

SOFTWARE LIBRE PARA TRANSMISIÓN DIGITAL EN ENLACES RADIO

Arnau Sánchez Sala, Joaquín Seoane Pascual

Fundación EHAS, UPM

arnau@ehas.org, joaquin@ehas.org

RESUMEN

Este documento describe la arquitectura de un sistema de comunicación digital con transceptores radio de voz en las bandas HF y VHF/UHF. Aunque el sistema está orientado principalmente al correo electrónico, se proponen también soluciones para el acceso a páginas web y mensajería instantánea. Se pone especial énfasis en el uso de tecnologías de bajo coste, alta eficiencia, y en la utilización de software libre y gratuito en su implementación.

Este proyecto se ha desarrollado en el marco del programa EHAS (Enlace Hispano Americano de Salud) [1], cuyo objetivo principal es contribuir a la mejora del sistema público de asistencia sanitaria en zonas rurales aisladas de países de América Latina por medio de las telecomunicaciones y la informática.

1. INTRODUCCIÓN

Las evaluaciones de necesidades realizadas en zonas rurales aisladas [2] han demostrado que el principal servicio, necesario y que claramente salva vidas, es la simple comunicación de voz en el ámbito local. Una vez satisfecha esta necesidad fundamental, sin embargo, la comunicación de datos se convierte también en un complemento valioso. La solución propuesta en el programa EHAS, pasa siempre por el uso de transceptores de voz comerciales, no preparados para la transmisión de datos.

El desarrollo de tecnología para la transmisión digital en enlaces radio no ha tenido un desarrollo comparable al de otros ámbitos (como pueda ser el acceso telefónico o la redes inalámbricas WiFi), por lo que los productos comerciales disponibles tienen, exceptuando los que alcanzan rendimientos muy pobres, un coste elevado, cuando no prohibitivo.

Una solución de bajo coste pasará no sólo por el aprovechamiento de los equipos de voz para datos, sino también por alcanzar una alta eficiencia que permita el uso de potencias más bajas en los transceptores (importante en estaciones que muchas veces se alimentan con placas solares) y antenas de menor altura (con el consiguiente ahorro en las torres).

Debido al bajo ancho de banda, se focalizará el esfuerzo en la transmisión de correo electrónico, sin desestimar la posibilidad de acceso a páginas web e incluso a mensajería (servicio muy interesante dado el poco ancho de banda que consume y la elevada interactividad que ofrece al usuario).

2. SUBREDES RADIO

Uno de los medios más fiables para garantizar la comunicación en zonas de cobertura de corta y media distancia sin visibilidad directa (en los que podría usar tecnología WiFi con mayores prestaciones), es la comunicación en la banda VHF/UHF (30-3000 Mhz), que permite conectar estaciones en un radio aproximado de 50km (según orografía) con una buena calidad de voz. La banda VHF/UHF (en adelante aludiremos a ella sólo como VHF) presenta una gran estabilidad, y no depende -a diferencia de onda corta- de las condiciones medioambientales o del momento del día..

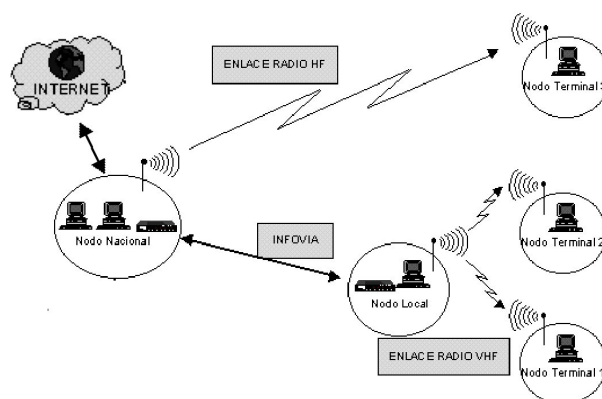


Figura 1: Redes radio en EHAS

La banda de HF (3-30MHz), permite comunicaciones de larga y muy larga distancia gracias a un fenómeno conocido como propagación ionosférica, consistente en la reflexión de las señales de radiofrecuencia en las capas altas de la atmósfera (la más importante de ellas situada a 250 km de altitud). El principal inconveniente de esta banda es la baja calidad de los canales, puesto que las señales transmitidas se encuentran expuestas a efectos de absorción atmosférica, elevado ruido y un acusado *multipath* (multicamino). Además, las condiciones de transmisión dependen de muchos factores (momento del día, estación del año, actividad de las manchas solares, tormentas ionosféricas, etc).

Tanto en VHF como en HF, la topología más habitual de las redes EHAS es la centralizada (figura 1), en la que varios clientes se conectan a un mismo servidor (del que normalmente dependen administrativamente), y que habitualmente dispone de salida a internet. Todo ello sin excluir otras topologías más complejas, por ejemplo aquéllas en las que un servidor es, a su vez, cliente en otra subred.

3. EQUIPAMIENTO DE LAS ESTACIONES

3.1. Equipo informático

Una condición básica, para no comprometer el coste global de las instalaciones, es que los ordenadores utilizados no tengan requisitos especiales en cuanto a potencia de cálculo o capacidad; orientativamente, se propone como mínimo un equipo de gama media, con una CPU a 500Mhz, memoria RAM de 128Mb y disco duro de 10Gb.

3.2. Sistema operativo

GNU/Linux es un sistema operativo libre y gratuito [3] de uso general, seguramente el más popular de los diversos existentes. Y aunque sigue siendo cierto que los sistemas de licencia Microsoft aún copan el mercado entre los usuarios domésticos, también lo es que Linux ha experimentado un notable desarrollo en los últimos años (su uso como servidores en internet, en cambio, es generalizado desde hace mucho tiempo). Afortunadamente, las modernas distribuciones de Linux no presuponen un alto conocimiento informático como antaño, gracias al desarrollo de interfaces gráficas más amigables (los proyectos *Gnome* y *KDE* destacan entre ellas).

Aunque Linux se presenta como un sistema único en su núcleo, son diversas las distribuciones -comerciales o no- que han visto la luz en los últimos años (entre muchas otras: *Debian*, *Red Hat*, *Slackware*, *Mandrake* o *SuSe*). Entre ellas, para el desarrollo de este proyecto se ha elegido *Debian* [4], aunque en ningún momento se utiliza software que no esté disponible o no sea compatible con las otras distribuciones. Entre otras particularidades, *Debian* destaca por la enorme cantidad de software disponible (en su mayoría aplicaciones GNU, o sea, libres y gratuitas, tanto los binarios como las fuentes) y el carácter totalmente altruista (basado en el trabajo de voluntarios) que guía su filosofía lo hace especialmente atractivo para un proyecto de estas características.

3.3. Transceptores

Las prestaciones que ofrecen los transceptores comerciales son muy dispares en función del fabricante, así que se recomienda una selección cuidadosa de los equipos.

En lo que respecta a VHF/UHF, el elemento más importante es comprobar que los filtros de audio (tanto en entrada como en salida) puedan ser desactivados; de otro modo, el ancho de banda quedaría cortado en torno a los 3KHz (banda de voz), con una limitación en la velocidad de transmisión digital.

En HF el principal escollo son las diferentes calidades de recepción según modelos y marcas. Los equipos de gama baja seguramente funcionarán, pero con peores prestaciones que los de gama media/alta. Entre otros, un problema habitual de los equipos baratos es el desplazamiento en frecuencia que sufren las señales transmitidas debido a la poca exactitud de los osciladores internos.

En cualquier caso, un factor fundamental a tener en cuenta a la hora de elegir el transceptor (tanto en VHF/UHF como en HF), es comprobar que la interconexión entre radio y ordenador se puede hacer de forma sencilla y no molesta para el usuario: el equipo debe disponer de un puerto especial (situado normalmente en la parte trasera) en el que tengamos, al menos, de las líneas de audio (recepción/transmisión) y PTT (*Push-to-Talk*), que es la línea digital utilizada para la conmutación entre recepción y transmisión (las radios son siempre *half-duplex*, o reciben o transmiten).

4. ARQUITECTURA HARDWARE

La conexión entre cualquier medio físico y un equipo digital exige de un dispositivo diseñado para esa función. De igual modo que para transmitir en una línea telefónica necesitamos un módem, para acceder al medio radio necesitamos un módem que haga de interfaz entre la radio y el ordenador. En la terminología habitual de comunicaciones radio, los módems que realizan esta función son conocidos como TNC (*Terminal Node Controllers*). Los precios y prestaciones de estos equipos varían drásticamente según los fabricantes. En la gama más baja tenemos los usados por radioaficionados, de precio muy asequibles pero de pobre rendimiento; en la alta los módems comerciales, con buenos rendimientos pero con precios muy elevados, especialmente los de HF.

El objetivo de este proyecto es conseguir calidades comparables a la de los módems comerciales evitando en todo momento el uso de protocolos propietarios o hardware especializado. Ello se consigue, en primer lugar, asignando las tareas de modulación y demodulación al propio ordenador del usuario. La potencia de los procesadores modernos permiten que parte de la CPU se dedique al procesamiento de señal del módem, sin interferir en el trabajo del usuario (de una forma parecida a como hacen los modernos *softmodem* telefónicos).

No obstante, aún sigue siendo necesario un dispositivo por el que el ordenador pueda intercambiar audio con la radio; el hardware más económico, y que de hecho ya viene integrado en la mayoría de equipos, es una tarjeta de sonido, que servirá a la perfección -y sin una carga significativa- para las tareas de conversión A/D y D/A.



Figura 2: Placa de interfaz USB

De este modo, mientras las tareas de procesamiento de audio quedan encomendadas a la tarjeta de sonido, resta una función, la de activación del PTT, que no puede realizar una tarjeta de sonido por la carencia de salidas digitales. Esta función quedará reservada a una tarjeta especialmente diseñada en EHAS para este fin que denominaremos *placa de interfaz USB* (figura 2). Tradicionalmente las placas de interfaz para transceptores radio utilizaban los puertos serie o paralelo; en los últimos años, sin embargo, se ha experimentado el auge imparable del puerto USB, mucho más versátil y rápido, hasta el punto que son muchos los equipos (especialmente portátiles), que precinden ya de los tradicionales puertos para incluir únicamente USB. Por ello, la placa de interfaz EHAS ha sido diseñada especialmente para este puerto.

Las necesidades especiales de las estaciones EHAS en lo que respecta al uso de radios, baterías y paneles solares, hace necesaria la monitorización y control de diversos parámetros del sistema, entre otros el de nivel de batería, temperatura de la radio (con su correspondiente ventilador), cálculo del SWR (*Standing Wave Ratio*, factor de onda reflejada), etc. Todas estas funciones han sido también incorporadas en la misma placa de interfaz, en cuyo núcleo usamos el microcontrolador *PIC16F745* de Michochip. El esquema final del conexionado queda como sigue:

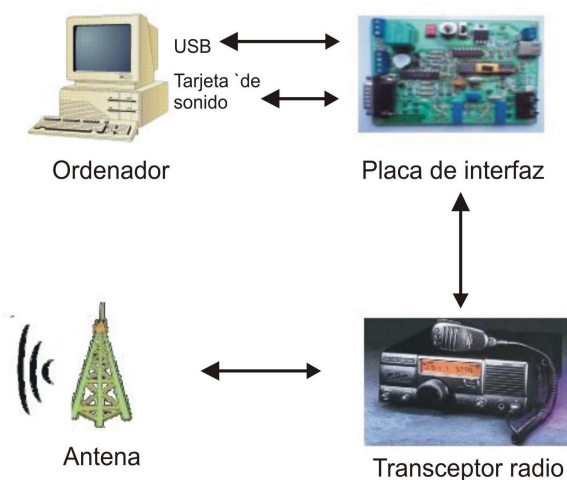


Figura 3: Equipamiento de una estación radio

5. PROTOCOLOS DE COMUNICACIÓN

Una vez decidida la arquitectura hardware, hay que analizar los protocolos que realizarán la función en cada uno de los niveles OSI.

5.1. Nivel físico

Para VHF/UHF se usarán canalizaciones (ancho de banda de canal) estándar de 12.5KHz; en HF la canalización típica es de poco más de 2 KHz. La tarea de recepción/envío de audio la llevaremos a cabo, en ambos casos, con *soundmodem* [5], un paquete libre que

proporciona la infraestructura para el acceso a tarjeta de sonido y que incluye un conjunto de módems (5 implementados por el momento) para la transmisión de información, que se escogen en función de la banda de trabajo.

Para VHF/UHF el módem escogido es *FSK-G3RUH* [6], una modificación libre de FSK que reduce considerablemente el ancho de banda final. Este módem permite llegar a una velocidad de señalización de 9600 bps para las canalizaciones estándar, una velocidad muy superior a la usada en anteriores proyectos (AFSK 1200 bps).

En onda corta, *soundmodem* usaremos un módem especialmente diseñado para esta banda, *newqpsk*, originalmente desarrollado para la placa Motorola DSP560002, y posteriormente traducido a lenguaje C y distribuido en Linux bajo licencia GNU/GPL. *Newqpsk* es un módem con tecnología OFDM (*Orthogonal Frequency Division Multiplexing*, modulación por división ortogonal de frecuencia). Esta modulación de espectro extendido consta de una serie de portadoras espaciadas en frecuencia para distribuir los datos por todo el ancho de banda del canal. La ortogonalidad se asegura con una determinada distancia entre portadoras que evitan que los datos se mezclen en el proceso de demodulación. Los beneficios de OFDM son su alta eficiencia espectral, la fortaleza a las interferencias de radiofrecuencia y una baja distorsión por *multipath* (multicamino), precisamente las características más habituales en canales HF.

Newqpsk usa 15 portadoras separadas 125 Hz (con un ancho de banda total de 2KHz), y cada una de ellas porta una modulación DQPSK (*Differential-quadrature PSK*, modulación diferencial PSK en cuadratura). La modulación diferencial implica que se trata de un módem no coherente, que aunque tiene una pérdida teórica de 3dB de relación señal-a-ruido en recepción respecto a los módems coherentes, permite mayor sencillez y menor carga de procesado de señal. La tasa de transferencia de cada portadoras es de 83.3bps (bits por segundo), lo que da una velocidad global de 2500bps. El modem incluye dos fases iniciales, de preámbulo y sincronización, tres niveles diferentes de FEC (*Forward Error Correction*, corrección de errores en recepción) con el algoritmo BCH, y diversidad espacial y temporal (*interleaving*) para aumentar la resistencia al fenómeno del *fading* (desvanecimiento) temporal y frecuencial.

5.2. Nivel de enlace

El protocolo más usado en el software libre para la transmisión digital radio es AX.25 [7], la versión para enlaces radio del protocolo X.25. Este protocolo permite múltiples conexiones punto-a-punto a nivel de enlace al mismo tiempo. El diseño específico para entorno radio se debe a que la típica forma de operar de X.25, donde las frecuencias se comparten, no es adecuada. Otra diferencia importante es que AX.25 asume que los extremos son del mismo tipo; con ello se eliminan las dos clases diferentes de dispositivos (DTE, DCE) de X.25.

5.3. Nivel de transporte

TCP/IP es sin discusión el protocolo estándar para la mayor parte de aplicaciones en Internet. Desgraciadamente, su protocolo de transporte (TCP) dista mucho de ser eficiente para enlaces radio, en los que tenemos altas tasas de error y propensión a la congestión, así que su uso debe ser desestimado. En la transmisión *off-line* de información (correo electrónico, habitualmente), el antiguo pero potente UUCP (*Unix-to-Unix Copy*) [8] es un protocolo mucho más adecuado, principalmente por su capacidad de continuar transmisiones rotas y por su fácil integración con el sistema de correo.

Para las aplicaciones en tiempo real (navegación en internet o mensajería), en cambio, UUCP no es protocolo útil. Además, las aplicaciones de este tipo suelen funcionar únicamente con conexiones TCP/IP. La solución propuesta en este caso consiste en la inclusión de *proxies* locales transparentes. El próximo apartado discute la arquitectura software y clarifica la implementación en los dos casos analizados.

6. ARQUITECTURA SOFTWARE

6.1. Acceso al correo electrónico

Aunque el protocolo UUCP se diseñó originalmente para la transmisión de ficheros entre máquinas, puede ser usado fácilmente para el transporte de correo (que no son, al fin, más que ficheros). La transmisión de correo electrónico por UUCP está consolidada desde hace años y los propios MTAs (*Mail Transfer Agent*, agente de transporte de correo) prevén su uso en la configuración estándar.

El paquete UUCP no consta de un protocolo único, y se permite elegir uno entre diversos sub-protocolos. Muchos de ellos no son adecuados para nuestro sistema, puesto que implementan funciones -como son la corrección de errores, control de flujo- que ya realiza el protocolo de enlace AX.25. La duplicidad de funciones en una pila de protocolos es obviamente muy negativa para el rendimiento final. Lo deseable es un protocolo de transporte que no realice ningún control sobre la transmisión (esto implica, por encima de todo, evitar paquetes de retransmisión). El protocolo más eficiente para el uso conjunto con AX.25, y que cumple las condiciones descritas, es el *protocolo y*. Las principales características de este protocolo son su bajo *overhead* (inferior al 0.6%), que como la mayoría de protocolos UUCP permita la reanudación de comunicaciones rotas, y que la transmisión de datos es unidireccional, la más indicada para comunicaciones radio *half-duplex*.

Otra característica deseable en canales de bajo ancho de banda es la compresión. No obstante, UUCP no incluye ningún mecanismo de este tipo, por lo que se añadió una capa adicional que implementa el programa BSMTP (*Batched Simple Mail Transport Protocol*, protocolo de transporte simple de correo en bloques) [9]. Este programa está pensado para funcionar conjuntamente con UUCP, y agrupa todos los datos a enviar en grandes bloques (*batches*) que posteriormente comprime.

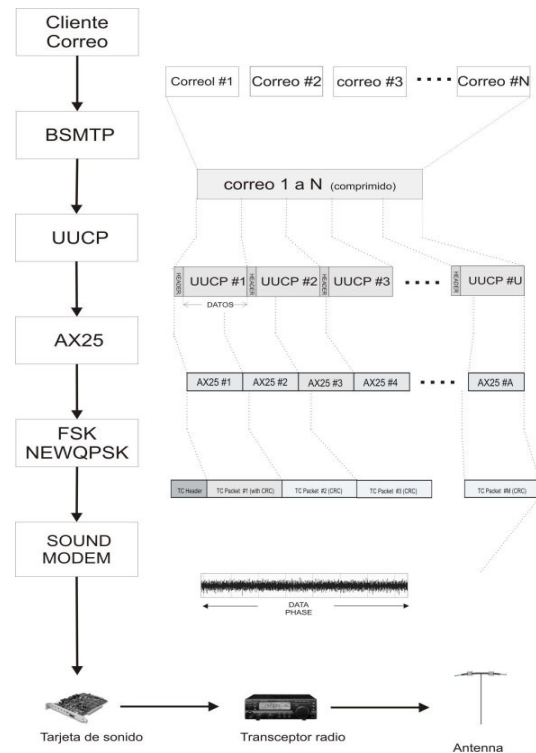


Figura 4: Acceso a correo electrónico

Por encima de UUCP/BSMTP instalamos como MTA (Agente de Transporte de Correo), el paquete *postfix* [10]. Otros MTA (*sendmail*, *exim*) son también compatibles con el sistema, pero la seguridad y fácil configuración de *postfix* han hecho preferible su uso.

Por último, en la capa más alta, la de aplicación, los usuarios pueden usar el cliente de correo que deseen, sin ninguna limitación especial, puesto que se comunicarán con el agente de correo con conexiones locales a los puertos estándar POP3 (para la recepción de correo) y SMTP (para el envío). La figura 4 resume gráficamente la estructura descrita.

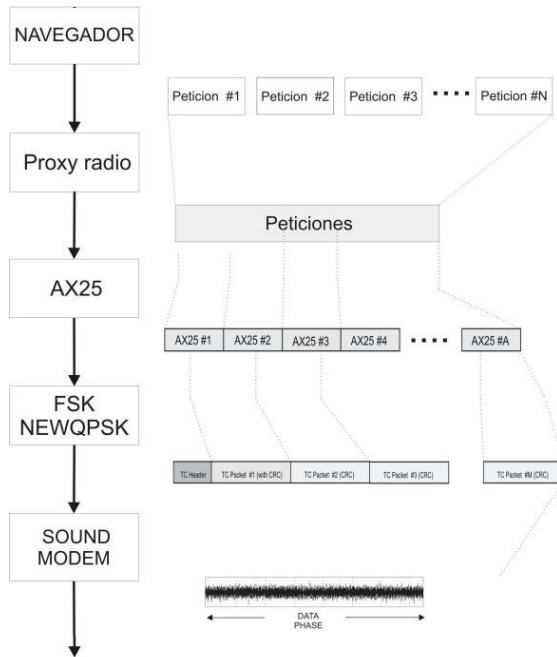
6.2. Acceso a servicios TCP/IP

Las aplicaciones de navegación en páginas web y de mensajería usan comúnmente TCP/IP, y aunque la posibilidad de usar TCP/IP sobre AX.25 (en modo desconectado) está soportada por el núcleo de Linux, el hecho que sea TCP -y no AX25- quien gestione el enlace (lo que incluye tamaño de paquetes, de ventana y los temporizadores), repercute muy negativamente en la velocidad final de la comunicación.

Como alternativa a TCP/IP se propone la creación de un enlace punto-a-punto AX25 entre la estación cliente y el servidor que le proporciona la conexión a Internet. En este enlace se encapsulan los accesos a puerto TCP/IP de la máquina local y se redirigen a la remota. Para ello, en la estación radio cliente se instala un *proxy* que escucha en los puertos TCP configurados (a los que conectará el usuario) y los redirige a otro puerto TCP de la máquina servidor.

Para que este sistema no sirva únicamente para la comunicación entre el cliente y el servidor (y no entre cliente e Internet, que es lo que finalmente buscamos), el puerto TCP redirigido conecta a un proxy HTTP del servidor. En nuestro sistema usamos el más popular y potente que existe en software libre, el paquete *squid* [11]. Todas las aplicaciones con soporte para *proxy http* funcionarán de forma transparente en nuestro sistema (ver figura 5).

Figura 5: Acceso a Internet con proxy local



7. MEJORAS INTRODUCIDAS

Hasta aquí se ha descrito un conjunto de aplicaciones libres ya existentes que trabajando conjuntamente ofrecen un sistema de transmisión digital perfectamente funcional para la transmisión de correo electrónico y el acceso a internet. Algunas de estas aplicaciones, sin embargo, presentan serias carencias que reducen la velocidad de transmisión de usuario. La evaluación del sistema permitió identificar cuáles eran los puntos flacos y se procedió a su mejora (gracias a que en el software libre todo código es accesible y modificable). Los más importantes son los cuatro que se detallan a continuación.

7.1. Sistemas robustos de codificación

Los canales con altas tasas de error (hasta del 10% en canales HF) hacen prácticamente obligada la inclusión de un sistema de codificación potente que permita la corrección de errores en recepción (FEC), aun a costa de disminuir la tasa de transmisión debido a la redundancia de información añadida.

Hemos visto que en la banda VHF/UHF se usaba el módem FSK de *soundmodem*, que originalmente no cuenta con ningún sistema de corrección de errores. Por ello se implementó un sistema de codificación

convolucional en transmisión y un decodificador *viterbi* en recepción. Las tasas de redundancia son configurables entre un factor 1 (sin redundancia) hasta un factor 3. Además se añadió el habitual sistema de *interleaving* (entrelazado) que distribuye los errores a lo largo del paquete, mejorando de forma sustancial la relación entre la tasa de errores (BER, *Bit Error Rate*) y la relación señal-a-ruido (SNR).

Para la banda de HF, los requisitos de corrección de errores son aún más importantes. La transmisión digital de datos sobre un canal de voz de 3 KHz ha sido siempre problemática, como ya hemos visto. Por ello, en vez de un sistema convencional de códigos convolucionales, se optó por un sistema más potente, fruto de trabajos relativamente recientes, y conocido como *turbocódigos* [13]. Los turbocódigos constituyen una potente técnica de corrección de errores que mejora a las anteriormente conocidas, y se usa habitualmente en sistemas de comunicación que necesitan un significativo ahorro de potencia o en los que, como en la banda HF, la relación señal a ruido (SNR) es extremadamente baja.

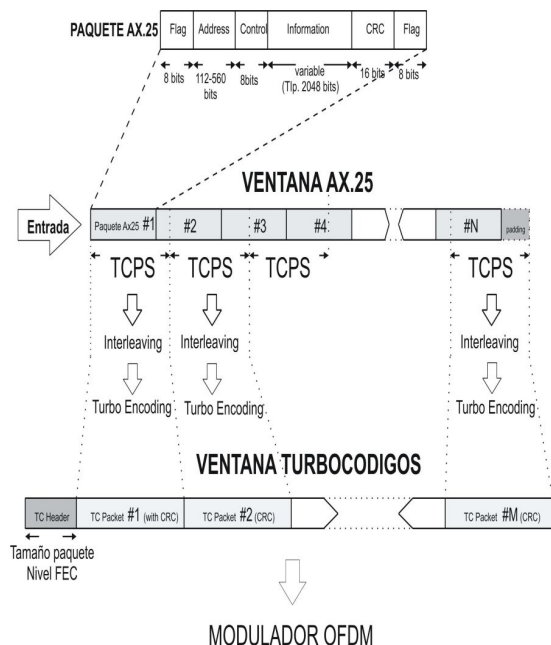


Figura 6: AX25 con codificación turbo

La implementación original de *newqpsk* (el módem para HF) empleaba para la corrección de errores un simple código de bloque BCH, que fue reemplazado por los turbocódigos (figura 6). A la implementación habitual de los turbocódigos se añadió una mejora en la estimación de ruido del canal (parámetro básico para la decodificación), la adición a cada paquete de un código CRC de validación de información (por motivos de eficiencia, ya que permite terminar la decodificación una vez se comprueba que es un paquete válido) y un mecanismo de *puncturing* [14] con el que se consiguen tasas de transmisión variables entre 1 y 3 (en su implementación inicial, los turbocódigos sólo permiten un nivel 3 de redundancia).

Los turbocódigos trabajan con iteraciones SISO (*Soft-in, Soft-out*) y el algoritmo log-MAP, esto es, que el demodulador debería entregar valores *soft* (no valores 0 ó 1 digitales, sino un gradiente entre ellos) para alcanzar la máxima eficiencia. Por ello, el demodulador de *newqpsk* fue modificado para que entregara valores *soft* (de 8 bits) al decodificador turbo. Además, igual que se hizo con el módem FSK, se implementó un sistema de *interleaving* configurable, que bien puede ser aplicado a toda la ventana de información o por paquetes individuales. Esta última opción es la más conveniente para máquina de baja capacidad de cálculo, ya que la decodificación y la recepción se hacen de forma paralela y no hay que esperar a la recepción de la ventana entera.

El nivel necesario de redundancia de los turbocódigos depende de la calidad del canal en cada momento. En onda corta las características del canal cambian de forma muy rápida e imprevisible, así que se añadió en *newqpsk* un mecanismo de autoadaptación para que, en función de la tasa de error en recepción, el transmisor ajuste el nivel de redundancia.

7.2. Protocolo ARQ

Se conoce con el término ARQ (*Automatic Repeat Request*, petición automática de repetición) al conjunto de técnicas con las que un protocolo confirma la recepción de las tramas de información y es capaz de pedir la retransmisión de las tramas perdidas. Los tres sistemas ARQ [15] más usuales a nivel de enlace son descriptor a continuación:

- Parada y Espera (*Stop-and-Wait*): Se envía una única trama y se espera a la confirmación de recepción (ACK) del receptor. Un esquema altamente ineficiente.

- Retroceso a N (*Go-back-N*): El más utilizado, basado en la ventana deslizante en la que una estación manda una máximo de paquetes en función del tamaño de la ventana. Cuando el receptor recibe una trama fuera de secuencia lo descarta y envía un comando de rechazo de paquete (REJ), hasta que la trama se recibe correctamente. Todos las tramas que lleguen con número de secuencia mayor son descartados.

- Rechazo Selectivo: Si el receptor detecta la pérdida de uno o varios paquetes, pide la retransmisión únicamente de éstos. La implementación es más compleja, puesto que el receptor debe almacenar los paquetes recibidos fuera de secuencia hasta que los anteriores sean correctamente recibidos. Su uso se justifica únicamente en canales con elevadas tasa de error.

La implementación Linux de AX25 (sistema *Go-back-N*) no es suficientemente eficiente para canales HF, así que fue necesario el desarrollo de un mecanismo de rechazo selectivo. El estándar AX.25 definido por la TAPR incluye un comando SREJ (Selective Reject) para el rechazo individual de paquetes (comando que nunca fue incluido en el núcleo de Linux), pero esta solución no es del todo eficiente, ya que la ventana de respuesta a un comando SREJ es más pequeña que el tamaño máximo permitido, pues sólo incluye los paquetes perdidos.

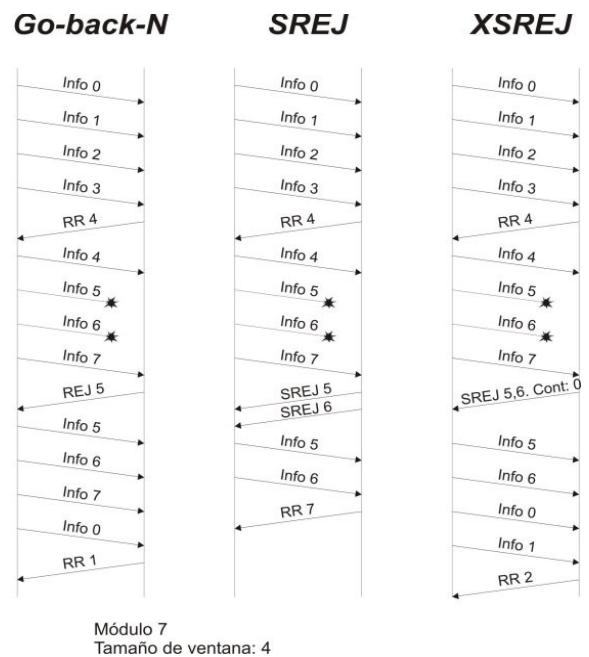


Figura 7: Mecanismos ARQ

Por esta razón, en vez de implementar el comando SREJ, hemos definido uno nuevo (XSREJ, eXtended Selective Reject) que informa al emisor de los paquetes perdidos y del punto en el que puede continuar la transmisión, lo que permite usar la ventana de transmisión en su totalidad (figura 7). En un sistema ARQ basado en el comando XSREJ, el máximo número de paquetes queda limitado a la mitad del número de la ventana. Como el modo estándar de AX.25 establece módulo 8 (lo que impone a un tamaño de ventana de 4, excesivamente restringido), se selecciona el modo extendido -módulo 128- que permite hasta 64 paquetes por ventana.

7.3. Protocolo Round-Robin

Las redes radio EHAS comúnmente tienen una tipología clientes/servidor en la que se comparte una única frecuencia para transmisión y recepción. Un acceso de este tipo es poco eficiente en canales ruidosos, en los que se producen un gran número de colisiones y tiempos muertos. Además, este medio de acceso presupone que los clientes se oyen entre sí, lo que no siempre es cierto (especialmente en onda corta).

La solución propuesta es la de añadir un mecanismo conocido como Round-Robin Polling al nivel de enlace (AX25), por el que el servidor asigna turnos a cada una de las estaciones cliente activas. Los clientes piden conexión y si son aceptados inmediatamente pasan a una cola de espera hasta que les llegue su tiempo asignado. Un sistema de estas características presupone que el tipo de información transmitida acepta esperas más o menos prolongadas. Esto no es un problema para el correo electrónico, pero puede hacer imposible el funcionamiento de sistemas de navegación internet y especialmente el de mensajería, que perdería toda interactividad.

La implementación de este protocolo mejora la velocidad media del sistema para la transmisión de correo electrónico, y permite un ahorro significativo en coste, tanto en el consumo de los transceptores como en la altura de las antenas.

7.4. Sistema distribuido de servidores

Por su largo alcance, en la banda de onda corta (HF) es habitual tener canales abiertos entre las diferentes subredes, de forma que un cliente tiene acceso a diversos servidores de la red. Originalmente, el sistema de correo estaba diseñado para que un cliente usara un único servidor configurado estáticamente (y así siguen siendo en VHF), pero en onda corta se ha desarrollado un aplicación (*gserversel*, selección genérica de servidor) que permite que un cliente establezca comunicación con cualquier servidor EHAS que tenga a su alcance, con un mayor aprovechamiento de los recursos disponibles.

En este sistema el cliente envía una petición genérica de conexión a los servidor de correo electrónico que es recibida por todos los servidores con los cuales tenga el canal abierto. Estos servidores (conectados entre ellos por Internet) intercambian información para decidir cuál es el más indicado para servir la petición (en función básicamente de dos parámetros, el número de clientes ya conectados a cada uno de ellos y la calidad del canal). El servidor radio elegido establecerá en este momento la conexión UUCP con el cliente, a la vez que, por un canal TCP/IP, intercambia el correo con el servidor central.

8. CONCLUSIÓN

El uso de software libre nos ha permitido desarrollar rápidamente y con pocos recursos un conjunto de soluciones de bajo coste para mejorar las condiciones de vida en zonas aisladas y desfavorecidas. El mayor logro del sistema es el de implementar con software libre los diferentes niveles de la transmisión hasta obtener soluciones altamente competitivas al nivel de módems comerciales, y a un coste mucho menor.

Se espera que el trabajo ya realizado y el futuro ayude a los sistemas públicos de salud de países en desarrollo, algunos de los cuales son plenamente conscientes de la importancia estratégica del software libre, a desplegar soluciones económicas, eficaces y controladas por ellos mismos.

[1] *Página principal del programa EHAS*,
<http://www.ahas.org>

[2] Andrés Martínez, *Evaluación de impacto del uso de tecnologías apropiadas de comunicación para el personal sanitario rural de países en desarrollo*, Universidad Politécnica de Madrid, 2003.

[3] *GNU General Public License*.
<http://www.gnu.org/copyleft/gpl.html>.

[4] *Página principal de Debian*,
<http://www.debian.org>

[5] Thomas Sailer, *Multiplatform soundcard packet radio modem*,
<http://www.baycom.org/~tom/ham/soundmodem/>.

[6] James Miller, *9600 baud packet radio modem design*,
<http://www.amsat.org/amsat/articles/g3ruh/109.txt>, 1994.

[7] William Beech, Douglas Nielsen y Jack Taylor, *AX.25 link access protocol for amateur packet radio v2.2*. Tucson Amateur Radio Corporation, 1997.

[8] Ian Lance Taylor. *Taylor Uucp*,
<http://www.airs.com/ian/uucp.html>.

[9] Roland Rosenfeld, *Batched SMTP mailer for sendmail and postfix*,
<http://www.spinnaker.de/debian/bsmtpd.html>.

[10] Wietse Venema, *The Postfix Home Page*,
<http://www.postfix.org>.

[11] *The Squid Page*, <http://www.postfix.org>

[12] Paul S. Cannon, Matthew J. Angling y Bengt Lundborg, "Characterization and Modeling of the HF Communications Channel", *Electromagnetic Metrology*, The International Union of Radio Science, 2002.

[13] C.Berrou, A.Glavieux y P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo Codes", *Proc. IEEE Int. Conf. on Commun. Ginebra, Suiza*, 1993.

[14] Fan Mo, S.C. Kwatra y Junghwan Kim: "Analysis of Puncturing Pattern for High Rate Turbo Codes", *Proceedings of IEEE Military Communications Conference*, 1999.

[15] V. P. Pribylov y G. A. Chernetsky, "Throughput efficiency of Automatic Repeat Request Algorithm with Selective Reject in Communication Links with Great Signal Propagation Delay", *Proceedings of the IEEE-Russia Conference (MEMIA)*, 2001.

[16] S.C. Cook, J.B. Scholz y T.C. Giles, *Dynamically Adaptive Digital Communication Systems for Improving Throughput and Error Rates*, Proc. of IEAust Communications, 1990.

[17] T.C.Giles, *On the Design of HF Modems*, University of Adelaide, Australia, 1995.

[18] Timothy J. Riley, *Third Generation HF Modem Testing*. Institute for Telecommunication Science, 2000.

[19] Ramy Card, Eric Dumas y Frank Menel, *Programación Linux 2.0: API de sistema y funcionamiento del núcleo*, Gestión 2000, 1997.