DESIGN AND IMPLEMENTATION OF A LORA MESH NETWORK WITH AN OPTIMIZED ROUTING PROTOCOL

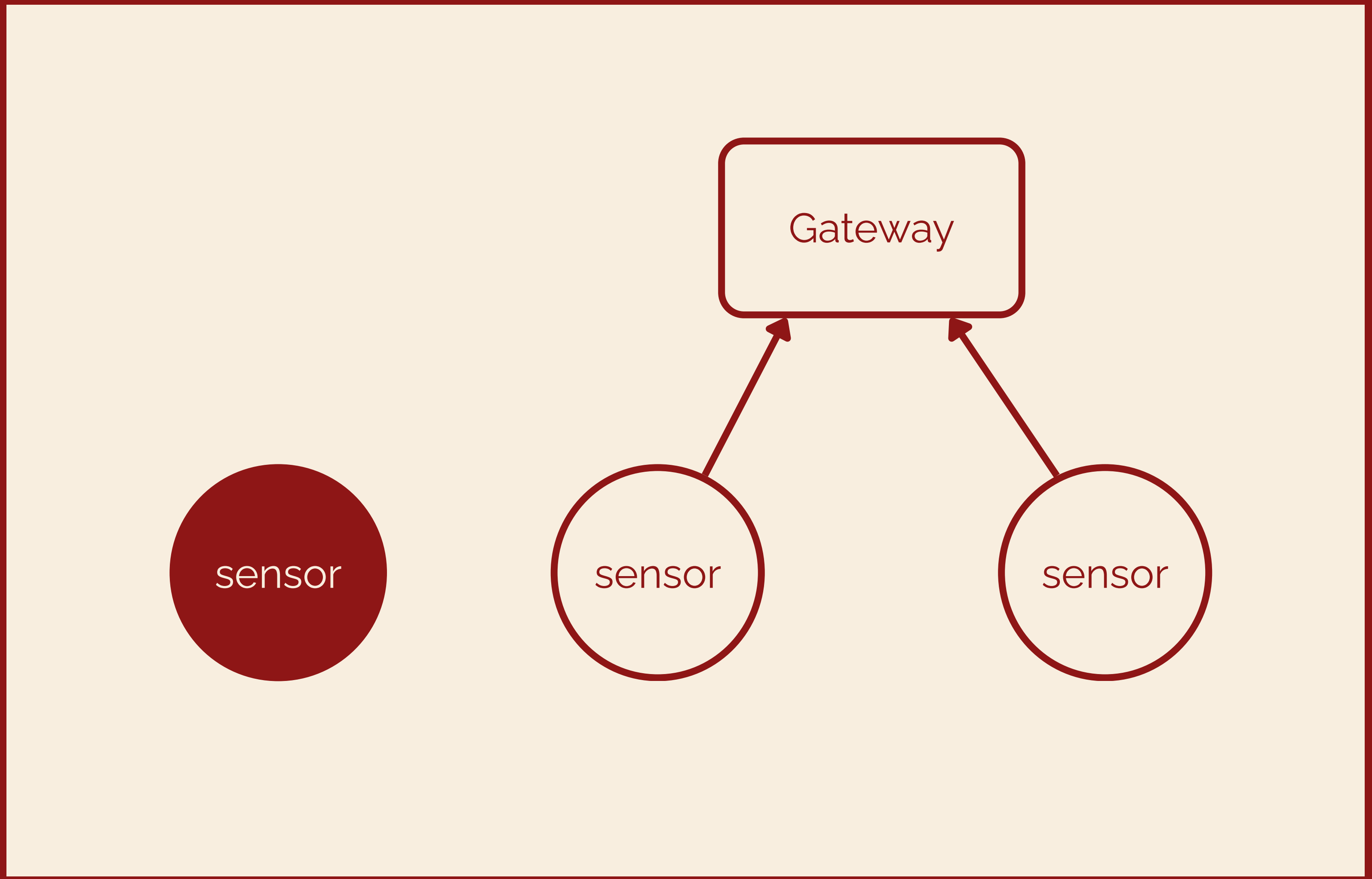
Internship Research

GitHub
ncwn/xMESH

Presented by
Nyein Chan Win Naing (st123843)

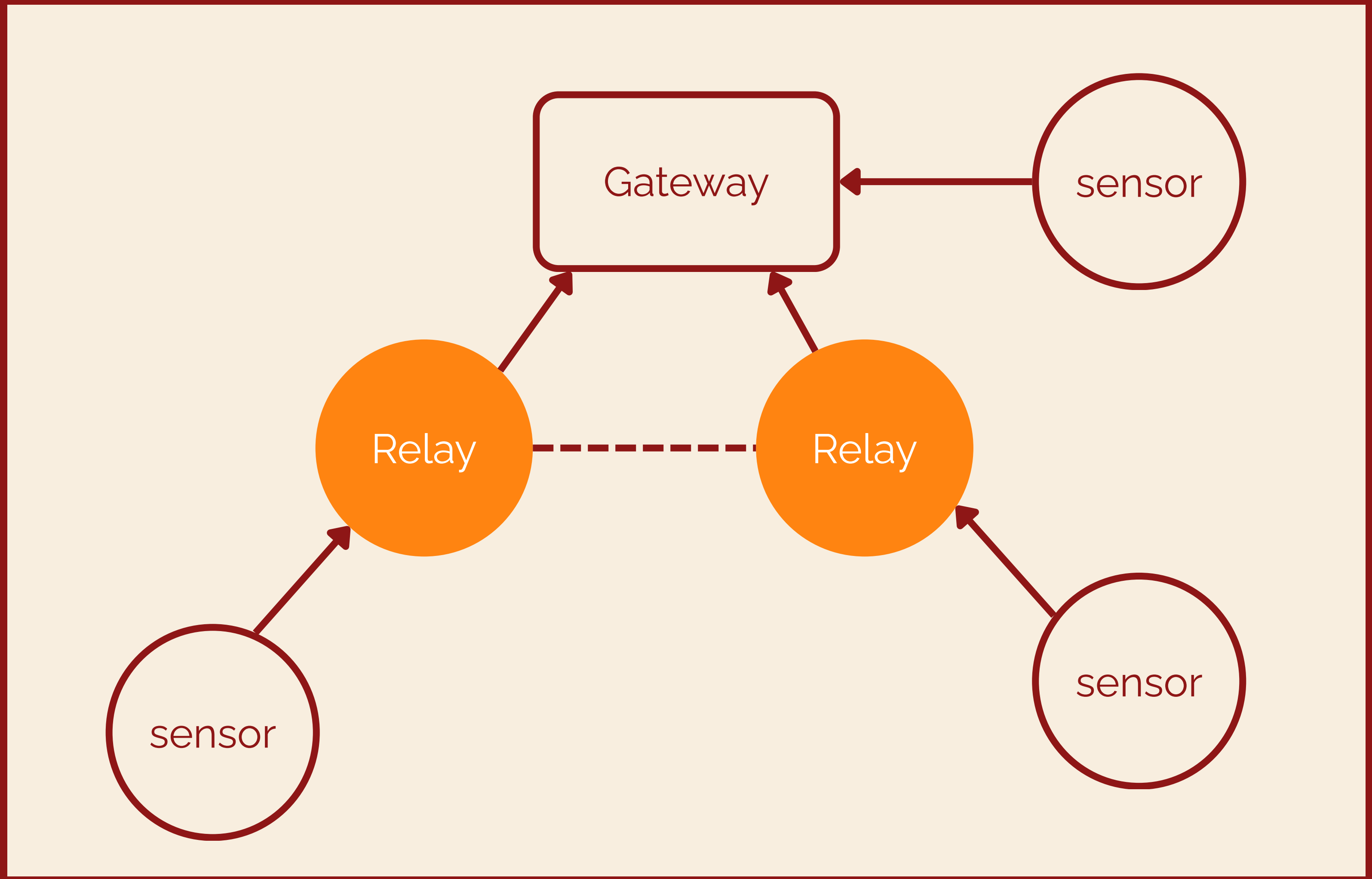
WHAT IS LORA MESH?

LoRaWAN



- Single hop only
- No relay capability
- Coverage limitedt.

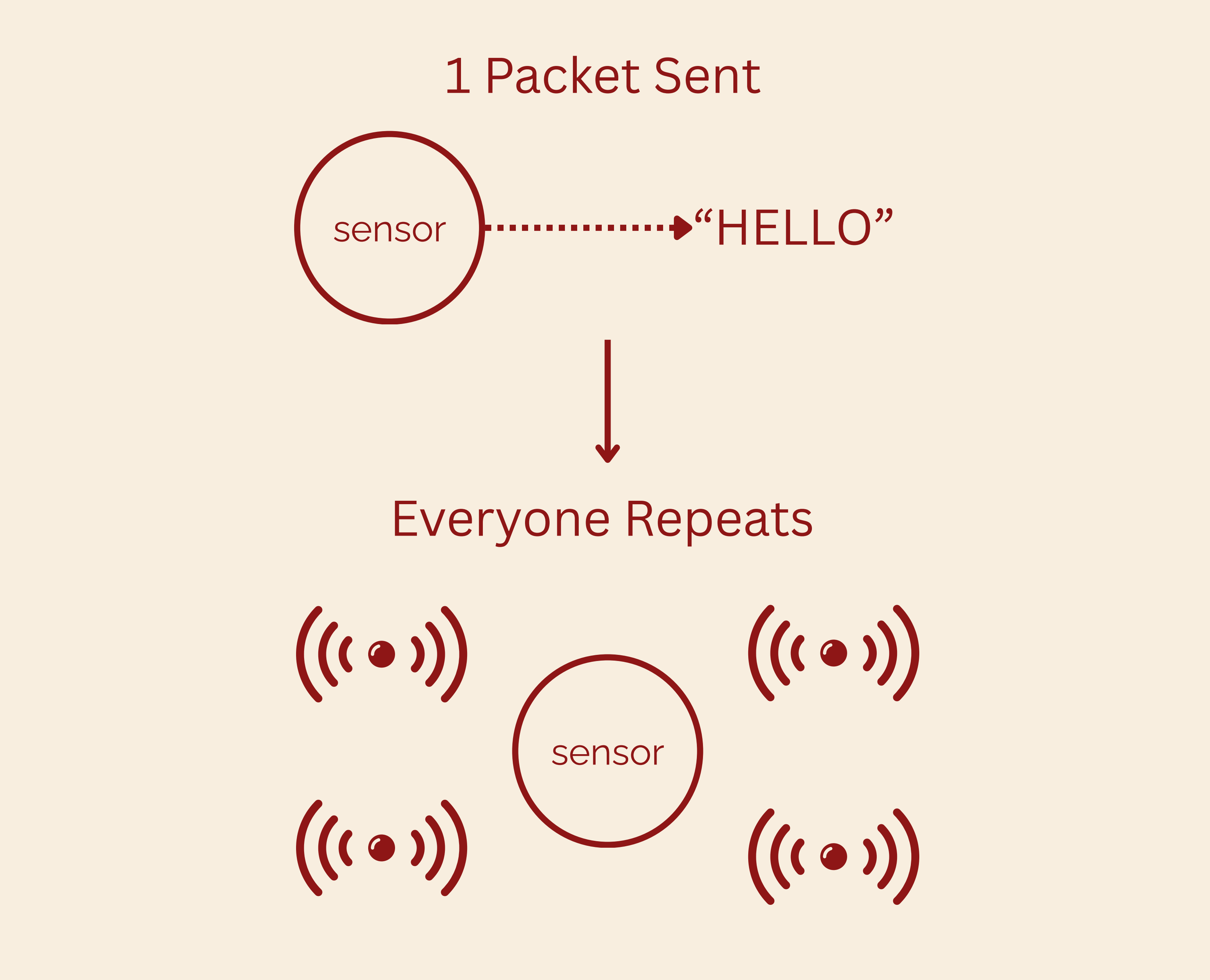
LoRa Mesh



- Multi-hop relay
- Extended coverage
- Self-healing paths

THE PROBLEM: BROADCAST STORM

Flooding = Chaos



Nodes	Transmissions per Packet	Problem
3	3-5	OK
5	10-25	Heavy
10	100+	**Fail**

THIS VIOLATES 1% DUTY CYCLE REGULATION!

SOLUTION: PROTOCOL COMPARISON

Protocol 1: Flooding

Baseline (Worst)

- Boradcast
- No Routing
- High Waste

Compare

Protocol 2: Hop-Count

Standard

- Unicast
- Fixed HELLO
- Hop Count Metric

Compare

Protocol 3: Gateway-Aware Cost

Contribution

- Smart
- Adaptive
- Quality-aware

Validate

PROTOCOL 1: FLOODING

Step 1: Sensor broadcasts

[SENSOR]

↓ Data

BROADCAST

↓ ↓

[RELAY] [GATEWAY]

Step 2: Relay rebroadcasts

[SENSOR]

↓

[RELAY]

↓ Data Again

BROADCAST

Step 3: Gateway Terminates flood (Does not rebroadcast)

- Simple implementation
- High reliability
- High overhead - Does Not Scale

PROTOCOL 1: DUPLICATE DETECTION

Every packet has: Source Address + Sequence Number

Packet ID = (Source: 0xBB94, Sequence: 42)

DUPLICATE CACHE (Ring Buffer)

Slot 1: (0xBB94, 41) ✓
Slot 2: (0xBB94, 42) ← NEW! Add here
Slot 3: (0x1234, 15) ✓
...
Slot 20: (oldest gets replaced)

IF already in cache → DROP

IF not in cache → PROCESS

PROTOCOL 2: HOP-COUNT ROUTING

Each node maintains a "phone book" (routing table)

[SENSOR]

↓ 1 Hop

[RELAY]

↓ 1 Hop

[GATEWAY]

SENSOR's Routing Table:

Dest	Via	Hops
GATEWAY	RELAY	2
RELAY	DIRECT	1

RELAY’s Routing Table:

Dest	Via	Hops
GATEWAY	DIRECT	1
SENSOR	DIRECT	1

PROTOCOL 2 - HELLO PACKETS

Every 120 seconds, each node broadcasts: "I'm here!"

t=0s	t=120s	t=240s	t=360s
↓	↓	↓	↓
HELLO	HELLO	HELLO	HELLO

FIXED INTERVAL (always 120 seconds)

HELLO Packet Contains:

"I am: Node 0x8154"
"I can reach: Gateway (1 hop), Sensor (1 hop)"

PROBLEM: Sends HELLOs even when nothing changes!

PROTOCOL 2 - THE LIMITATION

THE PROBLEM WITH COUNTING HOPS ONLY

[SENSOR]

PATH A (1 Hop)

PATH B (2 Hops)

WEAK SIGNAL
RSSI: -139 dBm

STRONG SIGNAL
RSSI: -90 dBm

[RELAY]

[GATEWAY]

Protocol 2 chooses Path A (1 hop) but loses 67% of packets!
Path B (2 hops) would deliver 95% of packets!

PROTOCOL 3 - OVERVIEW

TWO MAJOR IMPLEMENTATIONS

Multi-Metric Cost Function

Consider:

- Hop Count
- Signal (RSSI)
- Noise (SNR)
- Reliability
- Gateway Load

Trickle Adaptive Scheduling

Reduce HELLO packets when network is stable

60s → 600s

PROTOCOL 3 - COST FUNCTION

COST = W1×Hops + W2×Signal + W3×Noise + W4×ETX + W5×Load

W1×Hops	W2×Signal	W3×Noise	W4×ETX	W5×Load
1.0	0.3	0.2	0.4	1.0
HOPS	RSSI	SNR	ETX	LOAD

LOWER COST = BETTER ROUTE

Example:

- Direct path (weak signal): Cost = 2.95
- Via relay (good signal): Cost = 2.36 ← WINNER!

PROTOCOL 3 - COST FACTORS

W1: HOP COUNT (1.0) "How far?"

- More hops = Higher cost

W2: SIGNAL STRENGTH (0.3) "How strong?"

- Strong (-70 dBm) = Low cost
- Weak (-130 dBm) = High cost

W3: SIGNAL QUALITY (0.2) "How noisy?"

- Clean signal (+10 dB SNR) = Low cost
- Noisy signal (-15 dB SNR) = High cost

W4: RELIABILITY (0.4) "Do packets arrive?"

- ETX= 1.0: All packets arrive
- ETX= 2.0: Half the packets lost

W5: GATEWAY LOAD (1.0) "Is gateway busy?"

- Busy gateway = Penalty → Use another

PROTOCOL 3 - TRICKLE TIMER

PROTOCOL 2 (Fixed):

120s ↓ 120s ↓ 120s ↓ 120s ↓ 120s ↓ 120s ↓ 120s ↓ 120s ↓

ALWAYS the same - even when nothing changes!

PROTOCOL 3 (Trickle):

60s ↓ 120s ↓ 240s ↓ 480s ↓ 600s ↓

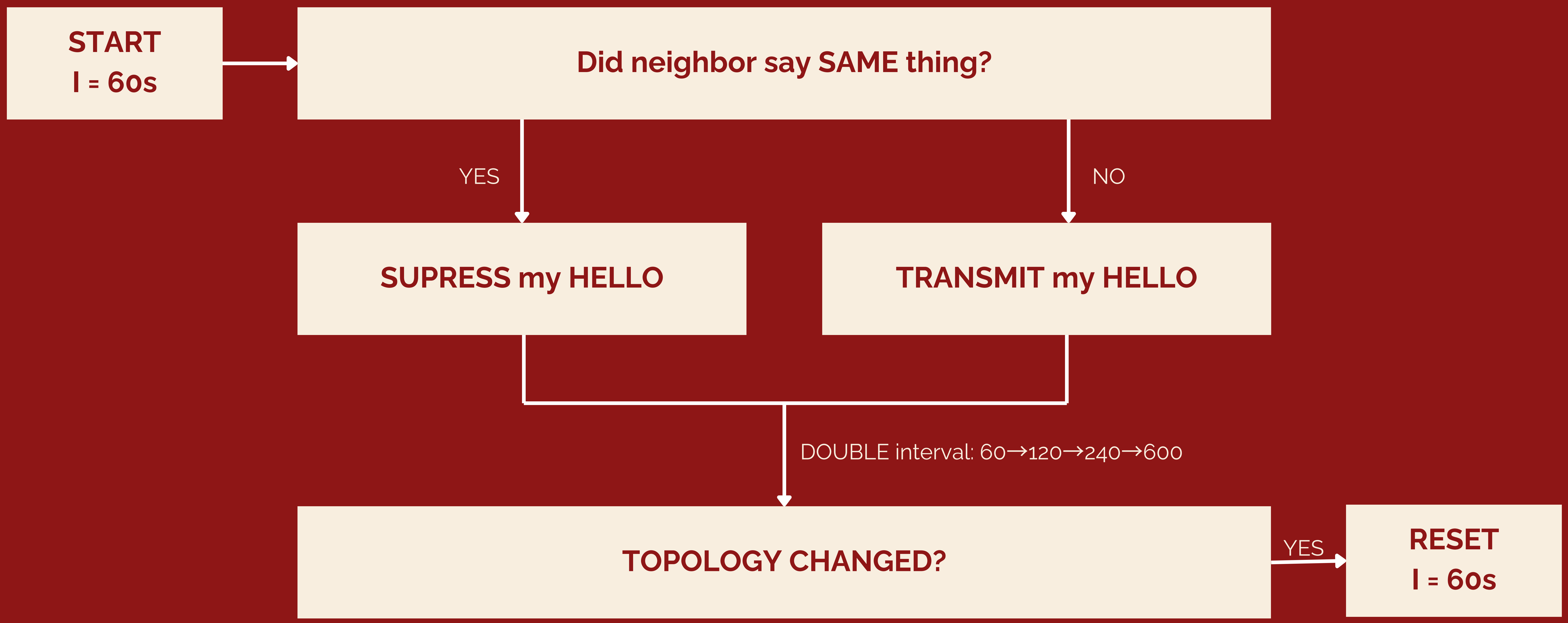
EXPONENTIAL BACKOFF (doubles each time)

Stable? → Longer intervals (fewer HELLOs)
Changed? → Reset to 60s (fast recovery)

RESULT: 31-33% FEWER HELLO PACKETS!

PROTOCOL 3 - TRICKLE STATE

TRICKLE DECISION FLOWCHART



PROTOCOL 3 - ZERO-OVERHEAD ETX

Traditional ETX:

[SEND] → [RECV] → [ACK]

EXTRA TRAFFIC

My Approach:

[SEND with seq#] → [RECV sees seq#]

↓

[DETECT GAPS]

Received: 1 2 3 - 5 6 7

↑

Gap! Packet 4 was lost!

ETX = Total / Successes = 7 / 6 = 1.17

NO EXTRA PACKETS - Uses existing sequence numbers

PROTOCOL 3 - GATEWAY LOAD BALANCING

WITHOUT Load Balancing:

[S1] [S2] [S3]

↓ ↓ ↓

[Gateway 100% Busy!]

WITH Load Balancing:

[S1] [S2] [S3]

↓ ↓ ↓

GW1 55% GW2 45%

HOW IT WORKS:

- Gateways report their load in HELLO packets
- Sensors add penalty to busy gateways
- Traffic naturally shifts to less busy gateways

VALIDATED: 45% / 55% TRAFFIC SPLIT

PROTOCOL 3 - FAST FAULT DETECTION

Protocol 2 (Library Default):

[NODE FAILS]

maximum 600 seconds to detect, Routing table only update every 600s

[DETECTED]

Protocol 3 (My Safety HELLO):

[NODE FAILS]

180s
(1st miss)

180s
(2nd miss)

[DETECTED]

maximum 360 seconds to detect, Immidiate Routing table update

40-50% FASTER THAN LIBRARY DEFAULT

HARDWARE SETUP

HARDWARE ARCHITECTURE

Heltec WiFi LoRa 32 V3:

- ESP32-S3 dual-core 240MHz
- SX1262 LoRa transceiver
- OLED display 128×64

Sensors (Protocol 3):

- PMS7003: PM2.5 air quality
- NEO-M8M: GPS location

Node Roles:

- [S] SENSOR - Collect and send data
- [R] RELAY - Forward traffic
- [G] GATEWAY - Receive and upload

Total Hardware 5 Boards

NETWORK TOPOLOGY TESTED

3-Node Linear:

[S] → [R] → [G]

4-Node Diamond:

[R]

[S] → → [G]

[R]

5-Node Mesh:

[R] [G]

[S] → →

[S] [G]

TEST ENVIRONMENTS:

INDOOR:	Lab, 5-10m spacing, 14 dBm, SF7 Direct connectivity, 96.7-100% PDR
OUTDOOR:	Two tests conducted: • AIT Campus: 105-383m, SF7, 14dBm (21-32% PDR) • Long-range: 935m, SF9, 20dBm (75% PDR)

TOTAL: 20 hardware tests

MULTI-HOP VALIDATION (935M TEST)

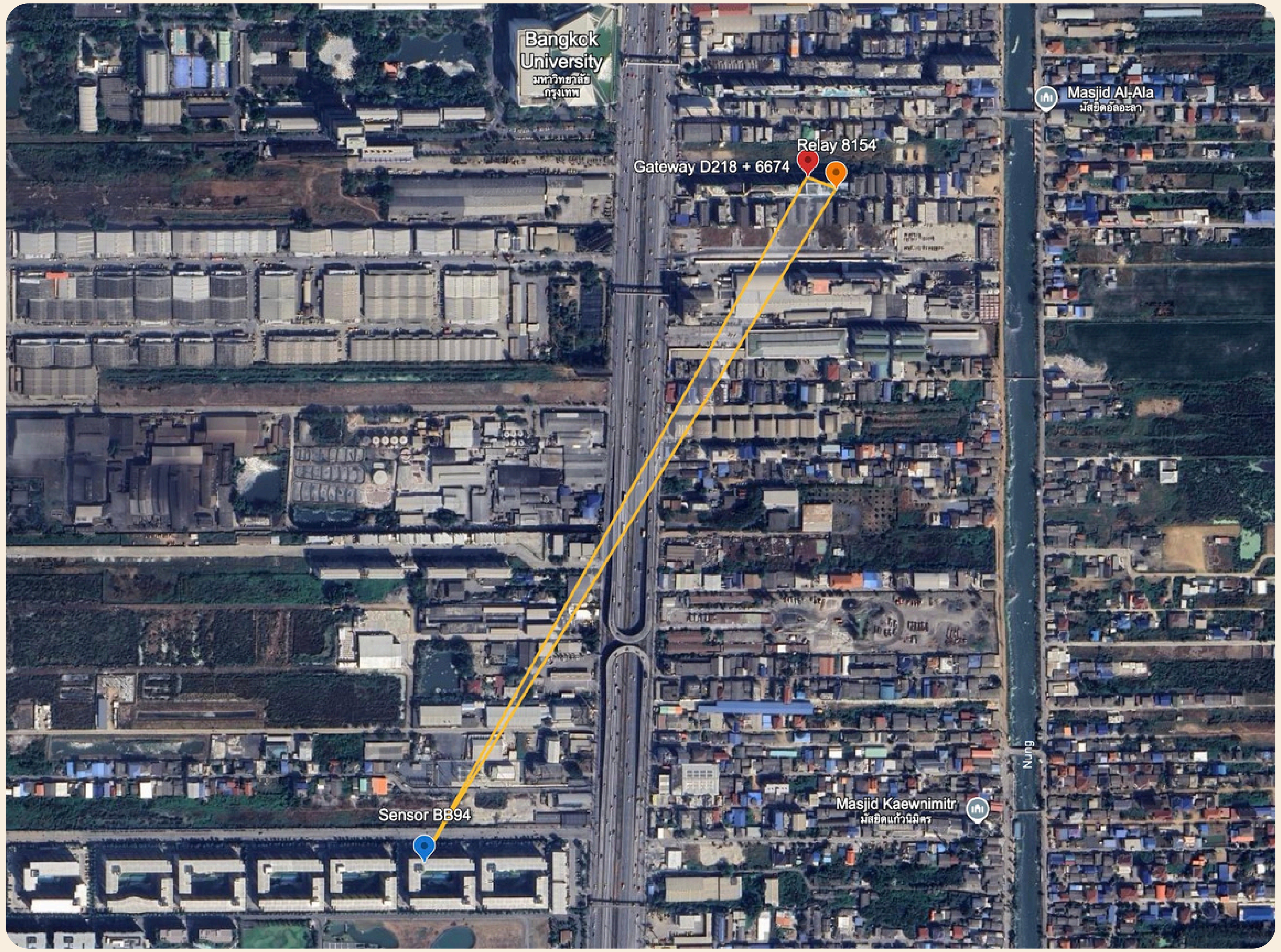
Test Configuration:

- Distance: 935 meters
- SF: 9 (higher range)
- TX Power: 20 dBm
- Duration: 60 minutes

KEY RESULTS:

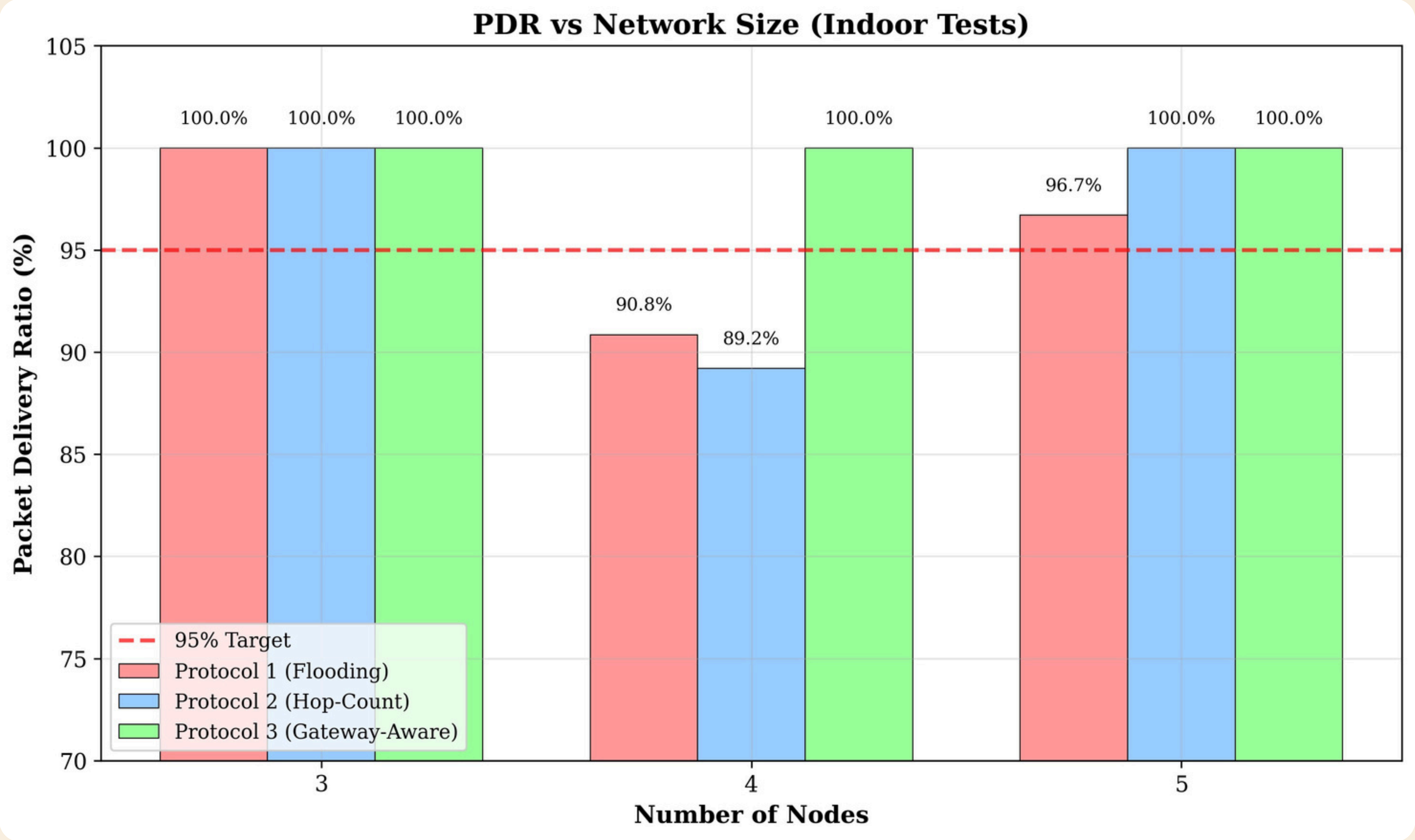
- Relay forwarded 70% of traffic
- 3-hop path chosen over weak 2-hop (cost 3.28 vs 3.95)

Direct Path Only: 33% PDR (very poor)
Via Relay (2-3 hops): 75% PDR (2.27× better!)



PROTOCOL 3 CORRECTLY CHOOSES THE BETTER PATH

RESULTS - PDR COMPARISON



ALL PROTOCOLS MEET >95% TARGET INDOORS

Protocol 3 achieves this with LESS overhead

PACKET DELIVERY RATIO (INDOOR)

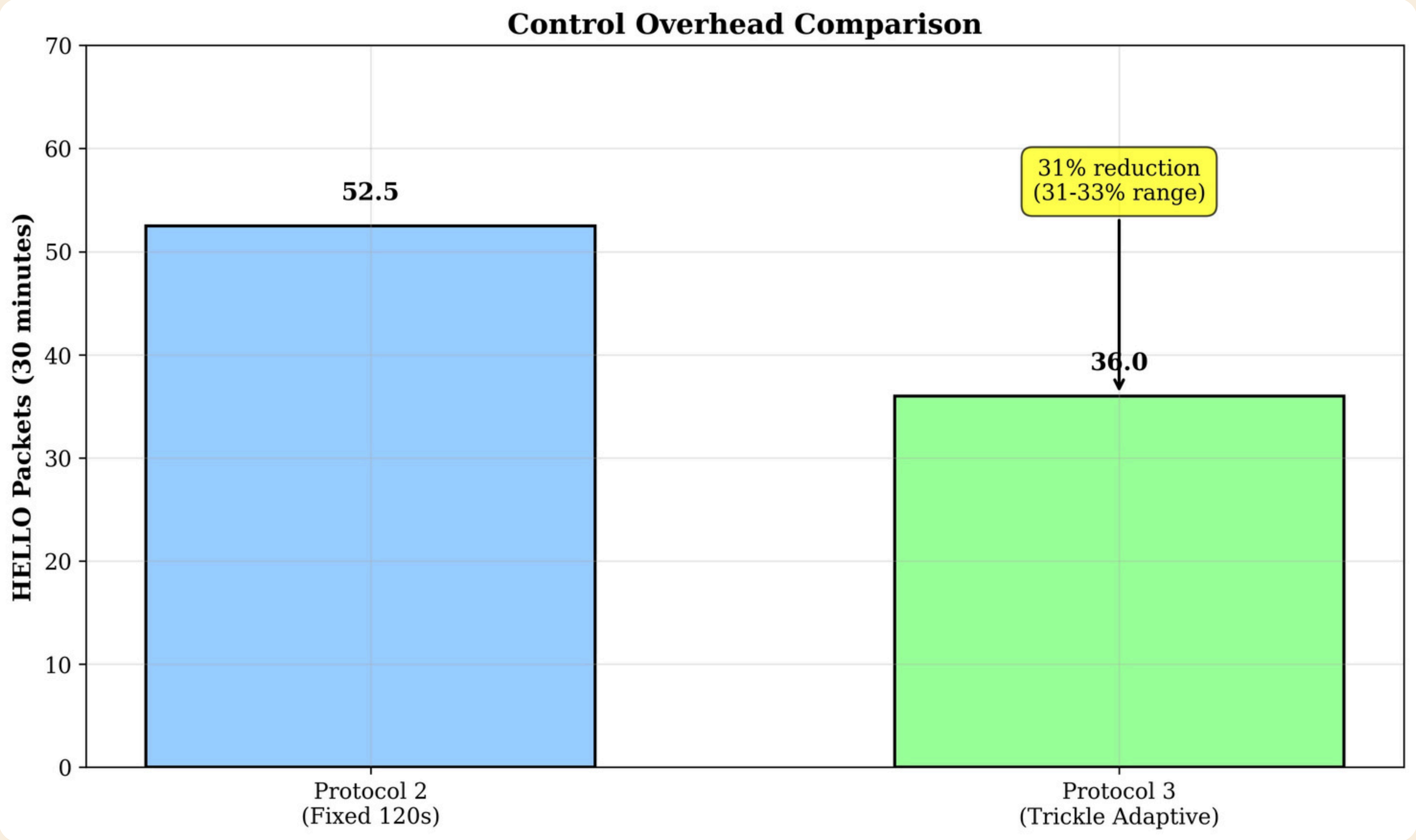
RESULTS - HELLO OVERHEAD

Protocol 2 (Fixed 120s): 52 HELLOs
Protocol 3 (Trickle): 36 HELLOs

REDUCTION: 52 → 36 = 16 fewer packets

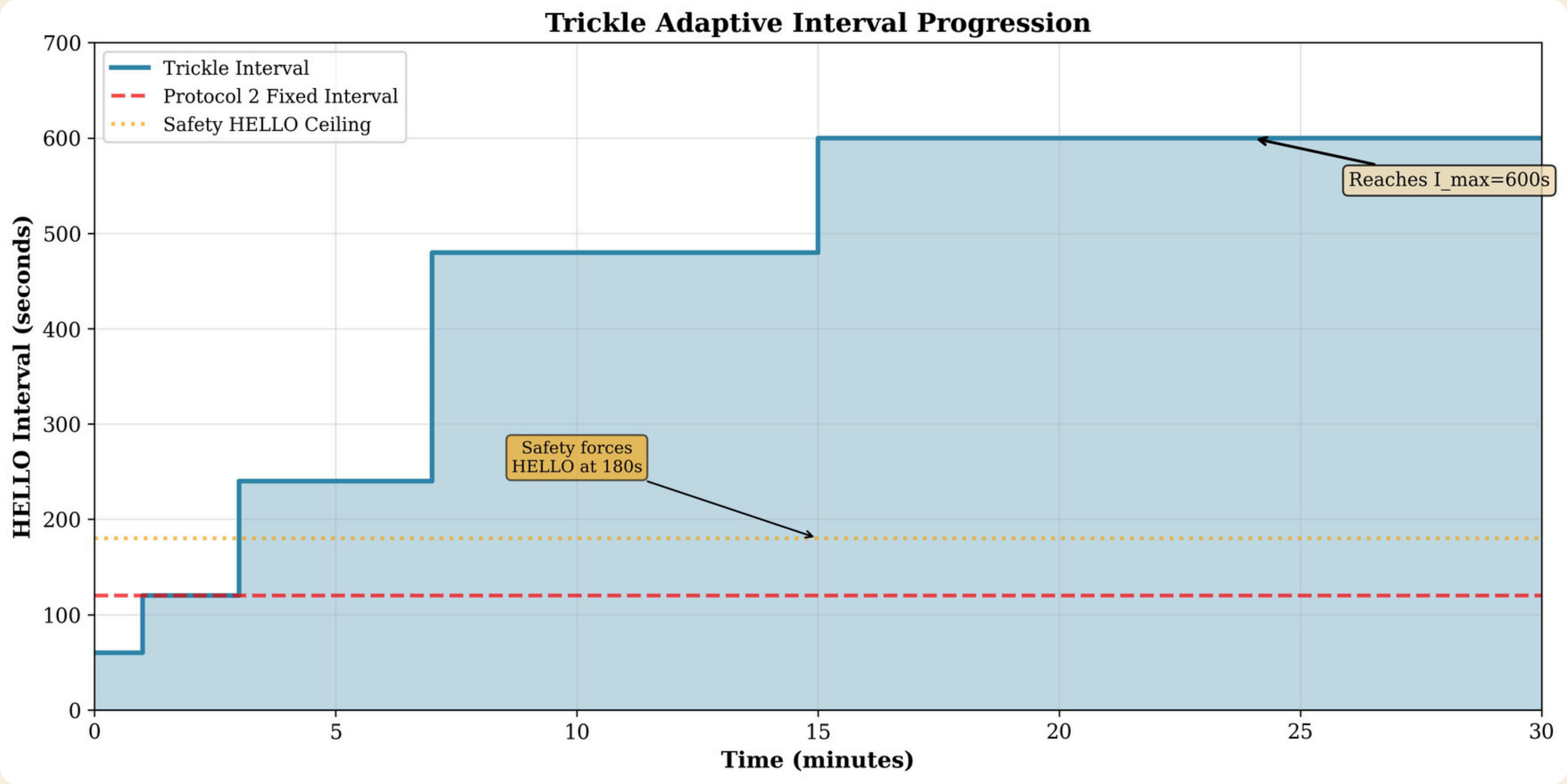
IMPROVEMENT: 31% reduction

Trickle internal suppression: 85-90%
(Limited by 180s safety ceiling)



HELLO PACKET OVERHEAD (30 minutes)

RESULTS - TRICKLE PROGRESSION



STABLE NETWORK:
- Starts at 60 seconds
- Doubles: 60 →...→ 600
- Reaches max

TOPOLOGY CHANGE:
- Resets immediately to 60 seconds
- Fast rediscovery of new routes

SAFETY CEILING:
- every 180s
- Ensures fault detection

TRICKLE INTERVAL PROGRESSION

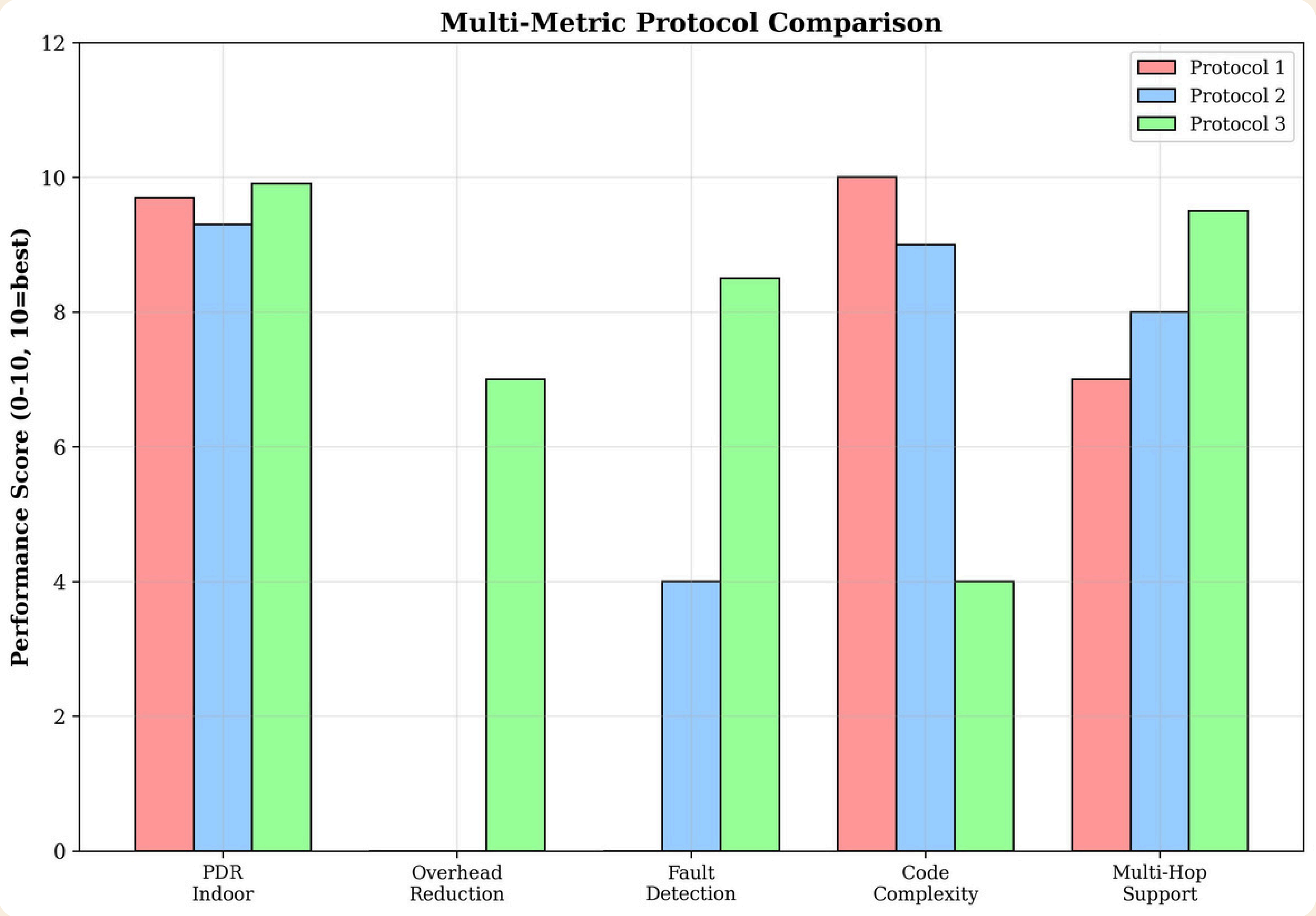
RESULTS - PROTOCOL COMPARISON

Protocol 3 WINS on:

- Overhead reduction (31-33%)
- Multi-hop support (cost-aware path selection)
- Fault detection (40-50% faster)
- PDR (maintains 96.7-100%)

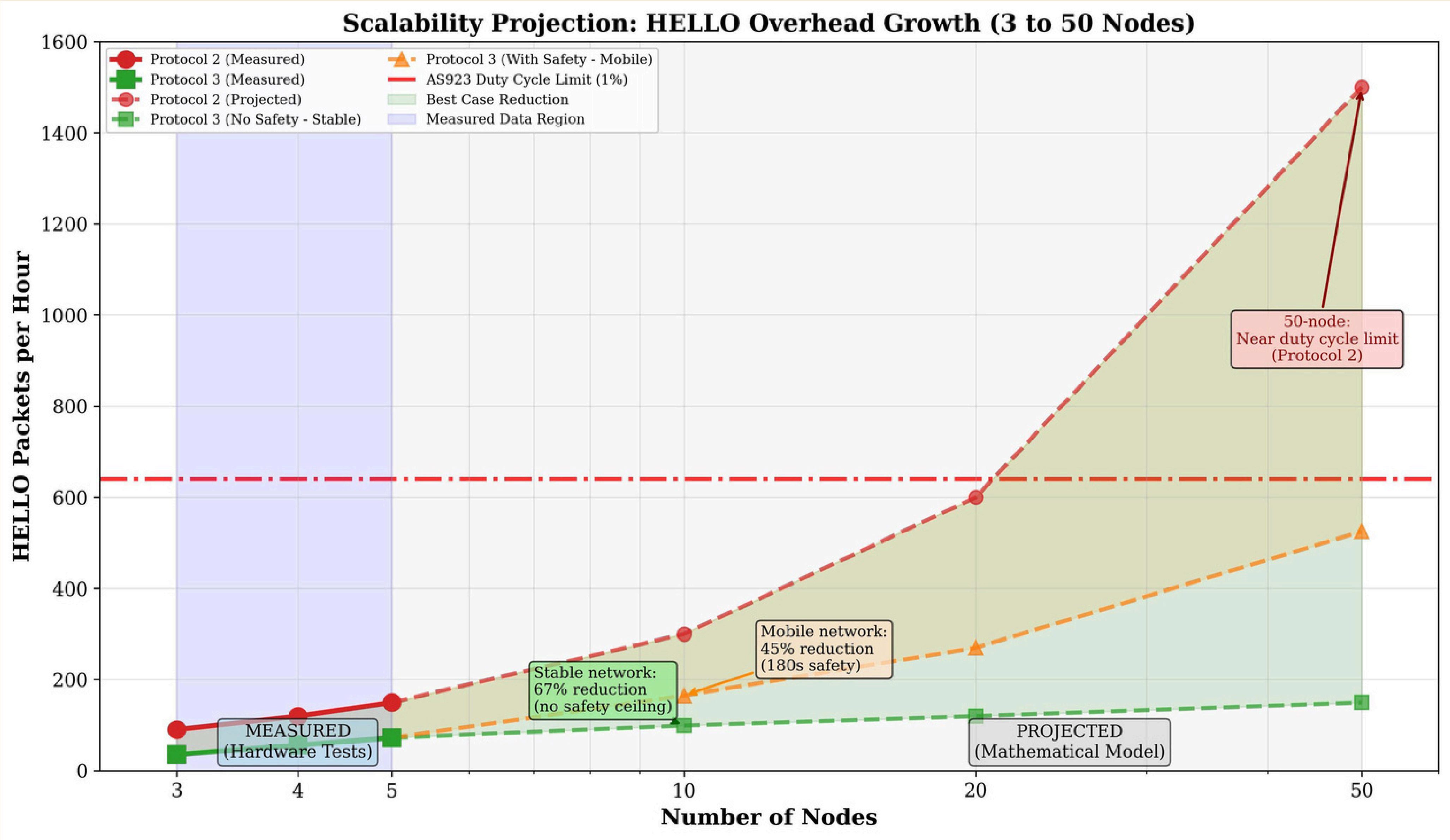
Trade-off:

- Higher code complexity
- More sophisticated debugging required



MULTI-METRIC PROTOCOL COMPARISON

RESULTS - SCALABILITY PROJECTION



MEASURED (3-5 nodes):

- Solid data points
- Protocol 3 achieves 31-33% reduction

PROJECTED (10-50 nodes):

- Protocol 2: Linear growth → hits duty cycle limit
- Protocol 3 (Green): 67-90% reduction (stable)
- Protocol 3 (Orange): 45-65% reduction (mobile)

SCALABILITY PROJECTION

INTEGRATED SENSOR DATA

PM SENSOR (PMS7003)

2.5=8 μg/m³
10=10 μg/m³
AQI: "Good"

GPS MODULE (NEO-M8M)

Location: 14.0775˚N, 100.6130˚E
Satellites: 8 (good fix)
(AIT Campus confirmed)

```
[15:55:13.722] [TOPOLOGY] Routing table size changed: 0 → 3
[15:55:13.726] [COST] New route to 8154 via 8154: cost=2.98 hops=1
[15:55:13.731] [COST] New route to BB94 via 8154: cost=3.98 hops=2
[15:55:13.735] [COST] New route to 6674 via 8154: cost=3.98 hops=2
[15:55:13.742] [TRICKLE] Topology change detected – resetting to I_min for fast convergence
[15:55:13.746] [Trickle] RESET – I=60.0s, next TX in 37.7s
[15:55:31.748] Link BB94: First packet (seq=40), initializing ETX tracking
[15:55:31.753] Link BB94: RSSI=-120 dBm, SNR=-14 dB, ETX=1.00, Seq=40
[15:55:31.756] Link quality: SNR=0 dB, Est.RSSI=-120 dBm
[15:55:31.758] RX: Seq=40 From=BB94
[15:55:31.762] PM: 1.0=6 2.5=8 10=10 μg/m³ (AQI: Good)
[15:55:31.767] GPS: 14.0775°N, 100.6130°E, alt=23.0m, 8 sats (Good)
[15:55:31.771] [GATEWAY] Packet 40 from BB94 received
[15:55:31.774] ✓ Packet validation passed
[15:55:31.779] [HEALTH] Neighbor BB94: Heartbeat (silence: 57s, status: HEALTHY)
[15:55:43.497] [90] Heartbeat – Node D218 (GATEWAY) – Uptime: 90 sec
[15:55:43.501] TX: 0 | RX: 2 | FWD: 0 | Routes: 3
[15:55:43.504] ==== Routing Table (with Cost Metrics) ====
[15:55:43.506] Routing table size: 3
[15:55:43.509] Addr   Via    Hops  Role  Cost
[15:55:43.512] ------|------|------|------|------
[15:55:43.515] 8154 | 8154 |    1 | 00 | 2.98
[15:55:43.518] BB94 | 8154 |    2 | 00 | 3.98
[15:55:43.520] 6674 | 8154 |    2 | 01 | 3.98
[15:55:43.523] ==== Link Quality Metrics ====
[15:55:43.525] Addr   RSSI   SNR   ETX
[15:55:43.528] ------|------|------|------
[15:55:43.530] 8154 | -129 | -17 | 1.00
[15:55:43.532] BB94 | -120 | -14 | 1.00
[15:55:43.535] ================================
[15:55:43.539] ==== Network Monitoring Stats ====
[15:55:43.543] Channel: 0.000% duty-cycle, 0 TX, 0 violations
[15:55:43.547] Memory: 317/381 KB free, Min: 317 KB, Peak: 64 KB
[15:55:43.551] Queue: 0 enqueued, 0 dropped (0.00%), max depth: 0
[15:55:43.556] [Trickle] TX=1, Suppressed=0, Efficiency=0.0%, I=60.0s
[15:55:43.560] Routing table: 3 entries × ~32 bytes = ~0 KB
[15:55:43.563] ===================================
```

DEMONSTRATES REAL-WORLD IoT APPLICATION

CONTRIBUTIONS SUMMARY

1. FIRST TRICKLE + LoRaMesher INTEGRATION
   Result: 31-33% overhead reduction

2. ZERO-OVERHEAD ETX TRACKING
   Result: Quality tracking with no extra traffic

3. LOCAL FAULT ISOLATION DISCOVERY
   Result: Faults affect 10-30% of network, not 100%

4. HARDWARE-VALIDATED FRAMEWORK
   Result: 20+ tests on real ESP32 hardware

5. OPEN-SOURCE IMPLEMENTATION
   Result: Available for community use

LIMITATIONS

SCALE

- Tested: 3-5 nodes
- Larger networks (10-50) need validation

RSSI ESTIMATION

- Currently estimated from SNR
- True RSSI requires RadioLib modification

OUTDOOR PDR

- 75% at 935m (below 95% target)
- Physical layer limitation, not protocol

MOBILITY

- Static scenarios tested
- Mobile scenarios need future work

FUTURE WORK

Next Step:

- Integrate with AIT Hazemon sensor platform
- 10-50 node scalability testing
- Mobile scenario validation

RESEARCH EXTENSIONS:

- True RSSI extraction from RadioLib
- Energy profiling and optimization
- Security layer implementation
- Machine learning for adaptive weights

APPLICATIONS:

- Smart agriculture monitoring
- Environmental sensing networks
- Building automation systems

CONCLUSION

THE PROBLEM:
Traditional LoRa mesh doesn't scale (broadcast overhead, fixed control traffic)

MY SOLUTION:
Gateway-Aware Cost Routing with Trickle (smart routing + adaptive scheduling)

KEY RESULTS:

- 31-33% overhead reduction
- 96.7-100% packet delivery
- 2.27× PDR improvement via relay
- 40-50% faster fault detection
- Validated on real hardware

SCALABLE LoRa MESH NETWORKS ARE ACHIEVABLE!

ACKNOWLEDGMENTS

Prof. Attaphongse Taparugssanagorn (Chairperson) and Dr. Adisorn Lertsinsrubtavee (Co-chairperson)
For guidance and support throughout

AIT InterLAB
For hardware resources and facilities

Committee Members
For your time and valuable feedback

LoRaMesher Community
For the open-source foundation

THANK YOU