Optimal Frame Pattern Design for a TDMA Mobile Communication System Using a Simulated Annealing Algorithm

Chung-Ju Chang, *Member, IEEE,* and Chen-Hsiang Wu

Abstract—**Different relative positions of voice and data slots within the frame of a TDMA mobile communication system may result in different system data throughput. The optimal relative position, namely the frame pattern which can achieve maximum data throughput, is designed in this paper. This design method can be formulated as solving a combinatorial optimization problem. Generally, the global optimal solution of such a problem is hard to find using conventional methods, which may be computationally intractable. A reliable and effective method called a simulated annealing algorithm is applied to obtaining the global optimum. Numerical results reveal that there exist various optimal frame patterns for different ranges of traffic load and the optimal frame pattern can achieve great throughput improvement over a random frame pattern.**

I. Introduction

As is well known, cellular mobile communications render convenient service and are widely used. The growing public acceptance has saturated capacity of the first generation system, which is based on analog voice transmission. In order to improve spectrum efficiency, second generation systems are being introduced with digital transmission technology. Various systems with different architectures, frequency division multiple access (FDMA), time division multiple access (TDMA), and code division multiple access (CDMA), have been proposed for digital radio. Among these, the TDMA structure supports users *flexibility* of choosing their own desired rates of communications and proper level of coding according to the nature of the transmitted traffic. This integration potential makes TDMA welcomed and widely adopted [7], [8]. Also, in addition to the traditional voice service, data service applications like sending maintenance instructions for repair personnel are becoming common. Therefore, the TDMA system will be expected to provide integrated voice and data services.

Many efforts in the literature have been devoted to finding an efficient multiplexing strategy for integrating voice and data. In traditional FDMA mobile systems, integration is provided by handling each type of communications separately and thus establishing two effectively independent subsystems within each vehicle [9]. Since such an architecture duplicates equipment, wastes channel space, and responds poorly to traffic change, an efficient approach is proposed by [10], [11], and [15] where the integration of voice and data is implemented by placing data into the silence gaps of voice. This policy can upgrade a conventional voice-only system to an integrated voice and data system without compromising voice performance.

There are also two multiplexing approaches for integrating voice and data in the TDMA mobile system. One is the fixed boundary multiplexing scheme that partitions a TDMA frame into two subframes, where the voice subframe and data subframe are dedicated to voice and data, respectively. This scheme faces the problem of an insensitive response to the traffic change of voice or data. The other scheme is the movable boundary multiplexing scheme that allows a data user to borrow idle slots within the voice subframe. It can dynamically use free slots so as to improve utilization [2], [12], [16], [17].

Due to the difference of delay and quality requirements for voice and data traffic in the integrated services TDMA mobile communication system, different transmission modes for voice and data slots are used. A central assignment and synchronous transmission mode is usually provided for voice traffic, while a distributed assignment and contention transmission mode is adopted for data traffic. As such, the positions of available data slots within a frame (called the *frame pattern* in this paper) should affect the data performance of the system. Generally speaking, the system will gather more ready packets awaiting transmission for a frame pattern with longer intervals between two neighboring data slots, and this will result in higher probability of packet collision at heavy traffic load. Thus when heavy traffic occurs, the frame pattern with data slots merged together is expected to have better throughput than the other frame patterns, such as the one with data slots uniformly distributed over a frame. We shall explain the phenomenon more clearly in a later section. This paper provides an algorithm to determine the best data slot positions for a given data traffic load so that the performance criterion, system data throughput, is maximized.

In this paper, the optimal frame pattern for an integrated voice and data TDMA mobile communication system is designed. The design problem is formulated as a global optimization procedure in which the system data throughput as the cost

Manuscript received April 14, 1992; revised June 22, 1992 and October 20, 1992. This work was supported in part by the Telecommunication Laboratories, Ministry of Transportation and Communication under Contract TL-NSC-81-2102 and by the National Science Council, Taiwan, Republic of China under Contract NSC81-0404-E-009-002.

The authors are with the Department of Communication Engineering, National Chiao Tung University, Hsinchu, Taiwan 300, Republic of China.

IEEE Log Number 9207173.

0018–9598/93$03.00

function will be optimized by properly adjusting the positions of available data slots via an algorithm called simulated annealing. This simulated annealing algorithm is a reliable and effective approach to solving this combinatorial optimization problem; it is often more efficient than an exhaustive search yet more robust than gradient descent. The details of this algorithm will be introduced later and numerical examples will show that there are various optimal frame patterns for different ranges of traffic load. Incidentally, if the optimal frame pattern alters as the offered data traffic load changes, the adjustment of data slot positions to the optimal one can be carried out by an automatic time-slot reassignment protocol. A similar protocol can be found in [1].

In Section II of this paper, formulation of the frame pattern and a model for an integrated voice and data TDMA mobile communication system are given, and system data throughput for a given frame pattern is obtained. Section III is devoted to finding the optimal frame pattern by the simulated annealing algorithm. Some numerical illustrations are discussed in Section IV. The conclusion is presented in Section V.

II. Frame Pattern and Data Throughput

A. The Frame Pattern

In an integrated voice and data TDMA mobile communication system, the entire frequency band is usually divided into a number of duplex channel pairs. One channel of each pair is for the down link and is configured in TDM mode, while the other channel is for the up link and is configured in TDMA mode [8]. The TDMA link is usually synchronized with the TDM link, but with a half-frame offset to preclude the need for the mobile to transmit and receive simultaneously [1].

Within the TDMA link, a synchronous transmission mode is provided for voice but a contention mode is provided for data. In this paper, we assume that a movable boundary multiplexing scheme is adopted in the uplink channel, where the data traffic can use both its own nominal data slots and the unassigned voice slots but it can not use the silence gaps of assigned voice slots; we also assume that there are N independent slots in a frame of the TDMA link, and N_t slots among the N slots are assigned for data service while the remaining (N- N_t) slots are assigned for voice traffic. Thus N_t, the total number of slots available for data transmission at a given time, is the sum of nominal data slots and the unassigned voice slots. As shown in Fig. 1, let d_i denote the slot number in a frame of the ith available *data* slot, where $1 \leq i \leq N_t$, $1 \leq d_i \leq N$, and $d_1 < d_2 < \cdots < d_{N_t}$. The notation $d_i+1(d_i-1)$ refers to the slot immediately after (before) d_i, which may be a voice or data slot. Also, let x_i be the interdistance between two successive *data* slots d_{i+1} and d_i, where the notation d_{i+1} refers to the first *data* slot following d_i. Thus x_i is given by

$$x_i = \begin{cases} d_{i+1} - d_i, & \text{if } i = 1, 2, \cdots, N_t - 1. \\ d_1 + N - d_{N_t}, & \text{if } i = N_t. \end{cases} \tag{1}$$

Note that $x_i, x_2, \cdots, x_{N_t} \geq 1$. We call $(x_1, x_2, \cdots, x_{N_t})$ the *frame pattern* of the integrated voice and data TDMA mobile communication system, where $0 \leq N_t \leq N$. The frame pattern extracts the feature of interdistance of two successive data slots in a frame. We denote a particular frame pattern by $\boldsymbol{X}(N_t)$, and denote the set of all possible frame patterns by $\boldsymbol{U}(\boldsymbol{X}(N_t))$ which is formulated as

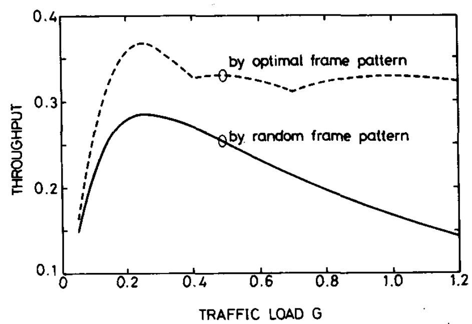


Fig. 1.

$$\boldsymbol{U}(\boldsymbol{X}(N_t)) = \{\boldsymbol{X}(N_t) = (x_1, x_2, \cdots, x_{N_t}) | \\ x_1, x_2, \cdots, x_{N_t} \geq 1, \text{ and } x_1 + x_2 + \cdots + x_{N_t} = N\}. \tag{2}$$

B. The Mean Data Throughput

Before obtaining the mean data throughput per slot, we assume the following.

1. The holding time of the voice call is much longer than the frame time so that the queueing behavior of data for a given N_t frame pattern can reach *steady state.*
2. The total data traffic, new and retransmitted, constitutes a *Poisson* process with mean arrival rate G packets/slot.
3. The slotted ALOHA random access protocol is assumed for data transmission.

Based on the above assumptions, the mean data throughput given a specific frame pattern $\boldsymbol{X}(N_t)$ in $\boldsymbol{U}(\boldsymbol{X}(N_t))$, denoted by $S_{\boldsymbol{X}(N_t)}$, can be given by

$$S_{\boldsymbol{X}(N_t)} = \frac{G}{N_t} \cdot \sum_{i=1}^{Nt} x_i \cdot e^{-G \cdot x_i}. \tag{3}$$

From (3), we can see that various frame patterns may result in different mean data throughput. Of all these patterns in $\boldsymbol{U}(\boldsymbol{X}(N_t))$, there must exist an optimal frame pattern which achieves the largest mean data throughput. However, there are many frame patterns for a given N_t slots in N, say 7.5394×10^{10} for $N = 60$, $N_t = 10$. A brute-force search for an optimal frame pattern is too difficult—possibly even impossible. The next section of this paper will be devoted to finding the optimal solution via an efficient and robust optimization algorithm—simulated annealing.

III. The Optimal Frame Pattern Design by Simulated Annealing

A. The Simulated Annealing Algorithm

For a given combinatorial optimization problem, there are many feasible solution states $\boldsymbol{X}$ which collect to form a solution state space $\boldsymbol{U}$, and there is a cost function C which assigns each $\boldsymbol{X}$ in $\boldsymbol{U}$ a cost $C(\boldsymbol{X})$. The ($\boldsymbol{U}$, $C(\boldsymbol{U})$) can be imagined as a land which consists of mountains and valleys whose height represents the corresponding cost. The goal of the optimization is to find the deepest valley, meaning the minimum cost. Traditional iterative improvement heuristics generate a random walk in the land and a move is adopted if the move maximally reduces the cost. This "greedy strategy" can easily trap the final solution into a local minimum. The deepest valley might be located far from the local minimum and can be reached only after hill climbing.

The simulated annealing, however, allows a move to climb hills, in addition to the welcomed move to lower cost. It controls the ability of this climbing in a manner analogous to physical solid annealing: raising the temperature so that the particles of solid are equipped with high energy to fully randomly arrange in the liquid phase, followed by a cooling process to decrease temperature so as to force the uniform particles into a regular lattice [4]. This simulated annealing algorithm is essentially a robust probabilistic algorithm to solve complex combinatorial optimization problems with good quality. Its robustness results from its guaranteed convergence to global optimum and its independence of the final solution on the initial point [3]. It is more efficient than an exhaustive search and more robust than gradient descent.

In this algorithm, an externally specified parameter T, analogous to the temperature in physical annealing, is used to indicate the ability to climb hills. The move to a higher cost point is sometimes permitted with probability positively proportional to T. As in physical annealing, T is initialized to a high value so that the simulated annealing can hardly distinguish valleys or mountains, rendering the initial point a practically complete freedom to visit everywhere in the land. After a number of walks, it is expected that a roughly lower region has been found and the freedom to wander should be restricted. As the algorithm progresses, T is gradually decreased and the search is systematically concentrated into regions likely to contain a global optimum, but still random enough to escape most local optimums. After T has decreased so small that the probability of escaping from the present valley is negligible, T is set to zero and the greedy strategy, in which no more hill climbing moves are allowed, starts to find the optimal solution in the final concentrated valley. The convergence to the global optimum is shown to be provable with probability one sense [13].

Based on the above concept and the references [3], [4], the general structure of the simulated annealing algorithm is stated as follows.

Simulated Annealing Algorithm

Step 0: [Initialize]

Initialize the effective temperature T_0 and randomly choose an initial state $\boldsymbol{X}_i$ from $\boldsymbol{U}$.

Step 1: [Generate a new solution state]

A new solution state $\boldsymbol{X}_j$ is generated by $\boldsymbol{X}_j = Generate(\boldsymbol{X}_i)$.

Step 2: [State Transition]

$\Delta c = C(\boldsymbol{X}_j) - C(\boldsymbol{X}_i)$

r = random number in (0, 1)

IF (r <$Accept(\Delta c, T)$) THEN

The new solution state $\boldsymbol{X}_j$ is accepted by $\boldsymbol{X}_i \leftarrow \boldsymbol{X}_j$.

ELSE

The new solution state $\boldsymbol{X}_j$ is rejected.

Step 3: [Inner Loop Criterion Checking]

IF (*inner loop criterion* is satisfied)

Executive Step 4 to decrease T.

ELSE

Repeat Steps 1–3.

Step 4: [Temperature Cooling]

A new lower T is updated by T = $Update(T)$.

Step 5: [Outer Loop Criterion Checking]

IF (*outer loop criterion* is satisfied) THEN

Execute Step 6.

ELSE

Repeat Steps 1–5.

Step 6: [Greedy Strategy]

$T = 0$

A greedy strategy starts to find the local optimum.

END

There are six parameters involved above, which perform within the algorithm in the following manner: (i) Initial effective temperature T_0 is chosen so that the initial acceptance probability of a move to a high-cost solution state is large, (ii) Function $Generate(\boldsymbol{X}_i)$ is to perturb a new solution state $\boldsymbol{X}_j$ from the present state $\boldsymbol{X}_i$, (iii) Function $Accept(\Delta c, T)$ is to determine the permission of the move from $\boldsymbol{X}_i$ to $\boldsymbol{X}_j$, (iv) *Inner Loop Criterion* is to check whether an equilibrium condition of the move at the temperature T has been reached, or whether a sufficient number of moves has been tried to reach an equilibrium condition at this temperature T, (v) Function $Update(T)$ is to decrease the present T to a new lower value, and (vi) *Outer Loop Criterion* is used to decide whether the T is low enough to stop the cooling process of the simulated annealing algorithm. These parameters characterize and qualify the simulated annealing algorithm and have to be decided before the simulated annealing is applied to a particular problem. We shall describe them more clearly in the next subsection. The four parameters in (i), (iv), (v), and (vi) are, additionally, called the annealing schedule for their influence on the annealing of T. Many efforts have been devoted to finding a generally good choice of these parameters to make the simulated annealing efficient and accurate [3], [5]. There is an adaptive annealing schedule developed in [3] which shows significant improvement in efficiency over a simplified schedule and a geometric schedule. The reader can refer to it for more details.

B. The Optimal Frame Pattern Design

In this subsection, we will determine the optimal frame pattern for the TDMA mobile communication system via the simulated annealing algorithm with the adaptive annealing schedule proposed in [3]. We define the solution state space to be the $\boldsymbol{U}(\boldsymbol{X}(N_t))$ given in (2) and the corresponding cost function $C(\boldsymbol{X})$ to be the negative of the mean data throughput $S_{\boldsymbol{X}(N_t)}$ given by

$$C(\boldsymbol{X}) = -\frac{G}{N_t} \cdot \sum_{i=1}^{N_t} x_i \cdot e^{-G \cdot x_i} \tag{4}$$

The six parameters of simulated annealing are chosen as follows.

(i) Initial T_0

Usually, the initial temperature T_0 is chosen high enough to approach infinity so that any kind of move is always permitted and the solution state space can be explored completely. However, a higher T_0 results in a longer sequence of T and hence slow convergence speed. Hence $T_0 = 10\sigma_\infty(C)$ is chosen here, where $\sigma_\infty(C)$ is the standard deviation of the cost of explored states at $T = \infty$. Notice that $\sigma_\infty(C)$ and the corresponding expectation $E_\infty(C)$ can be estimated from a number of randomly generated solution states since there is no preference among solution states at $T = \infty$. The procedure to determine the number of randomly generated solution states is to select a small positive integer number at first, randomly generate several sets of solution states according to the number, and then check whether all the above generated sets produce nearly the same estimate of the expected value and standard deviation. If all the sets produce the same estimate of the expected value and standard deviation, the number is decided; otherwise, the number of sets is increased and the above procedure is repeated. In our problem, $T_0 = 10\sigma_\infty(C)$ is large enough to maintain quite high acceptance probability.

(ii) NGenerate($\boldsymbol{X}_i$)

A new solution state $\boldsymbol{X}_j$ is generated from $\boldsymbol{X}_i$ by choosing a move from a move set $\mathcal{M}$ which contains all feasible moves. For a given applied problem, the feasible moves in $\mathcal{M}$ depend on the corresponding solution space and are required to be defined. Moreover, the $\mathcal{M}$ should be *complete* so that there exists a finite number of moves to connect any two arbitrary solution states. In our problem, we define move (i, j) in $\mathcal{M}$ to be the move which moves the present solution state $\boldsymbol{X}_i = (x_i, x_2, \cdots, x_{N_t})$ to a new solution state $\boldsymbol{X}_j = (x_1, x_2, \cdots, x_i + 1, \cdots, x_j - 1, \cdots, x_{N_t})$, where $i \neq j$, and $1 \le i, j \le N_t$. These $N_t(N_t - 1)$ moves in $\mathcal{M}$ guarantee the completeness. The above move is called a basic move. In addition to this, we also define a complex move which consists of a batch of basic moves. The complex move is equivalent to a batch of basic moves carried out in sequence at once, and its cost is evaluated only after the last move of the sequence has been completed. This move can "tunnel" through hills and thus speed up the convergence [3, pp. 99].

The adaptive generate function in [3] is adopted here. This method determines the selection probabilities of the moves from the data collected during the execution of the algorithm, and a new move is selected among all feasible types of moves according to the selection probability which is proportional to its quality factor. For the sake of fast convergence, a welcomed type of move is one which results in sizable perturbation of the cost while maintaining a reasonable chance of being selected [3, pp. 98]. To rank the desirability of a move m, its quality factor Q_m and selection probability P_m are therefore defined by

$$Q_m = \frac{\sum_{k=1}^{a_c(m)} S_k(m)}{a_t(m)} \tag{5}$$

and

$$P_m = \frac{Q_m}{\sum_{\mathcal{M}} Q_m} \tag{6}$$

where the $a_c(m)$ and the $a_t(m)$ are the numbers of accepted and attempted moves of type m, respectively, and the $S_k(m) = |C(\boldsymbol{X}_j) - C(\boldsymbol{X}_i)|$ is the absolute value of the change of cost produced by the kth accepted type-m move.

(ii) Accept($\Delta c, T$)

The acceptance probability should increase with the increase in T but decrease with the increase in cost change Δc after a move. Thus *Accept*$(\Delta c, T)$ should be monotonically increasing with T but monotonically decreasing with Δc. Here, we adopt the welcomed Metropolis rule in [3], [4] as the *Accept*$(\Delta c, T)$ in our problem. It is given by

$$\mathit{Accept}(\Delta c, T) = \min\left[1, \exp\left(-\frac{\Delta c}{T}\right)\right]. \tag{7}$$

(iv) Inner Loop Criterion

From the theoretical simulated annealing analysis in [3, pp. 123], the distance from collected data of cost $C(\boldsymbol{X})$ to global minimum cost C^*, $(C(\boldsymbol{X}) - C^*)$, is distributed in gamma density. Therefore, the inner loop criterion is to test whether a sufficient number of moves has been tried so that the distance is distributed with the gamma function. This task is carried out in three steps, as described below.

1. The shape parameter α and the scale parameter λ of the predicted gamma distribution $\Gamma(\alpha, \lambda)$ [14] at temperature T in terms of $E_\infty[C]$ and $\sigma_\infty(C)$ are determined. They are given in [3] and are rewritten as

$$\alpha = \frac{(E_T[C] - C^*)^2}{\sigma_T^2(C)} \tag{8}$$

and

$$\lambda = \frac{E_T[C] - C^*}{\sigma_T^2(C)} \tag{9}$$

where $E_T[C]$ and $\sigma_T(C)$ are obtained by

$$E_T[C] = E_\infty[C] - \frac{\sigma_\infty^2(C)}{T} \cdot \left(\frac{aT}{aT+1}\right) \quad (10)$$

and

$$\sigma_T(C) = \sigma_\infty(C) \cdot \left(\frac{aT}{aT+1}\right)^2 \quad (11)$$

and "a" is given by

$$a = \frac{E_\infty[C] - C^*}{\sigma_\infty^2(C)} \quad (12)$$

2. After the values of α and λ are obtained, the predicted gamma distribution is available. We then test the average cost $\overline{C}$ of the collected solution states by

$$|E_T(C) - \overline{C}| \leq \eta.\sigma_T^2(C) \quad (13)$$

The η is a parameter left for specification.

3. If the average test is passed, the Kolmogorov-Smirnov test is used to check the closeness to the predicted gamma distribution [6]. The distances from the n collected data of cost to the minimum cost are calculated and arranged in order of increasing magnitude. Let $\{y_i | 1 \leq i \leq n\}$ denote the sequence of ordered distance, and a cumulative distribution function (CDF) for the y_i's, denoted by $A_n(y)$, is defined by

$$A_n(y) = \begin{cases} 0, & \text{if } y < y_1 \\ k/n, & \text{if } y_k \leq y \leq y_{k+1}, \text{ for } 1 \leq k \leq n-1 \\ 1, & \text{if } y \geq y_n. \end{cases} \quad (14)$$

Let $F(y)$ denote the CDF of the predicted gamma distribution $\Gamma(\alpha, \lambda)$, and define $D_n - \max_y |F(y) - A_n(y)|$. If $D_n < \zeta$, these two distributions are said to be close enough. The ζ depends on the specified confidence coefficient and the number of collected data n. It can be obtained by looking up the table in [6]. The homogeneous theory developed in [3] proves that the convergence to global optimum is independent of the annealing schedule if an infinite number of iterations is executed to achieve equilibrium. However, for the sake of limited computation resources and for accuracy, an upper bound $N_{\max}$ and a lower bound $N_{\min}$ of iterations in the inner loop are additionally set. If the upper bound $N_{\max}$ of interactions are taken and yet the data points are not sufficiently close to the predicted gamma distribution, T is decremented anyway.

(v) Update(T)

This function is recommended by [3] and is given by

$$Update(T) = T \cdot \max\left[\exp\left(-\frac{\delta \cdot T}{\sigma_T(C)}\right), r_{\min}\right] \quad (15)$$

where δ specifies the decrement rate of T and $r_{\min}$ is used to make sure that too fast annealing is avoided. Both $r_{\min}$ and δ are left for specification.

(vi) Outer Loop Criterion

It is reasonable to stop the cooling process of the simulated annealing when the algorithm has reached a local minimum and T is so small that the probability to escape from the local minimum is negligible. The occurrence of this situation is signaled when we apparently find that all the accessed states are of comparable costs in the present temperature. We therefore define the following outer loop criterion: The difference between the maximum and the minimum costs among accepted states at the present temperature T equals the maximum change in an accepted move at the same value of T.

The simulated annealing algorithm works in our application with acceptable computation time. A practical example is presented in the next section.

IV. Numerical Examples and Discussions

In this section, numerical examples are provided to show some interesting phenomena of optimal frame patterns. The optimal frame pattern is obtained by implementing a simulated annealing algorithm on a VAX 8800 super minicomputer with the following simulated annealing parameters: $\eta = 0.75$, $\delta = 0.7$, $N_{\max} = 200$, $N_{\min} = 10$, $r_{\min} = 0.90$, and the confidence coefficient of the Kolmogorov–Smirnov test equals 0.95 (and hence $\zeta = 1.36/\sqrt{n}$ with n collected data of costs). The phenomena of optimal frame patterns are investigated while varying G and N_t.

A. Optimal Frame Pattern Phenomena

Fig. 2 shows the optimal frame pattern (FP*) for various traffic load G and their mean data throughputs versus G in the range of [0.0, 1.2] if $N = 40$ and $N_t = 10$. The FP* is searched every 0.00125 in the range of $G = [0.0, 1.2]$. We can see from the figure that the FP* = frame pattern (4, 4, 4, 4, 4, 4, 4, 4, 4, 4) when G is in the range of (0.0, 0.3825], FP* = (3, 3, 3, 3, 3, 3, 3, 3, 3, 13) when G is in [0.3838, 0.4125], FP* = (2, 2, 2, 2, 2, 2, 2, 2, 2, 22) when G is in [0.4138, 0.6925], and FP* = (1, 1, 1, 1, 1, 1, 1, 1, 1, 31) when G is in [0.6938, 1.20]. There indeed exists an optimal frame pattern for a given G and N_t, but none is always optimal for all G. Instead, there are various optimal frame patterns corresponding to different ranges of G. The phenomenon can be explained as follows. We know that the standard slotted ALOHA system has a maximum throughput 0.368 at traffic load $G = 1$ and, the heavier or the lighter the traffic load than $G = 1$, the smaller the throughput. In our system, as heavy traffic exists (e.g., $G \geq 1$), a data slot having larger interdistance with its last neighboring data slot will gather more ready packets, resulting in a smaller throughput. Thus the extremely nonuniform frame pattern (1, 1, 1, 1, 1, 1, 1, 1, 1, 31) where most data slots are close neighbors will perform the best. As the traffic gets lighter (e.g., $G < 0.6$), a data slot which has a suitable interdistance with its last neighboring data slot so as to operate at an accumulated traffic load near 1 will produce a maximum throughput. Thus the optimal frame pattern is the pattern which produces the most such slots. This justifies that Fig. 2 has an optimal frame pattern (2, 2, 2, 2, 2, 2, 2, 2, 2, 22) as G is around 0.5, and (4, 4, 4, 4, 4, 4, 4, 4, 4, 4) as

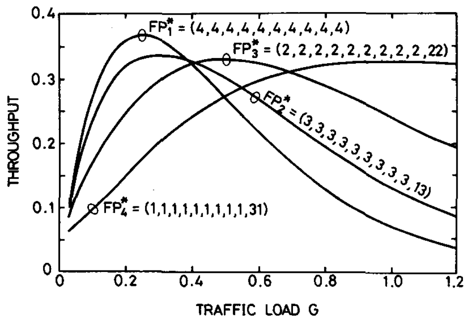


Fig. 2.

G is around 0.25. Also, we can see in our example that as the traffic load is getting lighter, most slots in the optimal frame pattern enlarge their interdistance with their last neighboring data slot, and finally, for small G, the optimal frame pattern is the pattern with data slots distributed over the whole frame as uniformly as possible.

A further investigation of Fig. 2 finds that the optimal frame pattern performs well in a preferred range and is not sensitive to a small change in G. In other words, the optimal frame pattern switches to the other one only at some discrete threshold of G and hence the total number of optimal frame patterns is finite. This fact is attributed to the reasons we used to explain the existence of an optimal frame pattern in some specific range of G in Fig. 2 and the restriction of integer solution for optimal frame pattern. Table I shows the optimal frame patterns for various N_t and G. Note that the total number of optimal frame patterns is 6, 4, 3 if N_t equals 5, 10, 15, respectively. Also notice that the finding of the optimal frame pattern (3, 3, 3, 3, 3, 3, 3, 3, 3, 13) in a small range of G at $N_t = 10$ identifies the value of careful search in simulated annealing. Furthermore, note that a maximum throughput 0.368 of a standard slotted ALOHA system occurs in the uniform optimal frame pattern (4, 4, 4, 4, 4, 4, 4, 4, 4, 4) at $G = N_t/N = 0.25$. This situation is as expected, since all the slots of the uniform frame pattern operate at an accumulated traffic load of 1 packet and thus the system still has a maximum throughput 0.368.

B. Throughput Improvement

Fig. 3 shows a throughput comparison between a system with optimal frame pattern assignment and a system with random frame pattern assignment if $N = 40$ and $N_t = 10$. (Note that a random frame pattern assignment is typical of a system which does not implement a time-slot reassignment protocol and which has no dedicated data slots.) Here, the mean throughput of the system with random frame pattern assignment is obtained by averaging the throughputs of frame patterns randomly selected from $\boldsymbol{U}(\boldsymbol{X}(N_t))$ over a quite large number. A great amount of throughput improvement is found, and the gain becomes larger with larger values of G. The gain ranges from 10% $\sim$ 120% and has nearly 25% improvement in moderate traffic at $G = (0.2, 0.6)$. The result is attractive and justifies the design of optimal frame pattern with time-slot reassignment protocol for our system.

TABLE I
THE OPTIMAL FRAME PATTERNS FOR VARIOUS RANGES OF TRAFFIC LOAD G AT $N_t = 5, 10,$ AND 15.

(A) $N_t = 5$

LOAD G	OPTIMAL PATTERN
(0.0000, 0.2188]	(8, 8, 8, 8, 8)
[0.2200, 0.2375]	(5, 5, 5, 5, 20)
[0.2388, 0.2888]	(4, 4, 4, 4, 24)
[0.2900, 0.4050]	(3, 3, 3, 3, 28)
[0.4063, 0.6925]	(2, 2, 2, 2, 32)
[0.6938, 1.2000]	(1, 1, 1, 1, 36)

(B) $N_t = 10$

LOAD G	OPTIMAL PATTERN
(0.0000, 0.3825]	(4, 4, 4, 4, 4, 4, 4, 4, 4, 4)
[0.3838, 0.4125]	(3, 3, 3, 3, 3, 3, 3, 3, 3, 13)
[0.4138, 0.6925]	(2, 2, 2, 2, 2, 2, 2, 2, 2, 22)
[0.6938, 1.2000]	(1, 1, 1, 1, 1, 1, 1, 1, 1, 31)

(C) $N_t = 15$

LOAD G	OPTIMAL PATTERN
(0.0000, 0.5062]	(2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3)
[0.5075, 0.6925]	(2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 12)
[0.6931, 1.2000]	(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 26)

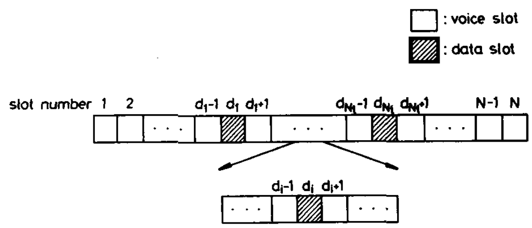


Fig. 3.

V. CONCLUSIONS

This paper develops a method for optimal frame pattern design in an integrated voice and data TDMA mobile communication system. The method is formulated as a combinatorial optimization problem, and a simulated annealing algorithm is applied to finding the global optimal frame pattern so that the system data throughput is maximum. The simulated annealing algorithm is reliable and effective. The numerical results show that there are various optimal frame patterns in different ranges of traffic load G and finite optimal frame patterns exist in global G range. The numerical results also show that the system can gain great throughput improvement by using optimal frame patterns.

VI. Acknowledgement

The authors wish to thank the anonymous reviewers for their valuable suggestions to improve the presentation of this paper, and to Prof. Po-Rong Chang for his valuable discussion about the simulated annealing algorithm.

References

[1] *Generic Framework Criteria for Universal Digital Portable Communication Systems,* FA-TSY-001013, Bellcore, Mar. 1990.

[2] D. P. Gaver and J. P. Lehoczky, "Channels that cooperatively service a data stream and voice message," *IEEE Trans. Commun.,* vol. COM-30, pp. 1151–1162, May 1982.

[3] F. I. Romeo, *Simulated Annealing: Theory and Application to Layout Problems,* Electronics Research Laboratory, University of California, Berkeley, 1989.

[4] E. H. L. Aarts and P. J. M. Van Laarhoven, "Statistical cooling: A general approach to combinatorial optimization problems," *Philips J. Res.,* vol. 40, no. 4, pp. 193–226, 1985.

[5] F. Catthoor and H. De Man, "Samirai: A general and efficient simulated-annealing schedule with fully adaptive annealing parameters," *Integration, The VLSI Journal,* pp. 147–178, 1988.

[6] P. G. Hoel, *Introduction to Mathematical Statistics.* New York: Wiley, 1962, pp. 345–349.

[7] A. A. M. Saleh and L. J. Cimini, Jr., "Indoor radio communication using Time-Division Multiple Access with cyclical slow frequency hopping and coding," *IEEE J. Select. Areas Commun.,* vol. 7, pp. 59–70, Jan. 1989.

[8] J. D. Parsons and J. G. Gardiner, *Mobile Communication Systems.* New York: Wiley, 1989.

[9] H. P. Stern, "Design issues relevent to developing an integrated voice/data mobile radio system," *IEEE Veh. Technol. Conf.,* pp. 57–64, June 1988.

[10] ——, "Design and performance analysis of an integrated voice/data mobile radio system," *IEEE Veh. Technol. Conf.,* pp. 300–305, May 1989.

[11] J. S. Dasilva, "Performance analysis of land mobile packet systems," Ph.D. dissertation, Dept. Syst. Comput. Eng., Carleton Univ., Ottawa, Ont., Canada, July 1980.

[12] M. J. Fischer and T. C. Harris, "A model for the performance of integrated circuit-and packet-switched multiplexing structure," *IEEE Trans. Commun.,* vol. COM-24, pp. 195–202, Feb. 1976.

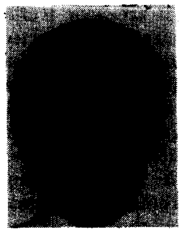
[13] E. Arts and J. Korst, *Simulated annealing and Boltzman machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing.* New York: Wiley, 1989.

[14] P. G. Hoel, S. C. Port, and C. J. Stone, *Introduction to Probability Theory.* Boston, MA: Houghton Mifflin, 1971, p. 129.

[15] D. J. Goodman, R. A. Valenzuela, K. T. Laird, and B. Ramamuirthi, "Packet reservation multiple access for local wireless communications," in *Proc. 38th IEEE Veh. Technol. Soc. Conf.,* May 1988, pp. 701–706.

[16] G. J. Coviello and P. A. Vena, "Integration of circuit/packet switching by a SENET (Slotted Envelope Network) concept," in *Conf. Rec. 1975 Nat. Telecommun. Conf.,* Dec. 1975, pp. 42-12 through 42-17.

[17] M. J. Ross, J. H. Gottschalck, and E. A. Harrington, "An architecture for a flexible integrated voice/data switch," in *Conf. Rec. 1980 Int. Conf. Commun.,* May 1980, pp. 21.6-1 through 21.6-5.



Chung-Ju Chang (S'85–M'85) received the B.E. and M.E. degrees in electronics engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1972 and 1976, respectively, and the Ph.D. degree in electrical engineering from National Taiwan University in 1985.

From 1976 to 1988 he was with the Telecommunication Laboratories, Directorate General of Telecommunications, Ministry of Communications, Republic of China, as a Design Engineer, Supervisor, Project Manager, and then Division Director. There, he was involved in designing digital switching system, ISDN user-network interface, and ISDN service and technology trials. In the meantime, he also acted as a Science and Technical Advisor for Ministry of Communications from 1987 to 1989. In August 1988, he joined the faculty of the Department of Communication Engineering and Center for Telecommunications Research, National Chiao Tung University, where he is currently Associate Professor. His research interests include performance analysis, broadband networks, and mobile radio networks.

Dr. Chang is a member of the Chinese Institute of Engineers (CIE).

Chen-Shiang Wu was born in Taiwan on November 3, 1967. He received the B.S. degree in electronics engineering from Catholic Fu Jen University, and the M.S. degree in communication engineering, from National Chiao Tung University in 1990 and 1992, respectively.

He is now with the Telecommunication Laboratories, Directorate General of Telecommunications, Ministry of Communications, Republic of China, as a design engineer. There, he is involved in digital subscriber loop technologies. His research interests include mobile communication and integrated services digital network.