論文 95-6-6-15

A Random Token Protocol for Voice/Data Integration in High-Speed Networks

Dong Soo Park*, Chong Kwan Un** Regular Members

고속망에서 음성과 데이타 통합을 위한 랜덤 토큰 프로토콜

正會員 박 동 수*, 殷 鍾 官**

ABSTRACT

We propose a voice/data integration protocol for high-speed networks based on a random token protocol. In this protocol, a TDMA-like service is provided for voice traffic and the random token protocol is used for data traffic. We use a framed approach with a movable boundary scheme. The protocol incorporates minimal network information and requires only limited synchronization like random access schemes. Therefore, the protocol is suitable for high-speed networks with frequent reconfiguration and also for mobile networks, where integrated voice and data service is required. We analyze our proposed voice/data integration protocol for both voice and data. We get the probability of voice clipping and the fraction of wasted bandwidth for voice performance. We also get the delay-throughput performance for data. We show that the fraction of speech loss can be maintained under a specified maximum by suitable choice of maximum size of a voice region in a frame and that the channel can be operated at a throughput close to unity. We also show that the protocol is robust and fair to data packets, requires little overhead to implement, and the voice performance is not affected by data traffic. In addition, we discuss numerical results obtained for various system parameters and verify them by simulation.

要 約

본 논문에서는 랜덤 토큰 프로토콜을 사용하는 고속망에서 음성과 데이타를 통합하는 프로토콜을 제안한다. 이 프로토콜은 가변하는 한계방식의 프레임 접근법을 사용하여 시분할 다중화 방식 같은 서비스를 음성 드래픽에 제공하고, 랜덤 토큰 프로토콜을 데이타 트래픽에 제공한다. 프로토콜은 최소의 망 정보와 랜덤 접근 방식과 같은 제한된 동기를 필요로 한다. 그러므로 제안된 프로토콜은 음성과 데이타가 통합된 서비스를 제공하는 자주 재구성되는 고속망과 이동망

*삼성전자
Samsung Electronics, Co., Ltd.
**한국과학기술원 전기 및 전자공학과
Dept. of Electical Engineering, KAIST
論文番號 : 95093-0304
接受日字 : 1995年 3月 4日

에 적합하다.

본 논문은 음성과 데이타 양쪽에 대하여 제안된 음성/데이타 통합 프로토콜을 분석한다. 음성의 성능을 위하여 음성 절단(clipping) 확률과 낭비되는 대역의 비율을 구하고, 데이타에 대하여 지연시간-효율(delay-throught)을 구한다. 본 논문에서는 음성 손실의 비율은 프레임에서 음성 영역의 최대크기를 적절히 선택한 구체화된 최대상황 아래서는 유지되고, 전송선이 1에 가까운 효율에서 동작할 수 있다는 것을 보인다. 또한 프로토콜이 에러 상황에 잘 견디고, 구현하는데 적은 오버해드를 필요로 하며, 데이타 패킷들에게 공정한 기회와 데이타 트래픽에 영향을 받지 않고 음성의 성능을 보장하는 것을 보인다. 덧붙여서 다양한 시스템 변수롬에 대한 결과를 살펴보고 시뮬레이션으로 이름 검증한다.

I. INTRODUCTION

Historically, voice and data communications have been handled by different networks. Today, data traffic is mostly carried by local area communications. The use of different networks for voice and data communications is due to the fact that voice and data signals have fundamentally different characteristics. Voice traffic usually requires a stringent delay for its real-time interactive application, although it can tolerate some loss. On the other hand, data traffic can tolerate longer and variable delays, but it requires an error-free delivery.

Recently, applications of local computer networks are steadily increasing. While interconnection of computers and resource sharing have been typical of their applications, their envisioned role lies in office automation which is receiving increased attention today[1]. Thus, much interest is being focused on such real time applications as interconnection of workstations with the capability of handling voice and data in the office environment. This requires a network capable of transimitting the various types of traffic between the stations[2]. Transmitting voice and data traffic over the same medium is desirable, because duplication of facilities can be avoided and the utilization of the network resources can be increased.

For voice communication in computer networks, the analog voice signal is first digitized and collected into packets at its source. Then the voice packets are transmitted over the network to their destination, where they are received and decoded to reproduce the original signal. Since the characteristics and transmission requirements of voice traffic are quite different from those of data traffic, a framed approach is often employed for integrating voice and data traffic. In this approach, a frame, which is repeated, consists of a voice subframe followed by a data subframe. The size of a frame may be fixed or variable, and the boundary allocation between the voice region and the data region may be fixed or movable.

The strong requirement for integration of voice and data over a single communication channel has stimulated to propose a number of integrated protocols for bus networks[1]~[10] and ring networks[11]~[15]. In the shared bus environment, most protocols are variations and extentions of carrier sense multiple access with collision detection(CSMA/CD)[2]. It is well known that CSMA/CD performs well for data traffic[16]. However, CSMA/CD provides poor performance for voice packets when voice traffic is added to the data[17]. Several integrated voice/data protocols based on CSMA/CD have been proposed in the last few years. These protocols combine the time division multiple access(TDMA) protocol for circuit-switched voice transmission and a CSMA-based protocol for packet-switched data transmission into one protocol[3]~[5]. Maxemchuck proposed a combined TDMA/CSMA protocol which does not incorporate a framed approach, where TDMA-like voice proposed transmissions are interspersed in with periods of data transmission and the order is detemined by each call generation time[3]. In this approach, a full packet must be transmitted even if the call is in the silent state and voice packets have high overhead to provide TDMA-like service. Meditch[4] and Chlamtac[1] used a fixed length frame and a fixed voice/data boundary scheme in their TDMA/CSMA protocols, in which synchronized clocks are needed to maintain the frame structure. Sharrock *et al.*[2] proposed a CSMA/CD-based, integrated voice/data protocol. They use a variable size frame and a movable boundary scheme, and no synchronized clocks are needed in their protocol. However, some modifications should be done to CSMA protocols in order to provide a TDMA-like service for traffic. In addition, no precise analyses have been done for the above protocols, and thus the behavior of protocols has been investigated via only simulation or approximate analysis.

In this paper, we propose a voice/data integration protocol for bus and radio networks, based on a random token protocol. The random token protocol is an upward extention of random access protocols[18]. It is similar to the p-persistent CSMA protocol except that the former has a scheduling period. In the p-persistent CSMA and the random token protocol, a ready user who has a packet to transmit senses the channel. If the channel is sensed busy, it persists until the channel becomes idle. Then it transmits its packet with probability p or waits with probability$(1-p)$ in the p-persistent CSMA. On the contrary, in the random token protocol, the user selects its scheduling time from a uniform distribution, waits until its scheduling time, and then transmits its packet if the channel is still idle or waits until the channel becomes idle otherwise. Thus the scheduling is obtained by random, implicit token passing[18]. Although it is not a strict demand-assignment protocol, the performance is superior to the random access schemes. In addition, priority functions can easily be invoked and voice/data integration can easily be done, since scheduling periods exist.

To avoid the capacity loss due to silences, we employ a movable-boundary scheme for our voice/data integration protocol in which the voice subframe size is determined by the number of active voice stations in talkspurts at the start of the frame. Access to the data subframe is provided to the data stations via the random token protocol. For voice, we determine the fraction of speech loss and wasted bandwidth by formulating a Markov chain for the number of ready voice stations at the frame boundary. For analysis of data performance, we employ a Markov chain by considering the voice region of frame as a long packet with priority. Simulation results are provided to validate our anslysis.

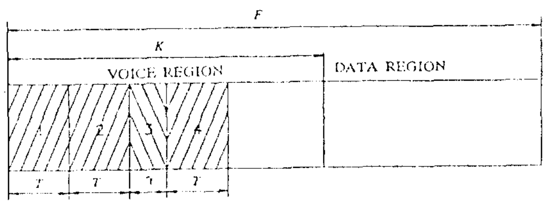
Following this introduction, in Section Ⅱ, we give a description of the voice/data integration protocol. In Section Ⅲ, we provide an analysis of our proposed protocol. In Section Ⅳ, we present the analytic and simulation results. Finally, we make conclusions in Section Ⅴ.

Ⅱ. VOICE/DATA INTEGRATION PROTOCOL

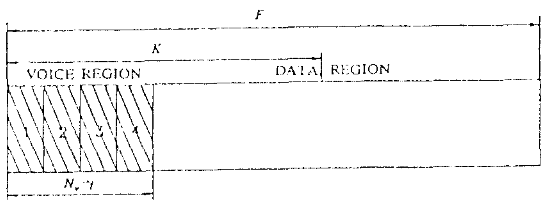
A. Overall Frame Structure

We consider a bus system with N_v active voice stations and N_d data stations, and assume that voice and data packets are of the same length. We assume that a station has no knowledge of the total number of stations in the network and of their logical ordering, nor of relative physical locations of stations. We further assume that the system is fully distributed. Thus, when executing a channel access protocol, a station can only utilize local information obtained by sensing the channel state: idle, successful transmission, or collision. The channel is sensed busy when successful transmissions of collisions occur. Channel time is partitioned into frames of variable length. The reason for the variance of frame size is that the choice of a node maintaining the frame structure for the network varies over time. Each frame is partitioned into one voice region and one data region, with the voice region occurring first. The boundary between the voice and data regions shifts from frame to frame according to the number of talkspurts of active voice calls during each frame. We also assume that time is minislotted with duration τ where τ is the end-to-end propagation delay. Fig. 1 illurstrates the frame structure with voice and data. The voice region contains one slot for each established voice call. The size of a slot is T where T is the packet transmission time. A station in a silent state sends a burst carrier during γ minislots to inform that it is silent. For convenience, T and γ are assumed to be integer multiples of minislots(i.e., integer multiples of τ).

Fig. 1 (a) shows a frame with both voice and data where K is the maximum of voice slots in a frame. If there are so many voice calls that the total number of voice slots is greater than K, then some packets of those calls are lost, and thus voice clipping occurs. It is also possible for the frame to be completely occupied by data as shown in Fig. 1 (b) if there are no calls. The slotted structure of the voice region results in a collision-free virtual circuit for each established voice call. The remainder of the frame time following the voice slots comprises of the data region. In the data region, data and call-related packets are transmitted using the random token protocol. Since the end of carrier(EOC) serves as a time reference for each station in the random token protocol, there is no need of system-wide synchronization clocks to maintain the frame structure. The voice and data regions are automatically delimited by the protocol.



(a)



(b)

F : FRAME SIZE K : MAXIMUM SIZE OF VOICE REGION
T : VOICE TRASMISSION TIME γ : CHANNEL JAMMING TIME

Fig. 1. Frame structure for voice/data transmission. (a) Case with three of four active voice stations in talkspurts. (b) Case with all four active voice stations in silence.

B. Voice Station Protocol

Let W be the channel bandwidth and R be the digitization rate of a vocoder in bits/s. Then $F=[W/R]$ is the frame length in slots where $[\alpha]$ is the largest integer smaller than or equal to α. For our protocol we require that $K \leq F$. Let t_j be the beginning of the frame j, ($j = 1, 2, \cdots$). We assume that a speech packet is created at the voice station and transferred into that station's buffer at t_j. If the station is in the talk state at t_{j-1} and t_j, the amount of speech collected will be sufficient to make a complete packet. If the station is silent at t_{j-1} and t_j, no speech packet is made. Otherwise, a partial packet is created with which a full packet is made by filling in its silent period. The packet assembled during frame (j-1) is transmitted in frame j.

Every voice station maintains three variables: *COUNT*, *ORDER*, and M_1. If a voice station wishes communication, it should obtain a voice slot. To obtain it, the caller must successfully broadcast a call request packet during the data region. If the call request packet is successfully transmitted, then it increments *COUNT* by one, which denotes the number of calls in the system (that is, the number of active users), and sets *ORDER* to *COUNT* where *ORDER* represents its position in the voice region of a frame. As a result, it obtains a voice slot of number *ORDER*. If the call request packet is not successfully transmitted, it canot obtain a voice slot. A new frame always begins every F solts. Whenever a new frame starts, every active voice station sets its variable M_1 to *ORDER*, and then decrements M_1 by one everytime it detects an EOC. It also counts the number of voice slots of size T. When M_1 equals one, the station transmits its voice packet following that EOC if it has a voice packet to transmit, or sends a burst carrier following that EOC if it is in the silent state. However, it does not transmit a voice packet which will be lost when the number of voice slots transmitted of size T is greater than K. When more than K voice stations are ready to transmit packets in a frame, only K stations whose *ORDER* values are smaller than the others can transmit packet. Then, stations whose *ORDER* values are greater than K stations having transmitted packets will lose their chances to transmit packets, resulting in packet losses and consequent clipping of corresponding speech. If the station wishes to stop communication, a call clear packet in which its *ORDER* value must be contained should be successfully broadcast. If a call clear packet is successfully transmitted, it decrements its *COUNT* by one, and sets its *ORDER* and M_1 to zero, and then the connection is cleared.

Everytime call request packets broadcast by other stations are received, every voice station increments its *COUNT* by one. Similarly, whenever call clear packets broadcast by other stations are received, every voice station decrements its *COUNT* by one. If the *ORDER* value of received call clear packet is less than its *ORDER* value, the station also decrements its *ORDER* value by one, so that its position number in a frame decreases as calls with lower position numbers terminate. It should be noted that all stations should update their variable *COUNT* whether they set up their calls(that is, virtual circuits) or not.

Initialization is performed in the following manner. The value *COUNT*=0 means that there is no voice call, so the channel is used by data users only. If a voice station wants communication and its *COUNT* value is zero, then it broadcasts a call request packet. If it is successfully transmitted, then it sets its *COUNT*, *ORDER*, and M_1 to one, and senses the channel. If the channel is sensed idle, it transmits its voice packet. Otherwise, it persists and transmits its voice packet at the EOC. In that way, a new frame is constructed. The voice station whose *ORDER* value is equal to one is identified as a head station. A head station should sense the channel every F slots after its transmission whether it has a packet or not. When the channel is sensed busy, a data packet is being transmitted on the channel, so it waits until the channel is idle. Thus the frame size is variable in duration between F and $F+1$ slots where the slot size is T. When the head station wants to terminate its communication, it broadcasts a call clear packet. Then the station whose ORDER value is equal to 2 becomes a head station. Every station can know the start of the frame by its timer since the next frame starts at least F slots after the start of the current frame.

C. Data Station Protocol

The data region is the frame time remaining after the voice region and delimited by one idle minislot. The protocol operating in this region is the random token protocol and is used for transmitting both data and callrelated packets. We assume that there are N_d data stations and that each station has a single buffer. Arrivals at a station are assumed to occur according to a Poisson process with rate λ when its buffer is empty.

Let τ be the maximum end-to-end propagation delay in the network and the time is slotted with the duartion of τ. For convenience of analysis, we shall assume that stations are synchronized to slot boundaries. We define a scheduling period(SP) which begins at the end of a transmission to be an interval of time dedicated to channel access resolution. In the scheduling period, each station waits for a period equivalent to a scheduling delay prior to attempting transmission. At the beginning of a scheduling period, each station calculates the scheduling delay k slots where $k \sim$ Uniform$[1 \cdots M-1] \in I$.

1) If, following this delay, the station has a packet for transmission and finds the channel idle, it transmits the packet.

2) If the channel is found busy, the user reschedules its transmission to the next scheduling period(that is, the user waits until the channel is idle and repeats step 2).

3) If, on the other hand, the station senses the channel idle for a period of$(M-1)$ slots following the transmission, the system enters s contention period, no schcduling delay is calculated so that stations at which a paket arrives will transmit at the beginning of the next slot boundary. Since the transmission of a voice packet always begins at EOC, we can say that the scheduling points of voice stations are always 0. For data stations, the scheduling points have the value larger than 0, and therefore, there is at least one idle minislot after EOC for data transmission, which delimits the voice and data regions in a frame. In other words, we can say that voice stations have a priority over data stations. Hence the analysis of data can be done by assuming that two priority classes exist in the system.

Ⅲ. ANALYSIS

A. Voice

We model a speech source by the three-state

Markov process shown in Fig. 2, based on statistical measurements[20]. In the idle state, no conversation takes place, but the source becomes active when its call request packet is successfully broadcast. Then the active source alternates between the talk and the silent states, and the active source becomes idle when its clear request packet is successfully broadcast.

In our analysis, we assume that N_v voice stations are continually active. The time interval spent in the talk and silent states is exponentially distributed with mean $1/\alpha$ and $1/\mu$, respectively. For our model we will use $1/\alpha$ =1.5s and $1/\mu$ =2.25s which are typical values estimated by experiments[7]. Thus the speech activity is $\mu/(\mu+\alpha)$=0.4.

Each voice station has a vocoder which digitizes speech at a rate of R bits/s. Silent periods in speech are suppressed. Each voice station can transmit only one packet in a frame and the duration of a frame is less than the maximum tolerable speech delay which is about 300ms for natural speech. We assume that over the duration of a frame, and active voice station makes at most one transition between its talk and silent states, and therefore we assume that short talkspurts are ignored and short silent periods are filled-in.

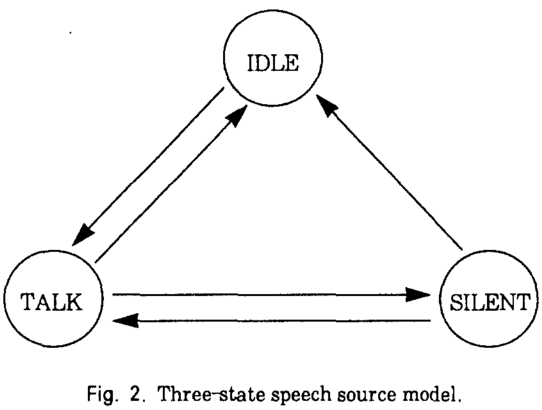


Fig. 2. Three-state speech source model.

Let X be the packet length in bits, W be the channel bandwidth in bits/s, and f the length of a frame in seconds where the actual length of a frame varies between f and $f+X/W$. Then we have

$$X = Rf. \tag{1}$$

A constraint on the choice of X, and, hence, on f is the maximum delay tolerable by speech signals [10]. The voice station whose packets suffer the longest delay is the station of the last slot of the voice region. Actually, the speech samples in that packet were collected at the beginning of the previous frame, which means they have alreay been delayed by f s at the beginning of the current frame. If $K = F$, then the frame can be completely used by voice stations, implying that the voice user using the last slot in a frame waits for a period of f from the start of the frame. Thus, the maximum delay suffered by speech is $2f$, ignoring the propagation delay and the variance of frame length. Therefore, the condition $f \le D_{max}/2$ where D_{max} is the maximum delay tolerable by speech excluding the propagation delay. Then we require that

$$X \le D_{max}R/2. \tag{2}$$

We denote by t_j the instant when frame j is generated by the head station. Let n_j be the number of voice stations that will have packets for transmission in frame j. We assume that all voice stations are independent. Since n_j depends only on n_{j-1}, the sequence $\{n_j, j = 1,2,\cdots\}$ forms an imbedded Markov chain.

Since the voice stations are independent, $\pi_i^{(V)}$ can also be determined [10] by

$$\pi_i^{(V)} = {}_{N_v}C_i\, z^i(1-z)^{N_v-i} \tag{3}$$

where i=0, $\cdots$, $N_{v,i}C_j = \frac{i!}{(i-j)!j!}$, and $z = \mu/(\mu+\alpha)$ is the probability that a voice station is in the talk state at an arbitrary instant.

When the maximum number of voice slots in a frame K is larger than N_v, all the voice packets are successfully transmitted. However, when K is less than N_v, voice clipping at some voice stations occurs if the number of active sations in the talk state is greater than K. Thus we have

$$P_{clipping} = \begin{cases} 0 & \text{if } N_v \le K \\ \sum_{n=K+1}^{N_v} \pi_n^{(V)} & \text{if } N_v > K. \end{cases} \tag{4}$$

When there are n active users in the talk state where $n \le K$, then there are (N_v-n) active users in the silent state, and thus $(N_v-n)\gamma$ minislots are wasted. Thus the fraction of wasted bandwidth of the voice region, W_b, is given by

$$W_b = \begin{cases} \sum_{n=1}^{N_v} \frac{(N_v-n)\gamma}{nT+(N_v-n)\gamma}\pi_n^{(V)} & \text{if } N_v \le K \\ \sum_{n=1}^{K} \frac{(N_v-n)\gamma}{nT+(N_v-n)\gamma}\pi_n^{(V)} & \\ \quad + \sum_{n=K+1}^{N_v} \frac{(N_v-K)\gamma}{KT+(N_v-K)\gamma}\pi_n^{(V)} & \text{if } N_v > K. \end{cases} \tag{5}$$

B. Data

Since the voice region of a frame consists of continuous voice slots, we can regard the voice region as a long data packet, whose arrivals take place regularly. Note that the voice region begins in the contention period or following an EOC, and that there is at least one idle minislot following an EOC for data transmission. Therefore, we can say that the integrated voice/data system is like a system using the prioritized random token protocol where a voice region has priority over data packets. In terms of the random token protocol, a voice region(i.e., a long data packet) has its scheduling point which is equal to 0 with probability one while data stations have scheduling points between 1 and M-1.

We assume that there are N_v active voice stations in the talk state and that there is no voice clipping, i.e., $N_v \le K$. Let T_1 be the length of the voice region. Let N be the number of active voice users in the talk state at the beginning time of a frame. Assume that N=n. Then, we have T_1=$nT+(N_v-n)\gamma$. We consider a finite population model, in which there are one user of voice whose packet length is T_1 and N_d users of data, where the voice user has high priority. Let C be the duration of collisions in minislots for data. Let S_d be the average data channel throughput defined as the fraction of channel time of

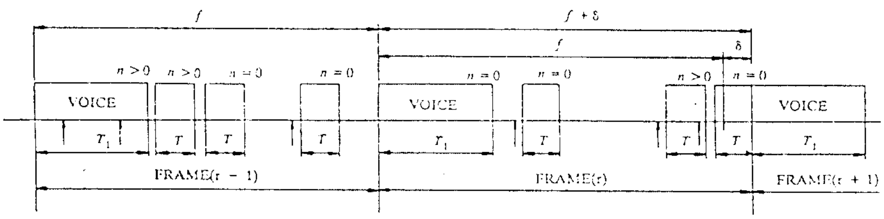


Fig. 3. Activity on the channel showing voice and data subframes.

a successful transmission and D be the average packet delay defined as the time between the arrival of a packet and the reception of it by a destination.

We construct a semi-Markov model by observing the system at the end of a transmission. Let $n(t)$ represent the numbers of backlogged data users in the system at time t. Then we define the state of the system at t_e as $(n(t_e))$ where t_e denotes the time of EOC at which there is no voice user. The activity on the channel is depicted in Fig. 3. Let $\{t_e^{(r)}\}_{r=-\infty}^{\infty}$ be the sequence of EOC's such that there is no voice user in the system at $t_e^{(r)}$. Let **P** be the transition probability matrix for each class. The steady-state distributions $\Pi = \{\pi_0^{(D)}, \pi_1^{(D)}, \cdots, \pi_{N_d}^{(D)}\}$, where $\pi_k^{(D)}$ is the steady-state probability of finding k packets at the end of a transmission, are obtained by solving a set of linear equations

$$\Pi P = \Pi \quad \text{and} \quad \sum_{k=0}^{N_d} \pi_k^{(D)} = 1. \tag{6}$$

In order to get the steady-state distributions Π, we should first obtain the transition probability matrix $P=\{P_{ij}\}$ where P_{ij} is the probability of transition from state i to state j. Before we obtain P, we define some components of P_{ij} and calculate them.

Let F be the frame length in minislots. Then a voice region begins every F minislots, i.e., the voice packet of size T_1 arrives every F minislots. Thus the arrival of voice packets is uniformly distributed between 0 and F. Hence the arrival probability process of a voice packet during x minislots, given that there are i voice packets(i=0, 1), is given by

$$\sigma_i(x) = \left(\frac{x}{F}\right)^i \left(1-\frac{x}{F}\right)^{1-i}, \quad i=0,1. \tag{7}$$

Let $q_{k,i}(x)$ be the probability of k arrivals in x minislots, given that there are i data packets in the system. Using the binomial distribution, we get

$$q_{k,i}(x) = {}_{N_d-i}C_k \cdot (1-e^{-\lambda x})^k \, (e^{-\lambda x})^{N_d-i-k}. \tag{8}$$

Let $\alpha_{i,t}$ be the probability that none of the i packets given at EOC is scheduled at scheduling points $1,\cdots,t-1$, and $\beta_{k,t}$ be the probability that none of the k packet arrivals which occurred during the time interval between the beginning of the scheduling period (=EOC) and t, is scheduled at scheduling points $1,\cdots,t-1$. From[18], $\alpha_{i,t}$ and $\beta_{k,t}$ are given by

$$\alpha_{i,t} = \left(1-\frac{t-1}{M_2}\right)^i, \quad 1 \le t \le M \tag{9}$$

$$\beta_{k,t} = \begin{cases} \left(1-\sum_{m=1}^{t-1}\frac{1-e^{-\lambda m}}{1-e^{-\lambda t}}\cdot\frac{1}{M_2}\right)^k, & 2 \le t \le M-1 \\ 1, & t=1 \end{cases} \tag{10}$$

where $M_2 = M-M_1$.

We also define $\beta_{k,t}'$ as the probability that none of the k packet arrivals is scheduled at scheduling points $1\cdots t$ with the same condition of $\beta_{k,t}$. Then, $\beta_{k,t}'$ is given by

$$\beta_{k,t}' = \left(1-\sum_{m=1}^{t}\frac{1-e^{-\lambda m}}{1-e^{-\lambda t}}\cdot\frac{1}{M_2}\right)^k, \quad 1 \le t \le M-1. \tag{11}$$

With these probabilities, let $P(\mathrm{ESP}=t \mid i,k)$ be the conditional probability that t is the earliest scheduling point(ESP) at which there is at least one packet ready for transmission, given that there were i packets at the beginning of the scheduling period and k arrivals occurred during the time interval between the beginning of the scheduling period and t. From [18], $P(\mathrm{ESP}=t \mid i,k)$ is given by

$$P(\mathrm{ESP}=t \mid i,k) = \alpha_{i,t}\cdot\beta_{k,t} - \alpha_{i,t+1}\cdot\beta_{k,t}', \ 1 \le t \le M-1 \tag{12}$$

We now define the conditional probability of a successful transmission, $S_i(t,k)$, with the following three conditions:

1) There are i packets in the system at the beginning of the scheduling period.
2) During the time interval between the beginning of the scheduling period and t, k packets arrived.
3) The earliest scheduling point at which at least one packet ready for transmission is t.

Then, it is given by

$$S_i(t,k)=\frac{{}_iC_1\cdot\frac{1}{M_2}\cdot\alpha_{i-1,t+1}\cdot\beta_{k,t}{}'+{}_kC_1\cdot\frac{1}{M_2}\cdot\alpha_{i,t+1}\cdot\beta_{k-1,t}{}'}{P(\mathrm{ESP}=t\mid i,k)}, \tag{13}$$

$$1\le t\le M\text{-}1.$$

Finally we define $AR(k_1,k_2)$ as the conditional probability that k_1 and k_2 arrivals of voice and data packets, respectively, occur in the cycle prior to a transmission which is initiated in the contention period. $AR(k_1,k_2)$ is given by

$$AR(k_1,k_2)=\sigma_{k_1}(M-1)\cdot q_{k_2,0}(M-1)\cdot\beta_{k_2,M-1}{}'$$

$$+\sigma_0(M-1)\cdot q_{0,0}(1)\frac{\sigma_{k_1}(1)\cdot q_{k_2,0}(1)}{1-\sigma_0(1)\cdot q_{0,0}(1)}, \tag{14}$$

$$(0\le k_1\le 1,\ 0\le k_2\le N_d).$$

With the above conditional probabilites we can get the transition probability P_{ij}. Consider first the case $n(t_e^{(r)})=i\neq 0$. Note that the imbedded points are EOC's at which there is no voice packet in the system. Thus the EOC at which the transmission of a voice packet ends is not an imbedded point. The time elapsed between $t_e^{(r)}$ and $t_e^{(r+1)}$ consists of a data transmission or data plus voice transmissions as we can see in Fig. 3. Let t_e' be the time of the first EOC following te(r) and L_m the time duration between t_e' and $t_e^{(r+1)}$ given that there is m voice packets(m=0, 1) at t_e', i.e., $L_m=t_e^{(r+1)}-t_e'$. Apparently $L_0=0$ and $L_1=T_1$. Then we have

$$P_{ij}=0,\qquad i\neq 0,\ j<i-1$$

$$\begin{cases}\sum_{k=0}^{N_d-i}\sum_{t=M_1}^{M-1}q_{k,i}(t)P(\mathrm{ESP}=t\mid i,k)\sum_{m=0}^{1}[S_i(t,k)\sigma_m(t+T+1)\\ \cdot q_{j-i-k+1,\,i+k}(T+1+L_m)+(1-S_i(t,k))\cdot\sigma_m(t+C+1)\\ \cdot q_{j-i-k,\,i+k}(C+1+L_m)],\quad i\neq 0,\ j>i\text{-}1.\end{cases} \tag{15}$$

Consider now the case $n(t_e^{(r)})=0$. At $t_e^{(r)}$, a contention period can start. If arrivals of data packets occur before their respective scheduling points during M-1 slots in the scheduling period, a contention period does not begin. Otherwise, a contention begins.

The transition probability that there are j_1 voice packets and j_2 data packets at t_e', given $n(t_e^{(r)})=0$, is given by

$$P_{j_1}[n(t_e')=j_2\mid n(t_e^{(r)})=0$$
$$=\sum_{k=0}^{j_2}\sum_{t=M_1}^{M-1}q_{k,0}(t)P(\mathrm{ESP}=t\mid 0,k)[S_0(t,k)$$
$$\cdot q_{j_2-k+1,k}(T+1)\cdot\sigma_{j_1}(t+T+1)+(1$$
$$-S_0(t,k))\cdot q_{j_2-k,k}(C+1)\cdot\sigma_{j_1}(t+C+1)]$$
$$+AR(1,0)\cdot q_{j_2,0}(T_1+1)\cdot\delta_{j_1,0}+AR(0,1)$$
$$\cdot\sigma_{j_1}(T+1)\cdot q_{j_2,1}(T+1)$$
$$+\sum_{k=2}^{N_d}AR(0,k)\cdot\sigma_{j_1}(C+1)\cdot q_{j_2-k,k}(C+1)$$
$$+\sum_{k=1}^{N_d}AR(1,k)\cdot q_{j_2-k,k}(C+1)\cdot\delta_{j_1,0},\quad 0\le j_1\le 1,\ 0\le j_2\le N_d. \tag{16}$$

Thus P_{0j} is given by

$$P_{0j}=P_0[n(t_e')=j\mid n(t_e^{(r)})=0] \tag{17}$$
$$+\sum_{k=0}^{j}P_1[n(t_e')=k\mid n(t_e^{(r)})=0]\cdot q_{j-k,k}(T_1+1),$$
$$0\le j\le N_d.$$

Thus, the transition P consists of [15] and [17].

Now we get the throughput and the delay. Since we have assumed that there are N=n

active voice users in the talk state at the beginning of a frame, we get the conditional throughput $S_d(n)$ and delay $D(n)$. Let $\bar{L}$ and P_s denote the average cycle length and the probabilty of a successful transmission, respectively. Then, from the theory of regenerative process, $S_d(n)$ is calculated as

$$S_d(n) = \frac{T \cdot P_s}{\bar{L}}. \tag{18}$$

Also, let $\bar{B}$ and $\bar{N}_d$ denote the expected sum of the backlog over all minislots in a cycle and the average channel backlog, respectively. Then we have

$$\bar{N}_d = \frac{\bar{B}}{\bar{L}}. \tag{19}$$

Then we can obtain $D(n)$ from Little's formula as

$$D(n) = \frac{\bar{N}_d}{S_d(n)}. \tag{20}$$

In order to get $S_d(n)$, we should obtain $\bar{L}$ and P_s. Let L_i denote the length of a cycle beginning with i packets in the system. For i〉0, L_i is given by

$$L_i = \sum_{k=0}^{N_2-i}\sum_{t=M_1}^{M-1} q_{k,i}(t) P(\mathrm{ESP}=t \mid i,k) \sum_{m=0}^{1} [S_i(t,k) \cdot \sigma_m(t+T+1) \cdot (t+T+1+L_m) + (1-S_i(t,k)) \cdot \sigma_m(t+C+1) \cdot (t+C+1+L_m)]. \tag{21}$$

For i=0, we let $L_0=R_1+R_2$ where R_1 gives the average cycle length if the contention period does not begin. R_2 gives the average cycle length when a contention period starts. The calculations of R_1 and R_2 are given in Appendix. Then the average cycle length is given by

$$\bar{L} = \sum_{i=0}^{N_d} \pi_i^{(D)} \cdot L_i. \tag{22}$$

The probability of a successful transmission is given by

$$P_s = \sum_{i=0}^{N_d} \pi_i^{(D)} [\sum_{k=0}^{N_d-i}\sum_{t=M_1}^{M-1} q_{k,i}(t) P(\mathrm{ESP}=t \mid i,k) \cdot S_i(t,k)] + \pi_0^{(D)} \cdot AR(0,1) \tag{23}$$

Consequently, the conditional throughput is obtained by substituting (22) and (23) in (18).

To compute $\bar{B}$ we define $Q_k(x)$ be the expected sum of backlogs over all slots in a period of x slots, given that there are k packets in the system at the beginning of that period. It is given by

$$Q_k(x) \triangleq \sum_{i=0}^{N_d-k}\sum_{t=0}^{x} (k+i) \cdot q_{i,k}(t). \tag{24}$$

Using $Q_k(x)$ and decomposing into states i=0 and i〉o, we obtain

$$\begin{aligned}\bar{B} = &\sum_{i=0}^{N_d} \pi_i^{(D)} \{ \sum_{k=0}^{N_d-i}\sum_{t=M_1}^{M-1} q_{k,i}(t) P(\mathrm{ESP}=t \mid i,k) \cdot (k+i \cdot t \\ &+ \sum_{m=0}^{1} [S_i(t,k) \cdot \sigma_m(t+T+1) \cdot Q_{k+i}(T+1+L_m) \\ &+ (1-S_i(t,k)) \cdot \sigma_m(t+C+1) \cdot Q_{k+i}(C+1+L_m)]) \} \\ &+ \pi_0^{(D)} [AR(1,0) \cdot Q_0(T_1+1) + AR(0,1) \sum_{m=0}^{1} \sigma_m(T+1) \\ &\cdot Q_1(T+1+L_m) + \sum_{k=2}^{N_d} AR(0,k) \sum_{m=0}^{1} \sigma_m(C+1) \cdot Q_k(C \\ &+1+L_m) \\ &+ \sum_{k=1}^{N_d} AR(1,k) \cdot Q_k(C+1+L_1)].\end{aligned} \tag{25}$$

Thus we can obtain the average channel backlog $\bar{N}_d$, and thus the average packet delay from [20]. Removing the condition on n, we have the throughput and the delay as follows.

$$S_d = \sum_{n=0}^{N_v} S_d(n)\pi_n^{(V)} \quad (26)$$

and

$$D = \sum_{n=0}^{N_v} D(n)\pi_n^{(V)}. \quad (27)$$

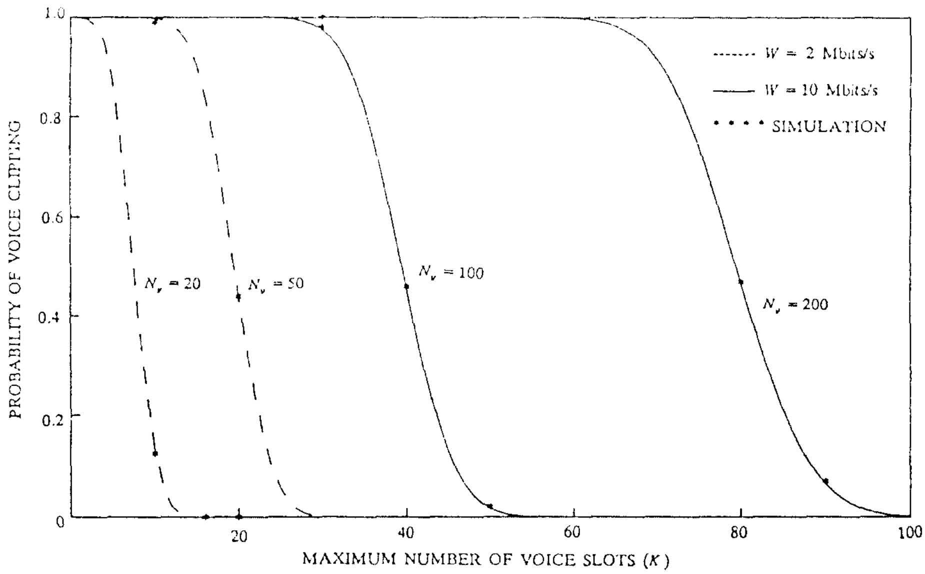


Fig. 4. Probability of voice clipping with various values of K (R=64 kbits/s).

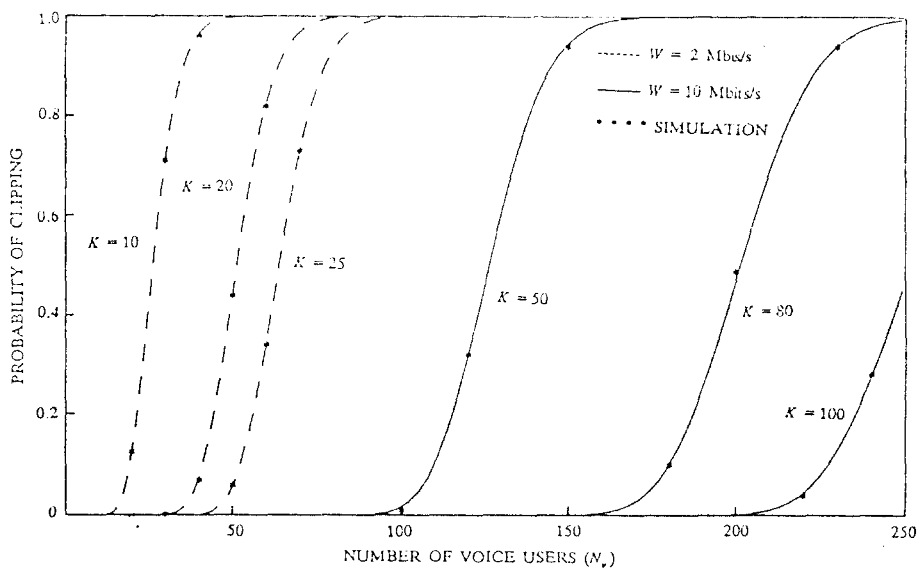


Fig. 5. Probability of voice clipping for various number of voice users (R=64 kbits/s).

Ⅳ. NUMERICAL RESULTS

In this section, we present numerical results to illustrate the characteristics of our proposed voice/data integration protocol with various system parameters. For voice performance, we get the clipping probability and the fraction of wasted bandwidth in the voice region. For data performance, we get the delay-throughput characteristics for various configurations. We also present simulation results to verify our analysis.

We choose D_{max}=100 ms and R=64 kbits/s. Then we get $X \leq 3200$ bits from [2]. Thus, we let X=1000 bits. Then we have f=15.62 ms from [1]. Since the packet transmission time is $T=X/W$=0.5 ms if we have W=2 Mbits/s, then we get 31 slots of size T in a frame. For W=10 Mbits/s, we have 156 slots in a frame. In Fig. 4, we show the clipping probability as a function of K. For W=2 Mbits/s, we have N_v=20 and 30, and for W=10 Mbits/s, we have N_v=100 and 200. We can see that the clipping probability is a decreasing function of K as expected. It is obvious that the more voice slots exist, the less clipping occurs.

In Fig. 5, we show the clipping probability as a function of N_v. For W=2 Mbits/s, we set K=10, 20, and 25, and for W=10 Mbits/s, we set K=50, 80, and 100. We can see that the more active users exist, the more clipping occurs. For the quality of voice signal, we require that $P_{clipping} \leq 0.005$. For $P_{clipping}$=0.005, we have N_v=20, 40, and 50, with K=10, 20, and 25, respectively, for W=2 Mbits/s. Thus, we can say that the number of active users for the good quality of voice signal is about twice the maximum number of voice slots in a frame. Therefore, $2K$ calls can be accepted for good voice communication. We can see this is also true for W=10 Mbits/s.

In Fig. 6, we show the fraction of wasted bandwidth in a voice region as a function of N_v. In our protocol, the stations in the silent

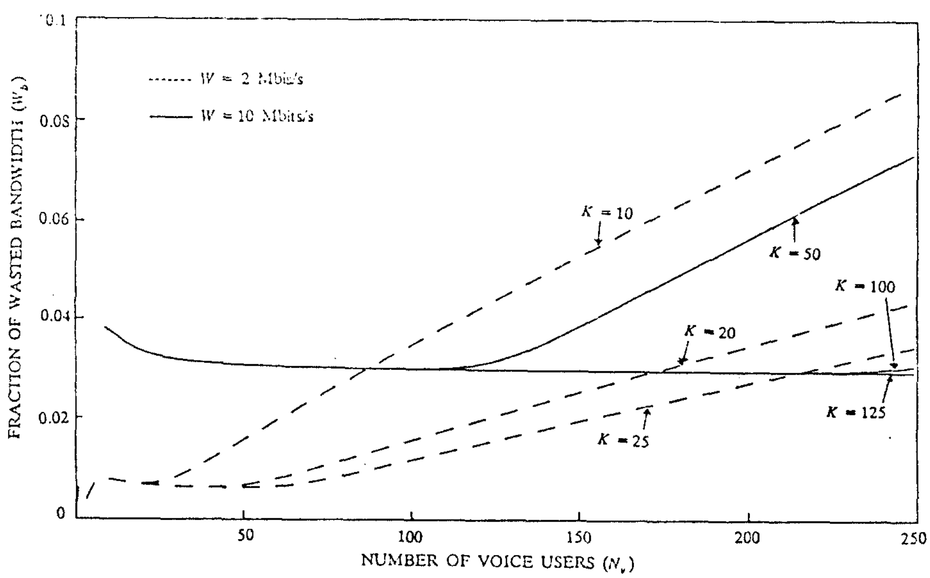


Fig. 6. Fraction of wasted bandwidth of voice region (R=64 kbits/s).

state transmit burst carriers for γ minislots to maintain the synchronization between active voice stations. Thus, some bandwidth is wasted due to this burst carrier. We set K=10, 20, and 50 for W=2 Mbits/s, and set K=50, 100, and 125 for W=10 Mbits/s. We can see that W_b slightly increases as N_v increases when N_v is less than $2K$. However, W_b is an increasing function of N_v when $N_v \geq 2K$. This is because the more stations send burst carriers when $N_v \geq 2K$. We can also see that the wasted bandwidth is below 2 percent for W=2 Mbits/s and below 4 percent for W=10 Mbits/s when we have N_v=$2K$. Thus, we can say that as more stations become active for communication, the more bandwidth is wasted and the more clipping occurs. Therefore, there exists an optimal number of active voice stations for a given system configuration, and we can say that the optimal number of active voice stations is about $2K$.

We now consider the data performance. In Fig. 7, we plot the average delay of data packets versus the total throughput(S_{v+d}) with N_v=20, N_d=10, W=2 Mbits/s, K=10, and M= 3, 6, 10, and, 12. We find that 20 voice stations consume about 25 percent of the channel capacity. From light to mid loads, the data delay is less than 3 slots. We also find that the maximum throughput that can be achieved is almost unity. We can see that there is little difference among four curves in Fig. 7 although the value of M=3 gives the best performance. Thus, we can say that the performance is less sensitive to M in the voice/data integration protocol than in the random token protocol.

In Fig. 8, we plot the average delay of data packets versus the total throughput(S_{v+d}) with N_d=10, W=2 Mbits/s, K=30, M=3, and N_v=2, 20, 30, and 40. We can see that the performance becomes better as the number of active voice stations decreases. For very small value of N_v, it is almost like the case where only

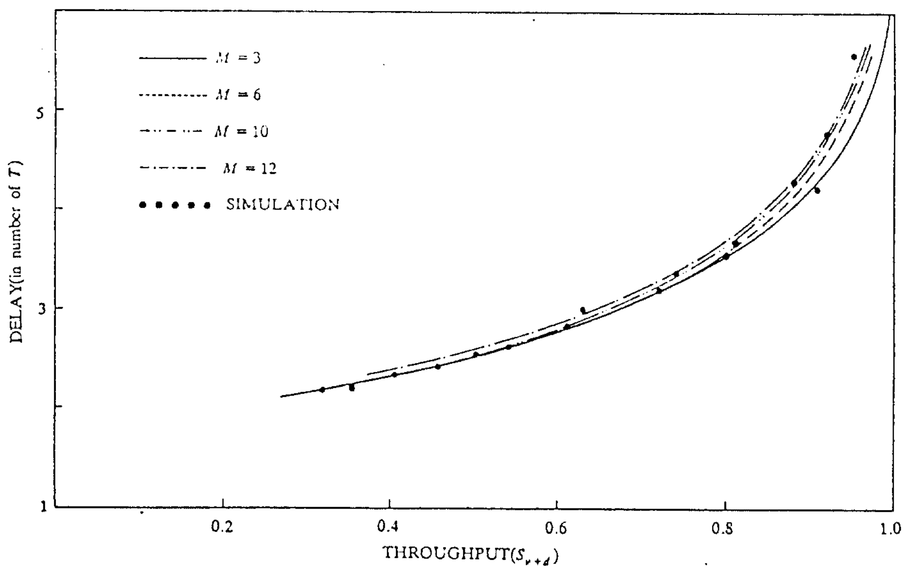


Fig. 7. Average delay versus total throughput (S_{v+d}) with various of M (R=64 Kbits/s, W=2 Mbits/s, N_v=20, N_d=10, K=10).

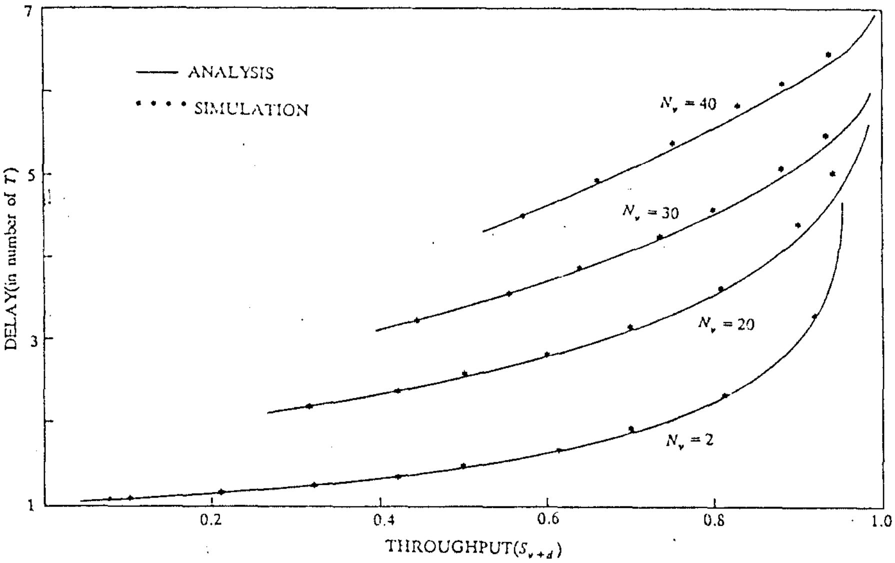


Fig. 8. Average delay versus total throughput(S_{v+d}) with various values of N_v (R=64 Kbits/s, W=2 Mbits/s, N_d=10, K=30, M=3).

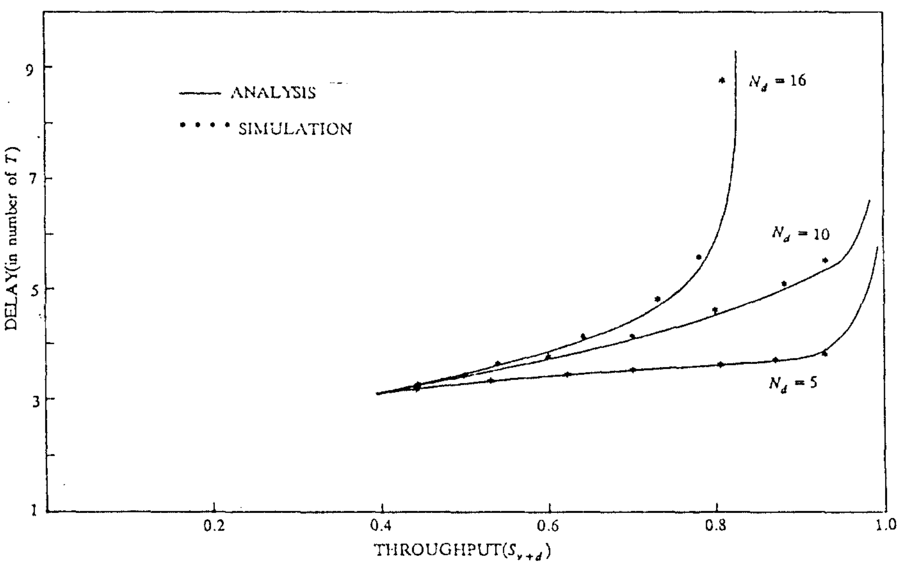


Fig. 9. Average delay versus total throughput(S_{v+d}) with various values of N_v (R=64 Kbits/s, W=2 Mbits/s, W=2Mbits/s, N_v=30, K=20, M=3).

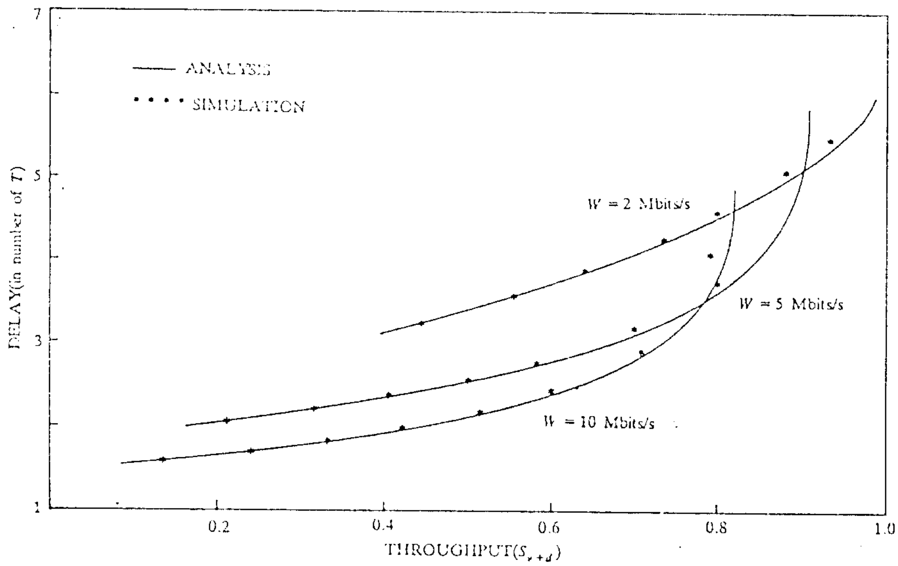


Fig. 10. Average delay versus total throughput(S_{v+d}) with various values of W (R=64 Kbits/s, N_v=30, N_d=10, K=30, M=3).

data users are in the network. As N_v increases, the voice throughput increases, and thus the data delay increases. The reason why the starting point of each curve in Fig. 8 is different is that the voice throughput is different from each other.

In Fig.9, we present the average delay of data packets versus the total throughput(S_{v+d}) with N_v=30, W=2 Mbits/s, K=20, M=3, and N_d=5, 10, and 16. We can see that the performance becomes better as less data users are in the network. Finally, in Fig. 10, we show the average delay of data packets versus the total throughput(S_{v+d}) to illustrate the effect of changing the channel bandwidth. We have N_v=30, N_d=10, K=30, M=3, and W=2, 5, and 10 Mbits/s. We find that as W is increased, the fraction of the channel bandwidth used by voice drops since the number of voice calls is fixed. With increasing W, we also find that the data delay decreases since the data region becomes longer. Hovever, we can see that the maximum achievable throughput decreases as the channel bandwidth increases. This is characteristics of a broadcast bus network.

Ⅴ. CONCLUSIONS

We have proposed and studied a voice/data integration protocol based on the random token protocol. We used a framed architecture with variable length and a movable boundary scheme. A TDMA-like service is provided for voice traffic while data traffic is served via the random token protocol. By a suitable choice of the maximum size of a voice region in a frame, we found that the fraction of speech loss can be maintained under a specified maximum. We also determined the fraction of wasted bandwidth due to the protocol and found that the channel could be operated at a throughput close to unity. For data performance, we obtained the delay throughput characteristics analytically for various system parameters. Further, we gave simulation results for validation. We found that the protocol is robust and fair to data packets, requires little overhead to implement, and the voice performance is not affected by data traffic. The protocol also incorporates minimal network information and requires only limited synchronization like random access schemes, and therefore it is suitable for local area networks and mobile radio networks.

APPENDIX

Here, we calculate the components of the average cycle length, R_1 and R_2, in the case that the cycle begins with zero numer of packets of each respective class.

$$R_1 = \sum_{k=0}^{N_d} \sum_{t=M_1}^{M-1} q_{k,0}(t) \cdot P(\mathrm{ESP}=t \mid 0,k) \cdot \sum_{m=0}^{1} [S_0(t,k) \cdot \sigma_m(t+T+1) \cdot (t+T+1+L_m) + (1-S_0(t,k)) \cdot \sigma_m(t+C+1) \cdot (t+C+1+L_m)] \qquad (A.1)$$

$$R_2 = \sigma_0(M-1) \cdot q_{0,0}(M-1) [M-1+\frac{\sigma_0(1) \cdot q_{0,0}(1)}{1-\sigma_0(1) \cdot q_{0,0}(1)} + \frac{\sigma_1(1) \cdot q_{0,0}(1)}{1-\sigma_0(1) \cdot q_{0,0}(1)} (T_1+1) + \frac{\sigma_0(1) \cdot q_{1,0}(1)}{1-\sigma_0(1) \cdot q_{0,0}(1)} \sum_{m=0}^{1} \sigma_m(T+1) \cdot (T+1+L_m) + \frac{\sigma_0(1) \cdot (1-q_{0,0}(1)-q_{1,0}(1))}{1-\sigma_0(1) \cdot q_{0,0}(1)} \sum_{m=0}^{1} \sigma_m(C+1) \cdot (C+1+L_m) + \frac{\sigma_0(1) \cdot (1-q_{0,0}(1))}{1-\sigma_0(1) \cdot q_{0,0}(1)} (C+1+L_1)] + q_{0,0}(M-1) \cdot \sigma_1(M-1) \cdot (M+T_1) + \sigma_0(M-1)[q_{1,0}(M-1) \cdot \beta_{1,M} \cdot \sum_{m=0}^{1} \sigma_m(T+1) \cdot (M+T+L_m) + \sum_{k=2}^{N_d} q_{k,0}(M-1) \cdot \beta_{k,M} \cdot \sum_{m=0}^{1} \sigma_m(C+1) \cdot (M+C+L_m)] + \sigma_1(M-1) \cdot \sum_{k=1}^{N_d} q_{k,0}(M-1) \cdot \beta_{k,M} \cdot (M+C+L_1) \qquad (A.2)$$

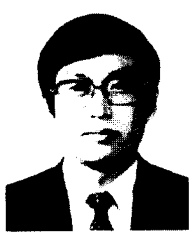
REFERENCES

1. I. Chlamtac, "An Ethernet compatible protocol for real-time voice/data integration," *Computer Networks and ISDN Systems*, vol. 10, pp.81-96, 1985.
2. S. M. Sharrock, S. Ghanta, H. C. Du, and K. J. Maly, "A CSMA/CD-based, integrated voice/data protocol with dynamic channel allocation," *Computer Networks and ISDN Systems*, vol. 18, pp.1-18, 1989/90.
3. N. F. Maxemchuck, "A variation on CSMA/CD that yields movable TDM slots in integrated voice data local networks," *Bell Syst. Tech J.*, vol. 61, pp.1527-1550, Sept. 1982.
4. J. S. Meditch and Y. Zhao, "Framed TDMA/CSMA for integrated voice-data local area networks," in *Proc. INFOCOM' 85*, pp.10-17, 1985.
5. S. Q. Li and J. C. Majithia, "Performance analysis of a DTDMA local area network for voice and data," *Computer Networks and ISDN Systems*, vol. 8, pp.81-91, 1984.
6. O. C. Ibe and D. T. Gibson, "Protocols for integrated voice and data local area networks," *IEEE Commun. Magazine*, vol. 24, pp.30-36, July. 1986.
7. J. O. Limb and L. E. Flamm, "A distributed local area network packet protocol for combined voice and data transmission," *IEEE J. Select. Areas Commun.*, vol. SAC-1, pp.926-934, Nov. 1983.
8. M. Fine and F. A. Tobagi, "Packet voice on a local area network with round robin service," *IEEE Trans. Commun.*, vol. COM-34, pp.906-

915, Sept. 1986.
9. F. Tobagi, F. Borogonovo, and L. Fratta, "Expressnet: A high-performance integrated-services local area network," *IEEE J. Select. Areas Commun.*, vol. SAC-1, pp.898-912, Nov.1983.
10. B. Mukherjee and J. S. Meditch, "Integrating voice with the *Pi*-persistent protocol for unidirectional broadcast bus networks," *IEEE Trans. Commun.*, vol. COM-36, pp.1287-1295, Dec. 1988.
11. A. Patir et al., "An optical fiber based integrated LAN for MAGNET's testbed environment," *IEEE J. Select. Areas Commun.*, vol. SAC-3, pp.872-881, Nov. 1985.
12. S. Q. Li and M. E. Zarki, "Dynamic bandwidth allocation on a slotted ring with integrated services," *IEEE Trans. Commun.*, vol. COM-35, pp.826-833, July. 1988.
13. T. Minami et al., "A 200Mbits/s synchronous TDM loop optical LAN suitable for multiservice integration," *IEEE J. Select. Areas Commun.*, vol. SAC-3, pp.849-858, Nov. 1985.
14. J. R. Brandsma, A. A. Bruekers, and J. L. W. Kessels, "PHILAN: A fiber-optic ring for voice and data," *IEEE Commun. Magazine*, vol. 24, pp.16-22, Dec. 1986.
15. L. M. Casey, R. C. Dittburner, and N. D. Gammage, "FXNET : A backbone ring for voice and data," *IEEE Commun. Magazine*, vol. 24, pp.23-28, Dec. 1986.
16. F. A. Tobagi and V. B. Hunt, "Performance of carrier sense multiple access with collision detection," *Computer Networks*, vol. 4, pp.245-259, 1980.
17. G. J. Nutt and D. L. Bayer, "Performance of CSMA/CD networks under combined voice and data loads," *IEEE Trans. Commun.*, vol. COM-30, pp.6-11, Jan. 1982.
18. I. Chlamtac and A. Ganz, "Evaluation of the random token protocol for high-speed and radio networks," *IEEE J. Select. Areas Commun.*, vol. SAC-5, pp.969-976, July. 1987.
19. F. A. Tobagi, "Carrier sense multiple access with message-based priority functions," *IEEE Trans. Commun.*, vol. COM-30, pp.185-200, Jan. 1982.
20. H. B. Kekre and C. L. Saxena, "Three-state Markov model of speech on telephone lines and optimal utilization of communication systems using TASI technique," *Comut. Elec. Eng.*, vol. 4, pp.235-250, Sept. 1977.

박 동 수(Dong Soo Park) 정회원

1985년 2월 : 서울대학교 전자공학과, 공학사
1990년 8월 : 한국과학기술원 전기 및 전자공학과, 공학박사
1987~1994년 6월 : 디지콤 정보통신 연구소
1994년 6월~현재 : 삼성전자 선임연구원



殷 鍾 官(Chong Kwan Un) 정회원

1964년 6월 : Univ. of Delaware 공학사
1966년 6월 : 동대학원, 공학석사
1969년 6월 : 동대학원, 공학박사
1964년~1969년 : Univ. of Delaware, Rcsearch Fellow
1969~1973년 : Univ. of Maine, 전자공학 조교수
1973년~1977년 : Stanford 연구소, 책임연구원
1982년 1월~1983년 6월 : 한국과학기술원 공학장
1983년 7월~1989년 6월 : 한국과학기술원 통신공학연구실장
1977년 7월~현재 : 한국과학기술원 전기 및 전자공학과 교수
IEEE Fellow, 뉴욕 학술원 정회원 한국과학기술 한림원 종신회원