Handling Hidden Terminals in Sensor Networks Over White Spaces

Andrew Martin
Computer Science, CUNY–Queens College
andrew.martin72@qmail.cuny.edu

Mahbubur Rahman
Computer Science, CUNY–Graduate Center & Queens College
mdmahbubur.rahman@qc.cuny.edu

***Abstract*—This paper presents a comprehensive analysis of the hidden terminal problem in Sensor Network Over White Spaces (SNOW), a low-power, wide-area network architecture utilizing TV white space frequencies. The hidden terminal problem, where nodes unaware of each other's existence cause transmission interference/collisions, poses significant challenges for the scalability of SNOW. We analyze traditional solutions, including Floor Acquisition Multiple Access (FAMA) protocols, highlighting their limitations in the SNOW context, as well as current research attempts to limit the impact of the hidden terminal problem. We then propose a novel distributed protocol based on information sharing and aggregation that enables nodes to collaboratively build conflict maps and communicate them to the base station without introducing significant overhead. Our approach integrates local conflict sensing, attachment-coded information sharing, and adaptive conflict resolution at the base station. This solution addresses both static hidden terminal scenarios and dynamic network changes while maintaining SNOW's energy efficiency and scalability.**

***Index Terms*—IoT, SNOW, hidden terminals, scalability**

I. Introduction

As wireless technologies continue to evolve, the demand for long-range, energy-efficient communication protocols has grown significantly. Large-scale, wide-area networks have numerous applications where thousands of sensors over long ranges can support numerous applications, including but not limited to sensing systems for farm monitoring, urban cities, etc. Sensor Network Over White Spaces (SNOW) [1] has emerged as a promising network architecture for a large-scale Internet of Things (IoT) deployment that leverages the powerful propagation characteristics of TV white space frequencies to enable long-range, low-power communication. However, as SNOW networks attempt to scale to accommodate thousands to hundreds of thousands of nodes, the hidden terminal problem becomes increasingly significant to the scalability of SNOW, as it will threaten network reliability and performance.

The hidden terminal problem occurs when two communication nodes, unable to detect each other's transmissions due to being out of range of each other, simultaneously send data to an access point, causing packet collision interference and thus, data loss. In traditional wireless networks, this problem is often addressed through handshake mechanisms such as Request-to-Send/Clear-to-Send (RTS/CTS) protocols. However, these approaches introduce significant overhead, making them unsuitable for the energy and scalability requirements of SNOW and are disabled by default in many wireless systems.

The hidden terminal problem primarily appears in SNOW through two primary gateways. **(1)** control channel congestion during mass node joining events, where multiple nodes compete for limited control channels, and **(2)** dynamic network changes that invalidate static assignments as nodes move or environmental conditions evolve. These challenges become more influential as the network scales, potentially limiting SNOW's practical deployment capacity.

This paper makes the following contributions:

1) Comprehensive analysis of the hidden terminal problem in the context of SNOW networks
2) A critical evaluation of traditional solutions and their applicability to SNOW
3) A consideration of current hidden terminal research and the types of minor optimizations that can be added to improve SNOW's reliability
4) A suggested dynamic cognitive subcarrier protocol that can leverage SNOW's design philosophies and encourage scalability

II. Background

A. TV White Space

B. What is TV White Space?

TV white space refers to frequencies in the television band unused by broadcasters. This spectrum has been declared usable by the Federal Communication Commission (FCC) for unlicensed devices operating as secondary users, provided they do not interfere with TV stations or licensed users [2]. TVWS has lower frequencies (54-698 MHz), so communication on these frequencies has excellent propagation qualities over long distances and through obstacles [3].

It should be noted that due to FCC mandates, in order to protect primary licensed users, an unlicensed device must query a spectrum database in order to learn about available TV white space channels in its area [4]. A typical TVWS setup utilizes a base station that consists of a signal transmitter that will register with the spectrum database and continuously share its GPS location in order to receive information from unoccupied channels in the TV spectrum at that time and location [5].

C. Snow Architecture

Sensor Network Over White Spaces (SNOW) is a low-power wide-area network (LPWAN) that leverages TVWS for long-range, energy-efficient communication. SNOW employs a Distributed implementation of Orthogonal Frequency Division Multiplexing (D-OFDM), where the base station (BS) splits available TVWS into narrow orthogonal subcarriers that can carry data to and from multiple nodes simultaneously [1].

The SNOW architecture consists of

- **Base Station (BS):** The BS is line-powered, internet-connected, and is equipped with two half-duplex radios, one for receiving asynchronous concurrent transmissions, and for sending data to different nodes concurrently. The BS bears the responsibility for subcarrier assignment, spectrum database queries, and overall network coordination
- **Sensor Nodes:** Power-constrained devices equipped with half-duplex radios, deployed in a one-hop star topology around the BS. Nodes rely on the BS for subcarrier allocation once joined into the network and do not perform spectrum sensing or database queries themselves.

Communication in SNOW follows two primary modes:

1) **Downward Communication:** The BS continuously transmits control information, including subcarrier assignments, acknowledgments (ACKs), and scheduling updates. This is done concurrently with uplink reception using a dual-radio setup. When a new node joins the network, it communicates via a dedicated join subcarrier. The BS assigns subcarriers based on node location and existing subcarrier usage to minimize hidden terminal interference.
2) **Upward Communication:** Nodes transmit asynchronously and independently on their assigned subcarriers. The BS uses a Global FFT (G-FFT) engine to continuously decode signals from all subcarriers, enabling concurrent reception from multiple nodes. After receiving a packet, the BS sends an ACK back on the same subcarrier, which also acts as a busy tone to suppress potential hidden terminal interference.

The BS assigns unique subcarriers to nodes when possible. When the number of available subcarriers is less than the number of nodes, the BS performs assignment to minimize interference between nodes and their hidden neighbors.

D. Hidden Terminal Problem

The hidden terminal problem is a fundamental challenge in wireless networks where carrier-sense mechanisms are employed for medium access control. In traditional wireless networks, nodes use Carrier Sense Multiple Access (CSMA) to detect if a desired communication channel is busy for transmitting. Nodes attempt to ”listen” before speaking/sending data. If a channel is busy, the device will wait until the channel is clear and attempt to send again at some other time. However, CSMA is not always sufficient for avoiding collisions with a BS. If nodes cannot detect each other’s transmissions due to out-of-range limitations or physical obstructions, then CSMA is not effective in mitigating the hidden terminal problem.

If there are three nodes, A, B & C, assume B is an access point. A communicates with B, and C communicates with B and vice versa, but A and B cannot ”see” or communicate with each other (this could be due to range or some sort of physical obstruction, where A & B although having a line of sight to the access point, will be obstructed to each other). If A & C send packets simultaneously to access point B, their messages will interfere and corrupt one another. A & C are unable to detect each other’s signals and cannot enable CSMA protocols, and this cannot be aware of the collision that they are engaging in. This situation represents the hidden terminal problem illustrated in Figure 1.

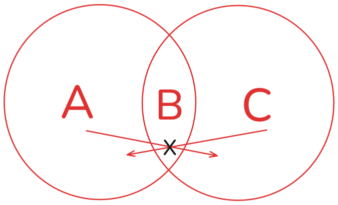

Fig. 1. A & C have their messages collide at the BS

It has been claimed that the Hidden Terminal Problem is one of the top problems that oppose a seamless wireless network [6].

E. Classical Solution/FAMA protocols

To mitigate the hidden terminal problem, early wireless multiple access protocols (MAC) utilized variations of handshake protocols. One prominent class of these protocols is known as Floor Acquisition Multiple Access (FAMA). The key idea behind FAMA is for a device to acquire the channel (or the ”floor”) before transmitting data, ensuring that a node doesn’t compete with nodes outside of its range that it may conflict with [7].

In a typical FAMA protocol, a node that wishes to transmit data initiates a handshake mechanism using Request-to-Send/Clear-to-Send protocol.

1) A node wishing to send data sends an RTS packet to the BS
2) If the channel is clear, the BS responds with a CTS packet
3) Upon receiving the CTS, the sender transmits its data
4) Other nodes that overhear the CTS defer their transmissions

This handshake resolves hidden terminal conflicts because even if a hidden node cannot hear the RTS, it will still hear the CTS from the receiver and avoid transmitting during the data exchange. In this way, FAMA extends the reach of CSMA-like behavior by leveraging the receiver’s ability to broadcast its intent to receive. This works as long as every node is within the base station’s range.

However, while FAMA protocols are effective in reducing collisions, they incur significant overhead. Every data transmission is preceded by at least two control packets (RTS and CTS), and the overall throughput decreases as the ratio of control to data packets increases. This makes FAMA less suitable for networks with short data packets, high contention,

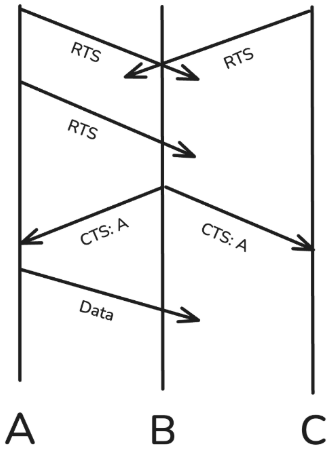

Fig. 2. A & C send out RTS's until they hear a CTS

or strict energy constraints, characteristics commonly found in large-scale sensor networks. By default, RTS/CTS exchanges are disabled in WLANs.

F. Hidden terminal Problem in Relation to SNOW

1) **Long-Range Communication:** TVWS frequencies enable communication over longer distances, increasing the potential for hidden terminal scenarios as nodes may be deployed across a wider area.
2) **Star Topology:** All nodes communicate directly with the BS in a single-hop manner, creating a dense communication pattern centered on the BS.
3) **Subcarrier Assignment:** The BS assigns subcarriers to nodes based on their spatial distribution, but if hidden terminal relationships are not accurately identified, nodes unknowingly hidden from each other may be assigned the same subcarrier.
4) **Half-Duplex Operation:** Nodes operate in half-duplex mode, preventing them from simultaneously transmitting and listening, which limits their ability to detect potential collisions.

SNOW particularly has two key hidden terminal bottlenecks:

1) **Control Channel Congestion During Node Joining:** In most scenarios considered for SNOW, it is assumed that nodes are slowly joining or are already joined to a BS with their location information. However, if the assumption is that a large number of nodes desire to join the SNOW network at once, they must discover the BS and obtain control channel information by passively listening during the downlink phase. However, during the uplink phase, nodes must wait for an idle control channel before transmitting their ID and location. Since all joining nodes compete for access to a single channel (or some set of channels), this process acts as a significant bottleneck as the network grows in scale, due to the nodes being unable to use the numerous advantageous orthogonal subcarriers, and can introduce hidden terminal scenarios.
2) **Dynamic Cognitive Subcarrier Assignment:** As nodes move or environmental conditions change, hidden terminal relationships evolve, requiring updates to subcarrier assignments. Computing optimal assignments in real-time becomes challenging as the network grows, since this problem is computationally equivalent to graph coloring, which is NP-complete.

G. Related work and Potential Directions for SNOW

This section reviews existing approaches to hidden terminal mitigation and explores how they might be adapted or extended for SNOW networks. We examine both established solutions and emerging techniques, identifying opportunities for novel applications in the SNOW context.

1) **Floor Acquisition Multiple Access (FAMA) Protocols** The hidden terminal problem has been addressed through various Floor Acquisition Multiple Access (FAMA) protocols since the early development of wireless networks [7]. The most prevalent solution is the Request-to-Send/Clear-to-Send (RTS/CTS) handshake mechanism, where nodes acquire channel control before data transmission. FAMA protocols are more clearly covered in the earlier section **Hidden Terminal Problem**.
   A traditional RTS/CTS protocol could be applied during SNOW's joining phase when nodes compete for control channels. However, the overhead concerns remain significant for large-scale deployments.
2) **Time Division Multiple Access (TDMA)/Polling Approaches** Earlier iterations of SNOW [8] utilized a polling method to facilitate communication that would largely mitigate the hidden terminal problem. When nodes exceeded the available subcarriers, nodes were grouped into sets of sizes corresponding to the number of available subcarriers, where each node had a unique subcarrier. Base stations used the downward phases to poll groups in a round-robin fashion, allowing for one group to transmit at a time during each upward phase, with nodes sleeping between their assigned transmission windows. This was always possible since the base station could utilize dedicated control channels for downward communication and avoid spectrum conflict.
   However, this introduces scalability limitations as SNOW grows larger, only allowing for some number of nodes to communicate simultaneously, creating an apparent bottleneck. Nodes or groups of nodes that wanted to communicate would be delayed in communication, while inactive groups and nodes were given time slots to communicate. Many IoT applications have sporadic, asynchronous traffic that doesn't align well with scheduled group transmissions, and SNOW, which depends on being a largely asynchronous-style network, doesn't implement this in the current iteration. Imple-

mentation of any kind of TDMA, or turn-based polling, isn't recommended as SNOW grows as a network.

3) **ZigZag Decoding** ZigZag is an 802.11 receiver design that mitigates hidden terminal interference by decoding collisions instead of avoiding them. It exploits asynchrony across successive collisions, strategically selecting interference-free chunks to iteratively reconstruct colliding packets. When two senders transmit simultaneously, their packets may overlap with different offsets due to random transmission jitters. ZigZag leverages these offsets to identify interference-free portions in one collision and uses them to decode and subtract interference in another. This process continues iteratively until both packets are fully recovered. By operating at the receiver without requiring modifications to the MAC layer or sender behavior, ZigZag achieves throughput comparable to scheduled transmissions while significantly reducing packet loss rates in hidden terminal scenarios [9].

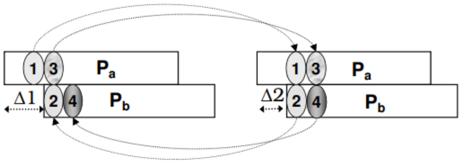

Fig. 3. This figure illustrates ZigZag decoding, a method where interference-free chunks of collided transmissions are used to iteratively decode overlapping segments. Nodes A and C transmit despite being hidden terminals, but ZigZag exploits timing offsets across collisions to resolve both signals. Adapted from Gollakota and Katabi [9].

However, chunk decoding is not viable in OFDM systems, because the time domain signals are modulated, transformed by IFFT, which means the packets are not chunk-decodable. To decode a packet, the entire interference-free packet must be processed through FFT, not just a section of the packet, regardless of whether it's interference-free or not [10]. ZigZag, as it stands, cannot be implemented into SNOW as SNOW is a D-OFDM system.

4) **Differential Overlap Decoding** The intuition behind ZigZag is not suitable for OFDM-based systems. Time domain signals in OFDM systems become modulated symbols transformed by IFFT. Which means that the chunk-by-chunk decoding schema of ZigZag fails, since these modulated symbols are not chunk-decodable. The entire packet must be processed via a FFT without interference. Differential Overlap Decoding (DOD) takes two colliding transmissions from two nodes, such that each node has different offsets to one another, such that the received collided messages are not equal. DOD takes the overall factors of modulation, channel, and collision into a system of linear equations and decodes by solving said equations. This system method in practice can be treated as a hidden terminal solution that results in a 3-5 dB network degradation [10].

   There is potential to add DOD on top of regular communications, handshaking methods on control channels, in order to mitigate hidden terminal effects. DOD would not be suitable for distributed networks that emphasize low power and low computation algorithms, since DOD requires the utilization of matrix inversions to decode interfered packets. However, SNOW operates in a single-hop star topology, with a line-powered base station, SNOW feasibly can implement a DOD-based approach. SNOW would need to add preamble signals to node communication in order to identify nodes in order to identify participants in a collision. SNOW does, however, utilize a per-packet ACK methodology; this no longer becomes guaranteed if slower algorithms are needed to decode messages. Some modification of the MAC layer and PHY layer is needed to accomplish this. This method has potential within SNOW as SNOW utilizes OFDM. This paper considers the improvements that DOD can provide via a simplified probabilistic simulation in the evaluation section.

5) **Hidden Terminal Conflict Sensing** In line with the idea of terminals sharing communication lists [**?**], terminals could share their communication attempts with another terminal. Specifically, in comparing failed attempts with hidden terminals. If a node can determine that it conflicted with a node hidden from it some number of times, it could switch to a channel, or use its shared list to a nearby terminal that is able to get through to request from the base station an updated subcarrier. This approach acknowledges the real-time circumstance-changing factors of wireless communication, a node may go to sleep for a long time and that channel can be reassigned, and when it wakes up, it may try to use it's old channel but may be conflicted, and thus still has a way to request a new channel or figure out that it's being conflicted. Identifying failed messages is simple because every received message is acknowledged by the base station. [1]. This can also assist in dynamic location updates because, as a node travels and listens and broadcasts information, it can update nearby terminals of its relative location without needing to actually calculate its coordinate position. This approach is considered and evaluated in this paper.

6) **Attachment Coding** FAST (Full-duplex Attachment System) contains a PHY layer protocol called Attachment Coding, which can be applied in OFDM-based WLANS. Channel information can be encoded in Attachment Coding, where Attachments can be attached to data transmission without reducing the decoding capacity of the data packets, which can be easily canceled out at receivers using interference cancellation [11]. This approach suggests that nodes can transmit control information across the network to each other, including to nodes out of direct range, which will allow for more complex and aware FAMA protocols. Nodes can

know to wait for a node to finish transmitting, even if they can't directly see its transmission. While this is efficient in OFDM systems, it conflicts with SNOW's D-OFDM architecture, where nodes are assigned isolated subcarriers, as opposed to utilizing/listening to the entire subcarrier spectrum at once. FAST assumes full-band signaling and full-duplex radios, which are not implemented in SNOW's energy-constrained, half-duplex nodes.

As full-duplex technology improves and potentially becomes more energy efficient, a future direction for SNOW could explore a PHY-layer per-subcarrier metadata encoding for lightweight conflict reporting inspired by FAST, or for nodes to listen to the entire subcarrier spectrum at once.

III. Proposed Approach: Information Sharing and Aggregation

A. Problem Statement

Traditional wireless networks use handshake protocols like RTS/CTS to mitigate hidden terminals, but SNOW does not implement RTS/CTS due to the overhead and scaling concerns. Instead, the BS currently attempts to assign subcarriers such that nodes known to be hidden from one another do not share the same channel. However, dynamic changes (such as new node joins, mobility, or environment changes) can invalidate or increase the severity of the hidden terminal groupings of these static assignments, and SNOW nodes do not communicate peer-to-peer to coordinate.

There is a need for a protocol to be adapted to SNOW that enables the following:

1) Accurate identification of hidden terminal relationships
2) Efficient communication of this information to the BS
3) Dynamic adaptation of subcarrier assignments
4) Minimal overhead on energy-constrained nodes

B. Joining improvements

Nodes, when joining on some set of control channels, should send redundant information packets with ID and location, still utilizing local CSMA, but by utilizing something like DOD or DNR, the BS may be able to decode these short join packets from hidden terminals.

C. Proposed Approach, Cognitive Protocol

This is a distributed protocol integrated with SNOW's MAC layer that enables nodes to collaboratively build a map of hidden terminal relationships and communicate this information to the BS. Our approach consists of three main components:

1) **Local Conflict Sensing** Nodes monitor transmission outcomes to detect potential collisions.
2) **Neighbor List Exchange** Nodes share information about nodes in their range that are detectable
3) **Conflict Resolution** The BS utilizes conflict sensing reports and neighbor list exchanges to update subcarrier assignments for nodes

D. Local Conflict Sensing

In SNOW, nodes expect an acknowledgment (ACK) from the BS for each successful transmission. This can be leveraged for detecting conflict:

1) Each node maintains a counter of its personal consecutive failed transmissions (no ACK received from BS)
2) When this counter exceeds a predefined threshold, the node flags a potential hidden terminal conflict and adds its personal conflict to its broadcasted neighbor list
3) The node includes this conflict information in subsequent transmissions to the BS once communication is reestablished and in broadcasted neighbor reports
This approach requires minimal additional broadcasted data at each node, just some counters indicating potential conflicts and neighboring nodes. This may require updated hardware; nodes in SNOW are currently half-duplex, they can only listen or broadcast, and this may need full-duplex radios, which can listen and broadcast at the same time.

E. Neighbor List Exchange

To identify hidden terminal relationships, nodes periodically broadcast a list of other nodes they can detect. This neighboring list can be low power to only reach some predefined range, so that it doesn't overpower data transmissions.

1) Each node maintains a neighbor list containing identifiers of nodes it has detected transmissions from.
2) During regular communication with the BS, nodes occasionally include this neighbor list as an attachment to their data packets. Since these packets are compact, this should be a small addition that carries critical information.
3) The BS aggregates these lists to build a comprehensive map of which nodes can and cannot detect each other. As well as which nodes are currently in active conflict (which cannot be captured based solely on location data, but this captures dynamic behavior, such as infrequent broadcast behavior).

To minimize overhead, nodes can encode their neighbor lists as bitmasks, where each bit represents whether a particular node is detectable. And the amount of conflicts it has personally experienced can be an integer byte appended to its report.

The design philosophy is to introduce minimal additions to SNOW: short listening periods and tiny control signals on the existing radio.

F. Conflict resolution:

1) The BS maintains a hidden terminal graph, where edges connect nodes that cannot detect each other.
2) When updating subcarrier assignments, the BS aims to assign different subcarriers to nodes connected by edges in this graph.
3) The BS prioritizes assignment updates for nodes reporting high conflict rates.

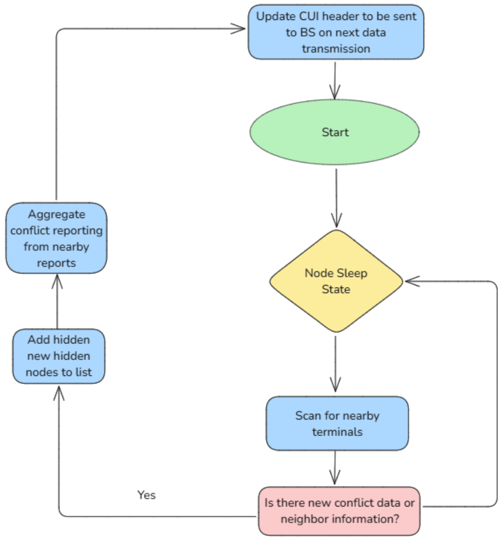
Update CUI header to be sent to BS on next data transmission
Start
Aggregate conflict reporting from nearby reports
Node Sleep State
Add hidden new hidden nodes to list
No
Scan for nearby terminals
Yes
Is there new conflict data or neighbor information?

Fig. 4. Node behavior

4) To avoid the ”ripple effect”[12], where channel reassignments cascade through the network, the BS performs holistic optimization rather than isolated changes.

These conflict reports are highly advantageous, a node can eventually get a report out to the BS or request help with assignments based on repeated confirmed conflicts. This data would be relatively small, a bit mask of nearby nodes and some byte-sized integer counters of nearby reported conflicts, which would be a header added to each data packet, which is normal in wireless networks. Base stations can match up times of conflicts and aggregate reports to identify key subcarrier conflicts between hidden nodes. The BS can also keep track of node behavior, if a node only transmits during a certain time, or once a day, etc., it can de-prioritize its subcarrier assignment. It is important that the BS maintains subcarrier assignments and that the nodes themselves do not attempt to switch subcarriers, in order to avoid exacerbating the hidden terminal problem. Uncoordinated subcarrier switching can lead to two major issues: ripple effects, where a node's decision to switch channels triggers a chain reaction of adjustments in neighboring nodes in an unideal manner. Channel Oscillation, where nodes alternate between an over-utilized (”bad”) channel and an underutilized (”good”) one, eventually causing the previously good channel to degrade due to crowding, and nodes then subsequently changing from the current channel (which is now over-utilized) back to the original channel (now underutilized)[12]. By enforcing centralized control at the BS, SNOW preserves subcarrier stability and avoids these systemic instabilities.

IV. Evaluation

A. *Simulation Framework and Methodology*

Simulations were conducted to evaluate the effectiveness of hidden terminal mitigation techniques in SNOW networks. The evaluation focuses on two primary approaches: Differential Overlap Decoding (DOD) for collision recovery and Dynamic Subcarrier Reassignment for proactive conflict avoidance. The simulations were implemented in Python using discrete-event simulation methodologies to model realistic network conditions and node behaviors.

B. *Experimental Setup*

DOD Performance Evaluation:

- **Network Scale:** 10,000 nodes with 29 available subcarriers
- **Traffic Model:** Packet generation with 0.002 packets per node per time step
- **MAC Layer:** CSMA with 50% transmission attempt probability
- **Retransmission Policy:** Maximum 2 retransmissions per packet with 80% retry probability
- **DOD Parameters:** 75% decoding success rate, 70% probability of different transmission offsets
- **Simulation Duration:** 1,000 time steps with deterministic traffic patterns for fair comparison

Dynamic Reassignment Evaluation:

- **Network Topology:** 1,000 nodes in 5,000×5,000 unit area with star topology
- **Mobility Model:** 30% mobile nodes with random waypoint movement (±10 units/step)
- **Transmission Range:** 300 units (1.13% area coverage per node)
- **Subcarrier Configuration:** 10 orthogonal subcarriers (100 nodes per subcarrier average)
- **Reassignment Delays:** 5 time steps for neighbor information propagation + 10 time steps for base station processing

C. *DOD Implementation Results*

1) DOD Algorithm Implementation: Our DOD simulation implements a probabilistic state machine that models the theoretical DOD process without requiring full OFDM signal processing. The implementation captures the following key mechanisms:

R1–R2 Collision State Tracking: The base station maintains a data structure `subcarrier_dod_R1_candidates[sc_id]` to store unique collision pair keys for each subcarrier.

1) **R1 Event (First Collision):** When two packets collide, a pair key `tuple(sorted([(node1, pkt1), (node2, pkt2)]))` is generated and stored with a timestamp.
2) **R2 Event (Second Collision):** If the same pair collides again, a DOD recovery attempt is triggered.

3) **Stale R1 Cleanup:** R1 entries older than 50 time steps are purged via a timeout mechanism.

Probabilistic DOD Success Modeling: The code models DOD success through independent probability checks:

- **Timing Offset Condition:** Each R2 event has a 70% probability of satisfying the different-offset requirement.
- **Decoding Success Condition:** Given a valid offset, there's a 75% chance the signal separation succeeds.
- **Combined Success Rate:** The theoretical DOD success rate is thus $0.7 \times 0.75 = 52.5\%$ for valid R2 collisions.

Algorithm 1: DOD Collision Resolution

```
Input : Subcarrier ID $sc\_id$, Collided Packet Pair Key $k$, Current Time Step $t$
if $k \in subcarrier\_dod\_R1\_candidates[sc\_id]$ then
    if random() < PROB_DIFFERENT_OFFSET_FOR_R2 then
        if random() < DOD_SUCCESS_PROB_GIVEN_R1_R2 then
            Increment packets_successful_via_dod by 2 // DOD success: both packets recovered
        else
            // DOD failed at second condition
            Schedule retransmissions or treat as a failed attempt
    else
        // DOD failed due to offset condition
        Schedule retransmissions or treat as a failed attempt
else
    // First time collision seen
    subcarrier_dod_R1_candidates[sc_id][k] $\leftarrow$ { "timestamp": $t$ }
```

2) Packet Delivery Performance Results: The simulation demonstrates DOD's effectiveness through statistical comparison. This seems in line to get a relatively flat increase in network throughput and packet delivery ratio, given that DOD suggests that the system modification can be treated as a 3 - 5 dB decrease [10].

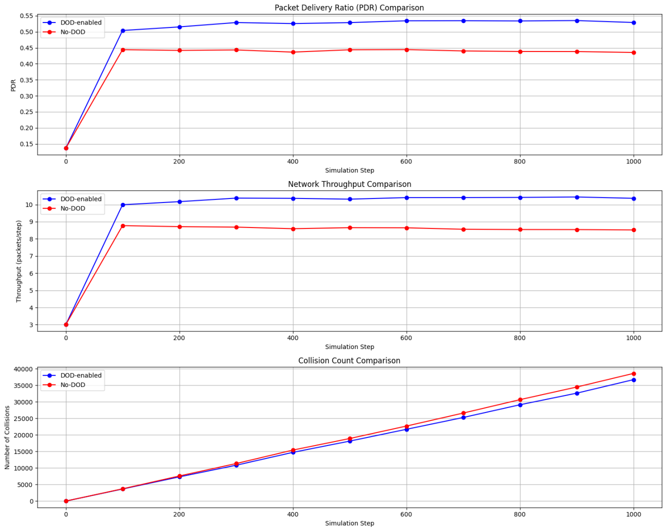

Fig. 5. Various network comparison metrics. It is seen that DOD's effectiveness can lead to a total Packet Delivery Performance improvement in the range of 10%.

3) Packet Loss Analysis: The simulation's packet loss breakdown reveals the distribution of failure modes in DOD-enabled networks:

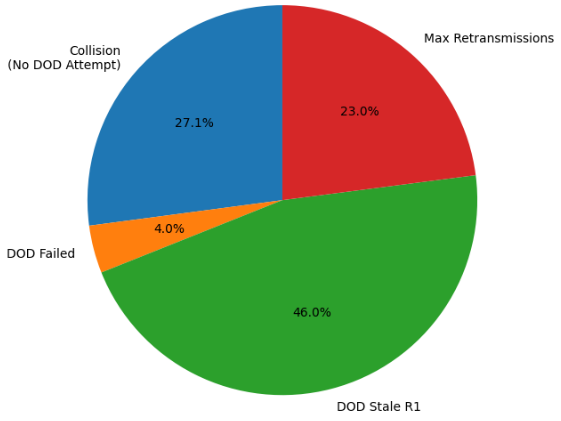

Fig. 6. Dod-enabled Network Packet Loss Breakdown

1) **Collision (No DOD Attempt):** 27.1% - Collisions involving more than two packets or single-occurrence collisions
2) **DOD Stale R1:** 46.0% - R1 collision candidates that exceeded the 50-step timeout before R2 occurred
3) **Max Retransmissions:** 23.0% - Packets dropped after exceeding retry limits
4) **DOD Failed:** 4.0% - DOD attempts that failed due to unsuccessful decoding or identical offsets

The stale R1 timeout represents the largest source of packet loss (46%), indicating that many collision pairs never get a second chance for DOD recovery. This suggests that the 50-step timeout may be too conservative and that traffic patterns don't naturally create R1-R2 collision sequences. Retransmission in the MAC protocol may need to be adjusted to create more R1-R2 retries. In addition, adaptive timeout strategies could significantly improve DOD effectiveness. Additionally, if more complex preamble strategies were used to identify all nodes within a collision via some combination of preamble and postamble signatures, then there is more potential to decode more complex collisions with more nodes.

D. Dynamic Subcarrier Reassignment Results

1) Hidden Terminal Detection and Mitigation: An important part of a network is adaptability to dynamic factors such as movement. This is the dynamic reassignment simulation, which implements finding hidden terminal relationships through spatial analysis rather than signal-based measurements. The algorithm builds a comprehensive neighborhood map for each node. Nodes identify all other nodes within transmission range (nodes in real circumstances would send occasional packets identifying themselves to neighbors). Nodes identify neighbors of neighbors, or a ”two-hop neighbor”, i.e, a device in a neighbor's list that is not in its immediate range. If a two-hop neighbor is using the same subcarrier as a target node, that is a potential hidden terminal to said node.

Algorithm 2: Detect Hidden Terminal Collisions

```
Input: Set of nodes, each with subcarrier, neighbor,
       and two-hop neighbor lists
foreach node $n$ in $nodes$ do
    foreach $t$ in $n.two\_hop\_neighbors$ do
        $t\_node \leftarrow$ nodes$[t]$ if
         $n.subcarrier == t\_node.subcarrier$
         then
            common_neighbors $\leftarrow$
             $n$.neighbors $\cap$ $t\_node$.neighbors if
             $common\_neighbors \neq \emptyset$ then
                Append $t$ to $n$.conflict_reports
```

This simulation was tested with 1000 nodes and 10 subcarriers. 30% of nodes were moving in a random walk fashion in a 5000 x 5000 grid. It is assumed that nodes can share their neighbor lists and conflict reports within 5 time steps, and that the base station can process, receive, and distribute a reassignment if necessary within 10 time steps.

Subcarrier reassignment greedily selects the optimal subcarrier for a node by examining nearby nodes within transmission range. It counts how many nodes are using each subcarrier and chooses the one with the least overlap to minimize interference. This is preferable to choosing a generally underutilized subcarrier or low-conflict subcarrier in order to maximize CSMA and maintain spatial subcarrier zones.

Algorithm 3: Subcarrier Reassignment Based on Local Density

```
Input: Node ID $n$, List of nearby node IDs $N$, Total
       number of subcarriers $S$, Transmission range $r$
nearby_nodes $\leftarrow$ [nodes$[n\_id]$ for $n\_id \in N$]
 subcarrier_proximity_counts $\leftarrow [0] * S$
foreach node $m$ in $nearby\_nodes$ do
    if $distance(n, m) \leq r$ then
        subcarrier_proximity_counts[$m$.subcarrier]++
n.subcarrier $\leftarrow$
 $\arg\min$(subcarrier_proximity_counts)
```

Subcarrier reassignment is triggered when a mobile node is detected in a hidden terminal collision and has not been recently assigned (to prevent oscillation), and the conflict information has propagated through the network (15 time step delay). This greedy local optimization approach assigns each node to the subcarrier with the fewest nearby users, minimizing local interference without global optimization.

2) Performance Results: In comparison to a baseline collision rate determined by initial random assignment, dynamic reassignment led to a collision reduction of 15% - 20% per time step in cumulative hidden terminal collisions.

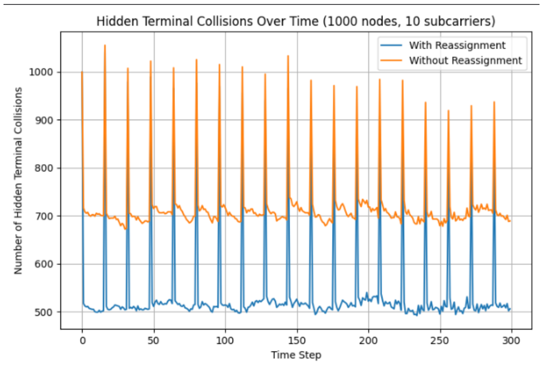

Fig. 7. Collisions per timestep

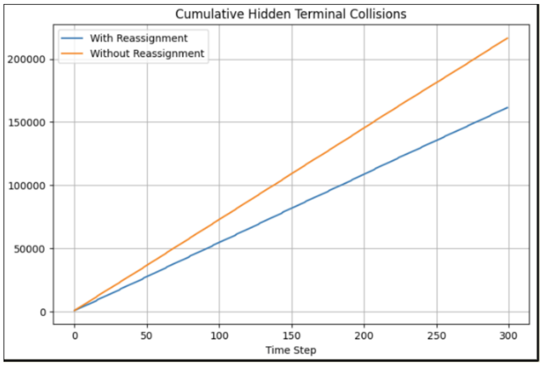

Fig. 8. Total collisions across the system as time step increases

With the given assumptions, this system shows promise in being able to assign local reassignments without base stations needing to individually check in with nodes and know the exact locations of a node; all it needs to do is view which nodes are nearby, and it gets a rough idea of the relative location.

E. Evaluation Improvements

Probabilistic models were used to simulate the network, it would be more realistic and ideal to use a more accurate system model, one provided in Qualnet or NS-3, that can simulate network conditions. There is no concept of packet length, OFDM modulation, or SNR/BER in the system, it's a binary model that evaluates on message received or message conflict. This is considered for future directions of SNOW dynamic subcarrier optimization improvements.

Better algorithms for subcarrier assignments can be explored, or proofs that this greedy heuristic algorithm has a close approximation to the optimal solution can be explored. The direction of using some sort of reinforcement model can also be explored, the base station can evaluate activity weight of a subcarrier, typical patterns of nodes, etc, and take that all into account when assigning subcarriers dynamically.

V. Conclusion

A. Summary of Findings

This research presents a comprehensive analysis of the hidden terminal problem in Sensor Network Over White Spaces (SNOW) and proposes practical solutions to enhance network scalability and reliability. Through theoretical analysis and simulation-based evaluation, several key insights emerge that can guide the future development of large-scale IoT deployments using TV white space frequencies.

B. Key Contributions

1) Comprehensive Problem Analysis: This paper's analysis reveals that while SNOW's existing location-aware spectrum allocation effectively minimizes hidden terminal conflicts in static scenarios, two critical bottlenecks persist: control channel congestion during mass node joining events and dynamic network changes that invalidate static subcarrier assignments. These challenges become increasingly significant as networks scale toward the millions of nodes that SNOW is designed to support.

2) Evaluation of Existing Solutions: Upon evaluating traditional approaches, including FAMA protocols, TDMA/polling methods, and collision recovery techniques, current literature aggregations demonstrate that traditional RTS/CTS handshaking introduces unacceptable overhead for energy-constrained SNOW nodes. Polling-based approaches create scalability bottlenecks that conflict with SNOW's asynchronous design philosophy. ZigZag decoding, while effective in 802.11 networks, is incompatible with OFDM-based systems like SNOW.

3) Collision Recovery Approach: Implementation and evaluation of Differential Overlap Decoding (DOD) for SNOW networks shows promising results, achieving approximately 10% improvement in packet delivery performance. DOD can leverage SNOW's powerful, line-powered base station to perform computationally intensive matrix operations that would be infeasible for energy-constrained sensor nodes. However, our analysis reveals that 46% of packet losses stem from stale collision pairs that timeout before a second collision occurs, suggesting opportunities for MAC layer optimization and adaptive timeout strategies.

4) Dynamic Cognitive Protocol Framework: Proposition and evaluation of a distributed information sharing protocol that enables nodes to collaboratively build conflict maps through terminal-to-terminal information exchange. Our simulation results demonstrate a 15-20% reduction in hidden terminal collisions through dynamic subcarrier reassignment based on local neighborhood density optimization. This approach maintains SNOW's energy efficiency while providing adaptive responses to network changes.

5) Design Principles for SNOW Hidden Terminal Mitigation: Based on current research, several key design principles emerge. **Centralized Intelligence** SNOW's architecture naturally supports centralized optimization at the base station, which should be leveraged rather than attempting distributed solutions that burden energy-constrained nodes. **Minimum Node Overhead** Any solution must preserve SNOW's low-power design philosophy. Our proposed bitmask-based neighbor reporting adds minimal overhead while providing critical network state information. **Adaptive Learning** The base station should learn from node behavior patterns, traffic characteristics, and mobility patterns to make intelligent subcarrier assignment decisions that prevent rather than react to conflicts.

Practical short-term improvements can include enhancing the joining process by implementing redundant join packet transmissions with DOD-based collision recovery to assist with control channel congestion. A lightweight conflict report via deploying lightweight bistmask-based neighbor reports to provide the BS with real-time network channel usage and geographic information. Adaptive timeout mechanisms should be explored to optimize DOD usage.

C. Conclusions and Future Works

The current evaluation relies on probabilistic models that, while informative, lack the complexity of real-world wireless environments. Future work should implement comprehensive network simulations using established frameworks such as NS-3 or QualNet that incorporate realistic channel models, interference patterns, and protocol stack implementations. These simulations should model actual OFDM signal processing, including FFT operations, channel estimation, and bit error rates under varying SNR conditions. Additionally, incorporating realistic mobility models, environmental factors such as weather and terrain effects, and dynamic spectrum availability would provide more accurate performance predictions for SNOW deployments.

The centralized nature of SNOW's base station presents unique opportunities for intelligent network management through machine learning approaches. Future work should develop predictive models that anticipate hidden terminal conflicts based on historical patterns, node mobility traces, and environmental conditions. Reinforcement learning algorithms could optimize subcarrier assignments by learning from network performance feedback, potentially discovering assignment strategies that outperform traditional graph coloring approaches. Additionally, anomaly detection systems could identify unusual network behaviors that may indicate security threats or hardware failures, enabling proactive network maintenance.

Current SNOW architecture assumptions, particularly half-duplex operation and isolated subcarrier assignment, may be relaxed as technology advances. Future research should investigate the feasibility and benefits of full-duplex capable nodes, which could enable simultaneous transmission and collision detection. Exploring hybrid modulation schemes that use chunk-decodeable formats for control channels while maintaining OFDM for data transmission could combine the benefits of both approaches. Additionally, investigating scenarios where nodes can monitor multiple subcarriers simultaneously, inspired by FAST's [11] attachment coding concepts, could enable more sophisticated conflict detection and coordination.

References

[1] A. Saifullah, M. Rahman, D. Ismail, C. Lu, J. Liu, and R. Chandra, "Low-power wide-area network over white spaces," *IEEE/ACM Transactions on Networking*, vol. 26, no. 4, pp. 1–16, 2018. Presented in IEEE/ACM Transactions on Networking.

[2] Federal Communications Commission, "Promoting more efficient use of spectrum through dynamic spectrum use technologies." FCC 10-198, ET Docket No. 10-237, Nov. 2010. Notice of Inquiry, adopted and released November 30, 2010.

[3] M. Rahman and A. Saifullah, "A comprehensive survey on networking over tv white spaces," *Pervasive and Mobile Computing*, vol. 59, 2019.

[4] FCC, "Unlicensed operation in the tv broadcast bands," 2010.

[5] M. Z. Islam, J. F. O'Hara, D. Shadoan, M. Ibrahim, and S. Ekin, "Tv white space based wireless broadband internet connectivity: A case study with implementation details and performance analysis," *IEEE Access*, vol. 9, pp. 97418–97432, 2021.

[6] H. Matsuura, S. Kubo, and T. Fujii, "Proposal and evaluation of a hidden terminal identification method using terminal-to-terminal information exchange for highly sensitive carrier sensing in lpwan," in *2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall)*, pp. 1–5, IEEE, 2022.

[7] C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Solutions to hidden terminal problems in wireless networks," in *Proceedings of the ACM SIGCOMM Conference*, (Cannes, France), pp. 39–49, ACM, 1997.

[8] A. Saifullah, M. Rahman, D. Ismail, C. Lu, R. Chandra, and J. Liu, "Snow: Sensor network over white spaces," in *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems (SenSys '16)*, (Stanford, CA, USA), pp. 272–285, ACM, 2016.

[9] S. Gollakota and D. Katabi, "Zigzag decoding: Combating hidden terminals in wireless networks," in *Proceedings of the 2008 ACM SIGCOMM Conference*, (Seattle, WA, USA), pp. 159–170, ACM, 2008.

[10] J. Cao, F. Yang, L. Ding, L. Qian, and C. Zhi, "Differential overlap decoding: Combating hidden terminals in ofdm systems," in *2013 IEEE Wireless Communications and Networking Conference (WCNC)*, (Shanghai, China), pp. 3732–3736, IEEE, 2013.

[11] L. Wang, K. Wu, and M. Hamdi, "Combating hidden and exposed terminal problems in wireless networks," *IEEE Transactions on Wireless Communications*, vol. 11, no. 11, pp. 4204–4213, 2012.

[12] E. Ahmed, A. Gani, S. Abolfazli, L. J. Yao, and S. U. Khan, "Channel assignment algorithms in cognitive radio networks: Taxonomy, open issues, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 795–824, 2016.