the contrary, the research community would likely benefit from a validation (or invalidation) of the adversarial models and corresponding algorithms surveyed here. Without such an evaluation, it may become increasingly difficult to justify algorithmic advances, regardless of their mathematical elegance.

IV. MODELS OF BYZANTINE ADVERSARIES AND RELIABLE BROADCAST

A fundamental problem in distributed systems is Byzantine Agreement. Since its introduction by Pease *et al* [74], this problem has received much attention. In the Byzantine Agreement problem there are a total of n processors in the system. Of these n processors, t of them may deviate from protocol in an arbitrary fashion; such processors are termed Byzantine or faulty. Furthermore, these processors are assumed to be controlled by an adversary and, therefore, they may act in concert to wreak havoc on the system. The remaining $n-t$ processors that are not controlled by the adversary are assumed to obey protocol and we call these processors correct. Note that while the processors corrupted by the adversary are fixed prior to the execution of any protocol (a static adversary), the correct processors do not know which are faulty. A processor p_i can set a value field to either 0 or 1 and must commit to a value $b_i \in \{0,1\}$ by the end of the protocol. There exists a processor known as the *dealer* (or the *source*) holding an initial value $v \in \{0,1\}$. A protocol that achieves Byzantine agreement is one which guarantees the following two properties:

1) All correct processors commit to the same value in $\{0, 1\}$
2) If the dealer is correct, then all correct processors commit to v.

While (1) and (2) are basic guarantees one would desire in such an adversarial setting, the problem of achieving them is non-trivial due to the arbitrary behavior of the faulty processors.

In this section, we survey results on Byzantine agreement in sensor networks. In this domain, we speak of nodes rather than processors and the problem has traditionally been labeled as *reliable broadcast* in the literature. In contrast to Section III, reliable broadcast in sensor networks typically (but not always) focuses on the problem of routing in a multi-hop network as opposed to focusing on communication amongst a small number of local parties. Furthermore, as we will see, the challenges that arise in multi-hop networks due to Byzantine faults are substantial even in the absence of jamming.

Berman et al. [75]

In a single-hop network, consider the case where each node suffers a Byzantine fault with constant probability $p < 1/2$; recall, this is the probabilistic fault scenario. Can reliable broadcast be achieved? And if so, what is the latency? These are the questions addressed by Berman *et al.* [75] who show that reliable broadcast can be achieved with probability at least $(1 - 1/n)$ and with a latency of $O(\log n)$ where n is the number of nodes.

The broadcast protocol is framed as follows. All nodes are partitioned into $\Theta(n/\log n)$ groups each of size $\Theta(\log n)$. These groups are then viewed as nodes belonging to a binary tree. The root node signifies the initial point of broadcast of the message where each node in this root group accepts the source's broadcast. Then, at each step, the message propagates along the edges of the tree to the children. The method of propagation involves the use of expander graphs. Specifically, communication from a parent group G_1 to a child group G_2 occurs along the edges of a bipartite expander graph defined in [76]. Each node then majority filters on the incoming messages in order to obtain the correct messages. The properties of the expander graph ensure that, given a constant probability of failure $p < 1/2$, a majority of the nodes in the child group will commit to the correct message with probability at least $1 - 1/n$. This propagation occurs from the root to the leaf level of the logical binary tree. At this point, the following action is taken by each node in each group in parallel. A node transmits its decision to other nodes within its group through the use of edge-disjoint matchings that together cover all edges corresponding to the group's complete graph on m nodes. This group action allows for the presence of erroneous messages to be "washed out" and, therefore, all correct nodes will have the correct message with high probability.

The broadcast at the root requires $O(\log n)$ time and each transmission from parent group to child group requires $O(1)$ time. Therefore, propagation to the leaf level requires $O(\log n)$ time. Finally, the group action requires $O(\log n)$ time per group; however, as it is run in parallel, the total running time of the broadcast protocol is $O(\log n)$.

The approach of using groups of size $\Theta(\log n)$ is what allows for the high-probability guarantees of this work. Note that, if the probability of a Byzantine fault is at least $1/2$, broadcast is guaranteed to fail with high probability by application of simple Chernoff bounds. While the main results of Berman *et al.* are given in the context of complete graphs, their algorithm can be applied to multi-hop networks assuming that the topology supports it. For instance, the authors suggest a tree topology where the nodes of the tree represent cliques of, say, logarithmic size and each group has all-to-all links with any children groups. However, such topologies seem somewhat restrictive and it is unclear whether such an algorithm would find utility outside of single-hop networks.

Pelc and Peleg [77]

Like the previous work, here Pelc and Peleg [77] consider the probabilistic fault scenario, reliable broadcast occurs logically along a tree topology rooted at the source, and success is guaranteed with high probability. However, in contrast to the previous work where failures were permanent, the failures here are assumed to be *transient*; that is, they occur for a single step and then may be corrected. A substantial portion of this work focuses on a message passing model where a node may send (possibly different) messages to each of its neighbors simultaneously and it may receive messages from all of its neighbors simultaneously. For the purposes of this survey, we focus on the results derived for the radio model which is a suitable fit for communications in a sensor network.

In the presence of Byzantine faults, the authors propose a simple reliable broadcast protocol. Let $k = \lceil c \log n \rceil$ and assume that the nodes $s = v_1, ..., v_n$ are ordered by their distance from the source node s. The protocol begins with the source node v_1 broadcasting its message for k time slots. Then v_2 majority filters on any messages it received, commits to the majority message, and broadcasts this message for k time slots. The process repeats for each consecutive node. As before, standard Chernoff bounds are used to give the high probability guarantee on successful reliable broadcast when the probability of a fault is $p < 1/2$. Conversely, the authors also demonstrate that for $p \geq 1/2$, it is not possible to guarantee reliable broadcast with high probability.

Finally, the authors examine bounds on the latency for achieving reliable broadcast with high probability. Let OPT denote the time required for reliable broadcast in a fault-free situation. Then, the authors provide a modified protocol that achieves a latency of $O(OPT \cdot \log n)$. Furthermore, it is also demonstrated that there exists a graph for which reliable broadcast cannot be achieved in $O(OPT + \log n)$ time. It remains an open question as to whether a lower latency can be achieved or $\Omega(OPT \cdot \log n)$ is indeed a lower bound.

The Grid Model

Before proceeding to other works on reliable broadcast, we first introduce a popular grid model for sensor networks that finds use in the remaining surveyed works. The grid model assumes that nodes are located at points in a 2-dimensional lattice or grid. The notation $p(x, y)$ denotes the node at location (x, y) on the grid; if the location has already been

established, p is simply used to denote the node. Every node has access to a single broadcast channel which allows for the transmission of a message within a radius r on the grid. For simplicity, the L_∞ metric is assumed where the distance $d(a,b)$ between two locations $a=(x_1,y_1)$ and $b=(x_2,y_2)$ is defined as $d(a,b)=\max\{|x_1-x_2|,|y_1-y_2|\}$. Results for the L_∞ metric can be extended to the more standard Euclidean metric where the broadcast neighborhood is a circle centered at the broadcaster.

When $p(x,y)$ broadcasts a value v, all nodes within a radius r of (x,y) receive v. The set of $(2r+1)\times(2r+1)$ nodes, including p itself, within radius r of $p(x,y)$ is the *neighborhood* of p and denoted by $N(x,y)$ or, when p is unambiguous, $N(p)$. A node is correct or honest if it obeys a prescribed protocol; otherwise, a node is said to be faulty, Byzantine, or adversarial if it is controlled by the adversary and, therefore, is free to deviate arbitrarily from any prescribed protocol. Nodes, correct or faulty, cannot spoof other nodes in the network. The adversary is free to select those nodes in the network it wishes to control subject to one constraint: *every neighborhood in the network contains at most t faulty nodes.*

There is a broadcast schedule that dictates when a particular node in the network is free to transmit over the single channel all nodes share. This schedule is assumed to be obeyed by all nodes, both faulty and correct, and prevents message collisions. In [78], a suggested schedule is provided where the node $p(x,y)$ transmits at time step $(x \mod (2r+1))\times(2r+1)+(y \mod (2r+1)) \mod (2r+1)^2$. A unique player is located at position $(0,0)$ on the grid. As in the Byzantine Agreement problem, this unique player is known as the dealer (or source) and is represented by $d(0,0)$ or d (or s). It is the dealer who initiates a broadcast of some value v.

A Popular Abstraction: Discussion of the Grid Model

Theoretical treatments of radio networks have attracted a measure of skepticism from practitioners [18], [19] and there are several aspects of the grid model that warrant discussion. The following issues are cause for concern:

Broadcast Region: Experiments with real-world deployments demonstrate that a broadcast region is not well-characterized by a circle of radius r with p located at the center [79], [80]. In fact, the region successfully receiving a broadcast is often dynamic, changing constantly depending on nearby sources of electromagnetic interference, power irregularities, features of the terrain, and weather conditions.

Network Synchrony: The model assumes the existence of a predefined schedule known to all nodes in the network. It is conceivable that such a schedule might be programmed into the nodes before deployment. However, use of a schedule presupposes a synchronous network where each node is able to determine its allotted time slot. Moreover, this synchrony must be maintained throughout the lifetime of the network so either there can be no significant clock drift over time or there must be a mechanism in place to maintain synchronization.

Network Topology: The use of a grid allows for an elegant analysis of the system. However, this layout is unrealistic for many settings.

Despite these concerns, there are a number of results in the literature that perhaps redeem the model. Some experimental work has demonstrated that the range of transmission is fairly reliable up to a certain distance δ [21]; therefore, it may be reasonable to take $r=\delta$. Other experimental studies have shown that communication is reliable so long as the signal-to-interference-plus-noise-ratio exceeds a threshold value [81], [82]; therefore, a transmission power above this threshold yields reliable communication. Broadcast inconsistencies due to weather or terrain are harder to rationalize; however, such failures may be handled through more robust transmission algorithms such as described in [83]. Moreover, the irregularity of the broadcast neighborhood has been studied [21], [79], [84], [85] along with a number of techniques for ameliorating the unreliability of the wireless medium. Regarding synchronization, this has been demonstrated experimentally without faults in [26] and, in the presence of an adversary, this has been examined in [67], [68]. In the course of this survey, we will see several works that generalize from the grid to arbitrary topologies.

Koo [78]

A number of works have examined the problem of reliable broadcast under a model where devices communicate via pairwise channels or multi-cast. However, in sensor networks, neither model is appropriate as a message is received by all devices within the radius of transmission and the work in [78] is the first to address reliable broadcast in this context. Koo places his results in the context of *radio networks* rather than sensor networks. The distinction is irrelevant to our purposes as these results apply equally to sensor networks which possess a transceiver in addition to sensing equipment. Radio networks are more general as they are not necessarily equipped with sensing apparatus and may not be subject to the same energy constraints as sensor networks.

While Koo provides both novel upper and lower bound results, it is perhaps the latter which is most important given subsequent improvements to the former and the idealized nature of the grid model discussed previously in Section IV. This result is stated as: Broadcast is impossible in the L_∞ metric if $t \geq \lceil \frac{r}{2}(2r+1) \rceil$.

In terms of Koo's upper bound, subsequent works have improved on this. Nonetheless, it is informative to review Koo's reliable broadcast protocol since it plays a role in later investigations into general sensor network topologies:

Reliable Broadcast Protocol (Koo, 2004)

- The dealer d broadcasts a value v.
- All players in $N(d)$ broadcast v to their neighbors and commit to v.
- For any player p not in $N(d)$, p commits to v when it receives $t+1$ message claiming v as the correct value from distinct nodes in $N(p)$.

There are two things to note about this protocol. First, a simple majority rule is employed to allow a node p to commit. From within $N(p)$, p may receive up to $2t+1$ values of

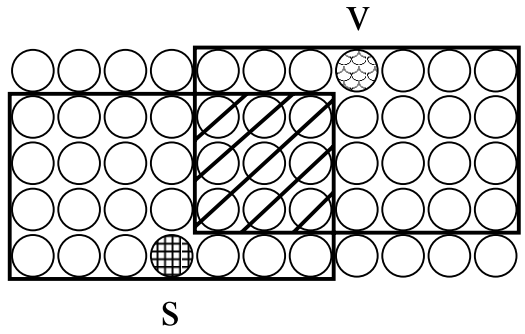

Fig. 1. An illustration of r^2 nodes (diagonal lines) in $N(v)$ that are closer to s than to v, where $r = 3$. This generalizes to any value r.

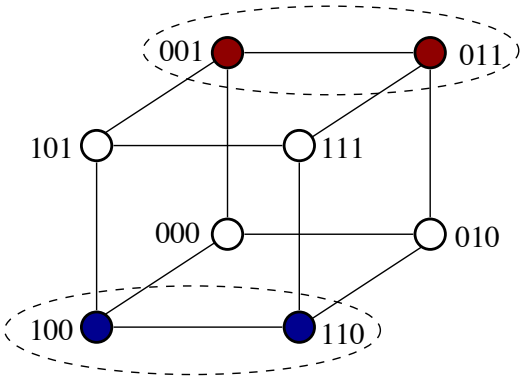

Fig. 2. An illustration of the cube H_3 and the sets $C_1 = \{001, 011\}$ and $C_2 = \{100, 110\}$ that correspond to calculating the value $LPC(H_3)$.

which at most t can be incorrect; therefore, receiving $t + 1$ identical values from unique nodes in $N(p)$ is sufficient to allow p to commit. Second, a node p only acts on information regarding how nodes within $N(p)$ behave. In contrast, one might consider a protocol whereby p gathers information from outside $N(p)$ and uses this to aid in its decision to commit to a received value. The situation is complicated since information from outside $N(p)$ must reach p via a node from within $N(p)$. If along this two hop path, either node forwarding information to p is faulty, problems may arise.

Koo's original protocol provides the following guarantee: if $t < \frac{r}{2}(r + \sqrt{\frac{r}{2}} + 1)$, then the protocol achieves reliable broadcast in the L_∞ metric. The proof of this result is a complicated inductive argument and we omit discussion of this since a subsequent result in [86] improves upon the upper bound utilizing an elegant proof. An important detail, which has bearing on a more realistic model of the broadcast neighborhood, is the extension of this result to the L_1 and L_2 (Euclidean) metrics. Finally, the authors show that, if $t < \frac{r}{4}(r + \sqrt{\frac{r}{2}} + 1) - 2$, then the protocol achieves reliable broadcast in the L_1 and L_2 metrics.

Pelc and Peleg [87]

Pelc and Peleg examine a generalization of the t-locally bounded fault model; that is, where each node contains at most t Byzantine nodes within its neighborhood. Specifically, they examine the broadcast protocol of Koo [78], which the authors label here as the *Certified Propagation Algorithm* (CPA), with the aim of establishing conditions for which CPA achieves reliable broadcast under arbitrary graphs in contrast to the grid model. The authors first define $X(v, s)$ to be the number of nodes in $v's$ neighborhood that are closer to s than v and then define the parameter $X(G) = \min\{X(v, s) \mid v, s \in V,\ (v, s) \notin E\}$. The first main result that the authors present is that, for *any* graph G such that $t < X(G)/2$, CPA achieves reliable broadcast. It is interesting to contrast this upper bound against the upper bound of $t < (r/2)(r + \sqrt{r/2} + 1)$ obtained in [78]. For example, Figure 1 illustrates a case where only r^2 nodes in $N(v)$ are closer to s. In this case, the bound by Pelc and Peleg gives $t < r^2/2$ which is a weaker bound than that given by [78].

The authors then turn to lower bounds in arbitrary graphs. To this end, they define a *t-local pair cut* to be a cut $C \subseteq V$ such that C can be partitioned into two disjoint sets $C = C_1 \cup C_2$ such that C_1 and C_2 have the property of being t-locally bounded. Then, the *local pair connectivity* of a graph G, denoted by $LPC(G)$, is the smallest non-negative t such that G possesses a t-local pair cut. The lower bound follows: for any graph G, reliable broadcast is infeasible for $t \geq LPC(G)$. For instance, the authors provide an example with the cube H_3 depicted in Figure 2. Here, $C = \{001, 011, 100, 110\}$ partitions H_3, and $C_1 = \{001, 011\}$ and $C_2 = \{100, 110\}$ are both 1-locally bounded; therefore, there is no reliable broadcast protocol for H_3.

Interestingly, it can be shown that there are graphs for which CPA fails to achieve broadcast while another protocol called the *Relaxed Propagation Algorithm* (RPA) succeeds. With RPA, a node broadcasts a received message immediately after it is heard, along with the entire sequence of nodes through which the message previously passed. A message m is accepted if it was received from: (1) the source directly or (2) $t + 1$ distinct neighbors who committed to m or (3) $2t + 1$ distinct neighbors who received it along $2t + 1$ node-disjoint paths originating from $2t + 1$ distinct nodes who committed to m. As we will see, RPA shares features in common with the improved reliable broadcast algorithm of Bhandari and Vaidya [86]. In any event, the existence of RPA proves that CPA is not the strongest broadcast protocol. Furthermore, assessing whether reliable broadcast is possible for $t \in [\lfloor X(G)/2 \rfloor + 1, LPC(G)]$ is problematic and the authors pose as an open question whether it is possible to tighten this interval by employing a different graph parameter.

Finally, the authors examine how knowledge of the graph topology is useful. Specifically, they prove the existence of a graph G for which reliable broadcast is possible when the topology is known, but impossible when the topology is unknown. This result has negative implications for ad hoc networks where devices may self-organize without having global knowledge of the network topology; indeed, the topology may change regularly rather than remaining static.

Ichimura and Shigeno [88]

In the previous work by Pelc and Peleg [87], for an arbitrary graph $G = (V, E)$, the parameters $X(G)$ and $LPC(G)$ were used to characterize the feasibility of reliable broadcast in the locally-bounded fault scenario using CPA. However, the

situation for $t \in [\lfloor X(G)/2 \rfloor + 1, LPC(G)]$ proved to be problematic. Here, Ichimura and Shigeno succinctly demonstrate two main results. First, they show that calculating $LPC(G)$ is NP-hard and so the utility of $LPC(G)$ is further limited. Second, the authors introduce a new parameter $\tilde{X}(G)$ based on the concept of a *maximum adjacency ordering*. Since the definition of $\tilde{X}(G)$ is fairly involved, we omit it here and refer the interested reader to the paper. However, using this new parameter, the authors show that CPA achieves reliable broadcast for $t < \tilde{X}(G)/2$ and cannot when $t > \tilde{X}(G)$. Importantly, it is also shown that $\tilde{X}(G)$ can be computed efficiently and that $\tilde{X}(G) \geq X(G)$ for any graph G. On the other hand, no relationship is established between $\tilde{X}(G)$ and $LPC(G)$, so a direct comparison in terms of the relative sharpness of these two parameters is not possible. Nonetheless, this parameter provides an interesting measure for achieving reliable broadcast using CPA; specifically, the range of uncertainty is $t \in [\lfloor X'(G/2) \rfloor + 1, X'(G)]$.

Bhandari and Vaidya [86], [89]

The last two works we surveyed, looked at generalizations of the problem posed by Koo [78] by investigating arbitrary graph topologies and examining the conditions under which CPA achieved reliable broadcast. Here, the work of Bhandari and Vaidya [86] returns to the original grid model and improves over the results of Koo [78] and an intermediate result by Vaikuntanathan [90] which lowered the upper bound to $t < (\frac{1}{\sqrt{2}} - \epsilon)r^2$ for a small constant $\epsilon > 0$. The presentation given in [86] is intricate and based on a complicated inductive argument which requires careful attention to the geometry of the grid. A much simplified and more elegant presentation was provided subsequently in a technical report [89] and for ease of exposition, we review this result. Regardless of the presentation, the main result achieved by Bhandari and Vaidya is an improved upper bound that matches the lower bound given by Koo [78].

Reliable Broadcast Tolerating Byzantine Faults: The presentation in [89] is particularly pleasant because, although it applies to the grid, the analysis itself allows us to leave behind the rigid grid layout of the previous work and consider a graph $G = (V, E)$ of arbitrary topology. Here, vertices correspond to nodes/players and an edge (u_1, u_2) implies $u_1 \in N(u_2)$ and vice versa. An d-cut is a cut $C = (S, V - S)$ containing the dealer d and this cut will be used to represent the frontier of broadcast. The set S, which we will refer to as the source side of the cut, denotes those nodes which have committed to the correct value v and the set $V - S$ denotes those nodes that have not committed. The problem of whether a correct node in $V - S$ will commit to the correct v can be viewed as a connectivity problem. If a node p has a sufficient number of vertex disjoint paths to nodes in S, then p will be able to cross over to the source side of the cut; that is, p will commit correctly. It remains to be seen what number of vertex disjoint paths is sufficient.

We briefly review the completeness argument given in [89] as it is a fairly intuitive but important result that serves as a foundation for several other works we survey later on. Let $p(x, y)$ denote the node p at location (x, y) in the grid. The perturbed neighborhood $PN(p)$ of $p(a, b)$ is defined as $PN(p) = N(a+1, b) \cup N(a-1, b) \cup N(a, b+1) \cup N(a, b-1)$. The following protocol for reliable broadcast in the presence of Byzantine faults is by Bhandari and Vaidya [89]. In this protocol, the message COMMIT(i, v) signifies that node i has committed to value v, and the message HEARD(j, i, v) signifies that node j has heard a message COMMIT(i, v).

Reliable Broadcast Protocol (Bhandari & Vaidya, 2005)

- Initially, the source s does a local broadcast of v.
- Each node $i \in N(s)$ commits to the first value it receives from s and does a one-time broadcast of COMMIT(i, v).
- The following protocol is executed by each node j (including those nodes in the previous two steps):
  - On receipt of a COMMIT(i, v) message from a neighbor i, j records the message and broadcasts HEARD(j, i, v).
  - On receipt of a HEARD(j', i, v), j records this message.
  - Upon receiving COMMIT or HEARD messages that 1) claim v as the correct value and 2) are received along at least $t + 1$ node disjoint paths that all lie within a single neighborhood, then node j commits to v and does a one time broadcast of COMMIT(j, v).

Proving that this protocol is correct is non-trivial and we refer the reader to [89] for details. To briefly summarize, the proof in [89] works by showing that for each node p in $PN(a, b) - N(a, b)$, there exist $2t + 1$ paths $P_1, ..., P_{2t+1}$ belonging to a single neighborhood $N(a, b + r + 1)$, each having one of the forms listed below:

- $P_i = (q, p)$ which is a one-hop path $q \rightarrow p$ or
- $P_i = (q, q', p)$ which is a two-hop path $q \rightarrow q' \rightarrow p$

where q, q', p are distinct nodes and q, q' lie in a single neighborhood $N(a, b + r + 1)$, and $q \in N(a, b)$ where, critically, nodes in $N(a, b)$ have committed to the correct message. The existence of these $2t+1$ paths, and the fact that each broadcast neighborhood has at most $t < (r/2)(2r + 1)$ Byzantine faults, is sufficient to prove that reliable broadcast is achieved. Figure 3 illustrates for a node p the set of nodes A_p where one-hop paths originate, and the sets of nodes which correspond to the two-hop paths of the form $q \rightarrow q' \rightarrow p$ where $q \in B_p$ and $q' \in B'_p$.

Crash Failures and the Euclidean Metric: The authors also consider the fault model where affected nodes do not behave maliciously, but instead suffer a permanent crash failure. While an adversary can select the location and number of failures, this scenario is far more benign and the authors demonstrate that reliable broadcast is possible for $t < r(2r + 1)$ crashes per neighborhood. As we have seen, in the Byzantine fault model, completeness is demonstrated by proving $2t + 1$ connectedness between an uncommitted node and those nodes that have committed to a value. Here, in the crash failure model, when an uncommitted node receives a value v, this value must be the correct one. Therefore, the issue is no longer about majority voting, but simply about reachability and it is this simplification that allows for twice

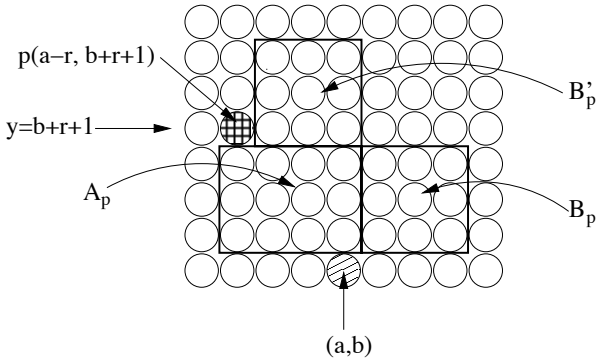

Fig. 3. An illustration of the sets A_p, B_p, and B'_p where $r = 3$ and $a, b = 0$. Node p is located at position $(a - r, b + r + 1)$. A two-hop path starts in B_p and passes through B' to reach p

as many faults to be tolerated. Finally, the authors translate their results into the L2 (Euclidean) metric and show that $t < 0.24\pi r^2$ Byzantine faults or $t < 0.48\pi r^2$ crash failures can be tolerated.

Koo et al. [91]

The results we have examined so far hold only in the case where the faulty nodes obey a broadcast schedule. Clearly, if such a constraint is abandoned completely, the adversary can simply execute a denial-of-service attack by having all faulty nodes broadcast continuously. However, an interesting question is whether broadcast can be achieved if the adversary is allowed to cause a limited number of message collisions. Another assumption upon which the previous results have relied is that faulty nodes cannot impersonate other nodes. Recall that, in order to commit, a correct node must receive messages from *distinct* nodes in a single neighborhood. Again, unlimited address spoofing makes reliable broadcast impossible to achieve; however, one might hypothesize that limited address spoofing might be tolerable. These are the issues dealt with by Koo *et al*. [91].

A Faulty Dealer with Collisions, Spoofing by Faulty Nodes: In the works we have surveyed so far, it has been assumed that the dealer is honest. In the absence of message collisions, the dealer cannot cause confusion over what value is being broadcast. This is because all nodes in the neighborhood $N(d)$ will immediately commit to the first value they hear from d; whether d is controlled by the adversary is of no consequence. However, now consider the following situation in which message collisions can occur:

1) The adversary uses a Byzantine node at location $(0, r + 1)$ to broadcast a null value as d broadcasts v.
2) The adversary then causes a node at location $(0, -r)$ to broadcast a null value as d broadcasts v' where $v' \neq v$.

In step 1, all nodes in the top r rows of $N(d)$ will receive only noise due to a message collision. However, all nodes in the bottom $r+1$ rows of $N(d)$ will commit to v. In the second step, the reverse will happen with the nodes in the top half of $N(d)$ (inclusive) committing to v'. Due to the collisions, the nodes in $N(d)$ will be unaware that conflicting values have been initiated by the dealer and, therefore, reliable broadcast will not be achieved.

The challenge of address spoofing is fairly obvious. Under the protocol by Bhandari and Vaidya, a node p will take action only upon receiving $t+1$ messages with identical values from distinct nodes in a single neighborhood. Therefore, a faulty node may simply claim to be $t + 1$ other nodes, all within a single neighborhood, and trick p.

A Robust Protocol: We assume that the total number of message collisions and address spoofing instances afforded to *each faulty node* is n_c and n_s, respectively. The protocol of [91] can be broken into two parts as follows:

1) **Agreement in $N(d)$:** all nodes in $N(d)$ will come to an agreement prior to propagating the message to other nodes.
2) **Robust Propagation:** the value agreed upon in the first step will be propagated in such a way that n_c message collisions and n_s spoofing attempts by the adversary do not prevent the correct nodes from committing to the correct value indefinitely.

To achieve agreement in $N(d)$, all nodes in the neighborhood emulate a peer-to-peer protocol appearing in [74], [92]. However, this protocol assumes that the network is fully connected. In contrast, the nodes in $N(d)$ are not fully connected (in a single hop sense) due to the limited broadcast radius. However, the authors show that it is possible for a node $i \in N(d)$ to set up a *virtual* connection to all other nodes in $N(d)$ by initiating the protocol B_{robust} below in the case that spoofing is allowed and nodes can detect collisions. In the following, B_{norm} is the protocol of Bhandari and Vaidya presented in [89].

Robust Protocol B_{robust}

1) In each case that node i would send a message m in B_{norm}, i sends m for a total of $t(n_c + n_s) + 1$ times.
2) In each case that i would perform an action when receiving a message m from node j in B_{norm}, i performs this action only when it receives the message $t \cdot n_s + 1$ times from j for the first time.

By using an almost identical argument to that given in [89], it can be shown that by having every node in $N(d)$ execute the above protocol, all nodes in $N(d)$ will come to agreement upon v within a bounded amount of time. Intuitively, if a node i has received at least $t \cdot n_s + 1$ instances of the same value v, then, given that there are at most t faulty nodes in $N(i)$, node i is assured that v is the correct value. Similarly, by sending the value $t(n_c + n_s) + 1$ times, node i guarantees that a listening neighbor will receive the v at least $t \cdot n_s + 1$ identical copies of the value, and thus can determine that v is legitimate. Now, for every instance that the peer-to-peer protocol would dictate that a peer send a message to all other peers, a node i in $N(d)$ uses the above robust protocol. Now that all nodes in $N(d)$ can agree upon a value, it remains to be seen how this value is propagated throughout the network. However, this also follows from B_{robust} using the same proof structure as was used to prove correctness of the protocol of B_{norm}. The authors also show how, with slight modifications to B_{robust}, reliable broadcast is feasible when nodes cannot detect collisions.

There is one more important point to be made regarding the protocol described in [91]. Note that the adversary can force a correct node to send $\Omega(t(n_c + n_s))$ messages. However, a faulty node sends only $\Omega(n_c+n_s)$ before its malicious actions are exhausted. Therefore, correct nodes are assumed to be able to send more messages than faulty nodes. In other words, correct nodes are required to possess more energy than faulty nodes. This is a rather strong assumption and it has motivated research into how to reduce the cost of reliable broadcast.

Bertier et al. [93], [94]

Until this point, the results we have surveyed have primarily focused on the *feasibility* of reliable broadcast in the context of obtaining upper and lower bounds on the number of faults. However, as we saw in the work by Koo *et al*. [91], the addition of message collisions to the grid model motivates the consideration of energy efficiency. In [93], Bertier *et al*. extend their initial findings in [94] and focus on this issue by examining the number of messages both the correct nodes and adversarial nodes can send. In particular, a correct node is allowed to send m messages while a Byzantine node may send m_f messages; these message budgets are motivated by the energy contraints of sensor networks. The costs, in terms of message efficiency, for achieving reliable broadcast are then expressed in terms of these budgets.

Reducing Transmission Costs: The authors begin by analyzing message efficiency when both m and m_f are known. It should be expected that for "small" values of m (relative to $t \cdot m_f$) reliable broadcast is impossible. Indeed, the authors show that for $m < m_0$, where $m_0 = \lceil \frac{2 \cdot t \cdot m_f + 1}{r(2r+1)-t} \rceil$, reliable broadcast cannot be achieved. In terms of an upper bound, the following protocol is presented:

Reliable Broadcast (Bertier *et al*., 2010)

1) The dealer d initiates a one-time action of broadcasting the value $2 \cdot t \cdot m_f + 1$ times and each node in $N(d)$ accepts the majority value.
2) For any node $p \notin N(s)$, if p receives a value, then it broadcasts this value $\frac{2\,t\,m_f+1}{\lceil (r(2r+1)-t)/2 \rceil}$ times. Node p accepts a value it receives at least $t \cdot m_f + 1$ times.

The authors go on to show that for $m \geq 2m_0$, this protocol achieves reliable broadcast for $t < r(2r + 1)$. Importantly, this bound on t exceeds that of the previous works we have examined by a factor of 2 in the Byzantine case. This increased tolerance is due to the fact that Byzantine faults can be overcome when correct nodes possess a higher message budget than faulty nodes as this allows for the correct message to be received in the majority. This result clearly improves on the message efficiency as it reduces the number of sent messages by a factor of $(1/2)(r(2r+1)-t)$ in comparison to the protocol by Koo *et al*. [91]. It is unknown whether reliable broadcast is feasible for the intermediate case where $m \in [m_0, 2m_0]$. An interesting additional case to consider is when m is still known but is heterogeneous; that is, when the message budget differs between correct nodes. Here, the authors demonstrate a modified protocol that guarantees reliable broadcast so long as $\Theta(r^3)$ correct nodes have a budget of $m' = \frac{2 \cdot t \cdot m_f + 1}{\lceil (r(2r+1)-t)/2 \rceil}$ and the remaining correct nodes have a budget of m_0. Notably, $m' \leq 2m_0$, implying that the heterogeneous case offers a strict improvement over the situation where m is homogeneous; this improvement arises due to a more careful analysis of how the correct value is propagated throughout the network.

When m_f is Unknown - A Reactive Protocol: Recall that in [91], Koo *et al*. discussed the problematic aspect of their protocol that requires correct nodes to incur a higher cost relative to the adversary. To address this issue, Koo *et al*. suggested that a reactive strategy, rather than a proactive strategy, might overcome this discrepancy. That is, rather than having correct nodes send redundantly using the protocol of [91], it might be possible to have nodes resend values only when malicious interference is detected. Such a strategy would conserve energy by not engaging in otherwise wasteful redundant transmissions.

Here, Bertier *et al*. devise a reactive strategy that relies on error-detecting codes. By employing results from *All-Unidirectional Error-Detecting* codes, the authors show how to encode messages such that a receiver can detect when a message has been tampered with by the adversary with high probability. Combining this feature with the protocol of Bhandari and Vaidya [86], the authors devise a reactive protocol that achieves reliable broadcast with probability at least $1-1/n$, where n is the network size, and requires no more than $m = 2(t \cdot m_f + 1)(2 \log n + \log t + \log m_{\max})(k + 2 + 2 \log k)$ transmissions per correct node where $t < (r/2)(2r + 1)$ and $m_{\max}$ is a loose upper bound on the budget of a faulty node.

This protocol incurs a higher message complexity than the bounded collision case due to the fact that m_f is largely unknown and collision detection is not employed. Of course, m_f is not *completely* unknown to the nodes as $m_{\max}$ is known to the correct nodes and is a parameter in the coding of the messages. However, $m_{\max}$ is allowed to exceed m_f by orders of magnitude and, noting the logarithmic term in the protocol's cost, the impact of this is fairly limited.

Finally, it is important to note that while all of these results improve on the tranmission costs, they do not address listening costs. In fact, the listening costs of the protocols presented by Bertier *et al*. appear to be asymptotically equal to those presented by Bhandari and Vaidya [86]. As discussed earlier, listening costs are significant and, therefore, reducing the amount of time a correct node must listen to the communication medium is an equally important problem.

King et al. [95], [96]

As we have seen, the issue of energy-efficient reliable broadcast was first raised in the work by Koo *et al*. [91] and pursued by Bertier *et al*. [93]. The work by King *et al*. [95], [96] addresses energy efficiency in the original model analyzed in [78], [86] where all nodes, both correct and faulty, obey a global broadcast schedule. Therefore, the main obstacle to reliable broadcast is again misinformation and routing failures caused by Byzantine nodes. In contrast to previous results, King *et al*. assume that the adversary is computationally bounded; in particular, they assume that the adversary cannot easily invert a secure hash function. As we will see, this

property essentially allows for Byzantine faults to be treated as crash failures. We first begin by describing a key subproblem.

The Bad Santa Problem: The authors begin by analyzing the *Bad Santa Problem* which is described as follows. A child is presented with n boxes, one after another. When presented with each box, the child must immediately decide whether or not to open it. If the child decides not to open a box, he is never allowed to revisit it. At least half the boxes have presents in them, but the decision as to which boxes have presents is made by an adversarial Santa who wants the child to open as many empty boxes as possible. The child must obtain a present, while opening the smallest number of boxes in expectation.

The authors apply this problem to a single-hop sensor network scenario where a receiver aims to awaken from the energy-efficient sleep state and listen to the communication channel. Each box in the Bad Santa problem corresponds to a time slot and the cost of opening of a box corresponds to the cost of listening. There is assumed to be a set of senders in the broadcast neighborhood of the receiver, each assigned their respective time slot by the global broadcast schedule. The adversary may select up to half of the senders to suffer a crash failure prior to the receiver making its choices about which boxes to open, and the receiver seeks to receive a message by listening to as few time slots as possible in expectation.

Importantly, an algorithm for solving the Bad Santa Problem must succeed with probability 1. Without this constraint, a simple random sampling algorithm is sufficient as this can provide a high probability guarantee that the child obtains a toy. The reason for insisting on a Las Vegas algorithm is the same as discussed in our examination of [66]. The authors show asymptotically tight upper and lower bounds for the Bad Santa problem. Surprisingly, the optimal expected number of opened boxes is $\Theta(\sqrt{n})$. An extension called the k-Stream Bad Santa Problem is also examined. In this case, $k+1$ streams of n boxes each are presented consecutively to the child who must obtain a toy by the last stream; the order of the full and empty boxes can differ arbitrarily from stream to stream. Here, the optimal expected number of opened boxes is $O(\log^{(k)}(n)+k)$ and $\Omega(\log^{(2k)} n)$. In particular, for $k=\Theta(\log^* n)$, the expected number of opened boxes is $O(\log^* n)$.

Reliable Broadcast Protocols: Using the solution to the single stream Bad Santa Problem, the authors provide a reliable broadcast protocol tolerating Byzantine faults that runs in two stages. In the first stage, the source propagates a fingerprint $f(m)$ of the message m it wants to broadcast. This fingerprint is assumed to be of size at least $\lg^2 |m|$ bits and generated using a secure hash function. Propagation of $f(m)$ is again done using the algorithm in [89] and every node is assured of committing to the correct fingerprint.

In the second stage, the source broadcasts m at a pre-specified time slot t_{start} and all correct nodes in $N(0,0)$ are assumed to receive m from the source and commit internally. Each node $q(x',y') \in N(0,0)$ then broadcasts its committal to m over the next $2r$ consecutive rounds. Recall that the proof of completeness given in [89] is constructive. That is, the set of nodes to which p should listen is specified exactly; call this set G_p. Node p selects nodes to listen to from G_p using the solution to the Bad Santa Problem and, upon receiving any message m', checks $f(m')$ against the fingerprint to which it committed in the first stage. If they match, p commits to m' internally and executes the broadcast instructions mentioned previously.

Let s be the number of times the adversary can create an input x', apply a hash function f to x' and check for a match between the output fingerprint $f(x')$ and some other fingerprint for which the adversary is attempting to generate a collision. Assuming $t < \frac{r}{2}(2r+1)$, the proposed protocol achieves reliable broadcast with probability of failure $O(s/|m^{\log|m|}|)$ where $|m|$ is the number of bits in m. In the second stage of the protocol, each node is awake for only $\sqrt{n}$ time slots. Furthermore, over both stages of the protocol, each node sends $O(n\log^2|m| + \sqrt{n}|m|)$ bits and receives an expected $O(n\log^2|m| + \sqrt{n}|m|)$ bits. A similar protocol that employs the solution to the k-Stream Bad Santa Problem achieves similar guarantees when $t < (1-\epsilon)\frac{r}{2}(2r+1)$ for any small but constant $\epsilon > 0$. In this case, each node sends $O(n\log^2|m| + k|m|)$ bits and receives an expected $O(n\log^2|m| + (\log^{(k)n})|m|)$ bits. Similar results are given for the cases where nodes suffer crash failures.

There are two important points to note. First, nodes must be awake as normal for the first stage. However, the transmission of a fingerprint should require significantly smaller time slots for moderately sized messages or larger. The authors argue that the network could be set up to alternate between the first stage, where nodes are constantly awake, and the second stage, where nodes are achieving significant power savings. Second, in the latter protocol, nodes are sending more bits by a factor of k; however, the expected amount of times the full message must be received decreases by far more. Therefore, there is a significant overall savings in expectation. To contrast these results with previous work, note that under the previous algorithms for reliable broadcast [86], [89], each node 1) is awake for $(2t+1)=\Theta(n)$ time slots, 2) broadcasts $\Theta(|m|)$ bits; 3) receives $\Theta(|m|)$ bits in the fail-stop model; and 4) can be forced by the adversary to receive $\Theta(n|m|)$ bits in the Byzantine fault model. Of course, these results on reliable broadcast allow for a probability of failure, but the authors also note that $|m|$ need not be large to make the probability of finding a message with the same fingerprint very small. For example, if $|m| = 1$ kB, the probability of a collision is already less than 10^{-30}.

Therefore, these algorithms save substantially on the overall amount of time a node must be awake in order to achieve reliable broadcast. However, there are trade-offs: the latency of the protocol is increased due to the need for two-stages and the need for synchronization is even more important here due to the use of t_{start} as an initial time step in the protocol. Furthermore, while this work makes progress on the issue of energy-efficient reliable broadcast, it remains to be seen whether these results can be applied to situations where the adversary can cause message collisions or spoof correct nodes.

Alistarh et al. [97]

As discussed in Section II-C, cryptographic techniques can offer protection against certain attacks in the wireless

domain. In particular, cryptographic authentication can prevent malicious behavior where messages are corrupted or correct nodes are spoofed. However, because sensor networks are energy-starved, there has been significant research into *light-weight* authentication. Here, Alistarh *et al*. examine reliable broadcast in the absence of such cryptographic authentication. As we saw earlier, Koo *et al*. [91] also addressed this problem and Alistarh *et al*. adopt the grid model. However, the model of [91] assumes that the instances of spoofing and message collisions are known *a priori* while Alistarh *et al*. do not make this assumption.

Non-Cryptographic Authenticated Broadcast: The authors provide two non-authenticated reliable broadcast protocols called NEIGHBORWATCHRB and MULTIPATHRB. Both protocols rely on a critical subroutine called the 1HOP-PROTOCOL which, in turn, is based on the 2BIT-PROTOCOL and we describe both subroutines now. The 2BIT-PROTOCOL addresses the problem of a sender which wishes to transmit two bits to a receiver in the presence of message collisions and spoofing attacks. This protocol is similar to the communication protocol between Alice and Bob used by Gilbert *et al*. [55]. Again, Alice and Bob can detect message collisions and the unforgeability of silence allows the players to detect interference and indicate this in a veto round. If a veto message is sent, then the 2BIT-PROTOCOL fails. By stringing together executions of the 2BIT-PROTOCOL, the entire message can be sent; this is the 1HOP-PROTOCOL as it accomplishes communication within a local broadcast neighborhood.

The first protocol NEIGHBORWATCHRB uses the 1HOP-PROTOCOL to achieve a multi-hop protocol by partitioning the grid into $r/2 \times r/2$ squares. Each node belongs to exactly one square and is aware of the other members of the same square. The devices in each square act in concert, either all broadcasting or all listening in unison; in this sense, a square becomes the new atomic unit in the grid and the authors refer to these as *meta-nodes*. Each meta-node is assigned a broadcasting time slot according to some global schedule. When a meta-node is scheduled, it executes a step of the 1HOP-PROTOCOL with neighboring meta-nodes; otherwise, for a significant amount of the time it is listening to neighboring meta-nodes. An interesting property of this protocol is that the fault-tolerance is fairly high; specifically, if $t < \lceil r/2 \rceil^2$, NEIGHBORWATCHRB achieves broadcast with an asymptotically optimal running time of $O(\beta D + \log |\Sigma|)$ where Σ is the set of all possible messages. The second protocol MULTIPATHRB achieves the optimal fault-tolerance $t < (r/2)(2r+1)$ by combining the 1HOP-PROTOCOL with the reliable broadcast protocol of Bhandari and Vaidya [89]. In contrast to the proactive protocol by Koo *et al*. [91], MULTIPATHRB is reactive which avoids the waste of unnecessary retransmissions.

Finally, the authors conduct simulations to evaluate their protocols under more realistic conditions. Notably, instead of deployment on the grid, nodes are placed uniformly at random in the plane. Both NEIGHBORWATCHRB and MULTIPATHRB exhibit promising performance. Not surprisingly, as node density increases, tolerance to both crash failures and Byzantine faults increases. Although the analysis of NEIGHBORWATCHRB does not admit optimal fault tolerance, this protocol achieves superior tolerance in practice over MULTIPATHRB, for high node density. Moreover, and in line with the theoretical analysis, NEIGHBORWATCHRB achieves significantly lower latency. Given the previous works we have surveyed, this paper is one of the few where the authors take significant steps to bridge the gap between their theoretical results and the practical performance of their protocols.

Two final comments are in order. Notably, while non-cryptographic authentication is achieved, both MULTIPATHRB and NEIGHBORWATCHRB require devices to spend significant time in the receive state so that they may detect interference or veto messages. Another issue that deserves some discussion is that of simulation itself, which allows for a evaluation of protocols while avoiding the typically high costs of deploying a sizable real-world test-bed. Here, the authors use the WSNet Worldsense simulator [98]. There are also a host of other popular simulators such as ns-3 [99], QualNet [100], OPNET Modeler [101], OMNet [102] which can be used in tandem with Castalia [103] and many others. We refer the interested reader to the surveys of Korkalainen *et al*. [104] and Imran *et al*. [105] for more information on available sensor network simulators.

Bhandari and Vaidya [106], [107]

Much of the previous work has focused on the locally bounded fault scenario in the grid model. Here in [106], the authors examine the probabilistic scenario in both the grid and in a network where nodes are distributed uniformly at random in the plane. In both cases, the network size is bounded by $\sqrt{n} \times \sqrt{n}$. The aim of this analysis is to identify the maximum probability of failure $p_{\max}$ along with the critical node degree d_{critical} for which reliable broadcast remains feasible.

Probabilistic Failures: As the authors are examining a model of random failures, the derived results are presented with probabilistic guarantees. In particular, reliable broadcast is asymptotically achievable if $Pr[\text{reliable broadcast succeeds}] \to 1$ as $n \to \infty$; alternatively, reliable broadcast is said to occur with high probability (w.h.p.).

The first major result is that $p_{\max} < 1/2$ is necessary and, letting $d_{\min} = O(1)$ denote the minimum node degree, $d_{\text{critical}} = O(d_{\min} + \frac{\ln n}{\ln \frac{1}{2p} + \ln \frac{1}{2(1-p)}})$. The second major result is the existence of a lower bound on the critical degree. The result centers around the simple observation that, if a correct node p has a neighborhood containing at least a $1/2$-fraction of faulty nodes, then p will commit to an incorrect value with probability at least $1/2$. By leveraging this observation, the authors prove that reliable broadcast is not asymptotically achievable unless the node degree is $\Omega(\frac{\ln n}{\ln \frac{1}{2p} + \ln \frac{1}{2(1-p)}})$. Therefore, the critical degree is $d_{\text{critical}} = \Theta(\frac{\ln n}{\ln \frac{1}{2p} + \ln \frac{1}{2(1-p)}})$.

Extensions - A Random Network and Crash Failures: The authors also examine the scenario where nodes are distributed uniformly at random in an $\sqrt{N} \times \sqrt{N}$ torus. Unlike the grid model, where node degree is uniform, the random network model allows for nodes with varying degree. Consequently, the notion of an average degree is employed and the *critical average degree* is the average degree corresponding to the minimum transmission range necessary for reliable broadcast

to be asymptotically achievable. The critical average degree is shown to be $O(\frac{\ln n}{\frac{1}{2}-p+\frac{1}{2}\ln\frac{1}{2(1-p)}})$. Finally, the authors analyze the scenario where nodes suffer crash failures and the critical degree is shown to be $\Theta(\frac{\ln n}{\ln \frac{1}{p}})$.

Summary of Results and Discussion

Table II summarizes key aspects and the main findings of the models examined in this section. There is a clear progression of results in this area and we take some time to discuss these now as well as discuss possible future research directions.

As mentioned in Section II-D, the work on reliable broadcast can be categorized by the fault model adopted: the probabilistic scenario or the locally-bounded scenario. Starting with the probabilistic scenario, we see initial work with single-hop networks by Berman [75] with transient failures, then work by Pelc and Peleg [77] (in tree topologies) with permanent Byzantine faults, and then finally the transition to multi-hop networks under the grid model by Bhandari and Vaidya [106]. The first two works share similarities in the propagation of the message via groups of devices for $p < 1/2$ and correctness is proved, in part, by using Chernof bounds. Curiously, this has been a technique adopted by several Byzantine fault-tolerant peer-to-peer systems, also under the probabilistic scenario. This is an example of a technique that spans both the wireless and wired domains of ad-hoc networks and we refer the interested reader to the excellent survey paper by Urdaneta *et al*. [108]. The third work is also closely related; while it does not rely as explicitly on groups of devices, such sets do exist and correctness again relies, in part, on Chernoff bounds, with a focus in the analysis on $n \to \infty$.

The body of work on the locally-bounded scenario is larger and primarily centers about the grid model, although arbitrary topologies are considered by Pelc and Peleg [87] and Ichimura and Shigeno [88]. We can see that initial work focuses on feasibility; specifically, establishing upper and lower bounds on t for which reliable broadcast can be achieved. Subsequently, the aspects of jamming, communication complexity, latency, and energy efficiency are addressed. In each of these cases, we see progress being made in tackling increasingly difficult adversarial models. The initial work of Koo *et al*. [91] shows that reliable broadcast is feasible with optimal tolerance so long as the instances of message collisions and spoofing are bounded and known ahead of time. Later, Bertier *et al*. [93] and Alistarh *et al*. [97] push things further by achieving reliable broadcast even when the adversary's budget is unknown. In contrast to the potentially inefficient proactive protocol of Koo *et al*. [91], both of these results provide reactive strategies that deliver the message as soon as the adversary terminates its interference; consequently, the message complexity is improved. Furthermore, Alistarh *et al*. [97] provide bounds on latency which, interestingly, bear resemblance to the latency results of Pelc and Peleg [77]. In terms of energy efficiency, again receive state costs seem to be mostly ignored with the exception of King *et al*. [95], [96].

In contrast to the works surveyed in Section III, the unreliability of wireless communication is largely ignored by these works and Bhandari and Vaidya address this practical issue in [109]. Along the same lines, it may be of interest to investigate reliable broadcast outside of the grid model and, even, outside of arbitrary graph topologies. For instance, one might examine reliable broadcast with adversarial interference under the SINR model. The experimentation of Alistarh *et al*. [97] demonstrates that theoretical analyses do not always match up with performance and it might be of interest to know whether more realistic interference and transmission propagation models can reduce this discrepancy.

Another interesting aspect that has not received much attention is that of mobility in the context of Byzantine fault-tolerant reliable broadcast. Several real-world sensor networks are deployed in scenarios where devices are not static such as monitoring wildlife on land [110] or collecting data on ocean currents [111] where devices may be mobile within a localized area. In some sense, the grid model accounts for mobility as a fault may be interpreted as the absence of a node as it moves within a local area. However, this aspect has not been examined explicitly and more precise models can surely be examined. For example, mobility might obey migration patterns or be dictated by predictable environmental phenomenon, in contrast to a worst case selection of faulty locations chosen by an adversary.

V. Conclusion

We have surveyed a number of results that deal with the basic task of communication in the presence of challenging adversarial interference. Much of the initial work has focused on feasibility with subsequent endeavors focusing on tackling increasingly powerful adversaries and bounding important metrics such as message complexity, latency, and energy-efficiency under a number of different sensor network models. By the sheer number of recent results, this is clearly a field of research that continues to evolve and there remain a number of interesting algorithmic problems waiting to be addressed. On the other hand, given the breadth of the results we have surveyed, future work in this area may benefit by an increased validation of the models being employed as the schism between theory and practice remains a largely unchallenged omission.

Acknowledgments

This work was supported in part by the Natural Science and Engineering Council of Canada (NSERC) under its Discovery program, and in part by the WCU (World Class University) program through the Korea Science and Engineering Foundation funded by the Ministry of Education, Science and Technology (Project No. R31-2008-000-10100-0).

References

[1] J. P. Walters, Z. Liang, W. Shi, and V. Chaudhary, "Chapter 17: Wireless Sensor Network Security: A Survey," in *Security in Distributed, Grid, Mobile, and Pervasive Computing*, Y. Xiao, Ed. Auerbach Publications, 2007.

[2] C. Karlof and D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," in *Proc. 1^{st} IEEE International Workshop on Sensor Network Protocols and Applications*, 2002, pp. 113–127.

[3] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor network security: A Survey," *IEEE Commun. Surveys & Tutorials*, vol. 11, no. 2, pp. 52–73, 2009.

TABLE II
A COMPARISON OF THE DIFFERENT RELIABLE BROADCAST RESULTS EXAMINED IN SECTION IV.

Result	Adversary Type	Spoofing	Collision Detection	Main Results & Advantages/Disadvantages
Berman *et al*. [75]	Byzantine probabilistic scenario	No	n/a	Feasibility of reliable broadcast when $p < 1/2$ with $O(\log n)$ latency. Analysis pertains to single-hop networks only. Extension to multi-hop networks is possible, but for constrained topologies.
Pelc & Peleg [77]	Byzantine probabilistic scenario	No	n/a	Reliable broadcast if and only if $p < 1/2$ and with $O(OPT \cdot \log n)$ latency. Propagation along tree topology. Deals with transient Byzantine and transmission faults rather than permanent faults. Transient transmission faults may be considered as a message collision without collision detection; such faults occur randomly.
Koo [78]	Byzantine locally-bounded scenario	No	n/a	Pioneering work on reliable broadcast in locally-bounded scenario for multi-hop networks using the grid model. Impossibility result when $t \geq (r/2)(2r+1)$ and upper bound of $t < \frac{r}{2}(r+\sqrt{\frac{r}{2}}+1)$. However, the model does not admit jamming or spoofing, and algorithmic results do not address latency or energy efficiency.
Pelc & Peleg [87]	Byzantine locally-bounded scenario	No	n/a	Examines fault tolerance of Koo's [78] reliable broadcast protocol, called CPA, in arbitrary graph topologies. Derive two graph parameters $X(G)$ and $LPC(G)$ which are useful in characterizing feasibility. Demonstrate CPA is not always the best protocol. The range of $t \in [\lfloor X(G)/2 \rfloor + 1, LPC(G)]$ is problematic.
Ichimura & Shigeno [88]	Byzantine locally-bounded scenario	No	n/a	Introduce a new graph parameter $X'(G)$ that gives another characterization of when CPA achieves reliable broadcast in an arbitrary graph. Characterization is incomplete for the range of $t \in [\lfloor X'(G/2) \rfloor + 1, X'(G)]$.
Bhandari & Vaidya [86], [89]	Byzantine locally-bounded scenario	No	n/a	A reliable broadcast protocol that tolerates $t < (r/2)(2r+1)$ is demonstrated; this is tight with the lower bound of Koo [78]. Technical report [89] contains a constructive and elegant proof. Authors also investigate crash failures. The model used still does not admit jamming or spoofing, and algorithmic results do not address latency or energy efficiency.
Koo *et al*. [91]	Reactive Jammer, Byzantine locally-bounded scenario	Yes	Results with and without collision detection	Demonstrate a proactive reliable broadcast protocol that tolerates a bounded number of message collisions and spoofing instances. However, the bound on the number of message collisions and spoofing instances must be known *a priori*. Correct nodes require more energy than adversarial nodes.
Bertier *et al*. [93], [94]	Reactive Jammer, Byzantine locally-bounded scenario	Yes	No	Examine feasibility and communication complexity of reliable broadcast when both correct and adversarial nodes have a bounded message budget. The demonstrated protocol allows correct nodes to have a smaller budget than that required by the protocol given by Koo *et al*. [91]. Listening costs are not addressed.
King *et al*. [95], [96]	Byzantine locally-bounded scenario	No	n/a	Look at energy-efficient reliable broadcast. Achieve quadratic improvement in energy efficiency over previous protocols with optimal tolerance, and achieves at least an exponential increase in energy efficiency for $t < (1-\epsilon)(r/2)(2r-1)$. The model used does not incorporate jamming or spoofing, and latency is not fully examined.
Alistarh *et al*. [97]	Reactive Jammer, Byzantine locally-bounded scenario	Yes	Yes	Non-cryptographic authentication in the presence of jamming and two reliable broadcast protocols. Simulation work is provided. Listening costs are not addressed.
Bhandari & Vaidya [106], [107]	Byzantine probabilistic scenario	No	n/a	Byzantine probabilistic scenario in multi-hop networks using the grid model. Feasibility result when $p < 1/2$ and the authors derive bounds on the critical degree and critical average degree. However, the model does not admit jamming or spoofing, and algorithmic results do not address latency or energy efficiency.

[4] J. Polastre, R. Szewczyk, and D. Culler, "Telos: Enabling Ultra-Low Power Wireless Research," in *Proc. 4^{th} International Symposium on Information Processing in Sensor Networks*, 2005.

[5] Crossbow, MICAz Wireless Measurement System. http://courses.ece.ubc.ca/494/files/MICAz_Datasheet.pdf.

[6] ——, Mica2 Wireless Measurement System. https://www.eol.ucar.edu/rtf/facilities/isa/internal/CrossBow/DataSheets/mica2.pdf.

[7] A. http://www.alertsystems.org/.

[8] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey," *Computer Networks*, vol. 38, pp. 393–422, 2002.

[9] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 52, no. 12, pp. 2292–2330, 2008.

[10] M. Tubaishat and S. Madria, "Sensor Networks: An Overview," *IEEE Potentials*, vol. 22, no. 2, 2003.

[11] D. Boyle and T. Newe, "Security Protocols for Use with Wireless Sensor Networks: A Survey of Security Architectures," in *Proc. Third International Conference on Wireless and Mobile Communications*, 2007, p. 54.

[12] H. Alwan and A. Agarwal, "A Survey on Fault Tolerant Routing Techniques in Wireless Sensor Networks," in *Proc. International Conference on Sensor Technologies and Applications (SENSORCOMM)*, 2009, pp. 366–371.

[13] Y. Wang, G. Attebury, and B. Ramamurthy, "A Survey of Security Issues in Wireless Sensor Networks," *IEEE Commun. Surveys & Tutorials*, vol. 8, pp. 2–23, 2006.

[14] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and G. Pantziou, "A Survey on Jamming Attacks and Countermeasures in WSNs," *IEEE Commun. Surveys & Tutorials*, vol. 11, no. 4, pp. 42–56, 2009.

[15] S. Schmid1 and R. Wattenhofer, "Algorithmic Models for Sensor Networks," in *Proc. 20^{th} IEEE International Parallel and Distributed Processing Symposium*, 2006, p. 160.

[16] P. von Rickenbach, R. Wattenhofer, and A. Zollinger, "Algorithmic Models of Interference in Wireless Ad Hoc and Sensor Networks," *IEEE/ACM Trans. Netw.*, vol. 17, pp. 172–185, February 2009.

[17] S. Dolev, S. Gilbert, R. Guerraoui, C. Newport, F. Kuhn, N. Lynch, and D. R. Kowalski, "Reliable Distributed Computing on Unreliable Radio Channels (Extended Abstract)," in *Proc. 2009 MobiHoc S^3 Workshop on MobiHoc S^3*, 2009, pp. 1–4.

[18] B. Raman and K. Chebrolu, "Censor Networks: A Critique of "Sensor Networks" from a Systems Perspective," *Computer Communication Review*, vol. 38, pp. 75–78, 2008.

[19] G. Chockler, M. Demirbas, S. Gilbert, N. Lynch, C. Newport, T. Nolte, "Reconciling the Theory and Practice of Unreliable Wireless Broadcast," in *Proc. International Workshop on Assurance in Distributed Systems and Networks (ADSN)*, 2005, pp. 42–48.

[20] M. H. Manshaei, Q. Zhu, T. Alpcan, T. Bąsar, and J.-P. Hubaux, "Game Theory Meets Network Security and Privacy," Ecole Polytechnique Fédérale de Lausanne (EPFL), Tech. Rep. EPFL-REPORT-151965, September 2010.

[21] G. Anastasi, A. Falchi, A. Passarella, M. Conti, and E. Gregori, "Performance Measurements of Motes Sensor Networks," in *Proc. 7^{th} ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, 2004, pp. 174–181.

[22] B. Cody-Kenny, D. Guerin, D. Ennis, R. S. Carbajo, M. Huggard, and C. M. Goldrick, "Performance Evaluation of the 6LoWPAN Protocol on MICAz and TelosB Motes," in *Proc. 4^{th} ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks*, 2009, pp. 25–30.

[23] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *Proc. 33^{rd} Hawaii International Conference on System Sciences (HICSS)*, 2000, pp. 3005–3014.

[24] S. Kumar, V. S. Raghavan, and J. Deng, "Medium Access Control Protocols for Ad Hoc Wireless Networks: A Survey," *Ad Hoc Networks*, vol. 4, pp. 326–358, 2006.

[25] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," in *Proc. Conference on Computer Communications (INFOCOM)*, 2002, pp. 1567–1576.

[26] Y. Li, W. Ye, and J. Heidemann, "Energy and Latency Control in Low Duty Cycle MAC Protocols," in *Proc. IEEE Wireless Communications and Networking Conference*, 2005, pp. 676–682.

[27] K. Whitehouse, A. Woo, F. Jiang, J. Polastre, and D. Culler, "Exploiting the Capture Effect for Collision Detection and Recovery," in *Proc. 2^{nd} IEEE Workshop on Embedded Networked Sensors*, 2005, pp. 45–52.

[28] I. Ramachandran and S. Roy, "Clear Channel Assessment in Energy-Constrained Wideband Wireless Networks," *IEEE Wireless Commun.*, vol. 14, no. 3, pp. 70–78, 2007.

[29] K. Srinivasan and P. Levis, "RSSI is Under Appreciated," in *Proc. Third Workshop on Embedded Networked Sensors (EmNets)*, 2006.

[30] F. Kuhn, N. Lynch, C. Newport, R. Oshman, and A. Richa, "Broadcasting in Unreliable Radio Networks," in *Proceeding of the 29^{th} Symposium on Principles of Distributed Computing (PODC)*, 2010, pp. 336–345.

[31] G. Chockler, M. Demirbas, S. Gilbert, C. Newport, and T. Nolte, "Consensus and Collision Detectors in Wireless Ad Hoc Networks," in *Proc. 24^{th} Annual ACM Symposium on Principles of Distributed Computing (PODC)*, 2005, pp. 197 – 206.

[32] D. Liu and P. Ning, "Multi-Level μTESLA: Broadcast Authentication for Distributed Sensor Networks," *ACM Transactions in Embedded Computing Systems*, vol. 3, pp. 800–836, 2004.

[33] R. Watro, D. Kong, S. Cuti, C. Gariner, C. Lynn, and P. Kruus, "TinyPK: Securing Sensor Networks with Public Key Technology," in *Proc. 2^{nd} ACM workshop on Security of Ad Hoc and Sensor Networks (SASN)*, 2004, pp. 59–64.

[34] Y. W. Law, J. Doumen, and P. Hartel, "Survey and Benchmark of Block Ciphers for Wireless Sensor Networks," *ACM Transactions on Sensor Networks*, vol. 2, no. 1, pp. 65–93, 2006.

[35] C. Karlof, N. Sastry, and D. Wagner, "TinySec: A Link Layer Security Architecture for Wireless Sensor Networks," in *Proc. 2^{nd} International Conference on Embedded Networked Sensor Systems (SenSys)*, 2004, pp. 162–175.

[36] Y. Xiao, K. V. Rayi, B. Sun, X. Du, F. Hu, and M. Galloway, "A Survey of Key Management Schemes in Wireless Sensor Networks," *Computer Communications*, vol. 30, pp. 2314–2341, 2007.

[37] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The Feasibility of Launching and Detecting Jamming Attacks in Wireless Networks," in *Proc. 6^{th} ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2005, pp. 46–57.

[38] A. D. Wood and J. A. Stankovic, "Denial of Service in Sensor Networks," *IEEE Computer*, vol. 35, no. 10, pp. 54–62, 2002.

[39] M. Li, I. Koutsopoulos, and R. Poovendran, "Optimal Jamming Attacks and Network Defense Policies in Wireless Sensor Networks," in *Proc. 26^{th} IEEE International Conference on Computer Communications (INFOCOM)*, 2007, pp. 1307–1315.

[40] C. Pöpper, M. Strasser, and S. Čapkun, "Jamming-Resistant Broadcast Communication Without Shared Keys," in *Proc. 18^{th} USENIX Security Symposium*, 2009.

[41] V. Navda, A. Bohra, and S. Ganguly, "Using Channel Hopping to Increase 802.11 Resilience to Jamming Attacks," in *IEEE INFOCOM Mini-Symposium*, 2007, pp. 2526–2530.

[42] X. Liu, G. Noubir, R. Sundaram, and S. Tan, "SPREAD: Foiling Smart Jammers using Multi-layer Agility," in *Proc. Conference on Computer Communications (INFOCOM)*, 2007, pp. 2536–2540.

[43] G. Alnifie and R. Simon, "A Multi-Channel Defense Against Jamming Attacks in Wireless Sensor Networks," in *Proc. 3rd ACM workshop on QoS and Security for Wireless and Mobile Networks*, 2007, pp. 95–104.

[44] T. Brown, J. James, and A. Sethi, "Jamming and Sensing of Encrypted Wireless Ad Hoc Networks," in *Proc. 7^{th} ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2006, pp. 120–130.

[45] S. Jiang and Y. Xue, "Providing Survivability Against Jamming Attack via Joint Dynamic Routing and Channel Assignment," in *Proc. 7^{th} International Workshop on Design of Reliable Communication Networks (DRCN)*, 2009, pp. 198–205.

[46] J. Deng, R. Han, and S. Mishra, "Defending Against Path-Based DoS Attacks in Wireless Sensor Networks," in *Proc. 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks*, 2005, pp. 89–96.

[47] N. Aschenbruck, E. Gerhards-Padilla, and P. Martini, "Simulative Evaluation of Adaptive Jamming Detection in Wireless Multi-hop Networks," in *Proc. 7^{th} Workshop on Wireless Ad hoc and Sensor Networks WWASN*, 2010, pp. 213–220.

[48] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy, "Denial of Service Attacks in Wireless Networks: The Case of Jammers," *To appear in IEEE Commun. Surveys & Tutorials*, 2011.

[49] J. Douceur and J. S. Donath, "The Sybil Attack," in *Proc. First International Workshop on Peer-to-Peer Systems (IPTPS)*, 2002, pp. 251–260.

[50] S. Lv, X. Wang, X. Zhao, and X. Zhou, "Detecting the Sybil Attack Cooperatively in Wireless Sensor Networks," *Proc. International Conference on Computational Intelligence and Security*, vol. 1, pp. 442–446, 2008.

[51] I. Saha and D. Mukhopadhyay, "Security Against Sybil Attack in Wireless Sensor Network Through Location Verification," in *Proc.*

International Conference on Distributed Computing and Networking (ICDCN), 2009, pp. 187–192.

[52] K.-F. Ssu, W.-T. Wanga, and W.-C. Chang, "Detecting Sybil Attacks in Wireless Sensor Networks Using Neighboring Information," *Computer Networks*, vol. 52(18), pp. 3042–3056, 2009.

[53] Q. Zhang, P. Wang, D. S. Reeves, and P. Ning, "Defending Against Sybil Attacks in Sensor Networks," in *Proc. Second International Workshop on Security in Distributed Computing Systems*, 2005, pp. 185–191.

[54] J. Newsome, E. Shi, D. Song, and A. Perrig, "The Sybil Attack in Sensor Networks: Analysis & Defenses," in *Proc. 3rd International Symposium on Information Processing in Sensor Networks*, 2004, pp. 259–268.

[55] S. Gilbert, R. Guerraoui, and C. C. Newport, "Of Malicious Motes and Suspicious Sensors: On the Efficiency of Malicious Interference in Wireless Networks," in *Proc. International Conference On Principles Of Distributed Systems (OPODIS)*, 2006, pp. 215–229.

[56] S. Gilbert, R. Guerraoui, and C. Newport, "Of Malicious Motes and Suspicious Sensors: On the Efficiency of Malicious Interference in Wireless Networks ," *Theoretical Computer Science*, vol. 410, no. 6-7, pp. 546–569, 2009.

[57] B. Awerbuch, A. Richa, and C. Scheideler, "A Jamming-Resistant MAC Protocol for Single-Hop Wireless Networks," in *Proc. 27^{th} Symposium on the Principles of Distributed Computing (PODC)*, 2008, pp. 45–54.

[58] A. Richa, C. Scheideler, S. Schmid, and J. Zhang, "A Jamming-Resistant MAC Protocol for Multi-Hop Wireless Networks," in *Proc. International Symposium on Distributed Computing (DISC)*, 2010.

[59] S. Gilbert, R. Guerraoui, D. Kowalski, and C. Newport, "Interference-Resilient Information Exchange," in *Proc. 28^{th} Conference on Computer Communications (INFOCOM)*, 2009, p. n. pag.

[60] N. Ahmed, S. Kanhere, and S. Jha, "Poster Abstract: Multi-Channel Interference in Wireless Sensor Networks," in *Proc. International Conference on Information Processing in Sensor Networks (IPSN)*, 2009, pp. 367–368.

[61] J. Komlós and A. G. Greenberg, "An Asymptotically Nonadaptive Algorithm for Conflict Resolution in Multiple-Access Channels," *IEEE Trans. Inf. Theory*, vol. 31, pp. 302–306, November 1985.

[62] S. Dolev, S. Gilbert, R. Guerraoui, and C. Newport, "Gossiping in a Multi-channel Radio Network: An Oblivious Approach to Coping with Malicious Interference," in *Proc. Symposium on Distributed Computing (DISC)*, 2007, pp. 208–222.

[63] ——, "Secure Communication over Radio Channels," in *Proc. Symposium on Principles of Distributed Computing (PODC)*, 2008, pp. 105–114.

[64] P. Turán, "On an Extremal Problem in Graph Theory," *Matematicko Fizicki Lapok*, vol. 48, pp. 436–452, 1941.

[65] D. Meier, Y. A. Pignolet, S. Schmid, and R. Wattenhofer, "Speed Dating Despite Jammers," in *Proc. IEEE International Conference on Distributed Computing in Sensor Systems*, 2009, pp. 1–14.

[66] V. King, J. Saia, and M. Young, "Conflict on a Communication Channel," Accepted to the *Symposium on Principles of Distributed Computing (PODC)*, 2011.

[67] S. Dolev, S. Gilbert, R. Guerraoui, F. Kuhn, and C. Newport, "The Wireless Synchronization Problem," in *Proc. 28^{th} ACM Symposium on Principles of Distributed Computing (PODC)*, 2009, pp. 190–199.

[68] S. Ganeriwal, C. Pöpper, S. Čapkun, and M. B. Srivastava, "Secure Time Synchronization in Sensor Networks," *ACM Transactions on Information and System Security*, vol. 11, no. 23, 2008.

[69] E. Bayraktaroglu, C. King, X. Liu, G. Noubir, R. Rajaraman, and B. Thapa, "On the Performance of IEEE 802.11 under Jamming," pp. 1265–1273, 2008.

[70] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming Sensor Networks: Attack and Defense Strategies," IEEE *Network*, vol. 20, no. 3, pp. 41–47, 2006.

[71] T. Voigt, H. Ritter, and J. Schiller, "Utilizing Solar Power in Wireless Sensor Networks," in *Proc. 28^{th} Annual IEEE International Conference on Local Computer Networks*, 2003, pp. 416–422.

[72] P. Corke, P. Valencia, P. Sikka, T. Wark, and L. Overs, "Long-Duration Solar-Powered Wireless Sensor Networks," in *Proc. 4^{th} Workshop on Embedded Networked Sensors*, 2007, pp. 33–37.

[73] P. Cardieri, "Modeling Interference in Wireless Ad Hoc Networks," *IEEE Commun. Surveys & Tutorials*, vol. 12, no. 4, pp. 551–572, 2010.

[74] M. C. Pease, R. E. Shostak, and L. Lamport, "Reaching agreement in the presence of faults," *Journal of the ACM*, vol. 27(2), pp. 228–234, 1980.

[75] P. Berman, K. Diks, and A. Pelc, "Reliable Broadcasting in Logarithmic Time with Byzantine Link Failures," *Journal of Algorithms*, vol. 22, no. 2, pp. 199–211, 1997.

[76] A. Lubotzky, R. Phillips, and P. Sarnak, "Explicit Expanders and the Ramanujan Conjecture," in *Proc. 18^{th} Annual ACM Symposium on Theory of Computing*, 1986, pp. 441–459.

[77] A. Pelc and D. Peleg, "Feasibility and Complexity of Broadcasting with Random Transmission Failures," in *Proc. 24th Symposium on Principles of Distributed Computing*, 2005, pp. 334–341.

[78] C.-Y. Koo, "Broadcast in Radio Networks Tolerating Byzantine Adversarial Behavior," in *Proc. 23^{rd} Annual ACM Symposium on Principles of Distributed Computing (PODC)*, 2004, pp. 275–282.

[79] G. Zhou, T. He, S. Krishnamurthy, and J. A. Stankovic, "Impact of Radio Irregularity on Wireless Sensor Networks," in *Proc. 2^{nd} International Conference on Mobile Systems, Applications, and Services*, 2004, pp. 125–138.

[80] D. Kotz, C. Newport, R. S. Gray, J. Liu, Y. Yuan, and C. Elliott, "Experimental Evaluation of Wireless Simulation Assumptions," in *Proc. 7^{th} ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, 2004, pp. 78–82.

[81] D. Son, B. Krishnamachari, and J. Heidemann, "Experimental Study of Concurrent Transmission in Wireless Sensor Networks," in *Proc. 4th International Conference on Embedded Networked Sensor Systems*, ser. SenSys, 2006, pp. 237–250.

[82] ——, "Experimental Study of the Effects of Transmission Power Control and Blacklisting in Wireless Sensor Networks," in *Proc. First IEEE Conference on Sensor and Adhoc Communication and Networks*. Santa Clara, California, USA: IEEE, October 2004, pp. 289–298.

[83] X. Zhang and S. B. Wicker, "Robustness vs. Efficiency in Sensor Networks," in *Proc. 4^{th} International Symposium on Information Processing in Sensor Networks (ISPN)*, 2005, pp. 225–230.

[84] A. Woo, T. Tong, and D. Culler, "Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks," in *Proc. 1st International Conference on Embedded Networked Sensor Systems*, 2003, pp. 14–27.

[85] M. Kubisch, H. Karl, A. Wolisz, L. C. Zhong, and J. Rabaey, "Distributed Algorithms for Transmission Power Control in Wireless Sensor Networks," in *Proc. 6^{th} Annual Communication Networks and Services Research Conference*, 2003, pp. 417–421.

[86] V. Bhandari and N. H. Vaidya, "On Reliable Broadcast in a Radio Network," in *Proc. 24^{th} Annual ACM Symposium on Principles of Distributed Computing (PODC)*, 2005, pp. 138–147.

[87] A. Pelc and D. Peleg, "Broadcasting with Locally Bounded Byzantine Faults," *Information Processing Letters*, vol. 93, no. 3, pp. 109–115, 2005.

[88] A. Ichimura and M. Shigeno, "A New Parameter for a Broadcast Algorithm with Locally Bounded Byzantine Faults," *Information Processing Letters*, vol. 110, no. 12-13, pp. 514–517, 2010.

[89] V. Bhandari and N. H. Vaidya, "On Reliable Broadcast in a Radio Network: A Simplified Characterization," CSL, UIUC, Tech. Rep., May 2005.

[90] V. Vaikuntanathan, "Brief Announcement: Broadcast in Radio Networks in the Presence of Byzantine Adversaries," in *Proc. 24^{th} Annual ACM Symposium on Principles of Distributed Computing (PODC)*, 2005.

[91] V. Bhandhari, J. Katz, C.-Y. Koo, and N. Vaidya, "Reliable Broadcast in Radio Networks: The Bounded Collision Case," in *Proc. 25^{th} Annual ACM Symposium on Principles of Distributed Computing (PODC)*, 2006, pp. 258 – 264.

[92] P. Berman and J. A. Garay, "Asymptotically Optimal Distributed Consensus," in *Proc. 16^{th} International Colloquium on Automata, Languages and Programming (ICALP)*, 1989, pp. 80–94.

[93] M. Bertier, A.-M. Kermarrec, and G. Tan, "Message-Efficient Byzantine Fault-Tolerant Broadcast in a Multi-Hop Wireless Sensor Network," in *Proc. 30^{th} International Conference on Distributed Computing Systems (ICDCS)*, 2010, pp. 408–417.

[94] ——, "Brief Announcement: Reliable Broadcast Tolerating Byzantine Faults in a Message-Bounded Radio Network," in *Proc. 22^{nd} International Symposium on Distributed Computing (DISC)*, 2008, pp. 516–517.

[95] V. King, C. Phillips, J. Saia, and M. Young, "Sleeping on the Job: Energy-Efficient and Robust Broadcast for Radio Networks," in *Proc. 27^{th} ACM symposium on Principles of Distributed Computing (PODC)*, 2008, pp. 243–252.

[96] ——, "Sleeping on the Job: Energy-Efficient and Robust Broadcast for Radio Networks," Accepted to Algorithmica, 2010.

[97] D. Alistarh, S. Gilbert, R. Guerraoui, Z. Milosevic, and C. Newport, "Securing Your Every Bit: Reliable Broadcast in Byzantine Wireless Networks," in *Proc. 22^{nd} ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, 2010.

[98] WSNet, http://wsnet.gforge.inria.fr/.
[99] ns-3, http://www.nsnam.org/.
[100] QualNet, http://www.scalable-networks.com/.
[101] OPNET Modeler, http://www.opnet.com/.
[102] OMNet, http://www.omnetpp.org/.
[103] Castalia, http://castalia.npc.nicta.com.au/.
[104] M. Korkalainen, M. Sallinen, N. Kärkkäinen, and P. Tukeva, "Survey of Wireless Sensor Networks Simulation Tools for Demanding Applications," in *Proc. 2009 Fifth International Conference on Networking and Services*, 2009, pp. 102–106.
[105] M. Imran, A. M. Said, and H. Hasbullah, "A Survey of Simulators, Emulators and Testbeds for Wireless Sensor Networks," in *Proc. International Symposium on Information Technology (ITSim)*, 2010, pp. 897–902.
[106] V. Bhandari and N. H. Vaidya, "Reliable Broadcast in Wireless Networks with Probabilistic Failures," in *Proc.* 27^{th} *Conference on Computer Communications (INFOCOM)*, 2007, pp. 715–723.
[107] ——, "Reliable Broadcast in Wireless Networks with Probabilistic Failures," University of Illinois at Urbana-Champaign, Tech. Rep., 2007.
[108] G. Urdaneta, G. Pierre, and M. van Steen, "A Survey of DHT Security Techniques," *ACM Computing Surveys*, vol. 43, no. 2, pp. 1–53, 2011.
[109] V. Bhandari and N. H. Vaidya, "Reliable Local Broadcast in a Wireless Network Prone to Byzantine Failures ," in *Proc. International Workshop on Foundations of Mobile Computing (DIAL-POMC)*, 2007.
[110] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. shiuan Peh, and D. Rubenstein, "Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet," in *Proc.* 10^{th} *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-X)*, 2002, pp. 96–107.
[111] K. Liu, Z. Yang, M. Li, Z. Guo, Y. Guo, F. Hong, X. Yang, Y. He, Y. Feng, and Y. Liu, "OceanSense: Monitoring the Sea with Wireless Sensor Networks," *Mobile Computing and Communications Review*, vol. 14, no. 2, pp. 7–9, 2010.

Maxwell Young received his B.Sc. in mathematics from Queen's University in 2001, a B.Sc. in computer science from the University of British Columbia in 2003, and a M.S. in computer science from the University of New Mexico in 2006. Currently, he is a Ph.D. candidate in the David R. Cheriton School of Computer Science at the University of Waterloo in Ontario, Canada. His research interests include security and distributed computing with a focus on adversarial fault tolerance in large decentralized networks.

Raouf Boutaba received the MSc and PhD degrees in computer science from the University Pierre & Marie Curie, Paris, in 1990 and 1994, respectively. He is currently a professor of computer science at the University of Waterloo. His research interests include network, resource and service management in wired, and wireless networks. He is the founder and editor in chief of the IEEE Transactions on Network and Service Management (2007-2010) and on the editorial boards of several other journals. He has received several best paper awards and other recognitions such as the Premiers Research Excellence Award, the IEEE Hal Sobol Award in 2007, the Fred W. Ellersick Prize in 2008, the Joe LociCero award and the Dan Stokesbury in 2009.