

Eliminating Digipeater (Source) Routing from AX.25

When the AX.25 protocol was developed, there was only one commonly available public network: X.25. There were a number of other dedicated network protocols such as SNA, DECNET, and a new kid on the block, IP.

The following paragraphs are from the AX.25 Version 2.0 specification, section 2.1:

“In order to provide a mechanism for the reliable transport of data between two signaling terminals, it is necessary to define a protocol that can accept and deliver data over a variety of types of communications links. The AX.25 Link- Layer Protocol is designed to provide this service, independent of any other level that may or may not exist.”

“This protocol follows, in principle, the CCITT X.25 Recommendation, with the exception of an extended address field and the addition of the Unnumbered Information (UI) frame. It also follows the principles of CCITT Recommendation Q.921 (LAPD) in the use of multiple links, distinguished by the address field, on a single shared channel.”

Two things stand out here:

- The AX.25 protocol is a link-layer protocol, designed to be independent of any other level.
- The UI frame was an addition to the AX.25 protocol. In fact, there was a UI frame in the X.25 protocol, but it was used solely for device-to-device signaling.

Again from the AX.25 Version 2.0 specification, section 2.2.13.3:

“The link-layer AX.25 protocol allows operation through more than one repeater, creating a primitive frame routing mechanism. Up to eight repeaters may be used by extending the repeater-address subfield. ...”

“It is anticipated that multiple-repeater operation is a temporary method of interconnecting stations over large distances until such time that a layer 3 protocol is in use. Once this layer 3 protocol becomes operational, repeater chaining should be phased out.”

It is important to note that the AX.25 Version 2.2 specification section 3.12.4 amends 2.2.13.3 as:

“Evolving consensus opinion is that repeater chaining belongs to a higher protocol layer. Consequently, it is being phased out of Layer 2, although backward compatibility is being maintained with a limit of two repeaters.”

Note that it was the original intent and continues to be the intent to eliminate repeater chaining from the link layer. What might be the reason for this?

1. Multiple repeats of the same packet on the same frequency substantially increase the potential for collisions. This is simple math proven out by studies such as the University of Hawaii packet radio studies and do not bear reproducing here. Because of topology, natural RF attenuators, etc., increasing the number of repeaters increases exponentially the potential for collisions.
2. Layer 3 and above networking allow for use of non-primary frequency repeating. Amateurs have done this in the analog world for years, linking repeaters using non-repeater frequencies and sometimes non-RF methods. Yet, when packet radio came about, it seemed that everyone had to see how many digipeaters they could get through. Of course, the effectiveness of this and the interference caused by this idea was not factored in. Unfortunately, this mentality cost most of the amateur packet radio networks their usability and most have disappeared as a consequence.

3. The digipeating algorithm used in most TNCs is rudimentary and requires knowledge of the local network topology by each network user. The algorithm also precludes the TNC sysop from controlling their station's operating characteristics as those characteristics are defined by the network users, not the sysop.

The only single-frequency, half-duplex network still in widespread use today is the APRS (Automatic Position Reporting System) network. APRS utilizes the AX.25 UI protocol for its primary transport mechanism and was originally designed completely around it. The APRS protocol has since expanded beyond the AX.25 limits such as SSID, but its roots are still very visible. One feature of the AX.25 protocol that Robert Bruninga accessed early on was the repeater field.

APRS originally used the "alias" capabilities of commonly available TNCs. "Common" aliases were created called RELAY (to be available in all digipeaters), WIDE, and TRACE (WIDE and TRACE to only be available in wide-area digipeaters). While this was a nice way to increase activity, it was soon found to cause a large amount of multiple repeats of packets forcing Mr. Bruninga to find a new way to handle common digipeating. He introduced two new concepts and, with the help of Kantronics, got them partially implemented. The first concept that was never universally adopted was to use the destination SSID to indicate what direction and how far a packet should travel. The second concept was the N-n algorithm that basically indicated that a packet should travel n hops from the source.

Note that the N-n algorithm does not cause a packet to be repeated n times. Rather, the idea is for the packet to go in all directions n hops. In a densely populated area (from a digipeater standpoint), this can cause an exponential number of repeats causing interference over a very large area. Unfortunately many of those repeats are created by the same digipeaters repeating the packet more than once. This is due to the broken or nonexistent implementations of duplicate checking in the digipeater programs.

A further attempt to reduce interference has been put forth by Mr. Bruninga this year by what he calls "the new N-n paradigm". In fact, this is simply a rework of his original algorithm with limits put in place to attempt to reduce interference. While it has helped to a certain degree in some areas, it has also caused a large amount of confusion amongst users and digipeater sysops. The "new paradigm" also leaves open the same holes that have dogged the APRS network (and any other AX.25 network) for years: the user dictates the repeater routing on a single, half-duplex channel.

So what is the solution? Eliminate multiple-repeater routing from the digipeat algorithm. While this is a fairly simple solution, it has met with a significant amount of resistance from those who believe that they should have the "right" to dictate where their packet goes, even though they do not have this "right" with any other amateur radio network. As demonstrated above, this is **not** a "right" with AX.25 and it was never meant to be abused as much as it has over the past 20 years.

At the bequest of a number of amateurs, I published the first version of the No-Source-Route (NSR) algorithm for UI digipeaters in the Winter 2005 PSR. There were a number of comments that caused additions to be made so that it could be implemented in today's networks with a minimum of adverse impact. One key point has remained constant, however: there is no provision for a user to choose their digipeater route (no source routing).

The final version was published in the Summer 2005 PSR and is also found at www.aprs-is.net. It is also shown here for review:

1. **The digipeater repeats any packets it sees from a station on its "must digipeat" list. This allows specific remote stations to make it into the "metro" LAN as deemed proper by the digipeater sysop. The digipeater will modify the path by simply appending its call with the H-bit set after the "OK" list digipeater call plus any path defined by the sysop.**

2. **The digipeater repeats everything it sees directly (no digipeated packets), stripping the entire path away and replacing it with just the digipeater's callsign with the H-bit set plus any path defined by the sysop.**
3. **The digipeater repeats any packets it sees last digipeated by a digipeater on its "OK" list unless it sees that it has been digipeated by a digipeater on its "exclude" list. This allows remote areas to make it into the "metro" LAN as deemed proper by the digipeater sysop. The digipeater will modify the path by simply appending its call with the H-bit set after the "OK" list digipeater call plus any path defined by the sysop.**

RELAY can be on the "OK" list allowing people to set up low-level RELAY alias digipeaters. Packets with RELAY in the path would only be digipeated if RELAY is in the first position and no place else.

The digipeater does full dupe checking (CRC or checksum) based on from call, unproto (not including the SSIS), I field length, I field data. The digipeater will not digipeat any packet where its callsign appears before or including the call with the H-bit set in the path. The depth of this dupe check would only need to be about 30 packets long.

The algorithm removes source routing completely from the UI protocol. It allows the digipeater sysop to define what makes up the local area network. These local area networks will normally be no more than 2 digipeaters deep, thereby restricting the amount of interference generated by repeated packets and also reducing the potential for collisions.

The basic NSR algorithm is as follows:

A NSR digipeater will repeat every packet heard directly (no H-bits set in the address field). The digipeater's callsign-SSID with the H-bit set will replace any existing path. A similar action occurs with a packet heard directly from a digipeater on the "OK" list. Any path after the "OK" digipeater's callsign-SSID will be replaced with the NSR digipeater's callsign-SSID with the H-bit set. All of this allows tracing of a packet's path while preventing disruption of adjacent non-NSR networks due to user attempts to source route.

Three important additions to the NSR algorithm have been made:

1. A station can be on the "must digipeat" list. This allows a sysop to specify a specific station's packets to be digipeated regardless of how they got to that point. A potential use for this is to pass packets from a "distant" weather office to local spotters. However, this may be better served by the gateways described later in this document.
2. A sysop can define path(s) that will be appended to packets digipeated based on standard criteria (direct heard, digipeater on "OK" list). This allows the sysop to install the NSR digipeater in a source-routed network thereby easing the transition to a NSR network.
3. A sysop can exclude packets repeated by specific digipeaters. The digipeat on the "OK" list rule is now able to be amended so that it reads "Digipeat if last digipeated by a digipeater on the "OK" list UNLESS it has already been digipeated by a digipeater on the "exclude" list." This allows the sysop to define a limit to where a packet can originate from thereby defining the LAN both inbound and outbound.

The only thing missing, on the surface, from the NSR algorithm is how to do routing over extended distances. That is missing because it should not be attempted at the link level on a single, half-duplex channel. Fortunately for APRS, there is a distance routing solution already implemented called gateways. Gateways provide full passage from the APRS RF network to the APRS backbone. Gateways provide sysop-defined passage from the backbone to the RF network. This limited gating to the RF network is necessary due to the massive amount of data passed on the backbone. The gateways also gate APRS message packets to RF if the receiving station is within range. Gateways in conjunction with NSR digipeaters provide for true worldwide

communications with the user only needing to know how to turn on their client. Most of the current APRS backbone consists of the Internet although RF backbones are implemented in some parts of the world.

A network implemented with NSR digipeaters will provide superior performance and reliability to local operations over any RF AX.25 network currently in place. This is because of the reduced bandwidth requirements and improved gateway performance since packets won't be appearing via RF from distant locations. Emergency communications are enhanced since the responders do not need to have any knowledge of network topology. Standard communications are also enhanced since distant stations are unable to disrupt local operations.

For APRS to expand beyond the limits it is currently experiencing, it is time to use a digipeater protocol designed specifically for an AX.25 UI network, not designed for an APRS network. By separating the link layer from the network layer, both bandwidth on the RF network and usability of the APRS global network is enhanced a minimum of twofold.

Peter Loveall AE5PL

Author of javAPRSSrvr, javAPRSIGate and javAPRSDigi

Author of the NSR algorithm