

White Paper – Why are Repeater Error Rates so High (up to 50%)

This paper examines several reasons that can cause a high Packet Error Rate (PER).

1. Collision-Dominated Airtime (Most Common)

A 50% PER is the classic footprint of two or more nodes transmitting with overlapping airtime such that half of all packets statistically collide.

Why it stabilizes around 50%

- If two nodes transmit with similar duty cycles and no coordination, each packet has roughly a 50% chance of overlapping the other's airtime window.
- This is especially true when: *txdelay* is too low (the default) for the network density
- Directed messages and ACKs overlap
- Repeaters forward at nearly the same time
- Two nodes share the same hop distance and forward simultaneously

Indicators

- RSSI/SNR look fine
- No CRC errors on the ones that do decode
- Loss is evenly distributed, not bursty
- Flood messages succeed more often than directed messages (because floods have randomized backoff)

2. Incorrect or Too-Aggressive *txdelay* / *direct.txdelay*

Too-low delay → nodes transmit too close together → deterministic collisions.

Why it lands at 50%

If two forwarders share a path and both fire within the same deterministic window, half of the packets get clobbered.

This is especially visible in:

- Dense neighborhoods
- Multi-hop DM ACK paths
- Repeaters with identical firmware timing

3. Repeater Timing Skew or Firmware Drift

If a repeater's internal timing is off (clock drift, ISR latency, or firmware bug), it may forward packets slightly early or late in a way that consistently overlaps with another node's airtime.

Symptoms

- Only packets involving that repeater show ~50% loss
- Floods may still succeed because randomized backoff masks the timing skew
- Directed messages through that repeater fail disproportionately

4. Two Nodes Using Identical Forwarding Logic (No Randomization)

If two repeaters or two hops share:

- Same hop count
- Same forwarding delay
- Same firmware build
- Same deterministic backoff logic

...they will transmit at the same time every time.

This produces a perfect 50% collision pattern.

This is extremely common in:

- Custom MeshCore builds
- Nodes using identical deterministic forwarding code
- Repeater pairs placed symmetrically in the network

5. Borderline SNR Where FEC Corrects Half the Frames

This is the RF-layer version of a 50% PER.

How it happens

- SNR is just barely above the FEC correction threshold
- Half the packets have enough symbol errors to exceed CR correction capability
- The other half decode cleanly

Distinguishing features

- You see CRC failures on the bad packets
- Increasing CR (e.g., CR7 → CR8) improves the rate
- RSSI may look fine, but SNR is marginal

6. Hidden Node Problem

Two nodes cannot hear each other but both can hear the repeater. They transmit “blind” and collide at the repeater.

Why it stabilizes at 50%

If both nodes have similar traffic rates, the repeater sees roughly half the packets collide.

Clues

- Nodes show good RSSI to the repeater
- Nodes show no RSSI to each other
- Floods succeed more often than DMs
- Increasing txdelay improves the situation

7. Antenna or Feedline Issue on One Node

A partially failing antenna (loose SMA, water ingress, cracked trace) can produce a stable 50% decode rate because:

- Some packets arrive during constructive multipath
- Others arrive during destructive multipath
- The pattern averages out to ~50%

This is less common but is real in actual deployments.

SUMMARY:

Symptom

Likely Cause

Loss is evenly distributed, not bursty	Collisions / txdelay too low
Loss only on DM or ACKs	Directed-path collision or repeater timing skew
CRC errors on half the packets	SNR borderline / FEC limit
Only packets through one repeater fail	Repeater timing skew or antenna issue
Floods succeed but DMs fail	Hidden node or deterministic collision
Increasing txdelay improves PER	Collision-dominated network

It is important for repeater owners to implement the *rxdelay*, *txdelay*, and *direct.txdelay* settings in White Paper #1. Universal adoption will reduce the number of collisions and make a better Mesh for everyone.