A new protocol for the integration of Voice and Data over PRMA

Parthasarathy Narasimhan & Roy D. Yates
WINLAB
Rutgers University
PO Box 909
Piscataway NJ 08855
email: *narasim@gandalf.rutgers.edu*
ryates@ece.rutgers.edu

October 20, 1995

Abstract

A new scheme for the integration of voice and data based on PRMA is proposed. The voice and the data subsystems are logically separated. The total available bandwidth is divided into three regions – voice information, voice contention, and data regions. The available bandwidth is dynamically partitioned between the above three regions subject to the fulfillment of the quality of service (QoS) requirements of the voice users. The voice subsystem has been modeled as a Markov Chain and an exact analytical method to compute the voice packet dropping probability is described. A non-linear programming problem is formulated to optimize the bandwidth allocated for the data users. Solutions to this non–linear programming problem that are very close to optimum have been obtained heuristically. Numerical results indicate that a significant amount of data traffic can be supported without sacrificing the voice capacity of the system.

1 Introduction

Goodman et al. introduced Packet Reservation Multiple Access (PRMA) [1], a statistical multiplexer of speech for wireless communication systems. PRMA exploits the silence periods in speech to accommodate more users than there are channels available. A speech activity detector classifies speech into talkspurts and silence periods. Unlike TDMA where the terminals are granted access to a channel (in the form of a time slot in a TDMA frame) for the entire duration of a call, PRMA voice terminals are granted access to the channel only during talkspurts and give up the use of the channel during silence periods. When a terminal moves from a silence period (state SIL in Figure 1) to a talkspurt, it has to contend for access to the channel. It does so by transmitting its oldest speech packet in a free time slot with a permission probability p. If no other terminal transmits in that time slot, then the contention is termed successful and the terminal get exclusive use of this time slot for the rest of its talkspurt. Congestion leads to access delays and since speech packets require prompt delivery, those that are delayed beyond a certain time, D_{max}, are discarded or dropped. Dropped packets affect speech quality and hence packet dropping probability is an important measure of PRMA performance. A packet dropping probability of 1% is considered acceptable [2] and we define the capacity of a PRMA system, $M_{0.01}$, to be the maximum number of voice terminals that can be supported so that the packet dropping probability is below 1%. An equilibrium point analysis of the performance of a voice–only PRMA system by Nanda et al. can be found in [2].

In PRMA, the base station broadcasts an acknowledgment message at the end of a time slot indicating whether it was able to decode correctly the packet transmitted in that time slot. If the packet was decoded correctly, then the base identifies the terminal that transmitted it. Otherwise, a NULL acknowledgment is transmitted. The mobile terminals listen to this acknowledgment message to decide on future courses of action. So, in PRMA, the receiver in the mobile terminal is required to listen continuously to monitor the acknowledgment messages broadcasted at the end of every time slot.

In order to reduce receiver activity so as to save battery life and to enable mobile–assisted handoffs [3], a variant of PRMA, called Frame Reservation Multiple Access (FRMA) was proposed in [4, 5]. In FRMA, the base broadcasts an acknowledgment once every frame. Mobile terminals wait until the end of a frame to get feedback on their actions in that frame. This approach also blends well with time–division duplexing (TDD) of uplink and downlink transmissions. An exact analysis of the performance of a voice–only FRMA system, along with the optimization of design parameters can be found in [4, 5].

Previous studies [6–9] on the integration of voice and data in PRMA have let the voice terminals and the data terminals contend for all the free slots. Nanda [6] presented an analysis of a PRMA system with integrated voice and data transmission using equilibrium point analysis. Each slot in a frame was classified either as reserved by a voice terminal or as an available slot. A contending voice terminal could contend for every available slot with voice permission probability p_v and upon successful contention, obtain a reservation for a time slot in every succeeding frame until the end of its talkspurt. A data terminal could also contend for every available slot with data permission probability p_d. But a data terminal that contended successfully did not obtain a reservation for that time slot. Instead, if a data terminal had more packets to transmit, it returned to the contending state. Otherwise, the data terminal departed from the system. In order to give the voice users priority over delay insensitive data users, p_d was chosen to be less than p_v. The system was modeled using two Markov Chains corresponding to the voice and data transmission subsystems respectively. Catastrophe theory was then used to find safe regions of operation to avoid instability. This study reported a decrease in the voice capacity of the system with the addition of data traffic.

Wong and Goodman [7] propose a variant of this scheme where the data sources generate bursts of one or more data packets. Data packets that contend successfully also reserve the next $R_d = \min(B, K)$ free slots to transmit their burst, where B is the length of their burst in packets, and K is a limit enforced by the system in order to prevent long bursts from degrading the performance of the other users. This improves the performance of the data users and the throughput of the system compared to Nanda's scheme described above.

Wu et al. [8] present a Markov analysis of Nanda's system [6]. To make the problem tractable, it is assumed that state transitions occur between the end of one frame and the beginning of the next. For the embedded Markov Chain describing this system, the transition probabilities were derived using an iterative technique. Jangi and Merakos [9] present another analytical method to analyze Reservation Random Access (RRA) protocols for Wireless Information networks which includes PRMA, among other RRA protocols. They have assumed that the transitions from silence to talkspurts and vice–versa occur at the frame boundaries and present a general Markov analysis of RRA protocols. Also, an iterative computational procedure based on matrix decomposition techniques is described to overcome high computational costs for large systems. They also present results of an integrated system similar to Nanda's [6] where the data users contend along with the voice users but with a different permission probability.

In this paper, we present an approach to the integration of voice and data sources in a FRMA system that allocates additional bandwidth for data users without sacrificing the capacity of the system to support voice users. For voice users, the QoS requirement of a packet dropping probability of at most 1% for the voice subsystem is maintained. In this protocol, the contention process of the voice terminals is independent of the data traffic. So any instability in the data subsystem does not affect the voice traffic. Priority to voice terminals is implicit but limited to the extent that the quality of service (QoS) requirements of the voice traffic are satisfied.

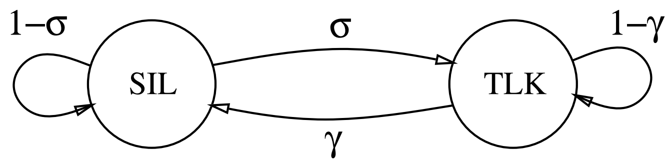

Figure 1: Two–state Markov model of speech activity

The rest of the paper is divided as follows. Section 2 describes the new scheme in detail along with the voice source model. Section 3 provides the analysis of the performance of the voice terminals. In Section 4, an optimization problem for the dynamic bandwidth partition is formulated. A solution to this problem would maximize the bandwidth allocated for the data terminals subject to keeping the voice packet dropping probability to at most 1%. Numerical results on the optimization of bandwidth allocations are provided in Section 5. In Section 6, a simple data subsystem is embedded in the data users' partition. Simulation is used to make a performance comparisons with results in [6,7]. Conclusions are presented in Section 7.

2 Description of the integrated FRMA system

The input to a voice terminal follows an alternating pattern of talkspurts and silence periods, as classified by a speech activity detector. The output of the speech detector is modeled as a two–state Markov process [10] as depicted in Figure 1. We assume the frame duration is sufficiently short that the probability of more than one transition (from TLK to SIL or vice–versa) is negligible. The probability that a talkspurt with mean duration t_1 ends in a frame of duration T is

$$\gamma = 1 - e^{-T/t_1} \tag{1}$$

That is, γ is the probability that a terminal in a talkspurt (state TLK) makes a transition to the silent state (SIL) in any given frame. Correspondingly, the probability that a silent gap, of mean duration t_2, ends during a frame is

$$\sigma = 1 - e^{-T/t_2} \tag{2}$$

For the voice–data FRMA protocol, the entire bandwidth available for the system will be divided into three regions, namely, the voice reservation region, the voice contention region, and the data region. The boundaries between these regions will be moved, dynamically, from frame to frame, in order to keep the voice packet dropping probability below

1%. There are no restrictions imposed either on characteristics of the data sources or the protocol that the data users should be using. Whatever the nature of the data subsystem, we believe the greater the bandwidth allocated to the data subsystem, the better its performance would be. The protocol for the FRMA system that supports both voice and data sources is described below.

- At the beginning of every frame, the base classifies every time slot as either
  - reserved by a voice terminal in which case the identity of the voice terminal that holds a reservation for this time slot is also broadcasted, or
  - available for voice contention in which case all voice terminals that are in a talkspurt but do not have a reserved slot yet can contend for these slots, or
  - a data slot for use of the data traffic.
- A voice terminal holding a reserved slot in a frame continues to hold its reservation by transmitting a packet in its reserved time slot. If the talkspurt ended in the previous frame, the terminal releases its reservation by not transmitting in its reserved slot.
- A voice terminal without a reservation, but with a packet to transmit is said to be in the contending state. Each contending voice terminal independently chooses to contend in each voice contention slot with a permission probability p that is broadcast by the base station once every frame. If exactly one terminal transmits a packet, then this contention is termed successful. The successful contending terminal is given a reserved time slot (if such a slot is available) for its exclusive use until the end of its talkspurt.
- If a voice terminal successfully contends for more than one voice contention slot in the frame, the base reserves one of the time slots to this terminal in an arbitrary manner. Thus, at the end of a frame, each voice terminal can hold a reservation for at most one time slot.

Voice terminals in the silent state that begin a talkspurt in the middle of a frame wait until the beginning of the next frame before contending for a reservation. A contending voice terminal that fails to obtain a reservation in a frame drops the oldest packet from its buffer. That is, a speech packet that is not transmitted successfully in a frame is dropped at the end of the frame. Data terminals do not get access to either the reserved slots or the voice contention slots. That is, we assume that an appropriate protocol for data users will be implemented using the remaining slots in each frame. Finally, no capture phenomenon is assumed, the channel is assumed to be error–free, and transmission delays are neglected.

3 Analysis of the voice subsystem

The analysis of the voice subsystem is a generalization of the analysis of the voice–only system presented in [4,5]. In the voice–only system, a slot that is not reserved by any voice terminal is available for contention. Hence, if the number of reserved slots at the beginning of a frame is R, then the number of voice contention slots is $N - R$. By comparison, in the integrated voice–data system, we control the number of voice contention slots from frame to frame to accommodate slots for the data users while maintaining the voice packet dropping probability below 1%.

Voice calls arrive into the system in a Poisson manner with an arrival rate of λ_v calls per second. The call holding times are distributed as i.i.d. exponential random variables with a mean call duration of $1/\mu_v$ seconds. We assume that the frame duration is so short compared to the inter–arrival time of voice calls that there can be at most one call arrival per frame. Hence the probability that a call arrives in a given frame is

$$\gamma_v = 1 - e^{-\lambda_v T} \tag{3}$$

The probability that a voice call will terminate in a given frame is

$$\sigma_v = 1 - e^{-\mu_v T} \tag{4}$$

Assume that all incoming calls are blocked if the number of voice calls exceed $M_{0.01}$ defined above.

At the beginning of a frame, let M be the number of voice calls in the system. Let R be the number of slots in the frame that are reserved by voice terminals. Note that the latter quantity is not the same as the number of voice terminals that are in a talkspurt since there may be some terminals that are in a talkspurt but do not yet have a reserved slot for their use. Let the total number of slots per frame be denoted by N, which is a constant.

Ideally, at the beginning of every frame, the number of voice contention slots and the permission probability for the contending voice terminals should be determined as a function of the system state. But at the beginning of every frame, C, the number of contending voice terminals is unknown. Hence, given the observable system state variables R and M at the beginning of each frame, the base station must choose the number of voice contention slots,

$$v = \mathbf{v}(R, M) \tag{5}$$

and the permission probability,

$$p = \mathbf{p}(R, M) \tag{6}$$

that the contending voice terminals will use in the frame. We will use a pair of vectors $(\mathbf{v}, \mathbf{p})$ to denote a voice contention policy that specifies the number of contention slots v and permission probability p for every R and M.

We now show that the voice subsystem can be modeled as a Markov chain with the state space described by (R, C, M) where at the beginning of a frame, R represents the number of slots reserved for voice terminals , C represents the number of voice terminals in the contending state, and M represents the number of voice calls in the system. We observe that a feasible state (R, C, M) must satisfy

$$0 \leq \quad R \quad \leq \min(M, N) \tag{7}$$

$$0 \leq \quad C \quad \leq M - R \tag{8}$$

$$0 \leq \quad M \quad \leq M_{0.01} \tag{9}$$

We will use S to denote the set of feasible states (R, C, M).

We assume that state transitions occur between the end of one frame and the beginning of the next. At the beginning of the n-th frame, let $R^{(n)}$ be the number of slots reserved for voice terminals, $C^{(n)}$ be the number of contending voice terminals, and $M^{(n)}$ be the number of voice calls in the system. Then,

$$\begin{aligned} R^{(n+1)} &= R^{(n)} + N_{CR}^{(n)} - N_{RI}^{(n)} \\ C^{(n+1)} &= C^{(n)} - N_{CR}^{(n)} - N_{CI}^{(n)} + N_{IC}^{(n)} \\ M^{(n+1)} &= \min(M^{(n)} + A^{(n)} - D^{(n)}, M_{0.01}) \end{aligned} \tag{10}$$

where

$N_{CR}^{(n)}$ is the number of contending voice terminals that contend successfully and hence obtain a reservation at the end of the n-th frame,

$N_{RI}^{(n)}$ is the number of voice terminals with reservations that complete their talkspurts during the n-th frame and hence move to the idle (SIL) state. This random variable is distributed as Binomial$(R^{(n)}, \gamma)$,

$N_{CI}^{(n)}$ is the number of contending voice terminals that complete their talkspurts during the n-th frame and move to the idle (SIL) state. This random variable is distributed as Binomial$(C^{(n)} - N_{CR}^{(n)}, \gamma)$,

$N_{IC}^{(n)}$ is the number of silent voice terminals that begin a talkspurt during the n-th frame and move to the contending state. This random variable is distributed as Binomial$(M^{(n)} - R^{(n)} - C^{(n)}, \sigma)$,

$A^{(n)}$ is the number of new voice calls arriving into the system during the n-th frame. We assume that the duration of a frame is so short compared to the duration of the inter arrival time of voice calls there can be at most one arrival of a voice call in a frame. So this random variable is a Bernoulli trial with parameter γ_v.

$D^{(n)}$ is the number of voice calls that complete during the n-th frame and depart from the system. This random variable is distributed as Binomial$(M^{(n)} - R^{(n)} - C^{(n)}, \sigma_v)$

where γ and σ are given in equations (1) and (2). From equation (10), we observe that $(R^{(n)}, C^{(n)}, M^{(n)})$ is a Markov state description. In addition, for any finite number of voice users, the system has a finite number of states. Since delayed voice packets are dropped, any choice of $(\mathbf{v}, \mathbf{p})$, such that $0 \leq p_i \leq 1$, $0 \leq v_i \leq N - i$, $i = 0, 1, \ldots, \min(M, N)$, would lead to the state $(0, 0, 0)$ being reachable from every other state. Thus, the Markov chain has a single communicating class and a unique stationary distribution $\pi(R, C, M)$ for any choice of $(\mathbf{v}, \mathbf{p})$.

In order to compute the transition probability matrix of the Markov Chain, we must derive the conditional probability distribution of N_{CR}, the number of contending voice terminals that succeed in obtaining a reservation at the end of a frame. Suppose the system is in state (R, C, M) at the beginning of an arbitrary frame. The number of voice contention slots is given by $v = \mathbf{v}(R, M)$ and the permission probability is $p = \mathbf{p}(R, M)$. Let J denote the number of successful transmissions by contending voice terminals during the frame. A successful transmission by a contending voice terminal occurs in a voice contention slot with probability

$$q = Cp(1-p)^{C-1}, \qquad C > 0 \tag{11}$$

and a failure (collision or idle) occurs with probability $(1 - q)$. Since the outcome in one voice contention slot is independent of the outcomes in all other voice contention slots, the number of successful transmissions J has a binomial distribution.

$$P\{J = j\} = \begin{cases} \binom{v}{j} q^j (1-q)^{v-j} & j = 0, 1, \ldots, v \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Since a contending voice terminal may have more than one successful transmission in a frame, N_{CR}, the number of contending terminals that succeed in making reservations, may be less than J, the number of successful transmissions. The probability distribution of N_{CR} can be expressed as

$$P\{N_{CR} = s\} = \sum_{j=s}^{v} P\{N_{CR} = s | J = j\} P\{J = j\} \tag{13}$$

We derive the expression for $P\{N_{CR} = s | J = j\}$ in Appendix A.

From the transition probability matrix derived above, we can derive the steady–state probabilities, $\pi(R, C, M)$ of the states of the Markov Chain. Speech packets that are not transmitted successfully in a frame are dropped at the end of the frame. The long term packet dropping probability is computed as the ratio of the average number of speech packets dropped in a frame to the average number of packets generated per frame.

Since each contending voice terminal that fails to obtain a reservation at the end of a frame drops a packet, the number of dropped packets in a frame equals the number of unsuccessful contending voice terminals. Each terminal that holds a reservation at the beginning of a frame successfully transmitted a packet in the previous frame since any terminal with a successful transmission in a frame continues to hold the reservation in the next frame. Hence R is the number of successful packets in the previous frame and if D is the number of packets that are dropped, the sum $R + D$ must equal the number of packets generated during the previous frame. Hence, the packet dropping probability is given by

$$p_{\text{drop}} = \frac{E[D]}{E[R + D]} \tag{14}$$

A contending voice terminal fails to make a reservation in an arbitrary voice contention slot with probability $1 - p(1-p)^{C-1}$. A contending voice terminal drops a packet at the end of a frame if it is not successfully transmitted in any of the v voice contention slots. Since the outcomes in the v voice contention slots are independent, an arbitrary contending voice terminal drops a packet with probability

$$\delta(R, C, M) = (1 - p(1-p)^{C-1})^{v} \tag{15}$$

with v and p chosen according to equations (5) and (6) respectively. Summing over all the C contending terminals, the expected number of dropped packets becomes

$$E[D|R, C, M] = C\delta(R, C, M) \tag{16}$$

Averaging over the stationary distribution of the system, the expected number of dropped packets per frame is

$$E[D] = \sum_{(R,C,M)\in S} E[D|R, C, M]\pi(R, C, M) \tag{17}$$

The expected number of voice packets generated per frame, $E[R + D]$, depends only on the probability distribution of the number of voice users in the system and the speech activity[1], α. So

$$E[R + D] = \sum_{m=1}^{M_{0.01}} m\alpha P\{M = m\} = E[M]\alpha \tag{18}$$

Since voice calls arrive as a Poisson process of rate μ_v and calls are blocked only when the there are $M_{0.01}$ voice calls in the system, the number of voice calls in the system has the truncated Erlang distribution

$$P\{M = m\} = \frac{(\lambda_v/\mu_v)^m/m!}{\sum_{n=0}^{M_{0.01}} (\lambda_v/\mu_v)^n/n!} \quad (0 \leq m \leq M_{0.01}) \tag{19}$$

Therefore, the packet dropping probability, is given by

$$p_{\text{drop}} = \frac{\sum_{(R,C,M)\in S} E[D|(R, C, M)]\pi(R, C, M)}{E[M]\alpha} \tag{20}$$

[1]defined as the proportion of time that a voice user is in a talkspurt

4 Selection of Design Parameters

A pair of vectors, $(\mathbf{v}, \mathbf{p})$, is said to be *feasible* for the voice subsystem if it satisfies the criterion that no more than 1% of the voice packets arriving into the system will be dropped. In choosing "good" values for the design parameters, we will restrict ourselves to the set of feasible solutions. This will guarantee that the QoS requirements of the voice terminals are satisfied. While attempting to satisfy the QoS requirements of the voice terminals, we would also like to get the "best" performance possible for the data users. At this time, we do not know what the operating parameters of the data subsystem (like the arrival process, the probability distribution of the service demand, the multi–access protocol that the data users will use, etc.). But we assume that whatever the operating parameters of the data subsystem, the performance of the data subsystem would be better if more bandwidth is allocated to it.

If the above assumption is true, then a pair of vectors, $(\mathbf{v}, \mathbf{p})$, is said to be *optimal* if it is feasible for the voice subsystem, and it maximizes the expected bandwidth available for the data subsystem. The expected number of slots per frame allocated to the data users is given by

$$E[N - R - v(R, M)] = \sum_{(R,C,M) \in S} (N - R - v(R, M))\pi(R, C, M) \tag{21}$$

So the problem of finding optimal values for the design parameters can be formulated as

$$\begin{array}{rl} \text{maximize} & E[N - R - v(R, M)] \\ \text{subject to} & p_{\text{drop}} \leq 0.01 \\ & 0 \leq p(R, M) \leq 1 \\ & 0 \leq v(R, M) \leq N - R \end{array} \tag{22}$$

In (22), $v(R, M)$ is shown to be a continuous variable. However, we must allocate an integral number of voice contention slots in any frame. This relaxation of the integer constraint on $v(R, M)$ is handled as follows. When a solution to (22) specifies a non-integral value for $v(R, M)$ in a frame with M voice calls and R reserved slots, we choose a random number $V(R, M)$ of voice contention slots such that $E[V(R, M)] = v(R, M)$. In particular, we define

$$f(R, M) = v(R, M) - \lfloor v(R, M) \rfloor. \tag{23}$$

so that

$$V(R, M) = \begin{cases} \lfloor v(R, M) \rfloor & \text{with probability } 1 - f(R, M) \\ \lceil v(R, M) \rceil & \text{with probability } f(R, M) \end{cases} \tag{24}$$

The transition probability matrix of the system, in particular equation (13), has to be modified to account for the randomization in (24). Hence the optimization problem becomes a non-linear programming problem in the continuous parameters $v(R, M)$ and $p(R, M)$ rather than an integer non-linear programming.

It is clear that if we reduce the number of voice contention slots, the mean access delay increases and hence voice packet dropping probability increases. Given any $(\mathbf{v}, \mathbf{p})$ such that $p_{\text{drop}} < 0.01$, the optimization procedure decreases the number of voice contention slots appropriately to increase the number of data slots. This procedure increases voice packet dropping probability and so we stop when $p_{\text{drop}} = 0.01$.

We know that $E[R]$ is the expected number of successfully transmitted voice packets in a frame. The expected number of voice packets generated by all voice terminals is $E[M]\alpha$. If exactly 1% of all voice packets are dropped, then, at optimum,

$$E[R] = 0.99E[M]\alpha \tag{25}$$

which is independent of the choice of $(\mathbf{v}, \mathbf{p})$. So the optimization problem (22) can be modified as

$$\begin{array}{ll} \text{minimize} & E[v(R, M)] \\ \text{subject to} & p_{\text{drop}} \leq 0.01 \\ & 0 \leq p(R, M) \leq 1 \\ & 0 \leq v(R, M) \leq N - R \end{array} \tag{26}$$

The decision variables in the above optimization problems are $(\mathbf{v}, \mathbf{p})$. The functions $E[v(R, M)]$ and p_{drop} are non–linear in the decision variables and do not have a closed form due to the presence of the stationary probabilities $\pi(R, C, M)$. This makes it difficult to solve the above problem using non–linear programming techniques that use derivatives. Computing the values of these two functions for a particular $(\mathbf{v}, \mathbf{p})$ involves the inversion of a transition probability matrix of the Markov Chain (which needs to be computed first) to get the steady–state probabilities.

The number of states in this Markov Chain is typically of the order of $O(NM_{0.01}^2)$. In order to keep the size of the optimization problem computationally small, we will assume the number of voice terminals in the system, M, is constant and solve for the "optimal" values for the design parameters for that value of M. For a constant number M of voice terminals, the number of states in the Markov Chain is $O(NM)$. We perform this optimization for each value of M such that $0 \leq M \leq M_{0.01}$. We then use these "optimal" solutions to compute exactly the performance of the voice subsystem with varying M due to call arrivals and departures.

These heuristic techniques have yielded approximate solutions to (26) that we believe are close to optimum. Even though the solutions to the optimization problem (26) is approximate, it should be noted that the corresponding performance measures (voice packet dropping probability and the expected number of data slots per frame) are computed exactly from the resulting Markov chain. Thus, the numerical results that are presented in the next section are exact, but are not necessarily optimal.

Source rate	32 KB/sec
Channel rate	432 KB/sec
Overhead	80 bits
Frame duration	11 ms
Number of slots per frame	11
Average talkspurt duration	1 sec
Average silence duration	1.35 sec

Table 1: Parameters of an 11 slot FRMA System

Properties of the optimal voice contention policy permit us to quickly calculate or estimate the impact of an increase in the system bandwidth. Let $(\mathbf{v}^*, \mathbf{p}^*)_{M,N}$ denote the optimal solution for the non–linear programming problem described in (26) with the number of slots per frame being N and the number of voice terminals being M. Then the following observations are true.

- Suppose M_0 and N_0 are such that $M_0 \leq N_0$. Then, the voice contention policy $(\mathbf{v}^*, \mathbf{p}^*)_{M_0,N_0}$ is optimal for any $N \geq M_0$ because the state space of the Markov Chain remains the same and the transition probability matrix depends only on the voice activity parameters and the decision variables.
- Let $D^*(M_0, N_0)$ denote the maximum available data bandwidth with M_0 voice terminals and N_0 slots per frame under the optimal policy $(\mathbf{v}^*, \mathbf{p}^*)_{M_0,N_0}$. For a system in which the number of slots is $N > N_0$, $(\mathbf{v}^*, \mathbf{p}^*)_{M_0,N_0}$ would restrict the voice users to a subset of N_0 slots while providing $p_{\text{drop}} \leq 0.01$. For the N slot system with voice contention policy $(\mathbf{v}^*, \mathbf{p}^*)_{M_0,N_0}$, an additional $N - N_0$ slots per frame would be available for data. Hence, under the optimal policy for a system with M_0 voice terminals and $N \geq N_0$ slots,

$$D^*(M_0, N) \geq D^*(M_0, N_0) + N - N_0 \tag{27}$$

Thus, any increase in the total available bandwidth translates to additional data bandwidth given the number of voice terminals does not change.

5 Bandwidth Allocations – Numerical Results

It is instructive to examine the bandwidth allocations as a function of the number of voice terminals M in the system. In particular, for a system with $N = 11$ slots described by

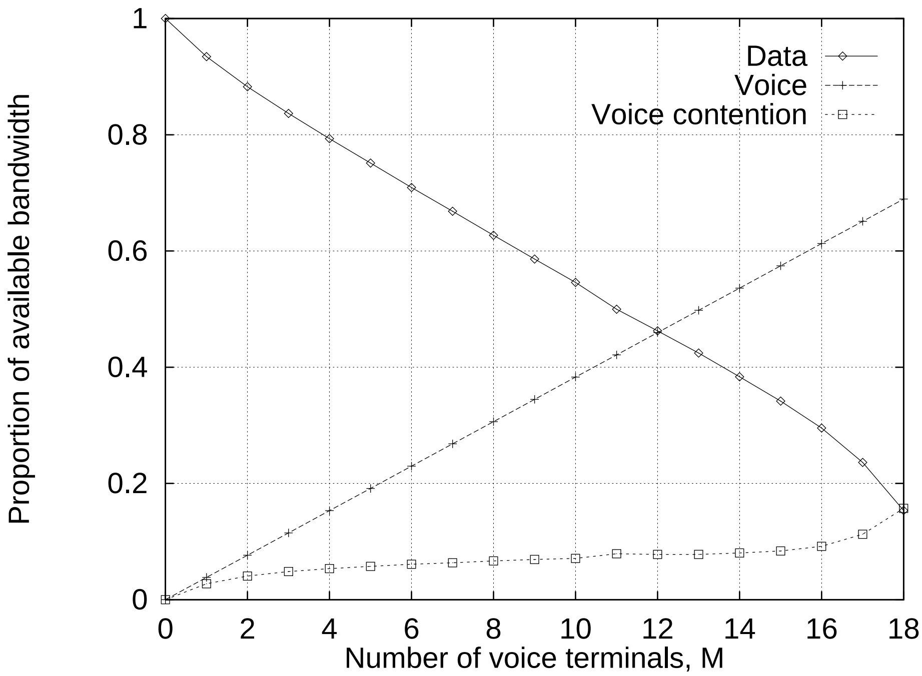

Figure 2: The proportion of bandwidth required by each of the three classes of traffic, namely, voice traffic, voice contention, and data as a function of the number of voice terminals. The number of voice terminals in the system, at each point in the graph, was assumed to be a constant. Parameters for this 11 slot system are given in Table 1.

Table 1, we have found approximate optimal solutions for (26). Figure 2 plots the proportion of the available bandwidth that is allocated for each of the three classes – voice traffic, voice contention, and data, as a function of M. The voice packet dropping probability is exactly equal to 1% whenever there was at least one voice terminal in the system. From [4], we know that $M_{0.01} = 18$. From the figure, we can observe that even when the voice subsystem is operating at its capacity, we can have a maximum data throughput of about 17%. This is due to the fact that in the voice–only system, when the number of terminals is constant at $M_{0.01}$, the voice packet dropping probability is around 0.0088 [4]. Since this is lower than the bound of 0.01, we are able to push the voice packet dropping to 0.01 and accommodate some data traffic.

Figure 3 plots, as a function of voice load, the proportion of bandwidth allocated to each of the three classes of traffic for the 11 slot system with a Poisson call arrival process. With a voice call blocking probability of 0.02, we can support about 11 Erlangs of voice traffic and about 50% of the channel can be utilized by the data users.

In order to compare the performance of our system with [6, 7], we also studied a 20 slot system with parameters described in Table 2. The capacity of the voice–only system was

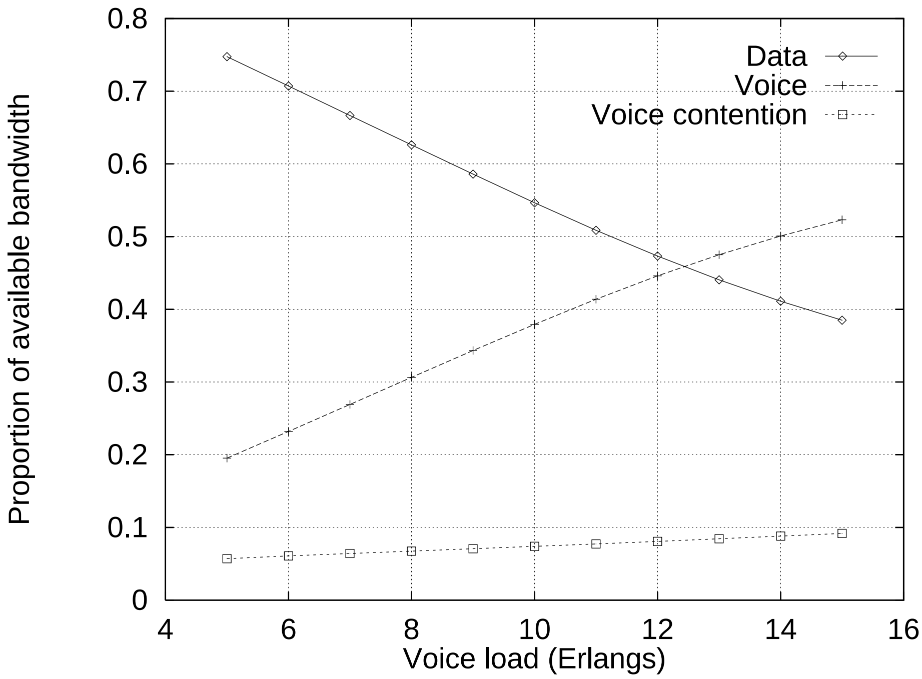

Figure 3: The proportion of available bandwidth required by each of the three classes of traffic as a function of the voice load on the system. Parameters for this 11 slot system are given in Table 1.

found to be 36 voice terminals, which is consistent with the results found in [6, 7]. Thus, there is no loss in capacity due to transmitting acknowledgments once per frame instead of once every slot.

In Figure 4, we plot the proportion of the available bandwidth that is allocated to the three classes of traffic. This figure is the analogue of the results in Figure 2, but for the 20–slot system. Both the 11 and 20 slot systems exhibit similar performance with regions of linear maximum data capacity followed by a sharp drop near $M = M_{0.01}$.

6 Performance of the Data Subsystem

So far, we have only considered the bandwidth allocations that result from our efforts to solve the optimization problem (26). We have found that significant capacity can be dynamically partitioned for the data users. In order to make comparisons with earlier work [6, 7], we will embed a simple data subsystem within the data users' partition. In this section, we evaluate by simulation the performance of the data subsystem.

Source rate	32 KB/sec
Channel rate	720 KB/sec
Overhead	64 bits
Frame duration	16 ms
Number of slots per frame, N	20
Average talkspurt duration, t_1	1 sec
Average silence duration, t_2	1.35 sec

Table 2: Parameters for a 20 slot FRMA System

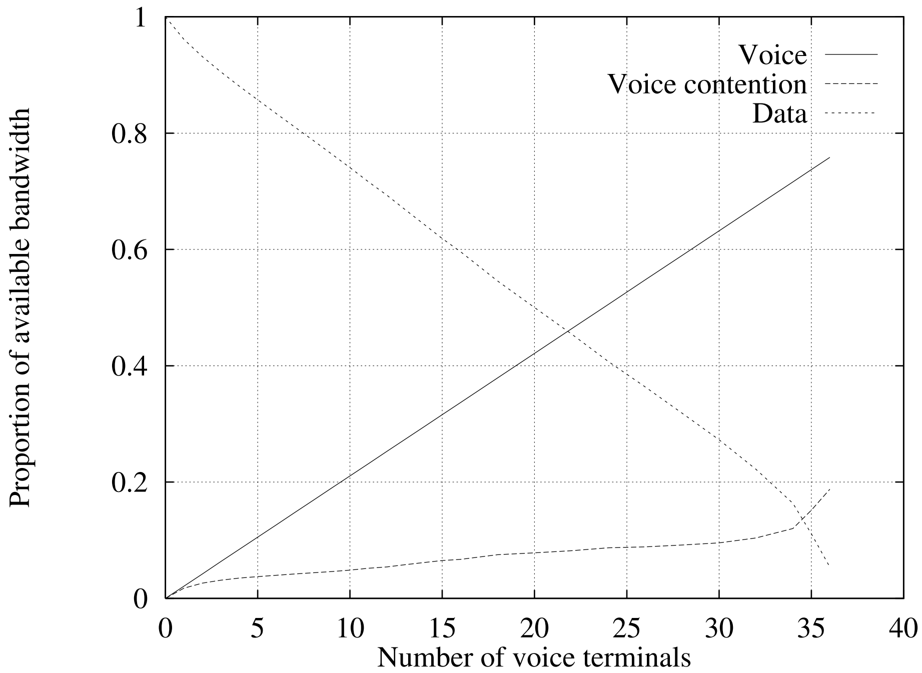

Figure 4: Proportion of available bandwidth allocated to each of the three classes of traffic. The number of voice terminals is constant, the decision variables for the voice subsystem were obtained from solutions to (26). Parameters for this 20 slot system are given in Table 2.

For the data subsystem, the arrival of data packets is modeled as a Poisson process with a mean rate of R_d bits per second at each data terminal. The number of data terminals is fixed and any data terminal with at least one packet in its buffer is considered backlogged. All other data terminals are considered to be in the idle state. The data subsystem cycles through two phases, each consisting of a number of frames.

- a contention phase – in which all backlogged data terminals contend for channel access. They do so by choosing to contend in a frame according to a data permission probability p_d, and then choosing one of the data slots at random. The contention phase ends when at least one data terminal contends successfully in a frame.
- a service phase – in which all successful data terminals are served according to a random order of service. During this phase, the base station identifies the data terminal that could transmit in each data slot. When a particular data terminal that has been assigned a data slot fails to transmit in that slot, then the base station assumes that this data terminal has moved to the idle state and removes this terminal from the service queue. The service phase ends when the service queue is empty.

The value of p_d and the choice of service discipline have an impact on the data subsystem performance. The dynamic partitioning of the voice and data subsystems permits us to optimize the data subsystem within the data users' partition. Moreover, the choice of data subsystem would have no effect on the QoS received by the voice users. The optimization of the data subsystem, however, is beyond the scope of this paper and could be the subject of a future study.

Following [6, 7], the number of voice terminals M_v, and the number of data terminals M_d, were both held constant at 20 in the simulation experiments. The buffer at each data terminal was assumed to hold 1000 data packets. As in [7], the data subsystem was assumed to be stable, as long as there were no buffer overflows at any data terminal for the duration of the simulation (200000 frames). The permission probability for the voice terminals and the size of the voice contention region were chosen from the solution to the problem (26) with $M = 20$.

Wong and Goodman [7] plot the average packet delay for a given data rate, R_d, at each data terminal with $M_v = 20$ and $M_d = 20$. They conclude that the maximum data rate that can be supported by their system is 3500 bps, compared to 1400 bps for Nanda's system [6]. In Figure 5, we plot the average packet delay for different values of R_d with $M_v = M_d = 20$ and compare the results[2] with those in [6, 7]. The integrated FRMA system can supported a significantly higher data rate with lower delays. Of course, one must keep in mind that

[2]The two curves for PRMA and IPRMA shown here were derived from Wong and Goodman [7].

the additional data capacity is due, in part, to the fact that even when the overall voice/data system is lightly loaded, the voice users will endure 1% packet dropping.

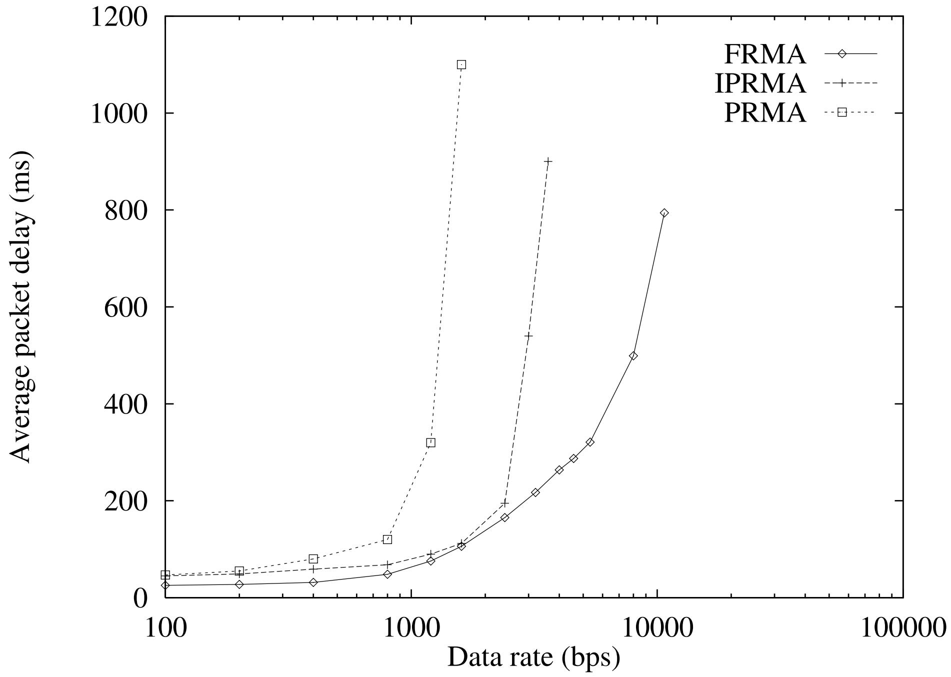

Figure 5: Average packet delay as a function of the data rate R_d at each data terminal. $M_v = M_d = 20$.

7 Conclusions

We have proposed a new scheme for the integration of voice and data over FRMA, a variant of PRMA. The voice and data subsystems are separated logically. Voice traffic is given priority that is implicit and limited by the satisfaction of its QoS requirement. In this case, the QoS requirements for the voice traffic is that the voice packet dropping probability not exceed 1%.

The voice sub–system has been formulated as a Markov Chain and an exact analysis of its performance (voice packet dropping probability) has been carried out. An optimization problem to maximize the bandwidth allocated to the data users while satisfying the QoS requirements of the voice users has been formulated. Since this optimization problem is difficult to solve using non–linear programming solution techniques, we have adopted heuristic approaches to obtain "approximate" solutions for the decision variables. But it should be stressed that the results summarized in the previous section are exact and we believe they are very close to optimum.

The capacity of the system to support voice users is not sacrificed in the integration of data users into the system. Also, since the voice and the data subsystems are logically separated, any instability in one will not affect the other. The numerical results indicate that the bandwidth allocated for the data users is quite significant. Overall, the new protocol does not sacrifice the simplicity of PRMA while improving the performance of the system.

In a preliminary study of the data subsystem using simulations, it was found that the integrated FRMA system described in this paper can support a higher data rate with lower delays than the integrated PRMA systems of Nanda [6] and Wong and Goodman [7].

A Conditional Distribution of the Number of Successful Terminals

Consider an arbitrary frame with R reserved slots, v voice contention slots, and C contending voice terminals. Let S denote the number of contending terminals that succeed in obtaining a reservation at the end of the frame. Let J denote the number of available slots in which there was a successful transmission. In this appendix, we show how to calculate the conditional distribution $P\{S = s|J = j\}$.

Let S_i denote the number of successful transmissions by user i ($i = 1, 2, \ldots, C$). so that $J = S_1 + \cdots + S_C$. Given $J = j$, consider one of the j slots in which a success occurred. Since all users have the same permission probability, each of the C users has a conditional probability of success $1/C$ in such a slot. Moreover, because the outcome in each of these j slots is independent of that in all other slots, the conditional distribution of $S_1, \ldots, S_C$ given $J = j$ is the multinomial distribution

$$P\{S_1 = s_1, \ldots, S_C = s_C|J = j\} = \begin{cases} \dfrac{j!}{s_1! \cdots s_C!} \dfrac{1}{C^j} & s_1 + \cdots + s_C = j \\ 0 & \text{otherwise} \end{cases} \tag{28}$$

Notice that $S = 0$ iff $J = 0$. Henceforth, we assume $J > 0$ and thus $S > 0$. Let A_s denote the set of all $(S_1, \ldots, S_C)$ for which $S = s$ so that

$$P\{S = s|J = j\} = \sum_{(s_1,\ldots,s_C)\in A_s} \frac{j!}{s_1! \ldots s_C!} \frac{1}{C^j} \qquad 1 \le s \le \min(j, C) \tag{29}$$

We observe that $S = s$ if there is a subset of exactly s users that have made at least one successful transmission. Let L_s denote the set of s element vectors $(m_1, \ldots, m_s)$ such that $\sum_{i=1}^{s} m_i = j$ and $m_i \geq 1$ for $1 \leq i \leq s$. Hence, $(s_1, \ldots, s_C) \in A_s$ if $s_1, \ldots, s_C$ contains a subsequence $(s_{l_1}, \ldots, s_{l_s}) \in L_s$ corresponding to the s users each with at least one successful transmission. Since there are $\binom{C}{s}$ possible subsets of s users,

$$P\{S = s | J = j\} = \binom{C}{s} \sum_{(s_{l_1}, \ldots, s_{l_s}) \in L_s} \frac{1}{C^j} \frac{j!}{s_{l_1}! \cdots s_{l_s}!} \qquad 1 \leq s \leq \min(j, C) \tag{30}$$

We define

$$M_j(s) = \sum_{(m_1, \ldots, m_s) \in L_s} \frac{j!}{m_1! \cdots m_s!} \qquad 1 \leq s \leq \min(j, C) \tag{31}$$

so that

$$P\{S = s | J = j\} = \binom{C}{s} \frac{M_j(s)}{C^j} \tag{32}$$

We note that $M_j(s)$ represents the number of length j ordered sequences of s distinguishable objects such that each object appears at least once in each sequence. We can recursively calculate $M_j(s)$. First, we note that if each object is not required to appear, then the number of length j ordered sequences of s distinguishable objects is s^j. Hence, $M_j(s)$ equals s^j less the number of ordered length j sequences in which one or more objects do not appear. Since there are $\binom{s}{i}$ subsets of i objects and since there are $M_j(s-i)$ ordered sequences of $s - i$ objects in which all $s - i$ objects appear at least once, we have

$$M_j(s) = s^j - \sum_{i=1}^{s-1} \binom{s}{i} M_j(s - i) \tag{33}$$

Finally, since $M_j(1) = 1$, we have a recursive method for the computation of $M_j(s)$.

References

[1] D. J. Goodman, R. A. Valenzeula, K. T. Gayliard, and B. Ramamurthi, "Packet reservation multiple access for local wireless communications," *IEEE Trans. Commun.*, vol. 37, pp. 885–890, August 1989.

[2] S. Nanda, D. J. Goodman, and U. Timor, "Performance of PRMA : A packet voice protocol for cellular systems," *IEEE Trans Veh. Tech.*, vol. 40, pp. 585–598, August 1991.

[3] P. Rauber, "Duplex schemes and receiver activity for packet reservation multiple access," Master's thesis, Rutgers, The State University of New Jersey, New Brunswick, NJ, 1994.

[4] P. Narasimhan, R. Yates, and D. J. Goodman, "Performance analysis of frame reservation multiple access," Tech. Rep. 80, WINLAB, Rutgers University, Piscataway, NJ 08854, July 1994.

[5] P. Narasimhan, R. Yates, and D. J. Goodman, "Analysis of frame reservation multiple access," in *Proc. International Conference on Universal Personal Communications, 1994*, (San Diego, CA), 1994.

[6] S. Nanda, "Analysis of packet reservation multiple access : Voice and data integration for wireless networks," in *Proc. GLOBECOMM '90*, (San Diego, CA), pp. 1984–1988, 1990.

[7] W. C. Wong and D. J. Goodman, "A packet reservation multiple access protocol for integrated speech and data transmission," *IEE Procedings – I*, vol. 139, December 1992.

[8] G. Wu, K. Mukumoto, and A. Fukuda, "Analysis of an integrated voice and data transmission system using packet reservation multiple access," *IEEE Trans. Veh. Tech.*, vol. 43, pp. 289–297, May 1994.

[9] S. Jangi and L. F. Merakos, "Performance analysis of reservation random access protocols for wireless access networks," *IEEE Trans. Communications*, vol. 42, pp. 1223–1234, February 1994.

[10] P. T. Brady, "A model for on–off speech patterns in two–way conversation," *Bell Syst. Tech. Journal*, vol. 48, pp. 2445–2472, September 1969.