1988

Efficient Algorithms for Hybrid Data/Voice Transmissions

Andrzej Kusiuk

Wojciech Szpankowski
Purdue University, spa@cs.purdue.edu

Report Number:

88-805

EFFICIENT ALGORITHMS FOR HYBRID DATA/VOICE TRANSMISSIONS

Andrzej Kusiuk
Wojciech Szpankowski

CSD-TR-805
August 1988

EFFICIENT ALGORITHMS FOR HYBRID DATA/VOICE TRANSMISSIONS

Andrzej Kusiuk†
Institute of Telecommunication
Technical University of Gdansk
80-952 Gdansk
Poland

Wojciech Szpankowski*
Department of Computer Science
Purdue University
West Lafayette, IN 47907
USA

Abstract

Efficient algorithms for hybrid data/voice transmissions in the PCM channel systems are presented. The main idea behind these algorithms is to use some statistical properties of the coded samples (e.g., repetitiveness) to improve the utilization of the channel. Two algorithms are discussed in details. In the first algorithm, some voice channels (slots) are used to data transmissions. This is achieved by discovering repetitiveness in the coded samples (e.g., the most frequent samples, repeating samples, and so on), and sending only one bit in the would-be voice slot, while the next bits are exclusively data bits. The second algorithm enables to increase the number of telephone conversations in the PCM channel by dynamic assignment of slots to voice and data transmissions. Finally, we present throughput performance evaluation of both algorithms. In particular, we show the improvement of the channel utilization for these algorithms in comparison to standard PCM channels without voice/data integration. In addition, we compare the two solutions to point out that the latter one is superior from the utilization view point, but is is also more sophisticated.

1. INTRODUCTION

Circuit switching is used widely for voice communication, and packet switching has been successfully demonstrated and implemented for data communications. Advantage of former is constant rate transmission for a long period of time. It is important for a voice transmission. Message switching and packet switching are designed for one-way delivery of messages. These methods are not suitable for real-time conversational interactions between people or computers. However, they have certain advantages over circuit switching, namely they can operate more efficiently with a higher trunk utilization.

† This paper was written when the author visited Department of Computer Science, Purdue University. He acknowledges support in part by NSF grant NCR 8702115.

* This work was supported in part by NSF grant NCR 8702115

An integrated approach to the switching of voice and data is attractive because it provides a more cost-effective utilization of communication resources, e.g., transmission and switching facilities. It also provides the possibility of interconnecting a broad community of user terminals. These considerations have motivated the telecommunication system planners and designers to investigate the concept of integrated switching and implementation of integrated communication networks. The concept of integrated switching which combines the features of circuit switching and packet switching has been introduced in [1] and is, in essence, a hybrid scheme.

In this paper we study statistical voice/data multiplexers for hybrid data/voice transmission in the PCM channels. We propose to use circuit switched systems for voice transmission and message switched for data transmissions [2], [3]. The main purpose of this paper is to report voice/data multiplexer solutions which offer efficient channel utilization and good performance for voice/data integration in the PCM digital channels.

Before we present the idea of our algorithms, let us have a look at other solutions proposed so far for voice/data integration. It seems that the first important paper presenting the analysis of hybrid voice/data communications is due to Fischer and Harris [4]. In this early solution, data signals have not been packed into voice conversations, however, data channels were statistically allocated (se also [5], [6]). The voice/data multiplexing systems (i.e., data and voice signals are mixed in the channel), were reported in [6], [7], [8], [9], [10]. In these systems, data are allowed to utilize additional time slots that may be available during silence periods in voice transmission, as it is done in TASI method [11], [12], [13]. H.H. Lee, in his paper [5], studied four statistical voice/data multiplexes that offer efficient channel utilization. In this system, two separate finite buffers are used to queue voice and data packets. Four versions of master frame management are considered: fixed boundary frame for voice and data traffic, movable boundary for data traffic, movable boundary for voice traffic and movable boundary for both voice and data traffics. In these solutions, the performance of statistical voice/data utilization multiplexer depends upon voice source characteristics, as well as characteristics of the used speech detector. In discrimination of speech by speech detector, fill-in and hangover are used to reduce signaling overhead and to ensure continuous flow of speech. When the fill-in and hangover duration becomes shorter, the mean silence and talkspurt durations become shorter, and the speech activity becomes lower. Hence, the performance of the statistical voice/data multiplexer would be better if shorter fill-in and hangover durations are used in speech detection. This fact suggests that fill-in or hangover durations should be kept as small as possible.

In our solution, this fill-in or hangover duration are reduced to zero by using some statistical properties of the coded samples. In particular, we shall consider repetitiveness of samples to enhance data transmissions. So, in this solution, the speech detectors are not used at all because speech activity and silence are not detected. The algorithm depends only on the repetitiveness of coded samples. Both algorithms are based on the idea that a sample received from voice conversation is coded and then compared with previous coded samples from the same channel. If this sample is identical with the previous one, it is not sent in the channel, and the channel may be used for data transmissions. This solution requires one overhead bit of information to indicate whether repeated or new sample is sent. We present two algorithms which are built around this idea. The first algorithm is called *conditional message switching with static channel allocation* because data are sent on the message switching bases, and they are transmitted only under the condition that a voice sample is repeated. The second algorithm called *conditional message switching with dynamic channel allocation* is also based on the same idea , however channels are dynamically allocated between voice and data, and therefore a pair of customers may change the allocated channel during a conversation. Moreover, by allowing dynamic allocations and taking into account statistics of silence and talkspurt periods the algorithm may process more voice conversations than the maximum number of channels. Naturally, the second algorithm suffers on the quality of conversations (e.g., some voice samples of already accepted call may be lost) but we shall prove that it is not significant.

This paper is organized as follows. In the next section, we shall discuss the first algorithm. In particular, we describe a block scheme for transmitter and receiver which work under this algorithm. In Section III we present the second algorithm together with technical solution for the transmitter and receiver. The last section is devoted to the performance evaluation of both algorithms. We finally note that numerical results for the performance evaluation are done for A-Low standard PCM, that is, PCM 30/32.

2. CONDITIONAL MESSAGE SWITCHING SYSTEM WITH STATIC CHANNELS ALLOCATION

In this section, we present the first algorithm for voice/data integration, which is further called *conditional message switching with static channel allocation* (in short CMSS). In addition, we offer a block scheme for the receiver and the transmitter which work under this algorithm.

The main purpose of data/voice integration in digital channels, is to improve the channel utilization. One possible solution to this problem, adopted in this paper, is to mix voice and data signals in the same channel. We note that telephone conversations can not be interrupted in the sense that once the connection is established, it must be "kept alive" until its termination. Nevertheless, using packetized voice, a channel can be "stolen" from the voice connection and given to data. Such a solution, however, may suffer in quality of voice transmissions [14]. Therefore, we would rather keep circuit switching for voice communications, while data signals are sent into the channel only when the voice communication can be suspended. This solution is called *conditional message switching* since message switching applied to data signals under the condition that the channel is available.

The CMSS algorithm is practically sound, since in the voice communication there is a lot of repetitiveness and we can take advantage of this by saving some communication capacity. This can be done in a number of different ways. For example, if one can detect the most frequent signals (coded samples), then by building on the both sides of the communication channel a standard (device), we may need to send only one bit of information indicating whether a standart sample is to be transmitted or not. Therefore, 7 bits are saved and can be used for data transmission. This solution, however, may not be very practical, since the silence level in the different channels is not the same, and this has significant impact on the quality of conversation.

We propose another solution, which avoids most of the drawbacks discussed above. Instead of detecting special frequent signals, we simply discriminate a sequence of consecutive

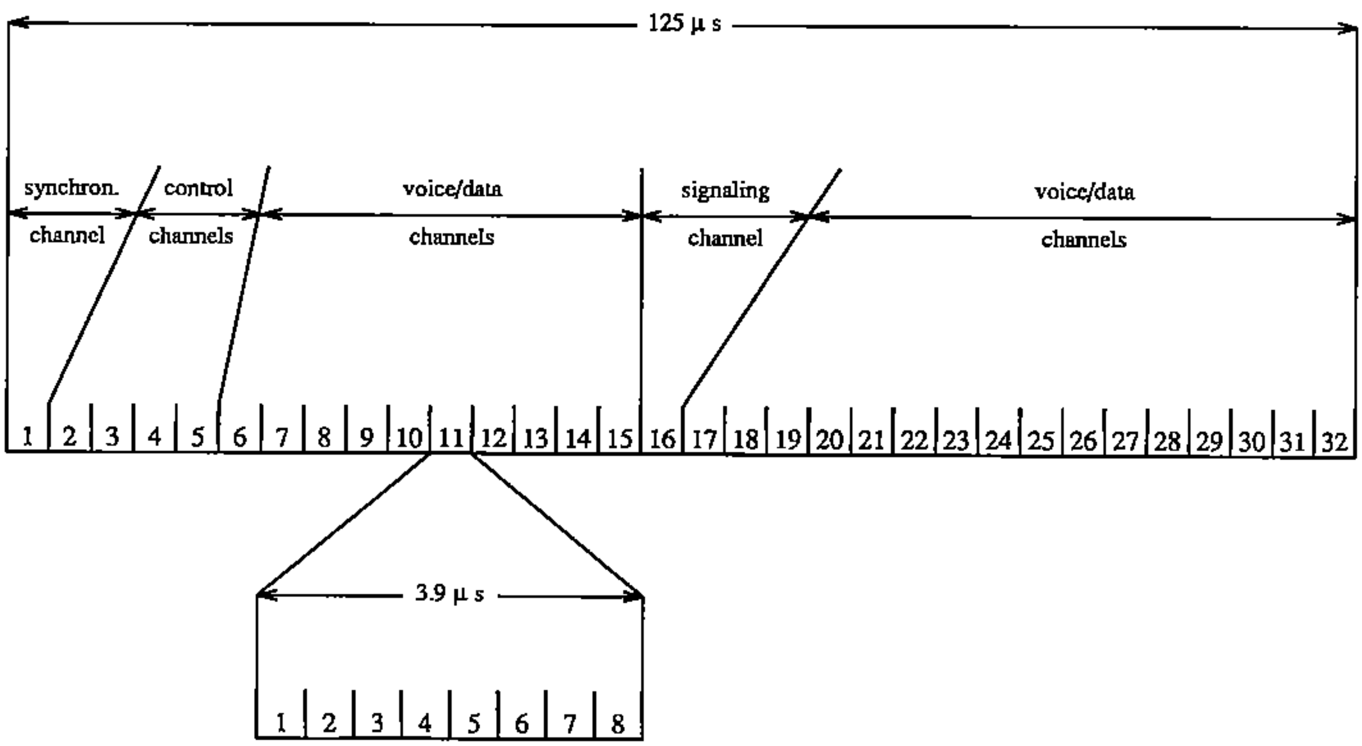

Figure 1. Modified frame of PCM 30/32 with voice/data integration.

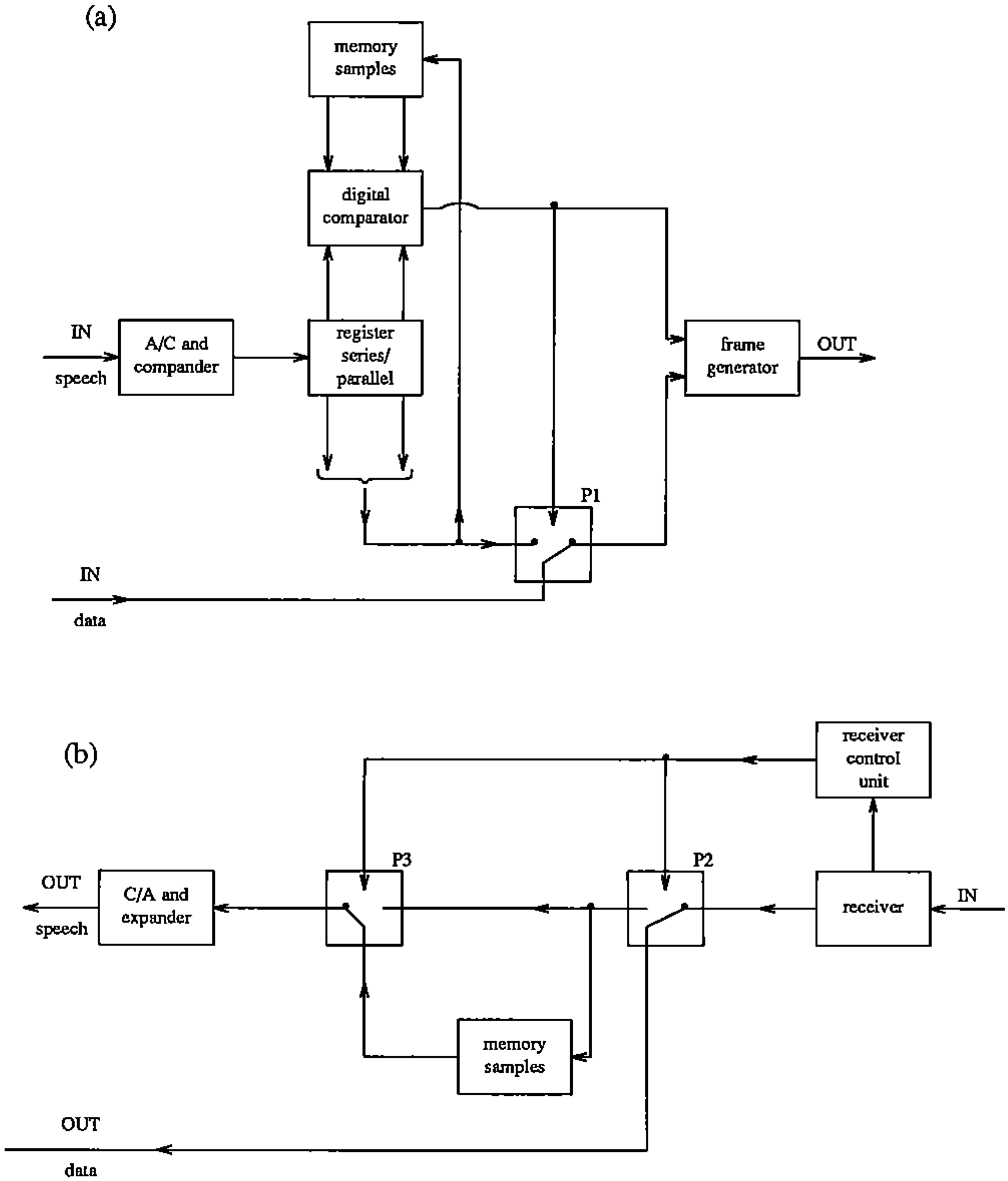

Figure 2. Block scheme for (a) transmitter, (b) receiver in the conditional message switching system.

identical samples. Only one sample out of the sequence is sent into the channel, while the other samples are reduced to one bit of information, and therefore seven bits are used for data transmissions (for engineering solutions see below). To achieve such a solution, we need to modify the basic structure of frame in the PCM 30/32 system.

In Figure 1, we show the modified structure of PCM 30/32 frame for integrated voice/data transmissions. As in PCM 30/32 there is one signaling and one synchronization channel, and thirty voice channels. But unlike the PCM 30/32, in our solution we have to reserve four control channels for the next 26 voice/data channels to carry information regarding the type of transmission (voice or data). Data transmissions may occur in any unused voice channel in a frame.

In Figure 2, we present the technical solution (i.e., block scheme) for receiver and transmitter of the CMSS system. On the transmitting side, a sample from the voice signal is coded and next it is compared in the digital comparator with previous coded samples from the same signal, as shown in Figure 2a. Previous samples are in the sample memory. The result of this comparison is used to send the samples (voice or data) into appropriate channels. If the compared samples are identical, switch P1 connects data buffers and one byte of data is transmitted into the channel. The result of comparison is sent into a frame generator and it is next transmitted by the digital channel. If the coded sample is different from the previous, it is sent into the generator frame with results of comparison, and next it is sent in the digital channel.

On the receiver side (Figure 2b) the frame from the channel is analyzed in the receiver, and the control slots (channels) are passed to the receiver control unit, which controls the P2 and P3 switches. If the control information indicates voice signals, then the switches are in the upper position. In addition, the voice sample is stored in the memory samples unit. If data transmission occurs, then the switches are in the lower position. This directly implies, as the figure suggests, that data signal is sent to a data user, while the voice user receives the sample from the memory unit.

Finally we note that our algorithm does not possess the drawbacks of the other solutions as discussed at the beginning of this section. In particular, we do not require speech detectors, so the voice transmission is not disturbed. In addition, the proposed algorithm improves the utilization of the channel as we shall discuss in detail in Section 4.

3. CONDITIONAL MESSAGE SWITCHING SYSTEMS WITH DYNAMIC ALLOCATION OF CHANNELS

The CMSS algorithm discussed in the previous section has two drawbacks, namely

- The number of voice users in the PCM channel is less than in the standard PCM 30/32 system (only 26 instead of 30).
- To increase the utilization of the channel one needs to increase data traffic in the channel.

In this section, we propose another algorithm that avoids most of the drawbacks discussed above. This solution is called conditional message switching with dynamic allocation of channels (in short CMSD), since both voice and data signals are dynamically assigned to different channels during a conversation, that is, an established pair of communicated users may use during a conversation different channels (slots) in consecutive frames.

It is known that in any conversation 60% of time is devoted to silence periods. Therefore, even without dynamic allocation (i.e., as in the first solution discussed in Section 2), at least 60% of the capacity can be gained by sending single impulses for repeated samples (i.e., silence samples). In addition, it is not difficult to prove (cf. [2]) that probability of transmitting different samples in all channels is very small, so it is a good chance that some voice channels can be used for other voice transmissions. Therefore, taking into account the statistical repetitiveness of samples, we may not only pack more data transmissions, but also increase the number of voice users. This can be explained in fact by the nature of the law of large numbers [17] which states, roughly speaking, that on the average the number of active users (i.e., simultaneous talkspurt), is much smaller than the number of users, and is approximately equal to the average number of users. So, designing the system from the average viewpoint rather than worst-case viewpoint (as it was adopted in our previous solution), allows us to connect more voice users to the system. This averaging approach is taken in our second algorithm.

The idea is to allow more simultaneous voice connections than the maximum number of channels. We note that once a voice connection is accepted, it *cannot* be dropped. Therefore, in very unfortunate situations when the number of *actual talkspurts* is larger than the maximum number of channels, some of voice samples are lost. We shall prove, however, that such a situation is very rare and the quality of voice transmission is still very high.

The details of the algorithm are discussed together with a description of a block scheme for receiver and transmitter, which work under the above algorithm, assuming the devices work in PCM 30/32 standard. In Figure 3, we present the modified structure of the standard PCM frame for 40 voice users. The frame consists of five control channels, two signaling channels,

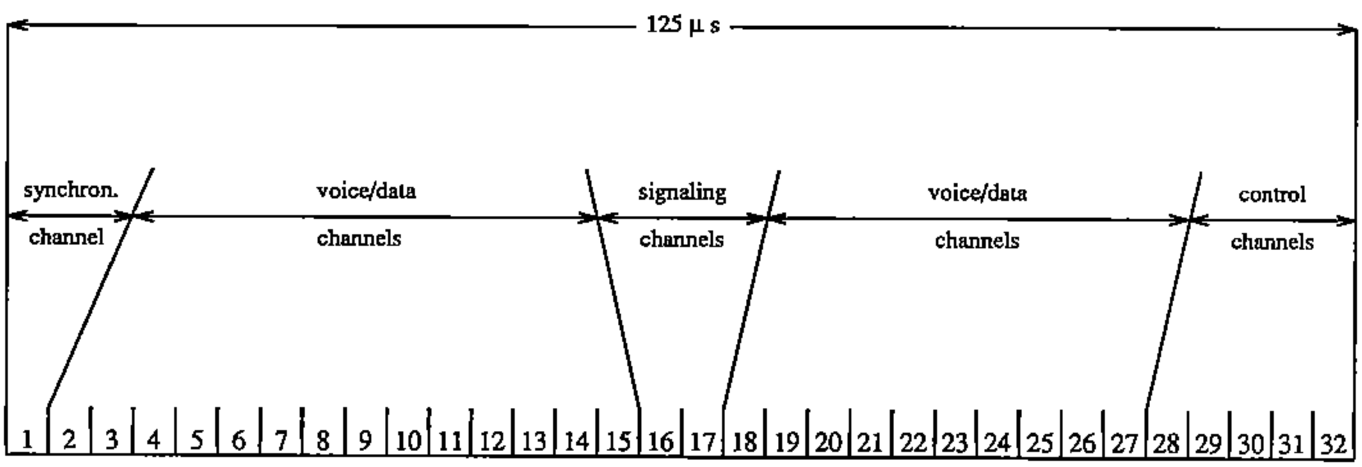

Figure 3. Modified frame of PCM 30/32 for dynamics allocation channels system.

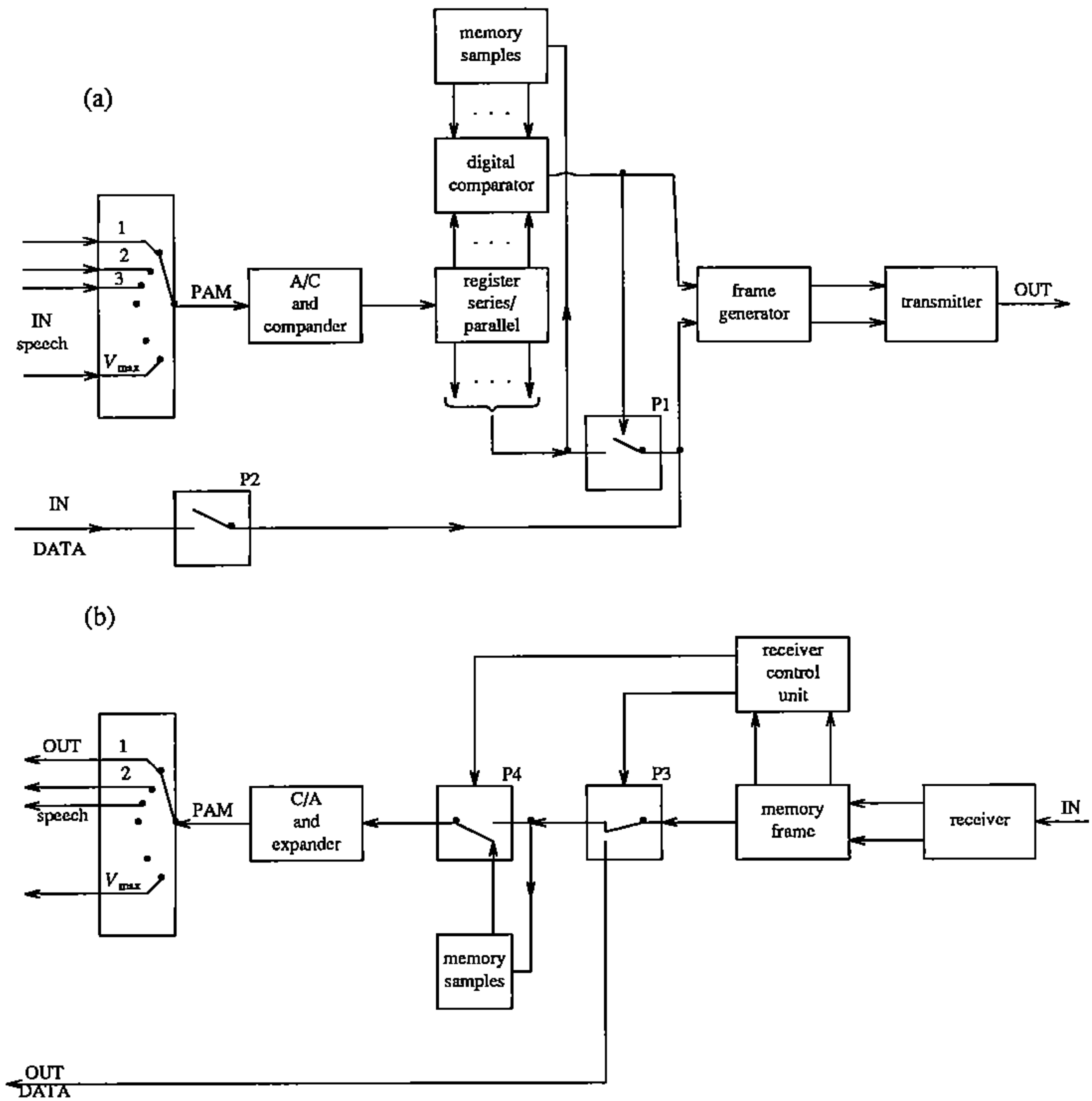

Figure 4. Block scheme for (a) transmitter, (b) receiver in the dynamics allocation channels system.

one synchronization channel and twenty four voice/data channels. We note that in our illustrating example with 40 users, two instead of one signaling channels are used. As in our previous solution, data signals are transmitting in unoccupied voice slots. We note, however, that in this solution voice users are assigned to consecutive slots (channels) in a frame, and only if some free slots are left, then they are assigned to data users.

Figure 4 shows the block scheme of transmitter and receiver for the conditional message switching with dynamic allocation of channel. As before, the set of users is divided into voice users and data users. Voice users have higher priority, while data users occupy only unused voice channels. In the previous algorithm, we have suggested a solution with signal compression by utilizing repetitiveness of voice data, that is, when a sample is repeated it is compressed to a single bit In the current solution, we apply both compression and *concentration* for voice users. This way more voice users may be connected to the channel as shown in Figure 4a. These voice users are cyclically scanned in a time T_1 which is shorter than the frame duration $T = 125\mu s$. In the second part, T_2 of the frame time T data users are served. To achieve this goal, we store all necessary information regarding the frame occupation in a block called frame generator. More specifically, in time T_1 the transmitter serves only voice users in the manner which almost mimics the first proposed solution (see Figure 2). There are, however, some differences. Since the number of voice users is larger than the maximum number of channels (i.e., concentration) we assume some losses, that is, a few samples in a call can be lost (for more quantitative analysis see Section 4). An accepted call cannot be dropped, and it is treated in exactly the same way as in Figure 2, that is, repeated samples are recognized and compressed to a single bit We mention again that voice users in this solution occupy *consecutive* slots (e.g., the same user may send signals in different slots in two consecutive frames). Once voice users are scanned in time T_1, then the rest of $125\mu s$ frame time is basically devoted to data users who try to transmit their information. Let us look at the frame occupation scheme at the end of time T_1. There are a number of channels, say 18, already assigned to voice users. So, the rest of the channels, say 6 for the situation in Figure 3, can be utilized by data users. As before, data users send their willness to transmit into the frame generator, which controls the occupation scheme for a frame. This block assigns the available channels to data users, form the frame, and sends signals to the transmitter.

On the receiver side (Figure 5b) the frame is first received and stored in the memory frame. We note here that every receiving frame is delayed by $125\mu s$ in comparison with transmitting frame. The information contained in control channels of a frame is used to control switches P3 and P4. Switch P3 is in the upper position for voice data, while P4 position depends upon the fact whether a new or repeated sample is received. As in the transmitting side, the time used for decoding is divided into two parts. In part T_1 voice channels are served, while in time T_2 (e.g., $T_1 + T_2 = T = 125\mu$) data users are received.

As a final note, we point out that the second algorithm avoids most of the problems encountered in the previous solutions as well as in our first algorithm. In particular, we may send between talkspurts not only data signals but also additional voice signals. Naturally , we pay for that by allowing some samples of a call to be lost. How much? The quantitative analysis of the algorithm is presented in the next section, and it indicates that losses can be very small if appropriate number of voice users is selected.

4. PERFORMANCE EVALUATION OF THE ALGORITHMS

In this section, we focus on the performance evaluation of the two algorithms discussed before. Our goal is to compare these two algorithms between themselves, and with a solution without integration. In particular, we shall compute the maximum throughput (i.e., capacity) of the channel available for data transmission. This parameter is our major measure for the quality of the algorithms, however, we also introduce some other performance characteristics.

Before we start our analysis, we need to agree upon a few things. The data transmission in the voice channels is of secondary importance, in the sense that the highest priority is given to the voice users. This implies that first we must assure high quality of voice transmission and, if possible, the rest of the channel capacity is given to data users. Therefore, to compute available capacity for data users, we concentrate in our analysis on voice transmission.

Let us begin the analysis from a detailed mathematical model description. This is done in a form of modeling assumptions, which are grouped into three categories namely, assumptions regarding the channel (M1-M2), regarding the set of users (M3-M5) and describing the service (M6). The first two assumptions are self explained and they describe the channel.

M1. There are N_{max} channels, each of them is noiseless. The propagation delay in a channel is neglected.

M2. All voice/data channels are randomly selected with equal probability.

The next two assumptions represent a set of users.

M3. The set of users is large (infinite) and collectively it produces the input traffic which is Poisson distributed with parameter λ.

M4. A call (message) which arrives when all channels are busy, is lost (i.e., loss system).

Finally, to analyze our algorithms, we need a probabilistic model for repeated samples, that is, we need to know what is the probability that the next sample exactly resembles the previous one. To the best of our knowledge, such statistics are not available in the open literature. We are in the process of gathering them, but for the purpose of this paper, we adopt the worst-case approach. Namely, it is known that silence periods occupy about 60% of a conversation time [15], [16]. Naturally, silence samples are repeated samples, however, in a real conversation, there are more repeated samples than silence samples. Nevertheless, adopting worst-case approach we assume:

M5. A new sample appears with probability p ($p = 0.4$ in our analysis). In addition, new samples occur independently (i.e., Bernoulli distribution for new samples).

At last, we traditionally (and for simplicity) assume exponential distribution of a conversation, i.e.,

M6. A conversation is exponentially distributed with parameter μ.

Now we are ready for the analysis of our two algorithms. Here is the plan for our analysis. In both cases, we adopt the voice transmission viewpoint, as explained at the beginning of this section. This implies that the probability of loss B cannot exceed the level recommended by CCITT (e.g., $B = 0.005$ for interstation channels). In our model, B is given, and under assumptions M1-M6 we compute traffic $A = \lambda/\mu$ in Erlangs and the channel utilization ρ. The latter parameter is defined as the ratio of the average number of busy channels to the maximum number of channels $N_{\max}$ (e.g., PCM 30/32, $N_{\max} = 32$ and $\rho < 30/32$). In addition, we consider voice channel utilization ratio ρ_V, which is defined as the ratio of the average number of channels occupied by voice (talkspurts) to $N_{\max}$. Finally, we compute the *normalized* capacity for data transmission C_D, that is, the maximum normalized flow of data the channel can carry. Naturally

$$C_D = \frac{N}{N_{\max}} - \rho_V \tag{1}$$

where N is the actual number of channels available for users (e.g., in PCM 30/32, $N = 30$). In sequel, we compare ρ and ρ_V as well as C_D for all analyzed algorithms.

Let us present details of our analysis and we first concentrate on the conditional message switching algorithm with static channel allocation (CMSS). We need some mathematics to present our results. Let X and Z represent the number of busy channels and the number of channels occupied by voice respectively. Under stationary conditions, and with our assumptions M1-M6, the system can be analyzed as M|M|N|N queueing model, whence [17]

$$Pr\{X=i\} = \frac{A^i/i!}{\sum_{k=0}^{N} A^k/k!} \quad i=0,1,\ldots,N \tag{2}$$

where $A=\lambda/\mu$. In particular, the probability of loss B is evaluated as $B=Pr\{X=N\}$. Since in our model B is given , one computes A from (2) and therefore, we denote it as $A(N,B)$. Knowing the traffic $A(N,B)$ (in Erlangs), we immediately obtain the channel utilization as

$$\rho = \frac{A(N,B)\cdot(1-B)}{N_{\max}} \tag{3}$$

We note that in CMSS algorithm built on PCM 30/32 standard we have $N_{\max}=32$ and $N=26$ (see Section 2). The next step is to compute the voice channel utilization ratio ρ_V. But, under our assumption M5, the voice traffic $A(N,B)$ drops to $p\cdot A(N,B)$, hence

$$\rho_V = p\rho \tag{4}$$

This formula indicates that the actual utilization of the channel is p times smaller, therefore, $\rho-\rho_V$ can be additionally utilized for data transmission. More precisely, let us compute the maximum normalized data throughput (i.e., capacity C_D) in the system. By (1), we note that

$$C_D = \frac{N_{CMSS}}{N_{\max}} - \rho_V = \frac{N_{CMSS}}{N_{\max}} - \rho + \rho - \rho_V = C_D^I + C_D^A \tag{5}$$

In (5) we split the normalized data capacity C_D into (almost algorithm independent) term C_D^I (I stands for *integration*) and algorithm dependent term $C_D^A = \rho-\rho_V$. Indeed, we note that the probability p in (4) depends, in fact, on the integration algorithm. For example, algorithms with silence detectors [18], [15], [20] have much smaller p compared to our two solutions.

Finally, one may ask about the distribution of the number of channels occupied by voice transmissions in the CMSS algorithm, that is, $Pr\{Z=j\}$. This distribution is of particular interest for the CMS algorithm with dynamic allocations (i.e., CMSD). To compute $Pr\{Z=j\}$, we first note that by assumption M5

$$Pr\{Z=j \mid X=i\} = \binom{i}{j} p^i (1-p)^{2-j}$$

and then (2) and the above imply, after some algebra, that

$$Pr\{Z = j\} = \frac{\left[\frac{p}{1-p}\right]^j \sum_{i=j}^{N} \binom{i}{j} \frac{A^i}{i!}(1-p)^i}{\sum_{k=o}^{N} \frac{A^k}{k!}} \tag{6}$$

Now we turn our interest to the second algorithm CMS with dynamic channel allocation (CMSD). This solution is more sophisticated since two types of losses must be considered. The first one, as in the previous algorithm, deals with a loss of a call, that is, whenever a new call arrives to a busy system, it is lost. However, in this solution, the number of maximum simultaneously connected voice users $V_{\max}$ is *larger* than the number of channels N. Therefore, some *samples can be lost* of an already *accepted* call. We denote this probability of sample loss as P_S.

Based on CCITT recommendations, the probability of loss for a call B cannot exceed a given level. Therefore, we fix B, as in the first algorithm. We note that this time $B = Pr\{X = V_{\max}\}$, where $Pr\{X = i\}$ is given by Erlang formula (2). From (2) for a given B, we compute the traffic $A(V_{\max},B)$ which depends on $V_{\max}$ (the maximum number of calls which can be accepted) and B. Knowing $A(V_{\max},B)$, we can easily evaluate the voice channel utilization ρ_V, since

$$\rho_V = pA(V_{\max},B)(1-B)/N_{\max} \tag{7}$$

Naturally, ρ_V from (7) is larger than the one computed in (4) for CMSS algorithm, because the traffic A is larger. But we pay for that by an additional loss in samples (and the quality of transmissions). To compute the probability of loss P_S, we note that a voice sample is lost if the number of simultaneously sent voice samples is larger than N (the actual number of channels) and smaller than the maximum number of accepted calls $V_{\max}$. Hence

$$P_S = Pr\{N < Z < V_{\max}\} \tag{8}$$

where the distribution of Z is given by (6). Finally, the capacity of data transmission C_D is

$$C_D = \frac{N_{CMSD}}{N_{\max}} - \rho_V \tag{9}$$

and C_D can be split into two terms as in (5). We note only that for PCM 30/32, the number of transmission channels N_{CMSD} in the second algorithm is 24, while N_{CMSS} for the first algorithm is 26.

In the conclusion, we compare the two proposed algorithms. In Table I, we show the voice utilization ρ_V as well as capacity of normalized data utilization ratio C_D for a system without integration and two algorithms, that is, CMSS and CMSD. We assume that both algorithms are implemented on the PCM 30/32 system. Comparing the voice utilization ρ_V for the three discussed solutions, one immediately notes that the best utilization ratio is achieved, as expected, in CMSD algorithm, while ρ_V for the other systems are comparable.

Service	N	V_{max}	B	ρ_V	C_D
Without Integration	30	30	0.01	0.23	0.00
			0.02	0.25	0.00
			0.03	0.28	0.00
			0.05	0.29	0.00
CMSS	26	26	0.01	0.20	0.61
			0.02	0.22	0.59
			0.03	0.23	0.58
			0.05	0.24	0.57
CMSD	24	40	0.01	0.36	0.39
			0.02	0.38	0.37
			0.03	0.38	0.37
			0.05	0.41	0.34

Table I. Comparison of algorithms for PCM 30/32 with $p = 0.4$.

It is a little smaller for CMSS, because only 26 out of 30 transmission channels are used for voice transmissions in CMSS solutions. Nevertheless, the static solution (CMSS) has the biggest potentials for data transmission, since the capacity C_D is the largest.

V_{max}	B	$\bar{P}_V$
40	0.01	0.00012
	0.02	0.00021
	0.03	0.00024
	0.05	0.00045
44	0.01	0.00089
	0.02	0.00165
	0.03	0.00183
	0.05	0.00304
48	0.01	0.00418
	0.02	0.00713
	0.03	0.00924
	0.05	0.01307

Table II. Probability of sample loss in CMSD algorithm for PCM 30/32 with $p = 0.4$.

We note also that the total normalized capacity for the static and dynamic allocation algorithms are equal to 0.81 and 0.75 respectively. It seems to us that the dynamic allocation algorithm is superior over the static solution (at least from the voice viewpoint), but one needs to remember that the former algorithm is much more sophisticated, and it additionally incurs losses in voice samples as shown in Table II. An important conclusion for Table II, is the fact that the probability P_S of loosing a sample with $V_{max} = 40$ is rather low, and should not at all disturb voice transmissions.

5. CONCLUSIONS

In this paper, we proposed two algorithms for voice/data integration. The first algorithm is simple in practical realizations and provides significant capacity of the channel for data transmission. If data transmission is intensive, then this algorithm also leads to a high channel utilization. The second discussed algorithm is more sophisticated, but the utilization of the channel by both voice and data is high. This solution, however, needs significant changes in telephone exchanges. In particular, it requires fast C/A converters, and in addition, it needs very reliable channels.

Both proposed algorithms differ significantly from the other solutions proposed so far in the literature. They do not use any silence detectors, and therefore, any silence period can be used for data transmissions. From the performance evaluation viewpoint, both algorithms are superior in comparison with other solutions. This would be even better visible if one can exactly estimate the probability of repeated samples. In the future we plan to carry out some experiments in real environment and estimate more accurately this probability.

REFERENCES

[1] K. Kümmerle, "Multiplexer performance for integrated line and packet switched traffic," in Proc. Int. Conf. Comput. Commun., Stockholm, Sweden, Aug. 12-14, 1974, pp. 507–515.

[2] A. Kusiuk, "Analysis of a node in message switching systems," Ph.D. Thesis, Technical University of Gdansk, 1988.

[3] A. Kusiuk and M. Zientalski, "A utilization of synchronic channels in the PCM system and its realization," register to patent 87.06.03, No. P-266081, Institute of Telecommunications, Technical University of Gdansk.

[4] M.J. Fischer and T.C. Harris, "A model for evaluating the performance of an integrated circuit-and-packet-switched multiplex structure," IEEE Trans. Commun., Vol. COM-24, pp. 195–202, Feb. 1976.

[5] E. Arthurs and B.W. Stuck, "Traffic analysis tools for integrated digital time division link level multiplexing of synchronous and asynchronous message streams," IEEE J. Select. Areas Commun., Vol. SAC-1, pp. 1112–1123, Dec. 1983.

[6] N. Janakinaman, B. Pagurek and J.E. Neilson, "Performance analysis of an integrated switch with fixed or variable frame rate and movable voice/data boundary," IEEE Trans. Commun., Vol. COM-32, pp. 34–39, Jan. 1984.

[7] M.J. Fischer, "Data performance in a system where data packets are transmitted during voice silence periods – single channel case," IEEE Trans. Commun., Vol. COM-27, pp. 1371–1375, Sept. 1979.

[8] T. Bially, A.J. McLaughlin and C.J. Weinstein, "Voice communication in integrated digital voice and data networks," IEEE Trans. Commun., Vol. COM-28, pp. 1478-1490, Sept. 1980.

[9] G.F. Williams and A. Leon-Garcia, "Performance analysis of integrated voice and data hybrid-switched links," IEEE Trans. on Commun, Vol. COM-32, pp. 695–706, June 1984.

[10] H.H. Lee and C. Kwan UN, "Performance analysis of statistical voice/data multiplexing systems with voice storage," IEEE Trans. Commun., Vol. COM-33, pp. 803–819, Aug. 1985.

[11] K. Bullington and M. Fraser, "Engineering aspects of TASI," Bell Sys. Tech. J., Vol. 41, pp. 1439–1473, July 1964.

[12] C.J. Weinstein, "Fractional speech loss and talker activity model for TASI and packet-switched speech," IEEE Trans. Commun., Vol. COM-26, pp. 1235–1257, Aug. 1978.

[13] R.V. Cox and R.E. Crochiere, "Multiple user variable rata coding for TASI and packet transmission systems," IEEE Trans. Commun., Vol. COM-28, pp. 334–344, Mar. 1980.

[14] J.C. McDonald, "Fundamentals of digital switching," Plenum Press, New York and London, 1983.

[15] C.J. Weinstein, "Fractional speech loss and talker activity model for TASI and for packet-switched speech," IEEE Trans. on Commun., Vol. COM-26, No. 8, pp. 1253–1257, Aug. 1970.

[16] R.L. Easton, P.T. Hutchinson and R.W. Kolor, "TASI-E", Communications System, IEEE Trans. on Commun., Vol. COM-30, No. 4, pp. 803–807, Apr. 1982.

[17] L. Kleinrock, "Queueing systems, Vol. I: Theory," Wiley, 1975.

[18] T. Suda, H. Miyahara and T. Hesegawa, "Performance evaluation of packetized voice systems in simulation study," IEEE Trans. on Commun., Vol. COM-32, No. 1, Jan. 1984.

[19] K. Spiram, P.K. Varshney and J.G. Shanthikumar, "Discrete-time analysis of integrated voice/data multiplexers with and without speech activity detectors," IEEE Journ. on Selected Areas in Commun., Vol. SAC-1, No. 6, Dec. 1983.

[20] C. Robegand, J.P Adoult, "Fast on-line speech/voice band data discrimination for statistical multiplexing of data with telephone conversations," IEEE Trans. on Commun., Vol. COM-34, No. 8, Aug. 1986.