Multi-Coil MI Based MAC Protocol for Wireless Sensor Networks

Computer Networks

Nishit Prajapati (202351106)
Samarth Acharya (202351125)
Pravit Soni (202351112)
Sivajeet Sabdakar (202351134)
Rajat Dhoot (202352325)

IIIT Vadodara
Gandhinagar Campus

November 2025

Ahmed, N., Zheng, Y. R., & Pommerenke, D. (2013). Multi-Coil MI Based MAC Protocol for Wireless Sensor Networks. Missouri University of Science & Technology.

Introduction

Checkout our Project Implementation: Github

We implemented **Multi-Coil MI Based MAC Protocol** for Wireless Sensor Networks

- Energy-efficient MAC protocol using Magneto-Inductive (MI) communication
- Three-dimensional coil system for directional transmission
- Suitable for both terrestrial and underwater sensor networks

Introduction

Project Methodology:

- Analyzed the research paper to understand the Multi-Coil MI MAC protocol architecture
- Evaluated implementation options and selected NS-3 network simulator due to hardware constraints
- Developed the simulation in NS-3.46, implementing the MAC protocol algorithm and comparing it with CSMA/CA

Understanding the Paper

Challenges in Wireless Sensor Network MAC Protocols:

- **Energy Wastage**
  - Idle listening: nodes continuously monitor the channel
  - Packet collision: simultaneous transmissions waste power
  - Packet overhearing: receiving packets not intended for the node
  - Control packet overhead: excessive handshaking increases energy consumption
- **Underwater Communication Limitations**
  - Slow propagation speed in acoustic communication
  - High path loss and multipath fading
  - Low data rates and high processing power requirements
  - Dynamic channel conditions and signal attenuation

Understanding the Paper

How the Paper Addresses These Challenges

- **Three-Coil System for Omnidirectional Communication**
  - Uses three orthogonal coils (X, Y, Z) to overcome directional limitations
  - All three coils listen simultaneously, but only one transmits at a time
  - REV packet sent three times (once per coil) to find optimal orientation
  - Automatically selects coil pair with strongest RSSI for communication
- **Five-State Energy-Efficient State Machine**
  - **Idle**: Ultra-low power mode (50 μA) - only receiver IC active
  - **Receive**: Packet decoding (200 μA)
  - **Channel Sensing**: Carrier detection before transmission (200 μA)
  - **Data Acquire**: Sensor data collection (250 μA)
  - **Transmit**: Packet transmission (1.12 mA)
- **Component-Level Power Management**
  - Individual control of microcontroller, receiver, transmitter, and sensors
  - Components turned OFF when not needed to minimize energy wastage

Understanding the Paper

How the Paper Addresses These Challenges

- **Efficient Packet Structure**
  - **REV (13 bytes)**: Handshake with coil information for optimal pairing
  - **ACK (5 bytes)**: Minimal acknowledgment with selected coil pair
  - **DATA (3-19 bytes)**: Variable length reduces control overhead
  - Each packet includes target ID matching to avoid unnecessary processing
- **Collision Avoidance Mechanisms**
  - Carrier sensing before transmission to detect channel availability
  - Directional nature of MI coils reduces interference range
  - Spatial reuse: different coil pairs allow concurrent transmissions
- **Magneto-Inductive Communication Advantages**
  - Works effectively in both terrestrial and underwater environments
  - Stable channel response without multipath fading issues
  - Magnetic permeability of soil/water similar to air
  - Lower propagation delays compared to acoustic communication

Algorithm Discussed in the Paper / Methodology

Packet Types and Structure:

Carrier	Preamble	Target ID	Packet ID	Tx ID	Tx Coil ID	EOF
1 byte	1 byte	4 bytes	1 byte	4 bytes	1 byte	1 byte

Reservation Packet

Carrier	Packet ID	Tx Coil ID	Rx Coil ID	EOF
1 byte	1 byte	1 byte	1 byte	1 byte

Acknowledgment

Carrier	Packet ID	Data Packets	EOF
1 byte	1 byte	1 byte – 16 bytes	1 byte

Data Packet

Algorithm Discussed in the Paper / Methodology

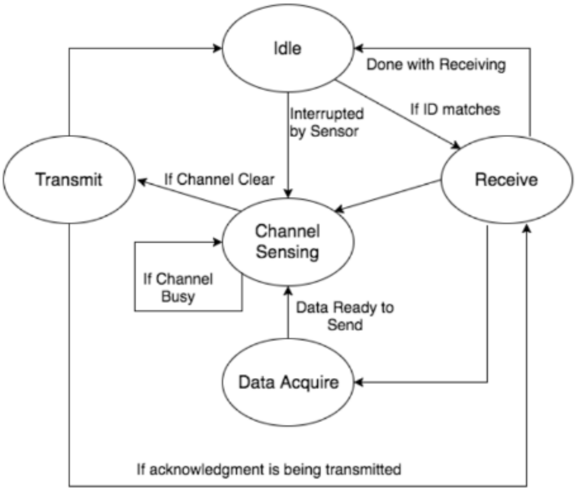
Five-State Machine with Component-Level Power Management:

TABLE III
ACTIVE AND IN-ACTIVE COMPONENTS IN EACH STATE

Working Mode	**Components**	**Power Mode**
Idle State	Micro-controller	Sleep
	Receiver Circuitry	ON
	Transmit Circuitry	OFF
	Sensors	OFF
Receive State	Micro-controller	ON
	Receiver Circuitry	ON
	Transmit Circuitry	OFF
	Sensors	OFF
Data Acquire State	Micro-controller	ON
	Receiver Circuitry	OFF
	Transmit Circuitry	OFF
	Sensors	ON
Channel Sensing State	Micro-controller	ON
	Receiver Circuitry	ON
	Transmit Circuitry	OFF
	Sensors	OFF
Transmit State	Micro-controller	ON
	Receiver Circuitry	OFF
	Transmit Circuitry	ON
	Sensors	OFF

Algorithm Discussed in the Paper / Methodology

State Transition Diagram:



Algorithm Discussed in the Paper / Methodology

Phase 1: Communication Initiation

- All nodes start in **Idle State** (50 μA) - listening on all three coils simultaneously
- When data is available, node transitions to **Data Acquire State** - collects sensor readings
- Node moves to **Channel Sensing State** - checks if channel is busy or clear
- If clear, enters **Transmit State** and sends **REV packet three times**
- Each REV (13 bytes) transmitted once per coil: X, Y, then Z
- REV contains: Target ID + Packet Type + Transmit Coil Information
- Three transmissions ensure at least one coil pair has strong signal, regardless of node orientation

Algorithm Discussed in the Paper / Methodology

Phase 2: Optimal Coil Pair Selection

- Destination node receives all three REV packets on its three coils
- Transitions to **Receive State** and compares RSSI values of all three receptions
- Selects coil pair with strongest signal strength (e.g., X-to-Z or Y-to-Y)
- Performs **Channel Sensing** and enters **Transmit State**
- Sends back compact **ACK packet (5 bytes)** using the selected best coil
- ACK contains: Packet Type + Selected TX Coil ID + Selected RX Coil ID
- Source receives ACK in **Receive State** - both nodes now know optimal coil pair

Algorithm Discussed in the Paper / Methodology

Phase 3: Data Transmission & Spatial Reuse

- Source performs channel sensing again, then sends **Data packet**
- Data packet (3-19 bytes) transmitted using pre-selected optimal coil
- Destination receives in **Receive State**, decodes packet, validates Target ID
- Both nodes return to **Idle State** after successful transmission
- Complete sequence: REV $\rightarrow$ ACK $\rightarrow$ DATA with minimal energy expenditure
- **Spatial Reuse**: Different node pairs can use different coils simultaneously
- Example: Node A-B communicate via Coil X while Node C-D use Coil Y concurrently
- Directional MI communication enables interference-free parallel transmissions

Implementation

What is Network Simulator 3 (NS-3)?

- Open-source discrete-event network simulator for research and education
- Widely used for simulating internet protocols, wireless networks, and MAC layers
- Written in C++ with Python bindings for scripting
- Provides detailed network stack simulation: applications, transport, network, MAC, and physical layers
- Key Features:
  - Supports WiFi, LTE, sensor networks, and custom protocol development
  - Real-time packet transmission, state management, and energy tracking
  - Built-in tools for performance metrics: latency, throughput, packet delivery
  - NetAnim support for network visualization
- **Why NS-3 for our project?**
  - Robust MAC protocol implementation framework
  - No hardware required - pure software simulation
  - Validated and trusted by research community

Implementation

mi-mac-demo.cc: Basic Protocol Demonstration

- **Purpose**: Demonstrate complete Multi-Coil MI MAC protocol workflow
- **Key Components Implemented**:
  - Three packet type enumerations: REV, ACK, DATA
  - Three coil IDs: X, Y, Z (orthogonal coils)
  - Five-state machine: Idle, Receive, Channel Sensing, Data Acquire, Transmit
- **Core Functions**:
  - `CalculateRSSI()`: Simulates signal strength for each coil (-45.5, -52.3, -48.7 dBm)
  - `SelectBestCoil()`: Compares RSSI values and selects strongest coil
  - `StateTransition()`: Tracks and displays state machine transitions
  - `SendPacket()`: Simulates packet transmission with structure details
- **Workflow Simulated**:
  - Source sends REV packet 3 times (once per coil)
  - Destination receives, selects best coil via RSSI comparison
  - ACK sent back using selected coil
  - DATA packet transmitted using established coil pair

Implementation

mi-mac-comparison.cc: Performance Comparison with CSMA/CA

- **Purpose**: Compare Multi-Coil MI MAC with traditional CSMA/CA protocol
- **Enhanced Features**:
  - `EnergyMetrics` structure: Tracks current consumption per state (Table II values)
  - Calculates total energy, packet count, and state transitions
  - Implements both MI MAC and CSMA/CA protocols for side-by-side comparison
- **CSMA/CA Implementation**:
  - RTS (Request to Send) → CTS (Clear to Send) → DATA → ACK handshake
  - Four packets vs. three packets in MI MAC (REV → ACK → DATA) for single communication pair.
  - Higher control overhead and energy consumption
- **Comparison Output**:
  - Side-by-side table: packets sent, state transitions, total energy
  - Energy savings percentage calculation
  - Key advantages: spatial reuse, directional communication, lower overhead
  - Validates paper's energy efficiency claims using actual current values

Results

Performance Comparison: Multi-Coil MI MAC vs CSMA/CA

Metric	Multi-Coil MI	CSMA/CA
Total Packets Sent	5	4
State Transitions	12	13
Total Energy (μJ)	11,183.84	14,058.40

Energy Savings: 20.4%

Results

Protocol Overhead Analysis:

- **Multi-Coil MI MAC**: REV ($3\times$) $+$ ACK $+$ DATA $=$ 5 packets total
- **CSMA/CA**: RTS $+$ CTS $+$ DATA $+$ ACK $=$ 4 packets total

- Despite sending more packets, MI MAC is more energy-efficient
- Reason: Optimized coil selection reduces retransmissions
- Lower power states used effectively
- Directional communication reduces interference

Results

Validation of Paper's Claims:

- Energy consumption matches Table II values:
  - Idle: 50 μA, Receive: 200 μA, Data Acquire: 250 μA
  - Channel Sensing: 200 μA, Transmit: 1120 μA
- State machine operates as designed (Figure 3)
- Packet structures match specifications (Figure 4)
- RSSI-based coil selection improves link quality
- Three-coil system ensures reliable communication

Results

Demonstrated Advantages of Multi-Coil MI MAC:

- **Energy Efficiency**: 20.4% savings over CSMA/CA
- **Spatial Reuse**: Different node pairs use different coils concurrently
- **Directional Communication**: RSSI-based selection optimizes links
- **Collision Avoidance**: Channel sensing + coil diversity
- **Lower Overhead**: Optimized handshake vs RTS/CTS
- **Dual Environment**: Works in terrestrial and underwater networks

Results

Simulation Effectiveness:

- NS-3 successfully simulated MAC protocol without hardware
- Complete protocol workflow demonstrated
- State machine behavior verified
- Energy metrics tracked accurately
- Performance comparison achieved
- Paper's theoretical advantages validated through simulation

Limitations & Future Work

Current Limitations:

- **Short Communication Range**: MI communication limited to 40-50 meters due to magnetic field attenuation
- **Coil Misalignment**: Significant signal degradation when coils are severely misaligned despite 3D coil system
- **Simulation-Only**: No hardware prototype implementation - results based on software simulation

Future Work:

- Hardware prototyping with actual MI coils and sensor nodes
- Multi-hop routing protocol for extended network coverage
- Adaptive power control to optimize range vs. energy trade-off
- Real-world testing in underwater and underground environments
- Integration with LoRa or other long-range technologies for hybrid networks

THANK YOU

Questions?

GitHub: https://github.com/NishitVSP/-Multi-Coil-MI-Based-MAC-Protocol-for-Wireless-Sensor-Networks