Low-cost data transmission over narrow-band radio channels

Arnau Sánchez Sala
Joaquín Seoane Pasqual
Francisco Javier Simó Reigadas
Andres Martínez Fernández
Fundación EHAS
E.T.S.I Telecomunicación. C.203-1
Ciudad Universitaria, s/n
28040 Madrid - Spain

Abstract- This paper presents a digital transmission system for narrow-band radio channels on the VHF/UHF and HF bands, providing efficient electronic mail transmission, web-browsing and chat communication. The system, focused in low-cost technologies and based completely on free software, has been developed in the EHAS (Hispano-American Health Link) project, which installs low-cost telecommunication systems and information services specially designed for rural primary health-care personnel that work in isolated areas in Latin American countries.

I. INTRODUCTION

More than half of the world population live in isolated rural area where public telephone network and even electricity are not available. Low density population makes difficult to afford the installation of telecommunication systems due to restrictions in power service, accessibility, maintainability and security. The EHAS [1] initiative is a viable proposal to face this problematic situation through telemedicine systems and services in rural areas of developing countries.

Primary care institutions in Latin America can be grouped into two categories: *Health Centers* (HC) and *Health Posts* (HP). A HP is a point of access to the health care system for a rural population. HP are typically located in towns of no more than 1000 inhabitants that have no telephone line and poor transport. A HC is usually located in a provincial or district capital and has telephone lines installed.

Several HP depend on a single HC, which together comprise a health *micro-network*, a basic primary care unit. These posts need better ways of communicating for consultation, to conveying epidemiological surveillance reports, ordering medical supplies and relaying information concerning acute epidemic outbreaks, medical emergencies or natural disasters. Communication and the exchange of information require health-care workers to travel from one facility to another, which can take hours or even days.

II. VOICE AND DATA NETWORKS

Our radio networks aim to break isolation and connect, for each micro-network, a Health Posts to its referential Health Center and all others Health Posts in the network.

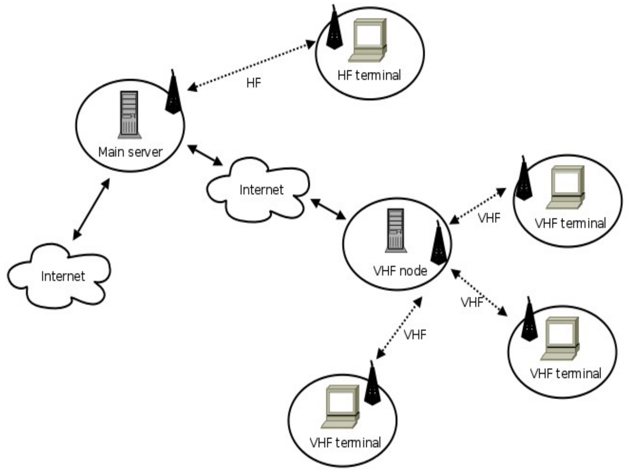

Fig. 1: Typical EHAS radio networks

The *VHF/UHF band* (30-3000 MHz) is usually the best alternative for short and half-distance voice-coverage without direct visibility (where *WiFi* networks do not apply). It is possible to interconnect stations in distances up to 70km (depending on the orography) with a good voice quality without depending on the environmental conditions.

The *HF band* (3-30MHz), on the other hand, allows long and very long distance coverage (depending on the frequency, thousands of kilometers) due to a phenomenon known as *ionospheric propagation*, which lies in the reflection of the radio signals in the higher layers of the atmosphere [2]. The main disadvantage is the low quality transmission, since transmitted signals are commonly exposed to distortion (atmospheric absorption, high noise, multi-path). Moreover, channel conditions are extremely variable, depending on the moment of the day, season, sun spots activity, ionospheric storms, among others factors.

Evaluations [1] done in isolated rural areas demonstrate that the principal service that saves human lives is the simple voice communication in the local area. Once satisfied this fundamental need, digital communication appears to be a valuable complement. Fig 1 shows a typical HF/VHF data EHAS network.

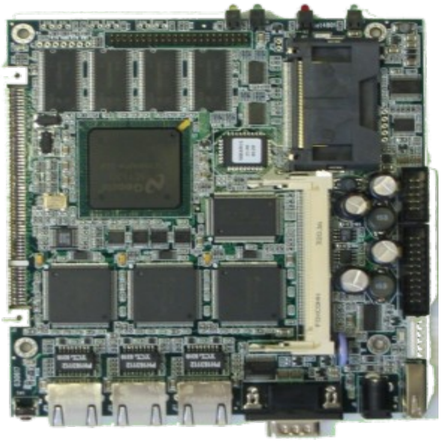

Fig. 2: Soekris Net4801 top view

In order to connect any physical medium to a digital equipment like a computer, we need a device known generically as *modem*. In radio terminology, these modems are known as *TNC* (Terminal Node Controllers). Prices and features of a TNC vary a lot depending on the manufacturer; commercial modems with high capabilities (high-speed, powerful error-correction schemes) are usually, specially HF modems, very expensive.

Our aim is to achieve comparable features of commercial modems with lower cost, avoiding proprietary protocols and expensive specialized hardware. The solution is using *software modems*, where the CPU does most of the task; on software modems code developing is faster and more versatile, easily maintained and modified. Moreover, as we use exclusively free software [3], we can run, study and modify freely previously existing applications, thus developing time is reduced. In fact, almost every layer involved in the communication system this paper presents has been tweaked to increase speed and robustness.

III. HARDWARE

1. *Process Unit*

We need a hardware capable of running a free modern operating systems (e.g. Linux, FreeBSD), powerful enough to modulate and do error-correction tasks. We choose a *Soekris Net4801 [4],* a compact (132x144mm), low-power (<5W), low-cost x86-compatible computer, based on a 266 Mhz Geode processor. This board is usually employed as WiFi router and/or Internet gateway but it fits for a radio-modem.

Although we are using a software modem, we still need some hardware to interface the audio. The easiest and cheapest hardware to do the A/D (analog/digital) and D/A (digital/analog) conversions is using a common sound card. The Soekris Net4801 has no integrated sound card, but a external one can be connected to the available USB 1.1 interface. Additional interface tasks can be accomplished with the 12-bit General Purpose I/O (GPIO) which provides 8 digital I/O lines and a TTL RS232 serial port. Power supply range is also a very attractive, as it works in a wide range of input voltage (6-20V DC) that matches perfectly in a common solar installation.

2. *Transceivers*

When selecting the VHF/UHF transceiver, it's important to check that input/output audio filters can be completely disabled (this option is usually known as *flat filters*). Other way, transmitted signal is cut on higher frequencies and data transmission is impractical. VHF/UHG transceivers should have at least a base-band bandwidth of 8Khz.

On HF transceivers the most important issue is the frequency shift that modulators introduce (cheap transceivers usually have low-accurate oscillators). Minimum frequency stability should be within ±*10 ppm.*

On both cases, check that radio/computer interconnection is feasible. Most radios (but not all of them) provide a rear connector to interface an external digital equipment. It has to support, at least, *audio Tx/Rx* and *PTT* lines (Push-to-Talk: used to set transceiver on transmission mode). VHF/UHF radios usually add some control digital lines (e.g. channel selection, ignition) while HF transceiver have a unique CAT (Computer-Aided-Transceiver) port to receive commands through a RS-232 serial port. Avoid transceivers that don't permit external control.

3. *Radio-interface board*

In order to control a transceiver (digital lines in VHF/UHF, CAT in HF transceivers) from a Soekris we designed a simple board connected to the GPIO, using some of the 8 I/O digital pins (3.3V). These lines will be used for several tasks: PTT activation, channel selection, radio state detection or ignition.

HF control is done (except for the PTT) with the previously mentioned CAT port on the transceiver and the TTL-level serial port on the GPIO. Be aware that there is no CAT standard defined among manufacturers, so software drivers must support each one individually.

IV. SOFTWARE IMPLEMENTATION

1. *Operating System*

GNU/Linux is a free Operating System for general usage. It runs the popular Linux kernel together with the GNU system tools. Although GNU/Linux represents a unique system core, there are many distributions --commercial or not-- being actively developed. Among them (Red Hat, Slackware, Mandrake, OpenSuse), we choose Debian for its high-quality standards and free software [4] commitment.

2. *Physical layer*

We use standard voice radios with no embedded digital transmission facilities, so we must take care of modulation/demodulation tasks. *Soundmodem* [5] is a free package designed for that task. It provides five different modems, each one with different modulations, speed and bandwidth usage.

For *VHF/UHF* networks we choose the FSK-G3RUH 9600bps (bits per second) modem, which employs a modified version of a FSK modulation that reduces significantly the bandwidth usage. The original modem does not implemented any Forward Error Correction (FEC) mechanism, so we added a convolutional codification (speed configurable between 3200 and 9600bps) in transmission together with a Viterbi decoder in reception.

For *HF*, soundmodem provides the NewQPSK 2500bps, a slower but more robust modem specially designed for short-wave channels. NewQPSK is a parallel-tone modem that implements a OFDM (Orthogonal Frequency Division Multiplexing) modulation. OFDM spread spectrum technique uses a large number of frequency spaced carriers to distribute data bits over the channel. Orthogonality is achieved by using a specific frequency spacing that prevents the demodulator from mixing frequencies.

The benefits of OFDM are high spectral efficiency, hardiness to RF interference, and lower multi-path distortion [6]. Moreover, both transmitter and receiver can be easily implemented using efficient IFFT/FFT techniques. NewQPSK has 15 carriers spaced 125 Hz, using each one a differential quadrature PSK (DQPSK) to get a baud rate of 83.3 for a total 2500 bps raw-speed.

NewQPSK originally implements Forward Error Correction (FEC) using a simple BCH block codes, and has time and frequency diversity (using *interleaving*) to deallocate burst errors. Nowadays, most of commercial modems are still using convolutional codes with Viterbi decoding at receiver, followed by a Reed-Solomon decoder for final burst correction. Nevertheless, *Turbocodes* [7] are a more powerful error correction technique which outperforms all previous known coding schemes and being used in communication systems where a significant power saving is required or when the operating SNR is very low. They can only be used when buffering large data-blocks is feasible (not suitable for Voice over IP, for example), since Turbocoding process involves packet interleaving.

As we said, the original NewQPSK implementation employed a simple BCH block-code as FEC mechanism which we replaced by Turbocodes routines with a self-adjusting redundancy protocol [8], a channel noise estimation and CRC-checking. To take full advantage of Turbodecoding, DQPSK demodulator was modified to deliver soft value (original NewQPSK used hard values 0/1, as BCH block codes don't take advantage of soft values).

Multi-path propagation usually present on HF sky wave communications leads to frequency-selective fades; despite using a powerful error correction scheme as Turbocodes, many packets missed due to this repetitive loss of one or more of the signaling tones. Then, we implemented a interleaving order to distribute burst errors over the interleaving window. In HF we have implemented two different interleaving schemes: *packet* and *global* interleaving.

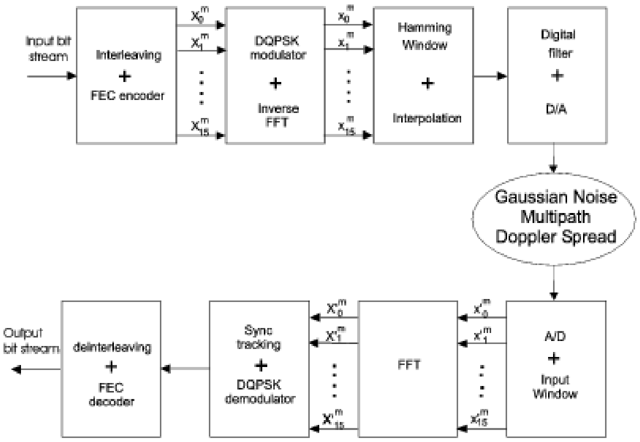

Illustration 3: NewQPSK block diagram

Global interleaving covers the whole transmission window (typically, about 15 seconds), giving best protection on good channels, however not suitable for channels where long fades propagate errors through the entire window transmission. Packet Interleaving processes each Turbocode packet separately and long fades and interferences only affect a portion of the window, showing better performance that global interleaving on typical HF channels.

3. *Link layer*

AX.25 [9] is a data link layer protocol derived from X.25 widely used on amateur packet radio networks.). Being responsible for transferring data between nodes and detecting errors, AX.25 is comparable to the Ethernet layer.

AX.25 is widely used on VHF/UHF data links and first tests showed good efficiency in our VHF networks. On the other hand we got very poor speed rates in HF due to packet retransmissions. It appeared that the basic ARQ (Automatic Request Query) scheme that AX.25 implements should be improved.

There are three common ARQ [10] schemes used at link level:

- *Stop-and-wait*: It is used with stop and wait flow control, where just one frame is transmitted at a time and every one must be acknowledged (ACK). A highly inefficient ARQ scheme.
- *Go-back-N*: This is the most common, based on a sliding window. A station may send a series of frames determined by window size. When receiving an out-of-sequence frame, the receiver discards that frame and all subsequent ones with a Reject command (REJ), until the damaged frame is received correctly.
- *Selective-reject*: The sender only retransmits frames that are negatively acknowledged or whose acknowledgment times out. The receiver must buffer incoming frames until the frame in error is retransmitted and accepted. This is less used than *go-back-N* because of its complexity and memory requirements, but minimizes retransmissions.

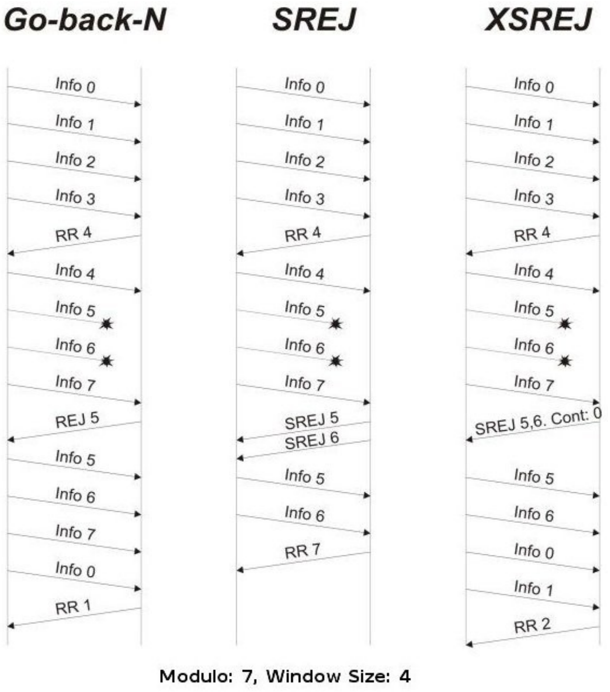

Fig. 4: ARQ schemes

Linux kernel's AX.25 implementations use the simple *go-back-N* scheme, so we needed to design and implemented a more efficient selective-reject mechanism. TAPR AX.25 Standard defines a *SREJ command* that requests an individual frame for retransmission. However, when one peer retransmits some frames which were lost, this transmission window contains only a few frames, and window is shorter than usual. As every transmission involves a lot of overhead (Tx/Rx commutation, radio delays, protocol headers), it is very inefficient to use retailed transmission windows.

To overcome these issues, we have defined a completely new AX.25 control packet (*XSREJ*, eXtended Reject) that informs the sender not only about one individual packet to retransmit, but of the complete list of them, thus the sender is able to assemble a full window on next transmission.

In a XSREJ-ARQ scheme, configuration parameters must avoid a possible overlapping between Tx and Rx window. To overcome it, the maximum window size should be no more than half of range of sequence numbers (maximum window size is 2^{N-1} for go-back-N and $2^{N/2}$ for XSREJ). Standard AX.25 is modulo-8, which permits a maximum window size of only 4. We enable extended AX.25 mode (modulo-128) so a window size up to 64 is accepted.

Medium access control was also modified. The CSMA-CA medium access control used in AX.25 (specially when working in noisy half-duplex links) experiences unavoidable collisions when more than one user is communicating with the server. As every station must hear the others in order to avoid collisions, more power than just the necessary to communicate with the server is transmitted. The solution has been adding a round-robin polling mechanism to AX.25. These scheme is not suitable for real-time data transmission (web-browsing, chat), as it introduces long wait delays, so it should only be enabled on pure e-mail networks. The server, which has the task of assigning sequential turns to every active clients. That way, clients can use directional antennas and save energy for power transmission (critical in isolated areas with solar-powered installations).

4. *Transport layer*

Arguably, TCP/IP is the standard transport protocol in Internet and almost any modern application use it. Unfortunately the TCP (Transport Control Protocol is not suitable for radio links: TCP, which provides end to end reliable delivery using a sliding window ARQ protocol with adaptive timeouts and window size, is designed for congestion control on reliable full-duplex links. As a result, the TCP layer experiences back offs and efficiency is not good on most channels.

V. SERVICES

1. *Electronic mail*

Electronic mail transmission is usually the highest priority in radio data networks, so an efficient alternative to use pure TCP/IP services as POP3, SMTP or IMAP over AX.25 was compelling. The *UUCP* (Unix-to-Unix Copy) [11] protocol, although regarded as obsolete, has been widely used to send files between UNIX computers over slow modem connections, and can be easily configured to transport e-mail messages.

Since AX.25 assures error correction and flow control, we need a transport protocol with no control over transmission (that means mainly no packet acknowledgments). Among all the UUCP protocol family, the *y-protocol* complies our needs: it is intended to be used in half-duplex communication lines which have its own error correction and flow control and adds very low overhead (less than 0.6%). It performs only error detection, but not error correction: when an error is detected, the line is simply dropped. Its also interesting the ability of resuming broken transmissions, a compulsory need to use in narrow-band transmissions.

UUCP does not include any compression mechanism, so we installed an additional layer, *BSMTP* (Batched Simple Mail Transport Protocol) [12], used in conjunction with UUCP to group and compressed e-mails.

Data exchange is made with configurable periodic connections between the Post Center and its micro-network server usually, but necessarily, situated on a Health Center. These servers connect to the main server in each country using Internet (or a HF link if no Internet connection is possible). If multiple connections are available, a priority connection order can be configured to route e-mails over the cheapest way (e.g. sending data through a HF connection before trying a costly telephonic modem call).

2. *TCP/IP through webproxy*

Any browser and modern IM (Instant Messenger) can be configured to use a web-proxy as Internet gateway. Our radio-modem provides a point-to-point TCP-AX.25 proxy that redirects local TCP ports to a remote machine over AX.25. That way, TCP/IP is overridden and radio transmission relies only on AX.25 protocol, increasing dramatically speed transference if we compare it to direct TCP/IP transmission. At server side, we use the popular *squid* [13] package as webproxy (configured to eliminate or reduce quality of images that pass the proxy)

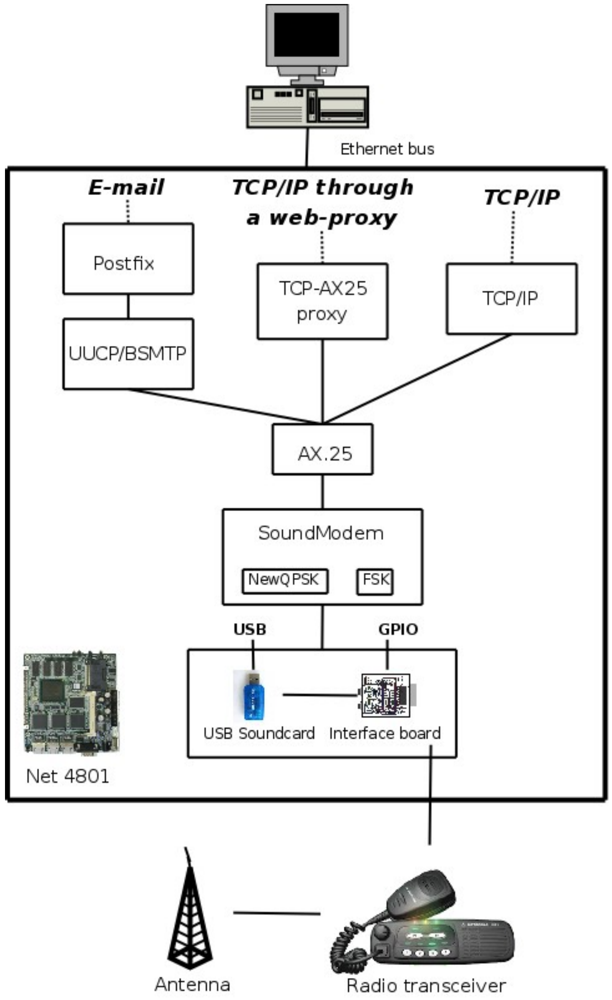

Fig. 5: Complete radio transmission system

3. *Direct TCP/IP access*

Despite the efficient problems already described, TCP/IP is still useful. TCP/IP over AX25 is fully implemented in the Linux kernel and we have enabled it in our radio-modem. Unexperienced users which don't want to manage a UUCP email system can use TCP/IP to access POP3 mailboxes or SMTP servers or connect Internet through a web-browser. In our our networks, direct TCP/IP it is only used for management tasks (e.g. ping tests, NTP synchronization, SSH remote access).

VI. CONCLUSION

Free software has allowed us to develop a complete and freely available [14] efficient communication system to improve life conditions in isolated and poor areas, using robust, low-cost, low-power consumption equipment. We hope that these efforts help the public health-care systems in third-world countries to deploy free, efficient, low-cost solutions to fit their communications requirements.

REFERENCES

[1] Andrés Martinez: *EHAS program: Rural telemedicine systems for primary health-care in developing countries*. In Proceedings of the International Symposium on Technology and Society, 2002.

[2] Paul S. Cannon, Matthew J. Angling, and Bengt Lundborg. *Characterization and modeling of the HF communications channel*. Electromagnetic Metrology, pages 597–624, 2002.

[3] Free Software Foundation. *The Free software definition*. URL: http://www.gnu.org/philosophy/free-sw.html

[4] *Soekris Net4801 specs*. URL: http://www.soekris.com/net4801.htm

[5] Thomas Sailer. *Soundmodem on modern operating systems*. In ARRL and TAPR Digital Communications Conference, 2000

[6] T. C. Giles. *On the Design of HF Modems*. PhD thesis, PhD in 2003 in the Technical University of University of Adelaide, Australia, 1995.

[7] C. Berrou, A. Glavieux, and P. Thitimasjshima. *Near Shannon limit error-correcting coding and decoding: Turbocodes*. In Proc. IEEE Int. Conf. on Commun. (Geneva,Switzerland), pages 1064–1070, May 1993.

[8] S. C. Cook, J. B. Scholz, and T. C. Giles. *Dynamically adaptive digital communication systems for improving throughput and error rates*. In Proc. of IEAust Communications, pages 192–196, 1990.

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

[10] V. P. Pribylov and G. A. Chernetsky. *Throughput efficiency of automatic repeat request algorithm with selective reject in communication links with great signal propagation delay*. In Proceedings of the IEEE-Russia Conference (MEMIA), 2001.

[11] Mark. R. Horton. *UUCP mail interchange format standard* . (RFC 976). http://www.ietf.org/rfc/rfc976.txt.

[12] Roland Rosenfeld, *Batched SMTP mailer for Sendmail and Postfix.* URL: http://www.spinnaker.de/debian/bsmtpd.html.

[13] *The Squid Page.* http://www.squid-cache.org

[14] EHAS Debian repository. http://debian.ehas.org/debian