Message-Efficient Byzantine Fault-Tolerant Broadcast in a Multi-Hop Wireless Sensor Network

Marin Bertier, Anne-Marie Kermarrec, Guang Tan

Message-Efficient Byzantine Fault-Tolerant Broadcast in a Multi-Hop Wireless Sensor Network

Marin Bertier, Anne-Marie Kermarrec, and Guang Tan*
INRIA/IRISA, Rennes, France, Email: {mbertier, anne-marie.kermarrec, gtan}@irisa.fr

Abstract

We consider message-efficient broadcast tolerating Byzantine faults in a multi-hop wireless sensor network. Assuming a grid network where all nodes have a communication range of r, and a single neighborhood contains at most t dishonest and collision-capable (bad) nodes, each with a message budget m_f, we investigate the minimum message budget m that each honest (good) node must have in order to achieve reliable broadcast. We consider three cases: (1) m_f is known in advance and m is homogeneous among all good nodes; (2) m_f is known in advance and m is heterogeneous among good nodes; (3) m_f is unknown. For the first two cases, we present possibility results and broadcast protocols that have message costs within twice the lower bound. For the third case, we present a coding scheme that helps verify the integrity of messages at a receiving node without using any cryptographic techniques. This code leads to a reactive local broadcast *primitive that has probabilistic reliability guarantees. Combined with a previously proposed scheme, it results in a broadcast protocol for $t < \frac{1}{2}r(2r+1)$ that guarantees reliability with high probability.*

1 Introduction

In the design of sensor networks, the impact of malicious behavior should be taken into account because the sensor nodes are often deployed in unattended and physically insecure environments. The low cost configuration of sensor nodes make traditional measures such as tamper-proof hardware not cost effective, and even standard cryptographic techniques may be too expensive for common devices [12]. Under this constraint, many reliability/security problems arise. In this paper we consider how to achieve reliable broadcast in a message-efficient way in such a network.

In the problem, referred to as *BFT-BCAST*, there is a base station serving as message source. The task is to deliver the correct message from the base station to all nodes in the network via multi-hop links, despite some faulty or malicious (called *bad* in this paper) nodes that may alter the message or cause collisions. For low cost consideration, it is desirable to accomplish the task assuming little or even no cryptography (for example, when (re)establishing keys). Assuming a non-collision and non-cryptography setting, Koo [13] first studies this problem and shows the maximum number of bad nodes per neighborhood, t, that can be tolerated by a broadcast protocol. In subsequent work [2,3], Bhandari et al. further prove that Koo's bound is a critical threshold. In [14], Koo et al. remove the non-collision assumption and show that the maximum tolerable t in a collision network remains the same as in a non-collision setting, provided that the bad nodes are collision-bounded.

We study a similar problem to that of [2, 13, 14], with an emphasis on message efficiency, an important system property that has not been considered previously. We assume that every node, either good or bad, has a total bound on the number of messages it can send. This tries to capture the fact that many network devices (for example the Smart Dust sensors [11]) are extremely constrained in energy, thus a finite message budget for a node to perform a task or an attack is a realistic assumption. This assumption differs from the one made by Koo et al. [14] which assumes that a bad node is bounded only in the number of collisions it can cause, but not bounded otherwise. The latter condition actually allows a simple treatment of collisions: if a bad node can cause at most β collisions, then a good node can simulate a collision-free transmission by repeatedly sending every message $\beta t + 1$ times. However, it is unclear what will happen if a good node's budget is lower than $\beta t + 1$, or whether this $\beta t + 1$ budget is necessary at all.

Our aim is to answer the following question: Given the message budget of every bad node $m_f \in \mathbf{N}$ and the maximum number of bad nodes per neighborhood, t, what is the minimum message budget each good node must have for BFT-BCAST to be possible, and if possible, how to achieve that? We show that, when m_f is known in advance, then the task can be achieved with a message cost within twice the lower bound, and allowing heterogeneous assignment of

*Author names are in alphabetical order; correspondence may be addressed to Guang Tan.

message budgets can further reduce average message cost. We also present a probabilistic solution for the case of unknown m_f.

Without assuming cryptographic mechanisms, the general approach for a broadcast protocol to overcome Byzantine faults is to send repetitive correct messages that outnumber potential false messages [2, 13, 14, 17]. To account for various kinds of faults, such protocols are often message costly, especially when the fraction of faulty nodes approaches the threshold. Therefore they are often envisioned to be used in some key parts of, rather than for all, communication tasks, such as (re)establishing cryptographic keys and sending message digests. Those parts of communication are particularly sensitive to failures, and thus deserve special care. On the other hand, they are less demanding in traffic, so a relatively heavy-weight protocol may be acceptable. Considering the most general application scenarios and also following the line of previous research, our protocols do not use any cryptographic techniques.

1.1 Related work

Reliable broadcast has been well studied for both point-to-point and radio networks. In [16], reliable broadcast in an arbitrary graph is considered. Upper and lower bounds in terms of graph theoretic parameters for the feasibility of reliable broadcast are presented, although no exact thresholds are given. Pelc and Peleg [17] consider random transmission failures for both radio and message-passing networks, where each node fails at each step with some constant probability. They establish feasibility conditions and estimate the complexity of almost-safe broadcasting for such a model. One of their assumptions is that a node cannot send different messages to different neighbors, which resembles the characteristics of radio transmission without collisions. A grid network with random but permanent node failures is considered in [4, 5]. Necessary and sufficient conditions on the required transmission range r as a function of node failure probability are derived. The problem of achieving consensus in a wireless network is studied in [8].

In [13], Koo considers achieving reliable broadcast in multi-hop radio networks in the presence of Byzantine faults. It is shown that the task is impossible for $t \geq \lceil \frac{1}{2}r(2r+1) \rceil$. Bhandari and Vaidya [2, 3] show that $\lceil \frac{1}{2}r(2r+1) \rceil$ is indeed an exact threshold of t, meaning that the task is always achievable for $t < \frac{1}{2}r(2r+1)$. Bhandari and Vaidya also present the threshold of t for the case of crash-stop failures. In [14], Koo. et al. additionally allow a bad node to cause a (known) bounded number of collisions, while placing no bound on the number of non-collision-causing messages that can be sent by a node. They show that despite the additional power available to the bad nodes, reliable broadcast remains solvable as long as $t < \frac{1}{2}r(2r+1)$.

Concurrent with Koo et al.'s work [14], Gilbert et al. [10] consider a *single-hop* and collision-bounded model in which a bad and a good node can send at most β and β' messages, respectively. Moreover, β is assumed to be unknown to good nodes in advance, and the source is assumed to be good. Different from this paper, their goal is to characterize the maximum ratio of disruption caused by the adversary to the cost of causing that disruption, and how long the adversary can delay the protocol without even performing a single broadcast. In [9], the authors place no restrictions on the adversary, which is allowed to disrupt communication and jam the airwaves in an arbitrary and unlimited fashion. Instead, they assume that each of the devices has access to multiple channels of communication. Based on such a model, they present algorithms that achieve ϵ-gossip and characterize their complexity.

1.2 Model and assumptions

The problem is to broadcast a message with value V_{true} from a base station, or source node, at $(0, 0)$ to all nodes in the network. We consider a network model similar to those described in [2–4, 13]. A total of n nodes are deployed on a grid (each grid cell is a 1×1 square). All nodes have an integer transmission radius r. A node's *neighborhood* is defined as the set of nodes within distance r of that node. We only consider the L_∞ metric, which means that a node's neighborhood is a square of side length $2r$ centered at itself. When no collision occurs, a message broadcast by a node is correctly received by all nodes within its neighborhood. To avoid edge effect we assume that the network is toroidal, and the network diameter $D \gg r$.

We adopt the locally-bounded adversarial model [13] where any single neighborhood contains $t < r(2r+1)$ bad nodes.[1] The bad nodes can alter the message and try to trick good nodes into accepting a wrong value. As in [2,3,13,14], we assume there is a pre-determined time-slotted schedule such that if all nodes follow the schedule then no collision will occur. However, a bad node may deviate from this schedule and cause message collisions. When two nodes and perform a local broadcast at the same time, their common neighbor nodes can receive a wrong message, or no message at all, without noticing anything abnormal. Let m and m_f be the message budget of a good and a bad node, respectively. We treat the base station as a special node that is not message-bounded.

Let A be a closed area on the plane, denote by (A) and $\{A\}$ the set of nodes in the interior and on the boundary of A, respectively, and let $[A] = (A) \cup \{A\}$. The set of nodes in a rectangular area $\{(x, y) : x_1 \leq x \leq x_2$ and $y_1 \leq y \leq y_2\}$ is denoted by $[x_1 \ldots x_2, y_1 \ldots y_2]$. When $x_1 =$

[1] Notice this bound is higher than that of the message-unbounded model [2, 13]. This bound can be achieved when good nodes have a higher message budget than bad ones.

x_2, we will just write $[x_1, y_1 \dots y_2]$; similarly, when $y_1 = y_2$, we write $[x_1 \dots x_2, y_1]$. A node is said to be *decided* if it has committed to, or accepted, a value, and *undecided* otherwise. A node set A is said to be decided if all the good nodes in A have accepted a value. Let $\mathcal{G}(A)$ and $\mathcal{B}(A)$ be the sets of good nodes and bad nodes of A, respectively.

As in [10, 12], we assume that the base station is always correct (a compromised base station often means the whole network becoming useless). With this assumption, a protocol is said to achieve broadcast in the network if the following two conditions hold: (1) Completeness: every good node in the network eventually accepts some value, and (2) Correctness: all good nodes accept V_{true}. Notice the difference from traditional definition of successful byzantine fault-tolerant broadcast, which also considers the possibility of a faulty source. This case can actually be handled separately by running a special protocol [14] for achieving agreement first among the source's neighborhood.

1.3 Contributions

Let $m_0 = \lceil \frac{2tm_f+1}{r(2r+1)-t} \rceil$, where $t < r(2r+1)$. This paper makes contributions for three cases[2]:

1. **(Known m_f and homogeneous m, Sections 2,3)** If m_f is known in advance and m is homogeneous among all good nodes, then the task is shown to be impossible for any $m < m_0$, while it is achievable if $m \geq 2m_0$. This result should be compared with the scheme suggested in [14] which requires $m = 2tm_f+1$, which is $\frac{1}{2}[r(2r+1)-t]$ times our budget. The key idea is to consider a concerted action of nearby good nodes in overcoming collisions, rather than isolated effort from individual nodes.

2. **(Known m_f and heterogeneous m, Section 4)** If m_f is known and m is allowed to be heterogeneous, then the task is possible when only $\Theta(r^3)$ good nodes have $m' = \frac{2tm_f+1}{\lceil (r(2r+1)-t)/2 \rceil} \approx 2m_0$ and all other good nodes have $m = m_0$. When $r \ll n$, which is common in practice, this may substantially reduce average message cost as compared with the homogeneous case. The improvement comes from a more careful analysis of the propagation pattern of V_{true}.

3. **(Unknown m_f, Section 5)** When m_f is unknown, then a coding scheme is proposed that helps verify the integrity of message at the receiver without using any cryptographic techniques. Borrowing ideas from *All-Unidirectional Error-Detecting* coding schemes [6, 7], this code leads to the design of a reliable reactive local broadcast primitive, which, combined with the multi-hop protocol proposed in [3], achieves reliable broadcast for $t < \frac{1}{2}r(2r+1)$ with high probability. This provides a probabilistic solution to an open problem suggested in [14].

[2]The first two results are reported in a two-page brief announcement [1].

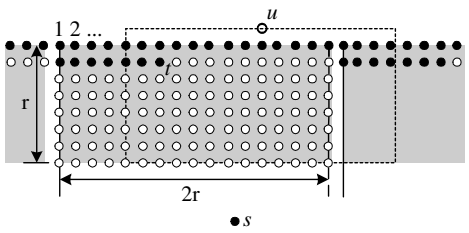

Figure 1: Impossibility of reliable broadcast.

2 A lower bound of m for reliable broadcast

In this section we show an impossibility result for reliable broadcast.

Theorem 1. *If $m < m_0$, then reliable broadcast is impossible.*

Proof. We let the adversary pick a stripe area of height r, extending in the horizontal direction; see an illustration in Figure 1. In this stripe area, for every interval of $2r+1$, which defines a rectangle (see the gray area shown in Figure 1), the adversary chooses to corrupt t nodes, starting from the top left node of the rectangle and in a left-to-right and then top-to-bottom order. In the figure, the bad nodes are shown in black while the good nodes are in white. We claim that none of the good nodes above the stripe area will be able to accept V_{true}.

We prove this by contradiction. Assume that u is the first node above the stripe area that accepts V_{true}. Then it must be able to make a majority decision from all the messages it receives. It is easy to verify that if u's neighborhood contains any good node from the stripe area (which is the necessary condition for u to ever receive a correct message), then u's neighborhood must cover exactly t bad nodes, and have $g \leq r(2r+1) - t$ good nodes in the stripe. In the worst case, the t bad nodes can corrupt up to tm_f messages by causing collisions, each resulting in a wrong value being delivered to u. To outnumber the tm_f wrong values, u will have to receive at least $tm_f + 1$ correct messages. This means that at least $2tm_f + 1$ correct messages have to be sent from the g good nodes in u's neighborhood. It follows that every good node has to send at least $\lceil \frac{2tm_f+1}{r(2r+1)-t} \rceil$ messages. Hence, it must be the case that $m \geq \lceil \frac{2tm_f+1}{r(2r+1)-t} \rceil$, which contradicts with the assumption $m < m_0$.

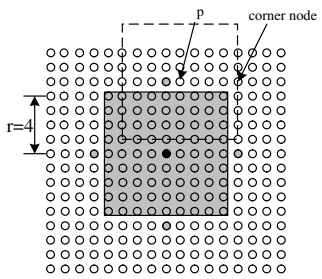

Figure 2: Impossibility of reliable broadcast for m slightly larger than m_0. $r = 4, t = 1, m_f = 1000, m_0 = \lceil \frac{2tm_f+1}{r(2r+1)-t} \rceil = 58, m = m_0 + 1 = 59$.

Therefore, there cannot exist one node above the stripe area that accepts V_{true}, meaning that the broadcast will fail. □

Next, we show that $m \geq m_0$ does not guarantee reliable broadcast in general. An example is shown in Figure 2, in which $r = 4, t = 1, m_f = 1000, m_0 = \lceil \frac{2tm_f+1}{r(2r+1)-t} \rceil = 58, m = m_0 + 1 = 59$. The bad nodes are distributed in the network in such a way that every neighborhood has exactly one bad node. First, the nodes in the neighborhood of the source node will be able to receive sufficient correct messages and accept V_{true}. Then, four additional nodes, in gray and located outside the gray square, can each receive $(r(2r+1)-t) \cdot m = 2065 > 2000 + 1 = 2tm_f + 1$ messages, thus being able to accept V_{true}. After that, no other good node will be able to receive enough correct messages that outnumber all possible wrong messages. Taking the node p as an example, it has only 33 decided neighbors, each being able to contribute 59 messages. Thus at most 1947 correct messages will be sent to p. The bad node in p's neighborhood, however, can alter up to 1000 messages by causing collisions, leaving only $947 < 1000$ copies of correct messages delivered to P. Consequently, p will not be able to tell the correct value from the messages received, and so the broadcast fails.

Generally, in the propagation of V_{true}, the nodes near the corners of the square area that has already accepted V_{true} have the fewest neighbors that can feed them V_{true}, so are the "weakest" under attack. Once we can solve this problem, the broadcast problem will be easy to solve.

3 Reliable broadcast for $m \geq 2m_0$

The main result of this section is the following theorem.

Theorem 2. *If $m \geq 2m_0$, then reliable broadcast can be achieved by some protocol.*

This result should be compared with the scheme suggested in [14] which requires every good node to have $m = 2tm_f + 1$ message budget, which is $\frac{1}{2}[r(2r+1) - t]$ times our budget. The basic idea is to let nearby good nodes cooperatively overcome collisions from relevant neighborhoods, rather than letting every single node to counter all possible interference from its own neighborhood. We first describe a protocol $\mathcal{B}$ that uses the condition $m \geq 2m_0$, and then prove that it guarantees reliable broadcast.

3.1 The protocol $\mathcal{B}$

1. **(Broadcast in the neighborhood of source)**: Initially, the source does a local broadcast of the message $2tm_f + 1$ times. Each neighbor i of the source accepts the majority value it receives from the source.

2. **(Broadcast in the rest of the network)**: every non-source node j, upon receiving a value, sends the accepted value $\frac{2tm_f+1}{\lceil (r(2r+1)-t)/2 \rceil}$ times. A node accepts a value once it receives such a value at least $tm_f + 1$ times.

3.2 Analysis

We show the correctness and completeness of $\mathcal{B}$.

Lemma 1. (Correctness) *No good node shall accept a wrong value by following the protocol $\mathcal{B}$.*

Proof. If a good node accepts a wrong value v, then following the protocol it must have received v at least $tm_f + 1$ times. But it has at most t bad neighbors, each of which can feed it a wrong value m_f times, thus it cannot have received more than tm_f wrong values. This means that v must indeed be the correct value V_{true}. □

The proof of completeness will use the following lemma.

Lemma 2. *Define node sets $A = [x \ldots x + r - 1, y \ldots y - r + 1]$, and $B = [x \ldots x + r, y \ldots y - r + 1]$. Assume $|\mathcal{G}(A)| \geq \lceil \frac{r(2r+1)-t}{2} \rceil$, and $|\mathcal{G}(B)| \geq \lceil \frac{r(2r+1)-t}{2} \rceil$. Then,*
(a) if A has accepted V_{true}, then the node set $A' = [x - 1 \ldots x + r, y + 1]$ will accept V_{true};
(b) if B has accepted V_{true}, then the node set $B' = [x \ldots x + r, y + 1]$ will accept V_{true}.

Proof. First, consider the node set A, as illustrated in Figure 3(a). For an arbitrary node from A', say P at $(x - 1, y + 1)$, its neighborhood will cover A entirely. Since $|\mathcal{G}(A)| \geq \lceil \frac{r(2r+1)-t}{2} \rceil$, and every good node in A will send the message $\frac{2tm_f+1}{\lceil (r(2r+1)-t)/2 \rceil}$ times, the total number of messages delivered to P will be at least $2tm_f + 1$. These messages could be altered or dropped at most tm_f times, resulting in at least $tm_f + 1$ correct messages being delivered to P. Taking a majority of all the received messages P will accept the correct value. The same argument can be applied to all other nodes in $[x - 1 \ldots x + r, y + 1]$. Thus (a) is proved. The case of B can be analyzed in a similar way, which proves (b). □

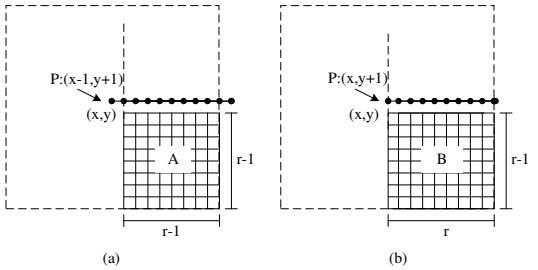

Figure 3: Propagation of V_{true} from areas with sufficient good nodes.

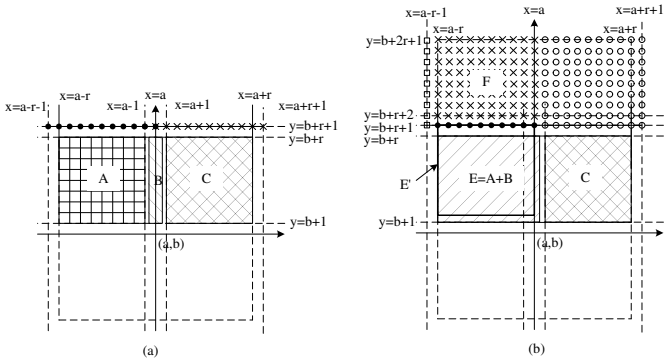

Figure 4: The propagation of V_{true} from the neighborhood of node (a, b) to its surrounding area.

Lemma 3. (Completeness) *Every good node is eventually able to accept V_{true}.*

Proof. The proof proceeds by induction.
Base Case: All good nodes in the neighborhood of the source are able to accept V_{true}. This follows trivially since the source broadcasts the correct message $2tm_f + 1$ times, thus being able to deliver at least $tm_f + 1$ correct messages to any node in its neighborhood.
Induction: We show that if the neighborhood of a node at (a, b), $[a-r \dots a+r, y-r \dots y+r]$, is able to accept V_{true}, then all nodes adjacent to that neighborhood are able to accept V_{true}. Due to the symmetry of network, it suffices to show that the set of nodes adjacent to one side of the neighborhood, that is, $[(a-r-1) \dots (a+r+1), y+r+1]$, can accept V_{true}.

Consider the four nodes sets, $A = [a-r \dots a-1, b+1 \dots b+r]$, $B = [a, b+1 \dots b+r]$, $C = [a+1 \dots a+r, b+1 \dots b+r]$, and $D = A \cup B \cup C = [a-r \dots a+r, b+1 \dots b+r]$, as illustrated in Figure 4. It can be seen that

$$|\mathcal{G}(A \cup B \cup C)| = |\mathcal{G}(D)| = |D| - |\mathcal{B}(D)| \geq r(2r+1) - t. \quad (1)$$

Now consider two cases:

1. Both $|\mathcal{G}(A)| \geq \lceil \frac{r(2r+1)-t}{2} \rceil$ and $|\mathcal{G}(C)| \geq \lceil \frac{r(2r+1)-t}{2} \rceil$. In this case, we apply Lemma 2(a) to A and B to obtain that both $[a-r-1 \dots a, y+r+1]$ (the line with filled circles) and $[a \dots a+r+1, y+r+1]$ (the line with crosses) by can accept V_{true}. Thus the node set $[(a-r-1) \dots (a+r+1), y+r+1]$ can accept V_{true}.

2. Either $|\mathcal{G}(A)| < \lceil \frac{r(2r+1)-t}{2} \rceil$ or $|\mathcal{G}(C)| < \lceil \frac{r(2r+1)-t}{2} \rceil$. Without loss of generality, assume that $|\mathcal{G}(C)| < \lceil \frac{r(2r+1)-t}{2} \rceil$. Then it must hold that $|\mathcal{G}(A \cup B)| \geq \lceil \frac{r(2r+1)-t}{2} \rceil$, because otherwise $|\mathcal{G}(A \cup B \cup C)| = |\mathcal{G}(A \cup B)| + |\mathcal{G}(C)| < r(2r+1) - t$, which contradicts with Eqn.(1). Let $E = A \cup B$, and applying Lemma 2(b) to E, we know that $[a-r \dots a, b+r+1]$ (see the line with filled circles) can accept V_{true}. Next, consider the node set $E' = [a-r \dots a, b+2 \dots b+r+1]$ which can be viewed by moving the set E upward by one unit. Since both E' and C are contained by a single neighborhood, and $|E'|+|C| = |E|+|C| = r(2r+1)$, it holds that $|\mathcal{G}(E')| \geq \lceil \frac{r(2r+1)-t}{2} \rceil$, because otherwise $|\mathcal{G}(E')| + |\mathcal{G}(C)| < r(2r+1) - t$, leading to $|\mathcal{B}(E')| + |\mathcal{B}(C)| > t$ which contradicts with the assumption that the number of bad nodes in a single neighborhood should be no more than t. Applying Lemma 2(b) to E' gives that $[a-r \dots a, b+r+2]$ can accept V_{true}. In a similar way, we can see that all nodes in the set $F = [a-r \dots a, b+r+1 \dots b+2r+1]$ (whose nodes are marked by crosses) will be able to accept V_{true}. Applying Lemma 2(b) to the set $[a-r \dots a-1, b+r+1 \dots b+2r+1]$ we obtain that the set $[a-r-1, b+r+1 \dots b+2r+1]$ (the line with small squares) will be able to accept V_{true}. Up to now, V_{true} has been accept by the nodes in $[a-r-1 \dots a, b+r+1]$. Following the same process as above, we can have all the nodes in $[a+1 \dots a+r+1, b+r+1 \dots b+2r+1]$ (whose nodes are marked by circles) accept V_{true}. Now we have that all the nodes in $[(a-r-1) \dots (a+r+1), y+r+1]$ can accept V_{true}, which completes the inductive step.

Combining the base case and the induction proves the lemma. □

Lemmas 1 and 3 together thus prove Theorem 2.

The necessary and sufficient conditions given in Theorem 1 and Theorem 2 for m can be stated in a more classic way: what is the maximum t that can be tolerated in reliable broadcast, given m and m_f?

Corollary 1. *Given m and m_f, any $t > \frac{mr(2r+1)-1}{2m_f+m}$ can cause broadcast to fail, while any $t \leq \frac{mr(2r+1)-2}{4m_f+m}$ can be tolerated by some broadcast protocol.*

4 Reliable broadcast for heterogeneous m

In this section we show that if nodes are allowed to have heterogeneous message budgets, then the average message cost of good nodes can be substantially reduced. The malicious behavior under consideration is the replacement of good nodes with bad nodes by the adversary, subject to the constraint of t and m_f on bad nodes.

Theorem 3. *If $\Theta(r^3)$ good nodes have message bound $m' = \frac{2tm_f+1}{\lceil (r(2r+1)-t)/2 \rceil} (\leq 2m_0)$ and the other good nodes have $m = m_0$, then reliable broadcast can be achieved by some protocol.*

The source of improvement over the homogeneous case is a more careful analysis of the propagation pattern of V_{true} in the network. In our proof of Theorem 2, and also in a sequence of previous papers [2, 13, 14], the inductive step requires a V_{true}-covered square region to expand into a larger square, so that the entire network can be eventually covered by V_{true}. The main challenge here is for the corner nodes (see Figure 2) near that region to accept V_{true}, since they have the fewest good neighbors that can feed them V_{true}. We have shown this to be the obstacle to reliable broadcast when every node has only $m = m_0$, and have demonstrated how this can be done with every good node having $m = 2m_0$. In this section we consider doing the task with only a fraction of the good nodes having $m = 2m_0$ while others having $m = m_0$.

Our new strategy is to use a circular, rather than a rectangular, area for the "growing body" of the V_{true}-covered region in induction. This strategy in effect eliminates the corner nodes problem: assume the circle is large enough, then any undecided node v adjacent to it has approximately a half neighborhood covered by the circle region, as compared with only a quarter neighborhood coverage in the case of a corner node near a square region. This significantly improves v's (worst-case) chance of accepting V_{true} because it can find more good and decided neighbors that supply V_{true} to them. On the other hand, this strategy requires transition into a geometric context as circles are no longer aligned with the integer nodes. We accomplish this by identifying a series of V_{true} propagation patterns.

We first describe a protocol $\mathcal{B}_{heter}$ that runs on a heterogeneous message budget configuration, as illustrated in Figure 5. In the cross-shaped area, all good nodes have a message budget of $m' = \frac{2tm_f+1}{\lceil (r(2r+1)-t)/2 \rceil} \approx 2m_0$, while all other good nodes have message budget m_0.

4.1 The protocol $\mathcal{B}_{heter}$

1. **(Broadcast in the neighborhood of source)**: Initially, the source does a local broadcast of the message $2tm_f + 1$ times. Each neighbor i of the source accepts the majority value it hears from the source.

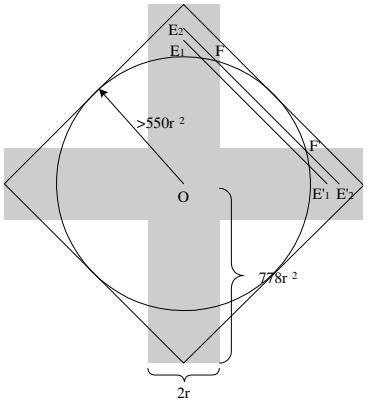

Figure 5: A message budget configuration which enables reliable broadcast. In the gray cross area, all (good) nodes have message budget $\frac{2tm_f+1}{\lceil (r(2r+1)-t)/2 \rceil} (\leq 2m_0)$, while all other (good) nodes have message budget m_0.

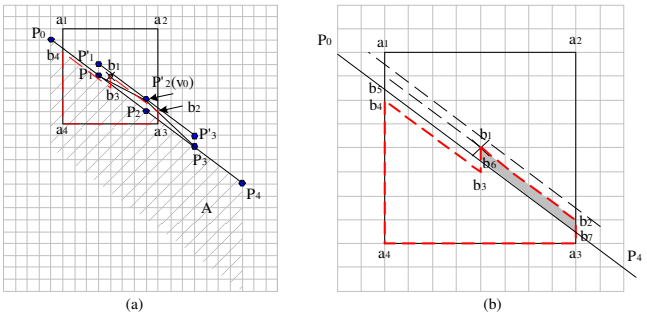

Figure 6: Committed line, basic message propagation pattern, and frontier.

2. **(Broadcast in the rest of the network)**: every non-source node j, upon accepting a value, sends the accepted value m times (or m' times if in the cross-shaped area). A node accepts a value once it receives that value $tm_f + 1$ times.

4.2 Analysis

We first show a sufficient condition for V_{true} to propagate to an undecided node.

Lemma 4. *If a node p has $r(2r+1)$ neighbors (either good or bad) that have accepted V_{true}, then p is able to accept V_{true}.*

Proof. The node p has at least $r(2r+1) - t$ decided good neighbors, and thus can receive at least $m_0(r(2r+1) - t) = 2tm_f + 1$ correct messages, meaning it can accept V_{true}. □

Next we describe a basic propagation pattern of V_{true} in the network. First we introduce several concepts. Define a *committed line*, denoted $\mathcal{L}(\rho, P_0, P_l), l \geq 1$, where

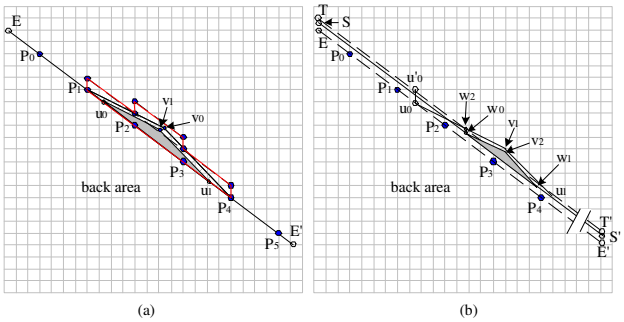

Figure 7: Shifted committed line, float committed line, and frontiers.

$\rho \in \mathbf{Z}$ and $-r \leq \rho \leq 0$, as a line segment such that: (1) its slope is ρ/r; (2) its left and right endpoints are two integer nodes $P_0 = (x_0, y_0)$ and $P_l = (x_l, y_l)$, respectively; (3) it contains a sequence of intermediate nodes $P_i = (x_0 + ir, y_0 + i\rho), 0 < i < l$; (4) let the line containing it be $y = f(x)$, then the good nodes in the area $\{(x, y) : x_0 \leq x \leq x_l$ and $f(x) - 2r \leq y \leq f(x)\}$, called the *back area*, have all accepted V_{true}. In Figure 6, the line segment $\overline{P_0P_4}$ is a committed line. The (shaded) back area A is a parallelogram sharing one edge with $\overline{P_0P_4}$ and having a height of $2r$.

A generalized form of a committed line is called a *shifted committed line*, which is the same as the former except that its two endpoints are not necessarily integer nodes and whose back area is defined as $\{(x, y) : x_0 \leq x \leq x_l$ and $\lfloor f(x) \rfloor - 2r \leq y \leq f(x)\}$. A further generalization is the *float committed line*, which does not necessarily contain the sequence of integer nodes $P_i = (x_0 + ir, y_0 + i\rho), 0 \leq i \leq l$. In Figure 6, if $\overline{P_0P_4}$ is moved along the line containing it so that either P_0 or P_4 becomes a non-integer node, then $\overline{P_0P_4}$ becomes a shifted committed line; furthermore, if $\overline{P_0P_4}$ (together with its back area) is moved to an arbitrary position in the plane, then $\overline{P_0P_4}$ becomes a float committed line.

The basic V_{true} propagation pattern is as follows.

Lemma 5. (Basic V_{true} propagation pattern) *Given a committed line $\mathcal{L}(\rho, P_0, P_l), l > 3$, moving the line $\overline{P_1P_{l-1}}$ upward by one unit yields a new committed line $\mathcal{L}(\rho, P'_1, P'_{l-1})$.*

In the following, we derive several variants of this propagation pattern that are increasingly easier to utilize in a continuous domain. Due to space reasons, the proofs are omitted here.

Let $\mathcal{L}(\rho, P_0, P_l), l > 3$ be a committed line. Draw a line with slope $(\rho+1)/r$ from P_1, and a line with slope $(\rho-1)/r$ from P_{l-l}, which intersect at v_0, called $\mathcal{L}$'s *frontier*. Recall that $[A]$ represents the node set in the interior and on the boundary of the closed area A.

Lemma 6. (Committed line propagation pattern) *Given a committed line $\mathcal{L}(\rho, P_0, P_l), l > 3$ with frontier v_0, then $[\triangle P_1P_{l-1}v_0]$ will accept V_{true}. Moreover, $|\overline{P_1v_0}| \geq (\lfloor|\mathcal{L}|/2\sqrt{2}r\rfloor - 1)r$ and $|\overline{P_{l-1}v_0}| \geq (\lfloor|\mathcal{L}|/2\sqrt{2}r\rfloor - 1)r$.*

Let EE' be a shifted committed line (Figure 7(a)) with slope ρ and whose length is larger than $4\sqrt{\rho^2 + r^2}$. Let u_0 and u_1 be two points on EE' such that $|\overline{Eu_0}| = |\overline{E'u_1}| = 2\sqrt{r^2 + \rho^2}$. Draw a line with slope $(\rho + 1)/r$ from u_0, and a line with slope $(\rho - 1)/r$ from u_1, which intersect at v_1, called EE''s *frontier*.

Lemma 7. (Shifted committed line propagation pattern) *Given a committed line $\mathcal{L}(\rho, P_0, P_l), l > 3$ with u_0, u_1, v_1 as specified above, then $[\triangle u_0u_1v_1]$ will accept V_{true}. Moreover, $|\overline{u_0v_1}| \geq (\lfloor|\overline{EE'}|/2\sqrt{2}r\rfloor - 2)r$ and $|\overline{u_1v_1}| \geq (\lfloor|\overline{EE'}|/2\sqrt{2}r\rfloor - 2)r$.*

Let EE' be a float committed line (Figure 7(b)) with slope ρ and whose length is larger than $6\sqrt{\rho^2 + r^2}$. Let w_0 and w_1 be two points on EE' such that $|\overline{Ew_0}| = |\overline{E'w_1}| = 3\sqrt{r^2 + \rho^2}$. Draw a line with slope $(-\rho + 1)/r$ from w_0, and a line with slope $(-\rho - 1)/r$ from w_1, which intersect at v_2, called EE''s *frontier*.

Lemma 8. (Float committed line propagation pattern) *Given a float committed line $\mathcal{L}(\rho, P_0, P_l), l > 3$ with w_0, w_1, v_2 as specified above, then $[\triangle w_0w_1v_2]$ will accept V_{true}. Moreover, $|\overline{w_0v_2}| \geq (\lfloor|\overline{EE'}|/2\sqrt{2}r\rfloor - 3)r$ and $|\overline{w_1v_2}| \geq (\lfloor|\overline{EE'}|/2\sqrt{2}r\rfloor - 3)r$.*

Define an *expanding line* as a line segment with a slope $h \in (-1, 0)$. Assume $\rho/r \leq h < (\rho + 1)/r$, where $\rho \in \mathbf{Z}$ and $-r < \rho < 0$. Figure 8(a) shows an example in which the expanding line EE' lies between two float committed lines: EE_1 with slope ρ/r, and EF with slope $(\rho + 1)/r$.

Lemma 9. *Assume an expanding line EE' sufficiently long and with a slope $\rho/r \leq h < (\rho + 1)/r$, where $\rho \in \mathbf{Z}$ and $-r < \rho < 0$. Draw a line segment EE_1 of length $37r$ with slope ρ/r, and a line segment $E'E'_1$ of length $37r$ from E' with slope $(\rho + 1)/r$, both beneath EE'. Then either EE_1 or $E'E'_1$'s frontier is above EE', with a distance to EE' (i.e., distance to its projection on EE') $d > 1.25$.*

Proof. (Sketch) Let $\angle 1 = \angle E_1EE', \angle 2 = E'EF, \angle 3 = E_1EF$ and $\angle 4 = \angle E'_1E'E$, as depicted in Figure 8(a). Since $\angle 4 = \angle 2$, it must hold that either $\angle 1 \leq \frac{1}{2}\angle 3$ or $\angle 4 \leq \frac{1}{2}\angle 3$. Due to the symmetry we only need consider the first case.

By Lemma 8, $|\overline{w_0v_2}| \geq (\lfloor 37r/2\sqrt{2}r\rfloor - 3)r > 10r$. As assumed, $\angle 1 \leq \frac{1}{2}\angle 3$, thus $\angle w_0w_2E = \angle 2 \geq \frac{1}{2}\angle 3 = \angle 1$. It follows that $|\overline{w_0w_2}| < |\overline{w_0E}| = 3\sqrt{\rho^2 + r^2} < 3\sqrt{2}r$. Therefore, $|\overline{w_2v_2}| > 10r - 3\sqrt{2}r > 5r$.

Since $d = |\overline{w_2v_2}| \sin\angle 2' = 7r \sin\angle 2 \geq 7r \sin\frac{1}{2}\angle 3$, we need to obtain a lower bound of $\frac{1}{2}\angle 3$ in order to bound

d from below. Figure 8(b) shows possible committed lines $EF_0, EF_1, \cdots, EF_r$, whose slopes are $0, -1/r, \cdots, -1$, respectively. It is easy to verify that the minimum $\angle 3$ corresponds to the angle $\angle F_r E F_{r-1}$. Let V be the projection of F_{r-1} on EF_r. It is the case that $|\overline{F_{r-1}V}| = \sqrt{2}/2$, so $\sin\angle 3 = |\overline{F_{r-1}V}|/|\overline{EF_{r-1}}| = \sqrt{2}/2|\overline{EF_{r-1}}| \geq 1/2r$. It follows that $\cos\angle 3 \leq \sqrt{4r^2-1}/2r$. Hence, $\sin\frac{1}{2}\angle 3 = \sqrt{\frac{1-\cos\angle 3}{2}} \geq \frac{1}{4r}$, and $d > 5/4 = 1.25$. □

Lemma 9 is used to prove the following result.

Lemma 10. *Let C be the circle of radius $R \geq 550r^2$ centered at the source node $(0,0)$. If $[C]$ has accepted V_{true}, then there exists some $\delta > 0$ such that $[C']$, where C' is a circle of radius $R + \delta$ centered at the source node, will accept V_{true}.*

Proof. Assume an expanding line EE' of length $74r$ whose end points are both on the circle C (see Figure 8(d)). Draw a ray from $(0,0)$ that is perpendicular to EE', and intersects with EE' and C at H and H_1, respectively. By Lemma 9, the float committed line EE_1 of length $37r$ right below EE' has a frontier point above EE'. Now move EE_1, together with its frontier v_2, along EE' until E reaches H, then the trajectory of v_2 will form a line segment v_2v_2'. Assume that the ray OH intersects with v_2v_2' at H_2. By Lemma 8, all nodes, if there are any, on the line segments $\overline{HH_2}$ are decided.

Let $L = |\overline{EE'}|$. It can be verified that $|\overline{HH_1}| = R - \sqrt{R^2 - |\overline{EE'}|^2/4} < 0.72 < 1.25 < |\overline{HH_2}|$, which means that H_2 is outside C. Let $\delta = 1.25 - |\overline{HH_1}| > 0.53$. Now move the expanding line EE' along the circle C, with EE''s endpoints remaining in touch with C, and subject to the slope constraint of an expanding line. During this process, $|\overline{HH_1}|$ will remain the same, so $\overline{H_1H_2}$ will remain no less than δ. The trajectory of $\overline{H_1H_2}$ therefore will sweep over a belt area in which all nodes will be decided, as shown as a shaded region in Figure 8(d). Note this belt region does not include the boundary line segment PP', where P (resp. P') is the intersection point between the line $y = x (x > 0)$ and C (resp. C'). By Lemma 7, it is easy to see that all nodes, if there are any, on $\overline{PP'}$ are decided. Similarly, $\overline{QQ'}$, where Q (resp. Q') is the intersection point between the line ($x = 0, y > 0$) and C (resp. C'), will be decided. By symmetry, the nodes in the ring of width δ outside and adjacent to C will be able to accept V_{true} , which means that all nodes in $[C']$ will accept V_{true}. □

Lemma 11. *If all nodes in the cross-shaped area shown in Figure 5 have accepted V_{true}, then $[C]$, where C is the circle of radius $R = 550r^2$ centered at the source node, will accept V_{true}.*

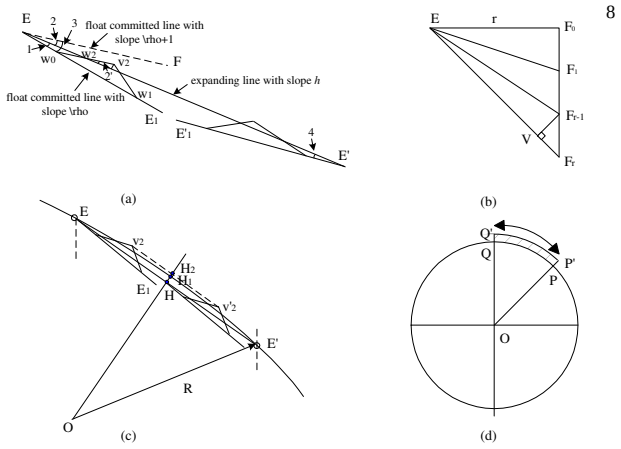

Figure 8: Expanding line and message propagation.

Proof. Following the same argument as for Lemma 3, we know that all the nodes in the cross area will accept V_{true}. Consider a committed line E_1E_1', as illustrated in Figure 5. By Lemma 5, E_1E_1' will yield a new committed line FF', where F and F' are two nodes at the edge of the cross area. Since FF' extends into the cross area, the extended line segment E_2E_2', where E_2 is a node on the y-axis and one unit above E_1, and E_2' is a node on the x-axis and one unit right after E_1', is a committed line. By induction and symmetry, all the nodes in the square of side length $\geq 778r^2$ will accept V_{true}. Thus all nodes within the circle of radius $R = 550r^2$ centered at the source node will accept V_{true}. □

With the above message budget configuration, V_{true} is guaranteed to "fill" the cross-shaped area, and so the rest of the network can accept V_{true}. Theorem 3 then follows from Lemmas 11 and 10, with similar correctness and completeness arguments to those of Theorem 2.

5 Reliable broadcast when m_f is unknown

In this section we consider reliable broadcast when m_f is unknown. Different from some previous work (e.g., [8,10]) in which some form of collision detector is required, we do not assume collision detection capability of the nodes. As a result, a receiving node cannot distinguish between a collision and the absence of transmission. In other words, the adversary has the ability to "cancel out" a message transmission (by, for example, predicting the shape of signal and sending an inverted signal [7]), without being noticed by the receiver. This generalization makes reliable broadcast more difficult because it is not obvious how transmission feedback can be exploited. The first challenge we face is therefore to make *local broadcast* reliable.

We provide a probabilistic solution to this first problem. We say that the broadcast is successful with a high probability if it succeeds with probability at least $1 - n^{-1}$, where n

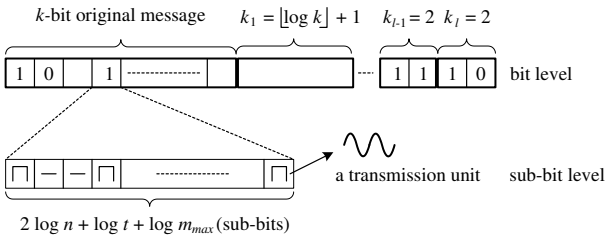

Figure 9: The two-level encoding scheme.

is the network size. Assume that the broadcast message has k bits, $\langle b_0 b_1, \ldots, b_{k-1} \rangle$. The good nodes know a very loose upper bound m_{max} for the bad nodes' budget. This bound may be an estimate of a practical device's energy limit, and may well be orders of magnitude higher than the real m_f. This bound is so loose that using the previous protocols with this knowledge is practically meaningless.

At the heart of our protocol is an encoding/decoding scheme that can detect errors when the message has been altered by the adversary through collisions. The idea is borrowed from *All-Unidirectional Error-Detecting* codes [6, 7] which are used in situations where it is possible to flip, for example, a bit from "0" to "1" but not vice-versa (except with a negligible probability). With such a code, the receiver is able to detect any number of bit-flipping errors.

The encoding is performed at two levels: bit-level and *sub-bit*-level (see Figure 9). A message is encoded into a sequence of bits, and each bit into a sequence of L sub-bits. As the basic transmission unit, a sub-bit has two states, $\sqcap$ and $-$, which represent the *presence* and *absence* of a signal (or a strong and weak signal) for a duration of a time slot, respectively. In the code, every bit 0 is represented by a sequence of $-$'s, while every 1 by a sequence of random sub-bits. At the receiver side, any such sequence containing at least one $\sqcap$ will be interpreted as a 1. This mechanism has the following consequence: the adversary can easily flip a 0 bit to 1, by inserting a $\sqcap$ in the sub-bit sequence of 0, but has some difficulty to turn a 1 into a 0, because to do so it has to know exactly which sub-bits are $\sqcap$'s and which are $-$'s. Taking one $\sqcap$ for $-$ will leave one $\sqcap$ intact in the sequence, while taking one $-$ for $\sqcap$ will lead to a transmission of signal that has nothing to cancel out, thereby generating a new $\sqcap$ sub-bit. Either will lead to the receiver decoding the sequence into a 1, which is correct. Due to the randomness of sub-bit generation for 1, guessing the whole sequence correctly will be increasingly difficult when the sequence length grows. In our design, $L = 2\log n + \log t + \log m_{max}$, which makes the attack success probability be $p_{biterr} = 2^{-L} = \frac{1}{n^2 t m_{max}}$.

At the bit-level, the coded message consists of the original message, denoted by S_0, with the appendix of a series of segments $S_1, S_2, \ldots, S_l$. The lengths of these segments are $k_0 = k, k_1, \ldots, k_l$, respectively, and satisfy $k_i = \lfloor \log k_{i-1} \rfloor + 1$. The segment S_i holds the number of 1 bits in the preceding segment S_{i-1}. The last two segments S_{l-1} and S_l each has two bits. It is easy to see that the last segment S_l can only be 01 or 10. At the receiver side, a node can verify the integrity of a message by checking the number of 1's in each segment. Since the adversary is only able to change 0 to 1 (except with a very small probability), and to maintain the consistency of bit 1 counting across the segments, any such changes will result in further changes, from the segment where the first changes take place through all segments until S_l. For S_l, the only change the adversary can possibly make is to turn one 0 into 1, resulting in the code 11. However, this code cannot happen in a correct code because the segment S_{l-1} can have at most two 1 bits. Therefore, once a message has been tampered with, the receiver will be able to detect an error almost surely, with a probability at least $1 - p_{biterr}$.

Let $K = \sum_{i=0}^{l} k_i$ be the coded message length for an original message of k bits, then transmitting a message takes $K \cdot L$ consecutive time slots. We define such a sequence of time slots as a *message round*. With the error detecting code, it is possible to realize a reliable local broadcast primitive with probabilistic guarantees. The local broadcast uses a negative acknowledge (NACK) message, which has the same length as a normal message, but with different content that is understood by the protocol. When a receiver detects an error in the message, it broadcasts an NACK message to its neighborhood. The receipt of an NACK message, either correct or corrupt, indicates a transmission failure. Upon detecting a failure, the sender re-transmits the message. A node repeats transmitting a message until it receives no NACK messages in subsequent consecutive $(2r+1)^2 - 1$ message rounds. In the worst case, a node needs to transmit a message tm_f times to deliver V_{true} to all its neighbors, with a probability at least $1 - t \cdot m_{max} \cdot p_{biterr} = 1 - n^{-2}$.

Having made local broadcast reliable, we can now run the protocol proposed in [3] on top of our reactive local protocol to achieve reliable multi-hop broadcast. This protocol can tolerate up to $\frac{1}{2}r(2r+1) - 1$ bad nodes per neighborhood. We call the combined protocol $\mathcal{B}_{reactive}$. Regarding the message overhead of good nodes, we have the following result.

Theorem 4. *In the protocol $\mathcal{B}_{reactive}$, any good node needs to transmit no more than*

$$m = 2(tm_f+1)(2\log n+\log t+\log m_{max})(k+2\log k+2)$$

times, where $t < \frac{1}{2}r(2r+1)$, to achieve reliable broadcast with a probability at least $1 - n^{-1}$.

Proof. At the message level, a node needs at most $tm_f + 1$ transmissions to make sure that every neighbor receive an

integral message with probability at least $1-n^{-2}$. Before it receives the message, it may need to transmit up to tm_f+1 negative acknowledge messages. So a node may transmit $2(tm_f+1)$ times in the worse case. Every message involves KL sub-bits, each of which may need one transmission. Observe that $K = k+\lfloor\log k\rfloor+1+\lfloor\log(\lfloor\log k\rfloor+1)\rfloor+1+\cdots \leq k+2\log k+2$. Hence in total a node needs to transmit at most $m = 2(tm_f+1)(2\log n+\log t+\log m_{max})(k+2\log k+2)$ to ensure reliable broadcast with probability at least $1-n^{-2}$. Considering the network diameter $D < n$, the whole network broadcast will succeed with probability at least $1-n^{-1}$. □

Comparing Theorem 4 with the results in previous sections, one can see that when m_f is unknown, good nodes need a much higher message budget to achieve reliable broadcast than when m_f is known. This is to be expected since without the knowledge of the adversary's capability, one has to make the worse case estimate when dealing with attacks.

In comparison with the *I-code* proposed in [7], our scheme has a lower coding overhead, since for a message of length k, our scheme generates a code of length $k+O(\log k)$, whereas I-code yields a length $2k$. On the other hand, our scheme has a higher per-attack penalty since the integrity verification is on a message basis, which means that every bit flipping attack from the adversary causes the whole message to be re-transmitted, while the I-code verifies message bit by bit, meaning that only the flipped bit needs to be re-transmitted. Final comparison on message efficiency thus calls for a refined model that takes into account message length and per-message attack rate. This might be a subject of future study.

6 Conclusions

We have studied the problem of reliable broadcast tolerating Byzantine faults in a message-bounded radio network. Given the communication range, the message bounds of nodes, and the maximum number of faulty nodes per neighborhood, we show how large m should be in order for broadcast to be reliable.

For the homogeneous case of m, the presented results leave an uncertain region of $m \in (m_0, 2m_0)$ for which it is unclear whether the broadcast task is possible. It is therefore of interest to investigate tighter bounds for this problem. Allowing probabilistic placement of bad nodes in the network as in [4] may be another topic of future research.

Acknowledgement. We sincerely thank Seth Gilbert and Dan Alistarh for very helpful discussions.

References

[1] M. Bertier, A.-M. Kermarrec, and G. Tan. Brief Announcement: Reliable Broadcast Tolerating Byzantine Faults in a Message-Bounded Radio Network. *Proc. of 22nd International Symposium on Distributed Computing (DISC 2008)*.

[2] V. Bhandari and N. H. Vaidya. On reliable broadcast in a radio network. *ACM PODC 2005*.

[3] V. Bhandari and N. H. Vaidya. On reliable broadcast in a radio network: A simplified characterization. Technical Report, CSL, UIUC, May 2005. Available at http://www.crhc.uiuc.edu/wireless/papers/bcastaddendum.pdf

[4] V. Bhandari and N. H. Vaidya. Reliable broadcast in wireless networks with probabilistic failures. In *INFOCOM 2007*.

[5] V. Bhandari and N. H. Vaidya. Reliable broadcast in a wireless grid network with probabilistic failures. Technical Report, CSL, UIUC, Oct. 2005.

[6] M. Blaum and H. van Tilborg. On t-Error Correcting/All Unidirectional Error Detecting Codes. *IEEE Transactions on Computers*, pages 1493 – 1501, 1989.

[7] M. Čagalj, S. Čapkun, R. Rengaswamy, I. Tsigkogiannis, M. Srivastava, J-P. Hubaux. Integrity (I) codes: Message Integrity Protection and Authentication over Insecure Channels. *Proc. of IEEE Symposium on Security and Privacy (S&P 2006)*, Oakland, USA.

[8] G. Chockler, M. Demirbas, S. Gilbert, C. Newport, and T. Nolte. Consensus and collision detectors in wireless ad hoc networks. *ACM PODC* 2005.

[9] S. Dolev, S. Gilbert, R. Guerraoui, and C. Newport. Gossiping in a multichannel radio network: An oblivious approach to malicious interference. In *Proc. of the the 21st International Symposium on Distributed Computing (DISC)*, 2007

[10] S. Gilbert, R. Guerraoui, and C. Newport. Of Malicious Motes and Suspicious Sensors: On the Efficiency of Malicious Interference in Wireless Networks. *ACM PODC 2006*.

[11] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister. System architecture directions for network sensors. *ACM ASPLOS 2000*, Cambridge, November 2000.

[12] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: attacks and countermeasures. *Proc. of First IEEE International Workshop on Sensor Network Protocols and Applications 2003*.

[13] C-Y. Koo. Broadcast in radio networks tolerating byzantine adversarial behavior. *ACM PODC 2004.*

[14] C-Y. Koo, V. Bhandari, J. Katz, and N. H. Vaidya. Reliable broadcast in radio networks: The bounded collision case. *ACM PODC 2006.*

[15] E. Kranakis, D. Krizanc, and A. Pelc. Fault-tolerant broadcasting in radio networks. *Journal of Algorithms* 39(1): 47-67, 2001.

[16] A. Pelc and D. Peleg. Broadcasting with locally bounded byzantine faults. *Information Processing Letters*, 93(3):109-115, 2005.

[17] A. Pelc and D. Peleg. Feasibility and complexity of broadcasting with random transmission failures. *ACM PODC 2005*.