REPORT:

Authors:
Nevio Benvenuto, Guido Bertocci, and **William R. Daumer** are members of the Digital Techniques Department at AT&T Bell Laboratories in Holmdel, New Jersey. They are involved in exploring various digital signal processing algorithms. **Duncan K. Sparrell** is a member of the Advanced Digital Signal Processing department at AT&T Bell Laboratories in Holmdel. He is responsible for system planning and testing. **Mr. Benvenuto** received the laurea in electrical engineering from the University of Padova, Italy, and the Ph.D. in electrical engineering from the University of Massachusetts. He joined AT&T in 1983. He alternates his duties at Bell Laboratories with those at the University of Padova, where he works on communication systems research. **Mr. Bertocci** received the B.S. in electrical engi-
(continued on page 22)

THE 32-kb/s ADPCM CODING STANDARD

Introduction

AT&T has played a major role in the development of national and international standards for 32-kb/s adaptive differential pulse code modulation (ADPCM). This paper highlights the process leading to the standards on 32-kb/s ADPCM recognized by the American National Standards Institute (ANSI) and the International Telephone and Telegraph Consultative Committee (CCITT). The paper briefly describes the algorithm itself and its performance. It concludes with a discussion of applications of ADPCM and the AT&T products and services that utilize ADPCM.

International Standards Activity in the CCITT

At the June 1982 meeting of the International Telephone and Telegraph Consultative Committee Study Group XVIII/WP2 in Geneva, Switzerland, a need for an international 32-kb/s coding standard was formally identified. An expert group was formulated and chartered[1] to recommend and fully specify a 32-kb/s algorithm of reasonable complexity that could offer a 2:1 reduction in bit rate from 64-kb/s pulse code modulation (PCM), yet maintain as much as possible the transmission performance of PCM. During that meeting and subsequent meetings of the expert group, agreement was reached on several key constraints that should be observed in designing such an algorithm:

- 32-kb/s bit-sequence-independent transmission channels would be available for transmission.
- The algorithm would be defined in terms of digital transcoding to and from 64-kb/s PCM, in either the $A = 87.6$ or $\mu = 255$ law format, with the coding and and transmission performance specified in CCITT Recommendations G.711 and G.712 respectively (thus, sampling at 8 kHz and encoding at 4 bits per sample were implicit requirements).
- For simplicity, the algorithm should not rely on side information, e.g., for parameter transmission or maintenance of frame alignment.
- For minimal transmission delay, all adaptation processes should be backward-acting in time.
- Particular attention should be paid to encoder/decoder mistracking recovery in the presence of digital transmission impairments, e.g., bit errors and slips.
- Adequate transmission performance should be maintained for nonvoice signals as well as voice. Nonvoice signals include voiceband data ($\leq$ 4800 b/s), signaling, analog facsimile, measurement signals, etc. [It was recognized at the June 1982 Study Group XVIII meeting that voiceband data performance at 9600 b/s would not be acceptable with the 32-kb/s ADPCM candidate algorithms under discussion. It is the authors' opinion that, to date, insufficient data have been presented to demonstrate adequate 9600-b/s voiceband data performance for general network use with 32-kb/s ADPCM in the presence of asynchronous tandem codings (i.e., at least three, preferably four) and added analog transmission impairments (e.g., random noise, nonlinear distortion, phase jitter, envelope delay distortion). This issue is currently being studied in CCITT Study Group XVIII in the context of digital circuit multiplication equipment.]
- Adequate transmission performance should be maintained in a network environment

which includes up to four asynchronous (interconnection at 4-kHz analog) and synchronous (interconnection with 64-kb/s PCM) tandem codings with added analog impairments (e.g., loss, random noise) and digital impairments (e.g., bit errors). [The current algorithm does not always meet these constraints with an end-to-end bit error rate (BER) criterion of 10^{-5} where, with some modems, only two or three asynchronous tandem codings are possible at a 4800-b/s rate.]

- The algorithm should include a technique to reduce cumulative distortion in multiple synchronous tandem codings.

At the June 1982 Geneva meeting, several contributions describing ADPCM algorithms were submitted by various organizations (including AT&T). It was generally felt that an ADPCM-based algorithm could satisfy the above constraints at 32 kb/s. However, it was recognized that the design of such an algorithm was much more challenging than the traditional approach of optimizing an algorithm for voice alone. In the ensuing 18 months, the expert group selected and fully defined an algorithm including an assessment of its performance.[2] This algorithm was formally approved in the October 1984 CCITT plenary session as an international standard, the detailed specification is contained in Recommendation G.721.[3] G.721 is generically defined at the bit level to ensure full compatibility and transmission performance when independently manufactured encoders and decoders are interconnected. The specification also includes a set of digital test sequences for compliance verification.

Domestic Standards Activities

In the year following the adoption of the ADPCM standard in CCITT, activities in ANSI Committee T1 (specifically subcommittee T1Y1.2) addressed the formulation of a North American standard that would not only include a 32-kb/s ADPCM coding algorithm but also a DS1 frame format specification. However, after intense discussion it was decided that the original constraint of bit-sequence-independent transmission channels was not appropriate for North American telecommunication networks. Due to 1's density constraints on T1 carrier, the original DS1 line coding [bipolar with zero-code suppression (BZCS)] did not allow long strings of 0's. A new DS1 line coding [bipolar with eight zero substitution (B8ZS)] was developed, but a large embedded base of digital banks, terminals, and multiplexers still exists without this capability.

Various proposals aimed at overcoming lack of bit sequence independence were considered. Finally, T1Y1.2 collectively decided that a slightly modified version of G.721 should be adopted to overcome this limitation. T1Y1.2 also agreed that G.721 was the recommended choice for bit-sequence-independent transmission channels. This T1Y1 algorithm[4] was at variance with G.721 in a very basic way. G.721 incorporates a 16-level midriser quantizer whereas the T1Y1 algorithm specified 15-level midtread quantization whereby the two innermost output levels are aligned at zero so that the 0000_2 and 1111_2 codes both represent a zero output level. (Other quantizer parameters such as decision, output, and step-size multiplier values were also changed in order to optimize performance.) This allows the suppression of the 0000_2 code word to satisfy 1's density constraints over adjacent 32-kb/s channels in a multiplexed format.

From a complexity viewpoint, this variation is simple but it rendered the T1Y1 algorithm incompatible with G.721. That is, a

16-level G.721 encoder could not be connected to a 15-level T1Y1 decoder without incurring severe distortion.

As the T1Y1 algorithm was undergoing the ANSI Committee T1 approval process, two problems with ADPCM equipment surfaced in the field. Unacceptable error rates were being observed with frequency-shift keying (FSK) modems during asynchronous, character-mode operation which is typical of users typing at a keyboard. In addition, long sequences of idle code into the decoder produced undesirable high-level output signals. These problems were inherent to both the G.721 and T1Y1 algorithms, since their root cause was traced to the operation of the adaptive predictor, which is a common element of both algorithms, and is unrelated to any aforementioned quantizer differences.

Subcommittee T1Y1 recognized these problems as requiring prompt resolution. Both problems were quickly analyzed and several algorithmic modifications were proposed.

In May 1986, T1Y1.2 formally agreed to modifications[5] that resolved these problems without significantly compromising performance for other signals. An added benefit of these changes was the ability to use a homing sequence that would drive the algorithm into a known state for testing purposes. The modified T1 ADPCM coding and frame format standard[6] was then set back in motion through the formal ANSI Committee T1 approval process.

CCITT Revisited

Since the modification to the T1Y1 algorithm addressed problems that were also inherent in the CCITT G.721 algorithm, the U.S. delegation reported to Study Group XVIII at Geneva in March 1986 on the cause of these problems and on progress in T1Y1 aimed at resolving them. Study Group XVIII also recognized the urgency of these problems and solicited contributions from its members. It chartered the U.S. delegation to report back in July with a proposal to modify G.721, including changes to the algorithm reflected in the bit level specification, new digital test sequences, and results verifying performance.[7]

In July the U.S. delegation fulfilled its mandate and presented Study Group XVIII with a proposal[8,9] that recommended modifying G.721 to solve all known problems and incorporating a 15-level quantizer, thereby eliminating all differences between the two algorithms and restoring the concept of a single, worldwide ADPCM coding standard. This proposal was formally accepted by Study Group XVIII and put forward for formal CCITT approval under accelerated procedures.

Algorithm Architecture

Figure 1 shows a simplified block diagram of an ADPCM encoder. Two major components form the algorithm: an adaptive quantizer and an adaptive predictor. The relationship between an encoder and a decoder is also depicted. The decoder is simply a subset of the encoder and transmits $r(n)$ as its output instead of $c(n)$.

The adaptive predictor computes an input signal estimate $\hat{x}(n)$ which is subtracted from the input signal $x(n)$ resulting in a difference signal $d(n)$. The adaptive quantizer codes $d(n)$ into a 4-bit codeword $c(n)$ which is sent over the transmission facility. At the receiving end an ADPCM decoder uses $c(n)$ to attempt to reconstruct the original signal $x(n)$. Actually, only $r(n)$ can be reconstructed; $r(n)$ is related to the original input signal $x(n)$ by

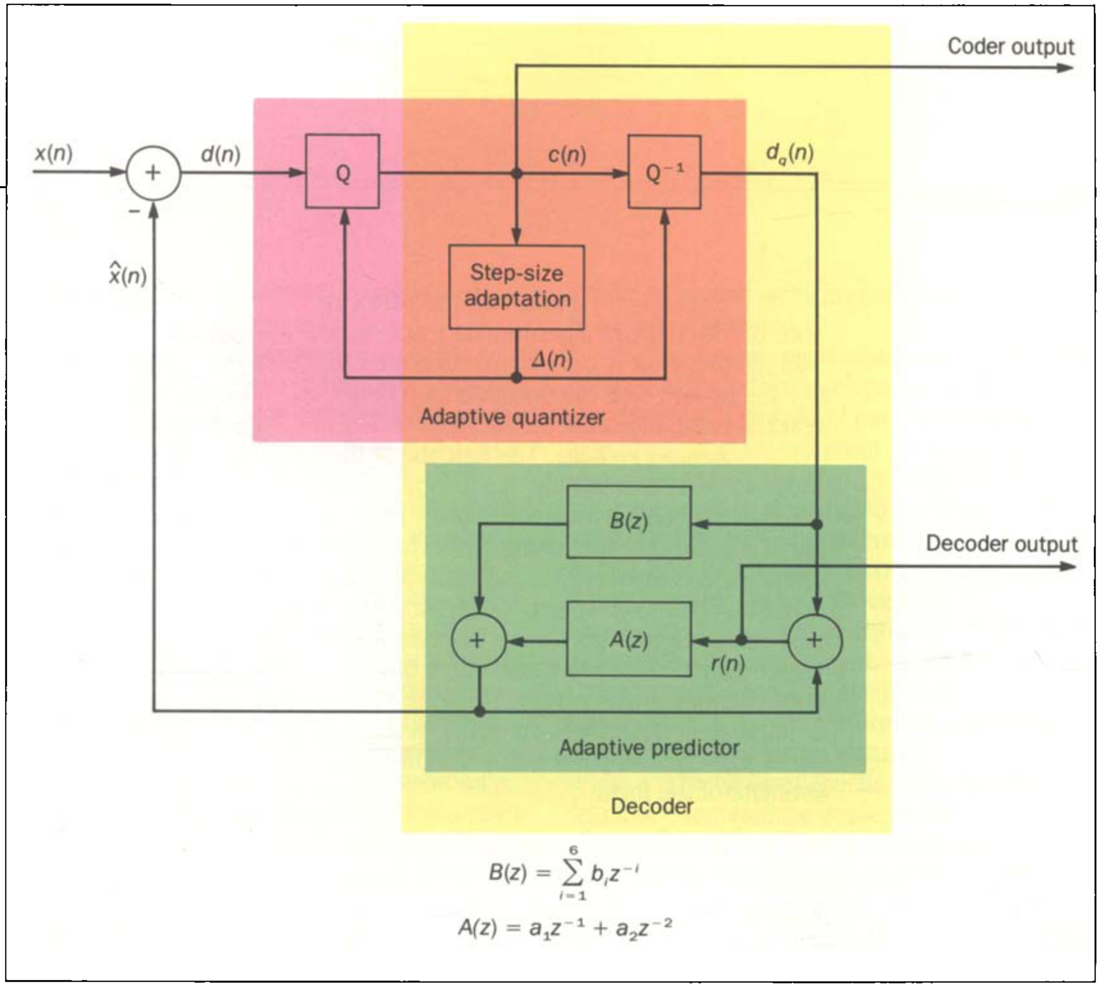


Figure 1. Block diagram of ADPCM encoder.

$$r(n) = x(n) + e(n) \quad (1)$$

where

$$e(n) = d_q(n) - d(n) \quad (2)$$

is the error introduced by the quantizer. A typical measure of performance is given by the signal-to-noise ratio

$$S/N = \frac{E[x^2(n)]}{E[e^2(n)]} = \frac{\sigma_x^2}{\sigma_e^2} \quad (3)$$

In equation (3) E denotes expectation. Introducing the concept of residual error power, $\sigma_d^2 = E[d^2(n)]$, the signal-to-quantizing-noise ratio is defined to be

$$S/N_Q = \frac{\sigma_d^2}{\sigma_e^2} \quad (4)$$

Then equation (3) can be rewritten as

$$S/N = G_p \cdot S/N_Q \quad (5)$$

where G_p, the prediction gain, is

$$G_p = \frac{\sigma_x^2}{\sigma_d^2} \quad (6)$$

Equation (5) states that the performance of the ADPCM algorithm depends on how well the predictor is performing, indicated by G_p, and on how well the quantizer is coding the residual error, indicated by the value of S/N_Q.

Adaptive Predictor. The function of the predictor is to estimate the input signal $x(n)$ and thus reduce the variance of $d(n)$, increasing G_p and thereby increasing S/N. The predictor is composed of an adaptive structure consisting of two poles and six zeros. A two-pole configuration was chosen because it allows easier control of decoder stability in the presence of transmission errors. However, in order to improve performance, six zeros were combined with the two poles giving a signal estimate of the form

$$\hat{x}(n) = \sum_{i=1}^{2} a_i(n) r(n-i) + \sum_{i=1}^{6} b_i(n) d_q(n-i) \tag{7}$$

The eight coefficients are adapted by using a simplified version of the gradient algorithm.[8]

Adaptive Quantizer. The quantizer is the system that encodes the residual error $d(n)$ into a digital signal $d_q(n)$ whose coded version $c(n)$ is transmitted. In practice, the quantizer consists of two parts, Q and Q^{-1}. Q produces $c(n)$ from the residual error $d(n)$ and the scaling factor $\Delta(n)$. Q^{-1} decodes $c(n)$ into a quantized residual error $d_q(n)$ by scaling $c(n)$ by $\Delta(n)$.

A Gaussian characteristic is used for the quantizer decision and output levels.[10] For the 4-bit quantizer used here it turns out that $S/N_Q \approx 20$ dB for optimal level Gaussian input signals.

Since the power of $d(n)$ may vary depending upon the input signal $x(n)$, the quantizer characteristic is scaled by a quantity $\Delta(n)$. Two additional scaling factors have been introduced, denoted Δ_u and Δ_l. $\Delta_u(n)$ is the estimate of the instantaneous level of $d(n)$ while $\Delta_l(n)$ is related to the long average of $\Delta_u(n)$ and reflects the value of σ_d. An adaptive mechanism is used to estimate Δ_u.[11] The step size Δ_u at time n is updated as follows:

$$\Delta_u(n) = \Delta_u^{\beta}(n-1) M(|c(n)|) \tag{8}$$

where values for M are given in Reference 6.

In equation (8) the leakage constant β has been introduced into the original algorithm to make the quantizer more robust to transmission errors.[12] It can be shown that for stationary signals such as voiceband data, improved performance is achieved when $\Delta(n)$ is set to a fixed value. One of the important characteristics of this algorithm is the ability to accommodate these signals as well as signals that are more nonstationary in nature, e.g., speech. By defining $y(n) = \log_2[\Delta(n)]$, $y_u(n) = \log_2[\Delta_u(n)]$, and $y_l(n) = \log_2[\Delta_l(n)]$, signals with widely different statistics can be effectively encoded by including a mechanism which expresses y as a linear combination of y_u and y_l; i.e.,

$$y(n+1) = a_l(n+1) y_u(n) + [1 - a_l(n+1)] y_l(n) \tag{9}$$

where $0 \leq a_l(n+1) \leq 1$. When a_l is 0 the quantizer adapts slowly and is considered to be in a locked state which is advantageous for stationary signals. Conversely, when a_l is 1 the quantizer adapts quickly and is considered to be in an unlocked state which is desirable in the case of rapidly varying signals.

Predictor Reset Mechanism. Because of the feedback architecture of the coder, when signals that result in very high predictor gains change, potential problems arise at transitions between different stationary (e.g., partial band) signals. This occurs mainly in the presence of tones (which comprise FSK modem signals) or similar signals when $G_p \approx 17$ dB and the quantizer is in the locked mode. Two mechanisms were added to the algorithm to prevent this problem.[6] One mechanism detects the presence of tones or similar signals (by means of a_2), and the other detects the transition from one partial band signal to another (by means of a_2 and d_q). When such a transition is detected all the predictor coefficients are reset to zero and the quantizer is forced into the unlocked mode.

Synchronous Tandeming. Another feature of this ADPCM algorithm is its operation to reduce cumulative distortion arising from successive synchronous tandem codings (ADPCM to PCM to ADPCM etc.). The ADPCM decoder modifies the PCM output codeword to ensure that $c(n)$ values generated by successive coders will match the values of the first encoder.

In this way, under ideal conditions, any number of synchronous tandem codings can be equivalent to a single coding. This is only the case when bit integrity is maintained and there are no bit errors in the intermediate 64-kb/s PCM streams. Bit integrity is lost if digital signal processing (e.g., digital loss) is used in the 64-kb/s PCM stream. However, performance with synchronous tandeming is better than or equal to that obtained in the asynchronous case. Differences in performance between the synchronous and asynchronous cases have yet to be fully quantified.

Implementation Considerations. It was recognized early in the development of the standard that a high-level equation description alone would not be sufficient for implementation. As a result, the standard algorithm has been specified in great detail to show exactly what must be done to implement the algorithm. In fact, this bit level description specifies everything from the bit precision of each parameter to the form of arithmetic (signed magnitude, 2's complement, or floating point) for each operation. As a result, if the bit level specification has been adhered to, encoder/decoder compatibility is ensured.

The bit level specification has been developed to minimize implementation complexity. For example, bit precision for each variable has been reduced as much as possible. Multiplications are avoided whenever possible and there are no divisions. Whenever multiplications or divisions would normally be required, logarithmic techniques are used. As a result, the most complex arithmetic operation is a 6 bit by 6 bit multiplication.

Even though the bit level specification describes exactly what must be done to implement the ADPCM standard, it does not specify how. This has been left to the ingenuity of the individual designer, suited to a particular application. Various implementations have been reported in the literature, ranging from special-purpose very large scale integrated circuits (VLSI)[13] to programmable implementations on commercially available DSP processors.[14]

Performance

Sample results that are typical of the performance obtained with the 32-kb/s

Figure 2. Voice subjective MOS results—asynchronous tandem codings.

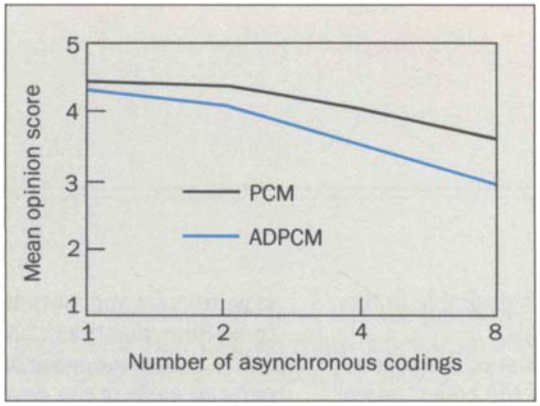
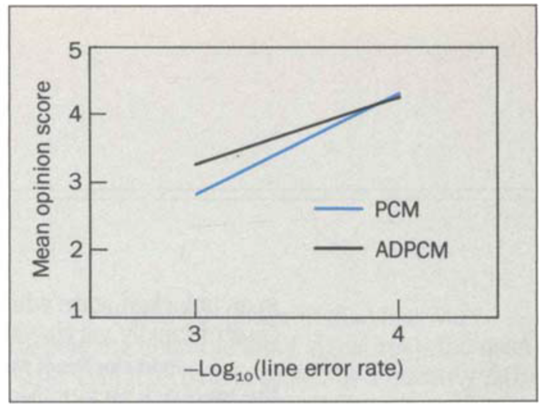


Figure 3. Voice subjective MOS results—line bit errors.



ADPCM algorithm are given here. (A complete set of results is given in Reference 9.) Figure 2 compares the tandem coding performance of ADPCM and PCM with speech. Mean opinion score (MOS) is plotted versus the number of asynchronous codings.

These results were obtained by having subjects listen to samples of encoded speech over calibrated handsets in an acoustically treated room (to maximize listener sensitivity) and rate what they hear according to a 5-point quality scale: excellent, good, fair, poor, and unsatisfactory.

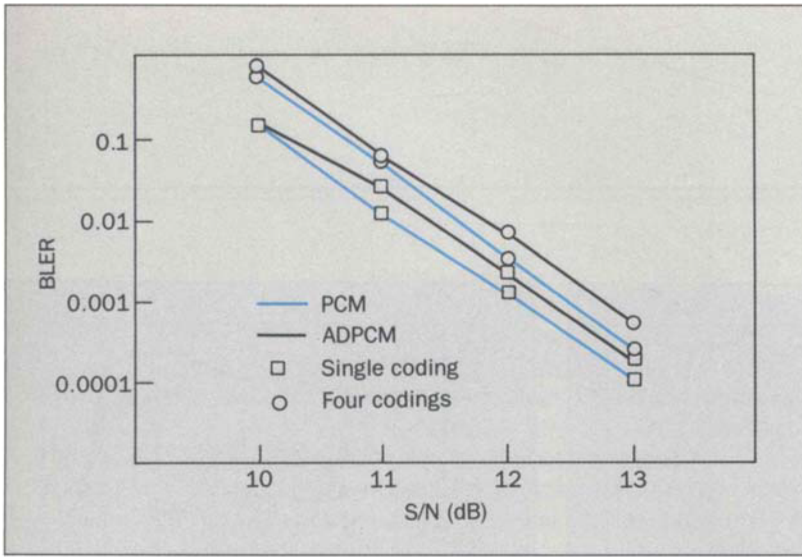
The figure shows that the performance with 32-kb/s ADPCM is comparable to 64-kb/s PCM for up to two asynchronous codings, slightly poorer for four codings and significantly worse with eight codings. Clearly, the deployment of asynchronous tandem codings of ADPCM in the network must be limited. CCITT and ANSI Committee T1Q1 have adopted a voice criterion which allows a maximum of four asynchronous ADPCM codings on an end-to-end connection if there is no other source of quantizing distortion. In addition, CCITT Recommendation G.113 allows one ADPCM coding in the national network on the national extension of an international connection. On the other hand, ADPCM is more robust than PCM in the presence of random bit errors as shown in Figure 3 where MOS scores are shown as a function of line error rate.

Illustrative voiceband data results (continuous carrier operation) are given in Figures 4 and 5 where block error rate (BLER) is plotted versus the level of random additive noise for the 2400-b/s V.26 and 4800-b/s V.27 modems respectively. Each block consists of 1000 bits. Test methodology is as in previous CCITT tests[2] with a single coding and four asynchronous codings, but with the added analog impairments contained in Table I.

With an acceptability criterion of a 10^{-2} BLER at an S/N of 24 dB, ADPCM provides an

Table I. Added Analog Impairments for Voiceband Data Tests

Analog impairments	Impairment level
Envelope delay distortion (relative to 1800 Hz)	4600 μs at 200 Hz, 2260 μs at 3200 Hz
Attenuation distortion (relative to 1000 Hz)	8.5 dB at 200 Hz, 5.6 dB at 3200 Hz
Second-order nonlinear distortion	36 dB
Third-order nonlinear distortion	38 dB
Phase jitter (at 120 Hz)	± 3° peak-to-peak
C-message weighted additive noise	Varied as a parameter



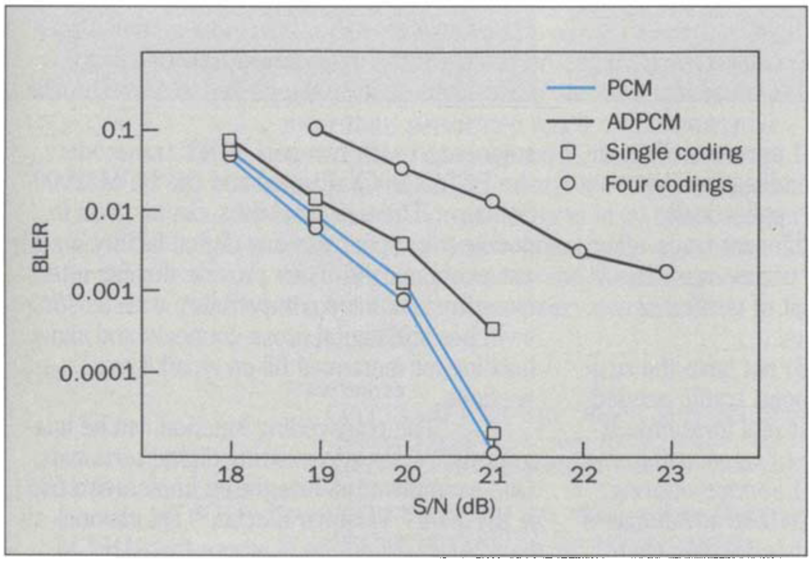


Figure 4. V.26 BLER results at 2400 b/s.

Figure 5. V.27 BLER results at 4800 b/s.

acceptable level of performance with both the V.26 and V.27 modems under these conditions with four asynchronous codings. As expected, the degradation with ADPCM relative to PCM is more pronounced with the higher speed V.27 signals. Performance of 9600-b/s V.29 is not acceptable for even one ADPCM coding.

However, bit error rate (BER) is recognized as an important criterion for some applications. In this case, the acceptability criterion is BER $< 10^{-5}$. This is almost always a more stringent constraint than BLER. Using this criterion, some modems provide acceptable performance with only two or three asynchronous codings at the 4800-b/s rate. In general, the impact on voiceband data performance is considerable even when limiting criteria are met.

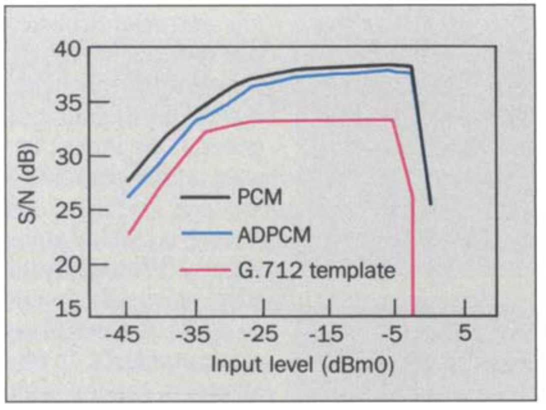
Classical transmission measurements such as S/N must be interpreted with care for adaptive signal processing algorithms such as ADPCM, since S/N typically depends on input signal statistics. In other words, such measurements, in general, cannot be used to predict performance for other signals with significantly different spectral and temporal characteristics. For completeness, Figures 6 and 7 respectively plot S/N versus input level for narrowband Gaussian noise (CCITT Recommendation G.712 Method I) and a 1004-Hz sine wave (Method II).

Applications

The specification of a standard 32-kb/s ADPCM algorithm opens the door to a host of applications in existing telecommunications networks.[15] These applications can be divided into three categories: telephone company use, end-customer applications, and new service offerings.

It is important to recognize that deployment of ADPCM must take into account performance limitations described above, particularly with regard to voiceband data. Because of these performance constraints, there is currently no industry agreement on any deployment strategy in the public switched network that can meet the service needs of different carriers.

By placing the transcoding function on customer premises, customers can double the



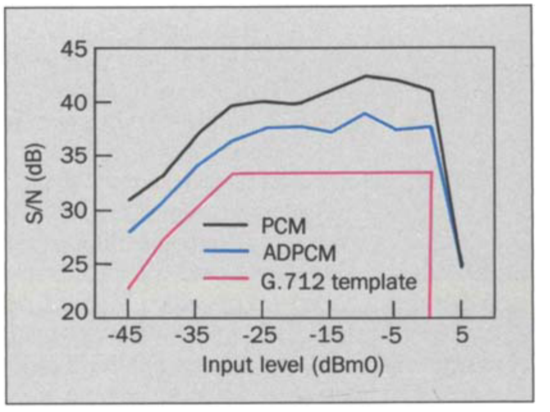


Figure 6. G.712 Method I performance—narrowband Gaussian noise.

Figure 7. G.712 Method II performance—1 kHz sine wave.

capacity of their leased DS1 services. This is the area where 384 kb/s bundling has the greatest impact.[15] In these applications, ADPCM exhibits slightly different trade-offs than in telephone company use because the prove-in is based on the cost of tariffed services instead of facility costs.

Many customers do not have the large cross-sections of point-to-point traffic needed to prove-in ADPCM at their end locations. If the telephone company offers transcoding as a service function on the DS1 service offering, many smaller customers can take advantage of ADPCM. This is particularly effective when multiple services are carried on the same DS1 (i.e., integrated access).

AT&T Products and Services

In 1984, AT&T introduced its first ADPCM transmission product, the AT&T Western Electric® BCM32000.[16] The BCM32000 is a stand-alone transcoder that provides the 2:1 compression function using the recently standardized 384-kb/s bundled frame format.[6]

The BCM32000 has recently been augmented with two new AT&T transcoders, the BCM32000X-Panded and the BCM32000-Solitaire. These transcoders can be used to double the capacity of any digital facility quickly and economically. They provide flexible interfaces for maximum compatibility with existing switches and digital cross-connects and allow hubbing for increased fill on small cross sections.

The transcoding function can be integrated directly into existing digital terminals. One example of an integrated implementation is the AT&T Western Electric® D4 channel bank ADPCM option,[17] where the ADPCM function is efficiently integrated into the common equipment of the bank. Only two new plug-ins are required to provide the ADPCM feature and transport all 48 D4 channels on only one DS1. The D4 with ADPCM option is fully compatible with the BCM32000X-Panded and the BCM32000-Solitaire. Similarly, the SLC® Series 5 carrier system will also provide an integrated 32-kb/s ADPCM capability for loop applications.

AT&T also offers services based on the 32-kb/s standard. The M44 service function[18] of the Accunet® T1.5 digital service provides for compressing either access or interoffice DS1 facilities. This interface is available for international services as well and was just recently used for the first time on trans-Atlantic facilities in conjunction with the international EPSCS (enhanced private switched communication service) offering.

Summary

AT&T played a major role in the development of 32-kb/s ADPCM standards. AT&T was instrumental in the development of the algorithm, performed much of the evaluative testing, and participated strongly in the standards arena. AT&T continues to be actively involved in industry forums in an effort to establish deployment constraints agreed to by the telephone industry and thereby maintain the quality of network services for our customers.

References

1. CCITT COM XVIII-R10, "Report on the Meeting Held in Geneva (18-22 June 1982)," source: Working Party XVIII/2 (Speech Processing), Geneva, July 1982.
2. CCITT COM XVIII-R28, "Report of the Meeting of Working Party XVIII/2 (Speech Processing) (Geneva, 21, 22 and 25 November)," source: Working Party XVIII/2 (Speech Processing), Geneva, December 1983.
3. CCITT Recommendation G.721, "32 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)," *CCITT Red Book*, Fascicle III.3, pp. 55-93, October 1984.
4. ANSI Committee T1, "Draft Proposed American Standard 32 KB/S ADPCM Algorithm and Line Format," Document No. T1Y1/LB 85-01, March 28, 1985.
5. ANSI Committee T1, Contribution to Subcommittee T1Y1.2, "FSK Enhancements and Idle Code Predictor Gain Modification for Proposed T1Y1 ADPCM Algorithm," source: AT&T Bell Laboratories, Document No. T1Y1.2/86-0027, March 1986.
6. ANSI Committee T1, "Draft Proposed American Standard 32 KB/S ADPCM Algorithm and Line Format," Document No. T1/LB 81, July 23, 1986.
7. CCITT COM XVIII-R17, "Report on the Meeting Held in Geneva (24-27 March 1986)," source: Working Party XVIII/8 (Speech Processing), Geneva, April 1986.
8. CCITT Study Group XVIII, "Report on the Meeting Held in Geneva (8-15 July 1986)," Temporary Document No. 26P, Annex 1 (modified G.721 algorithm), source: United States of America Delegation, Geneva, July 1986.
9. CCITT Study Group XVIII, "Report on the Meeting Held in Geneva (8-15 July 1986)," Temporary Document No. 26P, Annex 2 (verification test results), source: AT&T, Geneva, July 1986.
10. J. Max, "Quantizing for Minimum Distortion," *IRE Transactions on Information Theory*, Vol. IT-6, pp. 7-12, March 1960.
11. N. S. Jayant, "Adaptive Quantization with a One-Word Memory," *Bell System Technical Journal*, Vol. 52, No. 7, pp. 1119-1144, September 1973.
12. D. J. Goodman and R. M. Wilkinson, "A Robust Adaptive Quantizer," *IEEE Transactions on Communications*, Vol. COM-23, pp. 1362-1365, November 1975.
13. N. Sato et al., "A Single Chip Signal Processor for CCITT Standard ADPCM Codec," IEEE International Solid State Circuits Conference '85, *Conference Record*, p. 192, February 14, 1985.
14. A. Charbonnier, X. Maitre, and J. P. Petit, "A Digital Signal Processor Implementation of the CCITT 32 KBit/s ADPCM Algorithm," IEEE International Communications Conference '85, *Conference Record*, Vol. 1, pp. 1197-1201, June 23-26, 1985.
15. T. Kilm and D. K. Sparrell, "Introduction of 32 kb/s ADPCM Equipment into the North American Network," IEEE GLOBECOM '84, *Conference Record*, Vol. 2, pp. 803-807, November 26-29, 1984.
16. R. Adleman and D. K. Sparrell, "Taking a Big Bite Out of Bits," *AT&T Bell Laboratories Record*, pp. 4-8, August 1984.
17. C. N. Ade et al., "32-KB/S ADPCM in the Western Electric® D4 Digital Channel Bank," IEEE GLOBECOM '85, *Conference Record*, Vol. 3, pp. 1128-1131, December 2-5, 1985.

18. AT&T Technical Reference PUB 54070, "M44 Multiplexing, A Service Function of ACCUNET® T1.5 Service, Interface Specification," September 1986. Available from Director—Services, AT&T Communications, Bedminster, N.J. 07920.

Biographies (continued)

neering from Carnegie Mellon University and the M.S. in electrical engineering and computer science from the University of California at Berkeley. He joined AT&T in 1979. ***Mr. Daumer*** *received the B.S. and M.S. in electrical engineering from Drexel University. He joined AT&T in 1973.* ***Mr. Sparrell*** *received the B.S. and M.E. in electrical engineering from Rensselaer Polytechnic Institute. He joined AT&T in 1978.*

(Manuscript received September 29, 1986)