

# APRS 数字中继器算法

---

John Langner, WB2OSZ  
Rough Draft 1 - August 2022  
Yuluxk, BG2FFJ  
2025 年 4 月译制

业余无线电中继台通过转发其他电台的信号以扩大通信范围。

模拟语音中继台在一个频率上接收信号，同时在另一个频率上实时转发相同的信号。

AX.25 数字中继台（即“数字中继器”或 *digipeater*）采用“存储转发”（store and forward）方式：接收一个数据包，对其进行检查，必要时进行修改，然后再转发。通常会在同一个无线电信道上转发，但多端口的数字中继器也可以连接多个无线电信道，将一个信道上接收到的数据包转发到另一个信道。

[APRS 协议参考文档 1.0.1](#) 并未具体描述数字中继器应如何工作。我们所依赖的，仅是早期 TNC（终端节点控制器）所形成的先例，这些先例早在 WIDEn-N 机制出现之前就已存在。因此，当前存在许多实现方式，它们的行为并不完全一致。较新的实现方法已针对传统 TNC 的不足之处，探索出不同的改进方案，以克服其局限。

## 1 标准“TNC-2”监控格式

首先我们需要理解标准显示格式所表达的内容，这样才能理解示例和空中接收到的实际数据包

```
source > destination : information source >  
destination , digipeater1 : information  
source > destination , digipeater1, ... , digipeater8 : information
```

标准显示格式以两个或更多地址开始：

源地址	<p>-发起站。</p> <p>通常是业余无线电呼号，但也可能是一个战术呼号。它可以选择性地跟随一个“-”和一个数字（最大为 15），称为子站识别符（SSID），以允许最多 16 个电台使用相同的呼号。</p>
目的地址	<p>-在传统的连接模式分组中，这是一个特定的电台。</p> <p>在 APRS 中，它有多种用途。我们在此讨论中可以忽略它。</p>
经由路径	<p>-最多 8 个地址，表示数据包已经经过以及可能前往的路径</p> <p><u>当一个数字中继器地址后跟有“*”时，表示该中继器及其之前的所有中继器地址已被使用。它们显示了数据包已经被转发过的路径。</u></p> <p><i>（在 AX.25 帧的底层，每个中继器地址都有一个“已使用”H 位，用于指示该地址是否已被使用。）</i></p>

## 2 手动路由

假设我想要将一个数据包显式地通过 N2GH 中继器，再通过 W2UB 中继器进行中继。原始数据包将如下所示，列出了两个特定的中继器。注意，这里没有“\*”号，因此我们接收到的是原始（源）电台的信号：

```
WB2OSZ>APRS,N2GH,W2UB:something
```

N2GH 识别到自己的名称位于第一个未使用的中继器位置中，并转发该数据包。结果如下所示：

```
WB2OSZ>APRS,N2GH*,W2UB:something
```

N2GH 中继器在其地址上设置“已使用”标志（AX.25 协议中的“H”位），以表示它已被使用，不会再被用于后续中继决策。当你看到某个中继器名称后面带有“\*”时，你可以确定你听到的是它的转发信号。

同样的事情再次发生。W2UB 看到自己的名称出现在第一个未使用的中继器位置上，然后转发该数据包。你会看到呼号后带有“\*”，这说明你正在听它的传输：

```
WB2OSZ>APRS,N2GH,W2UB*:something
```

所有中继器地址都已被使用，因此该数据包不能再次被中继。

某些软件可能会将其显示为带有两个“\*”的形式：

```
WB2OSZ>APRS,N2GH*,W2UB*:something ← 错误
```

这种写法看起来可能更直观（表示两个地址都已被使用），但它是错误的。这不符合标准监控格式的规则：只有**最后一个**被使用的中继器标记为“\*”，而前面的地址默认已被使用。在格式中，永远不应该出现“已使用”的地址排在“未使用”的地址之后。

如果所有节点行为规范，你可以据此得出以下结论：

- 该数据包最初来自 WB2OSZ。
- 它被 N2GH 转发。（因此 N2GH 能听到 WB2OSZ。）
- 它被 W2UB 转发。（因此 W2UB 能听到 N2GH。）
- 你接收到的是来自 W2UB 的传输。（因为它后面有\*。）

来自 [AX.25 协议规范](#) 的内容：

*“目标电台可以通过检查地址字段来判断帧所经过的路径，并使用该路径返回帧。”*

第二句话在理论上可能成立，但在实践中并不总是可行。你可能遇到这样的情况：电台 X 能听到电台 Y，但 Y 却听不到 X，因此相同的反向路径并不适用。

正如我们之后会看到的，一些实现方式并不规范，因此我们实际上无法确定数据包到底经过了哪些路径。它们可能没有标识自身，或者在路径中留下无用的内容，造成信息模糊不清的情况。

### 3 什么会被中继？

显然，数字中继器不应转发它所接收到的所有内容。如果它那样做，任何在信道上被接收到的内容都会在所有可用的中继器之间来回弹跳。

因此，首先发送方需要构造一个合适的 *via* 路径；其次，中继器需要有适当的配置来指定其行为方式。

使用特定的电台名称通常并不理想。有哪些中继器可用？谁能听到我？谁能听到能听到我的中继器？如果我常用的中继器不可用会怎样？如果我在旅行，不知道周边情况又怎么办？

“别名”（Aliases）可以使中继器除了响应自己呼号以外，还能响应额外的名称。多个电台可以响应相同的别名。例如，当地的应急指挥中心（EOC）可能响应别名“EOC”，这样你就无需记住实际使用的呼号。位于华盛顿山顶的中继器可能响应别名“MTWASH”。有时移动中继器会响应“TEST”这个别名。

20 世纪的传统 TNC 灵活性有限，最多只支持 4 个别名。例如：

```
UIDIGI ON EOC,TEST
```

如果我发出如下内容：

```
WB2OSZ>APRS,EOC:something
```

它可能会被转发为：

```
WB2OSZ>APRS,KB1MKZ*:something ← 正确
```

别名总是会被中继器的呼号所替换。不应被如下形式转发，其中别名本身被标记为已使用：

```
WB2OSZ>APRS,EOC*:something ← 错误
```

别名应始终被中继器的呼号替换。这也回到了之前提到的一条规则：已使用的地址应反映数据包从源电台传输至你所经过的路径。

有两个不同的旧版 TNC 手册并未提及 UIDIGI 是否具备重复抑制功能（而对于 UITRACE 有明确说明），因此它们可能会在无线电信道中造成大量重复数据。重复抑制将在后文讨论。

## 4 新的 n-N 模式

大多数情况下，我们并不希望手动指定一系列特定的中继器。相反，我们通常希望指定任何中继器，或具有某些属性的中继器，例如地理区域。

在 APRS 的早期，中继器别名“RELAY”和“WIDE”被广泛使用。但自 2004 年以来，这些别名已被废弃，所有关于它们的使用应当早已被移除。所以我们不再讨论它们，因为那样只会引起混淆。

修复 144.39 APRS 网络

新 n-N 模式 <http://www.aprs.org/fix14439.html>

目前被接受的方法是以通用形式指定 APRS 中继器的类别，即 XXXn-N。

XXX	前缀，最多 5 个字符。通常是“WIDE”，但也可以用于地理区域、特殊事件或其他用途。例如，“MA”可以代表马萨诸塞州。移动中继器站有时会启用“TEST”。
-----	--

<i>n</i>	通常是 1，表示本地的短距离“填充”中继器。2 通常用于具有长距离的良好位置。理论上可以使用到 7，但实际上通常只使用 1 和 2。
<i>N</i>	剩下的数字表示数据包应被中继的最大次数，也称为跳数（hop count）。初始值通常在 1 到 7 之间递减，直到达到 0 时不再进行中继。

20 世纪的 TNC 并不提供太多灵活性。

```
UITRACE WIDE, 30
```

这意味着它会响应一个由以下部分组成的地址：

- 字符串 “WIDE”。
- 一个介于 1 到 7 之间的数字。
- 一个介于 1 到 7 之间的 SSID。

这种方式并不太具可定制性。它将匹配 49 种不同的组合，如 WIDE1-1、WIDE1-7、WIDE2-2、WIDE7-5 等。

似乎没有办法指定多个通用别名或在 SSID 前面指定数字。

传统的中继器配置命令对于 2004 年后的 APRS 已不再适用。较新的实现方式提出了不同的方案，为配置行为提供更多灵活性。

## 5 数据包失控

如果我们不小心，数据包的中继可能会完全失控。一个原始数据包可能会被多个中继器接收到，并被它们各自转发。越来越大的中继器环路会听到多个其他中继器的信号，并转发它们所接收到的每个数据包。原始电台可能会听到其他电台转发自己发送的数据包并再次转发，形成一个回路。

我们可以做几件事情来控制这种情况。

### 5.1 减少跳数

原始电台指定数据包可以被转发的最大次数。例如，我们可能从以下内容开始：

```
WB2OSZ>XXXX,WIDE1-3:whatever
```

配置为响应模式 WIDE1-N 的中继器会将跳数减少到 2，并插入它自己的呼号：

```
WB2OSZ>XXXX,WW1ABC*,WIDE1-2:whatever
```

另一个中继器会将跳数减少到 1 并发送：

```
WB2OSZ>XXXX,WW1ABC,WW2DEF*,WIDE1-1:whatever
```

请注意，只有最后一个中继器名称后面有一个“\*”。接下来的中继器会将跳数减少，并发送以下两种形式之一：

```
WB2OSZ>XXXX,WW1ABC,WW2DEF,W3GHI*:whatever
```

I 在第一个示例中，WIDE1-0 已经用完，因此我们可以丢弃它，使数据包变得更小。一些实现会在 WIDE1 用尽后依然保留其别名：

```
WB2OSZ>XXXX,WW1ABC,WW2DEF,W3GHI,WIDE1*:whatever
```

这会创建一个模糊的情况。我们是听到了 W3GHI 还是听到某个没有自我标识的其他电台？我认为第一个方法更有意义，并且符合 [AX.25 协议规范](#) 中关于中继路径的意图：

*“目的站可以通过检查地址字段来确定数据帧到达它的路径，并使用此路径来返回数据帧。”*

总之，中继器地址已经全部使用完，因此该数据包无法再被中继。

## 5.2 从空闲信道到开始传输的延迟

通常，当我们准备传输时，我们会等待一个空闲信道，然后等待一个随机的时间，以减少与其他人同时传输的机会。这是基于 SLOTTIME 和 PERSIST 参数的设置。

对于常见的默认值，我们会遇到以下概率的延迟：

延迟（毫秒）	概率
100	.25 = 25%

200	.75 * .25	= 19%
300	.75 * .75 * .25	= 14%
400	.75 * .75 * .75 * .25	= 11%
500	.75 * .75 * .75 * .75 * .25	= 8%
600	.75 * .75 * .75 * .75 * .75 * .25	= 6%
700	.75 * .75 * .75 * .75 * .75 * .75 * .25	= 4%
etc.	...	

如果在这个随机等待时间内检测到信号，我们就会重新开始等待过程。

在中继转发的情况下，当信道变得空闲时，我们会立即开始传输。中继器不会试图避免碰撞，而是立即同时开始在同一信道上进行传输。AX.25 协议规范将这些称为“加速”数据帧。由于 FM 捕获效应，最强的信号应当获胜。

传统 TNC 通常有一个名为 UIDWAIT 的参数，必须关闭才能使其正常工作。

使用 KISS TNC 作为中继器并不是一个好主意，因为 KISS 协议无法为单个数据包应用“nowait”功能。如果您使用 KISS TNC 和一个单独的应用程序作为中继器，它会等到经过良好配置的中继器同时传输时才开始，可能会与其他表现不佳的中继器产生冲突，从而造成额外的不必要的拥堵。

## 5.3 重复抑制

解决方案的第三部分是避免在一定时间内发送重复数据包，通常是 30 秒。中继器必须记住它传输的每个数据包，并且在 30 秒内不能重复传输相同的内容。比较时只涉及源地址、目标地址和信息部分。换句话说，检查两个数据包是否相同时，会忽略变化的中继器路径。

中继器不应该重复传输源字段中包含自身呼号的数据包，因为这可能会导致循环。

## 6 中继器算法总结

中继器的配置包括：

- 其自身的呼号。
- 一个可选的别名集合，表示它将响应的附加名称。
- 一个可选的“通用”地址集合，其中 SSID 是剩余的跳数。通常，对于短程中继器，这将是 WIDE1-N。它应该至少能够添加一个用于地理区域或其他特殊目的的地址。

## 6.1 决定是否符合中继条件

- (a) 查找中继器地址中的第一个“未使用”地址。如果未找到，返回“否”。
- (b) 源地址是否是我的站点地址？如果是，返回“否”。
- (c) 如果第一个未使用的中继器地址是我的站点地址，返回“是”。
- (d) 如果第一个未使用的中继器地址是我的别名之一，返回“是”。
- (e) 如果第一个未使用的中继器地址是某种通用形式，如 XXXXn-N，并且与我的配置规则匹配，返回“是”。

传统的 UITRACE 仅允许一个前缀，例如 “WIDE”，它将匹配“WIDE”、1 到 7、“-” 和 1 到 7。

一些较新的实现提供更大的灵活性，包括多个前缀以及区分“WIDE1”和“WIDE2”。

- (f) 否则，返回“否”。

## 6.2 抑制任何重复数据包

- (a) 如果我们最近（通常在 30 秒内）已经转发过此数据包，则返回“否”。

比较基于源地址、目标地址（不包括 SSID）和信息部分。它不包括中继器地址。

实现通常使用 16 位或 32 位哈希，而不是保存整个字符串，以减少存储空间和比较时间。

## 6.3 如果应重新发送且地址是我的呼号或我的别名之一

- (a) 将地址替换为我的呼号。
- (b) 将该地址标记为已使用。
- (c) 进行传输。

**或者，如果应重新发送且地址形式为 XXXXn-N:**

- (a) 如果  $N \geq 2$ ，N 值递减。在其前面插入中继器呼号，并在地址数少于 8 时标记为已使用。



例如:           W9XYZ >APRS,WIDE2-2  
变为:           W9XYZ >APRS,WB2OSZ\*,WIDE2-1

- (b) 如果  $N = 1$ ，我们不想将  $WIDEn-0$  保留在中继器列表中，因此通用地址被中继器呼号替代并标记为已使用。

例如:           W9XYZ >APRS,WIDE2-1  
变为:           W9XYZ >APRS,WB2OSZ\*

- (c) 如果  $N = 0$ ，跳数已经用尽，数据包不再被中继。这是一个错误情况，我们不应遇到。如果地址计数已经用尽，我们应当期望设置“已使用”H 位。这里存在至少一个有缺陷的实现会产生这种情况。

## 6.4 传输并记住它

- (a) 传输修改后的数据包，不在频道清空后添加正常的随机延迟。
- (b) 将其添加到最近重新传输的数据包列表中，并附上时间戳。