quency response and causes more severe ISI. Each equaliser, with the exception of VDFE, has 15 taps: for MLFE, \( N_p = 9 \) and \( N_b = 5 \). The VDFE inherently has 8 taps. The value of \( M \) in eqn. 3 is set at 3 for MLE and MLFE. (Simulations with \( M = 10 \) were also performed; but the results are not shown here because they are almost identical to those with \( M = 3 \).) The carrier phase offset \( \theta = 1.571 \), and the step size \( \mu \) for the adaptation is \( 5 \times 10^{-3} \). Assuming binary DPSK, the tap coefficients were initially obtained by using \( 10^4 \) training data. The bit error rate (BER) values were empirically estimated by processing \( 10^7 \) binary input data.

Figs. 4 and 5 show the BER values for channels A and B, respectively. For comparison, the BER associated with coherently detected DFE (DFE/coherent) is also shown. As was expected, the performance of the MLE is close to that of the LE. The VDFE performed better than the LE and the MLE for channel B, but somewhat worse than for channel A. The MLFE outperformed all the other noncoherently detected schemes; furthermore, it is almost comparable to DFE/coherent.

Conclusions: A decision feedback demodulation-based equaliser compatible with differentially coherent PSK has been proposed. This equaliser, called the MLFE, has a linear feedback structure, but the equaliser output is modified by using the decision feedback demodulation algorithm before feedback. Through computer simulations, it was observed that the MLFE can perform much better than other existing equalisers, and that its performance is even comparable to that of a DFE with coherent detection.

Fig. 4 Bit error probability against \( E_b/N_0 \) for channel A

Fig. 5 Bit error probability against \( E_b/N_0 \) for channel B

Dual FH/MFSK system over a Rayleigh fading channel

Chanbum Park and Jae Hong Lee

Indexing terms: Frequency agility, Frequency hop communication, Frequency shift keying

A dual frequency-hopped/multilevel frequency shift keying (FH/MFSK) system is proposed for a Rayleigh fading channel. The dual FH/MFSK system is achieved by frequency-hopping a carrier frequency of the FH/MFSK system proposed by Goodman et al. The numerical results show that the proposed system allows more users than does the conventional system.

Introduction: Frequency-hopped/multilevel frequency shift keying (FH/MFSK) system was proposed for wireless communications to increase the number of users [1]. It was shown that an FH/MFSK system...
system allows many users to share the same frequency band by assigning a distinct tone sequence called an address to each user [1, 2]. In this Letter, a dual FH/MFSK system is proposed for a Rayleigh fading channel. The dual FH/MFSK system is achieved by frequency-hopping a carrier frequency of the FH/MFSK system proposed in [1]. The numerical results show that the proposed system allows more users than does the conventional system without increasing the total system bandwidth.

FH/MFSK system: The block diagram of a transmitter for a conventional FH/MFSK system is shown in Fig. 1a. In the conventional FH/MFSK system proposed in [1], q-bit binary input data are transformed to message $X_k$ of $M = 2^q$ levels with duration $T$, for the kth user ($k = 1, \ldots, K$). The message $X_k$ is divided into L chips each with duration $T/L$. The kth user has a distinct L-chip code address of $M$ levels $R_k (R_{k1}, \ldots, R_{kM})$. The transmitted signal of the kth chip is given by

$$Y_k = X_k \oplus R_k (mod M), l = 1, 2, \ldots, L,$$

for the kth user. The number of chips per symbol in the dual FWMFSK system is $L_q$. The transmitted signals are the MFSK signal of the Ith chip is given by

$$&_{I} = \frac{1}{L_q} \sum_{l=1}^{L_q} H_{I} e^{j2\pi \frac{f_l}{f_0}}$$

where $f_l$ is the frequency of the lth chip and $H_{I}$ is the transmitted signal of the Ith chip.

The block diagram of a receiver for a dual FH/MFSK system is shown in Fig. 1b. The dual FH/MFSK system divides the total available frequency band into $Q$ frequency slots and uses $M'$-level FSK, where $M'$ times $Q$ is equal to $M$ of the conventional FH/MFSK system. In the dual FH/MFSK system, $Q$ frequency slots are used for frequency hopping the carrier frequency of the FH/MFSK signal. The number of chips per symbol $L'$ is reduced to $L_q/q$ in order to maintain the data rate, where $q$ is the number of information bits per symbol in the dual FH/MFSK system. In this case, the effect of time diversity in each symbol is reduced. To overcome this reduction in time diversity, the parallel hop FH scheme [3] is used as the form of multiple addresses assigned to each user. The number of addresses assigned to each user is $H$ ($< q/q'$), which is less than or equal to the reciprocal of the time diversity reduction factor. The time-frequency matrix of a transmitted signal is shown in Fig. 2 as an example for the 8-chip, 16-level conventional FH/MFSK system and the 4-chip, 4-level, 4-frequency-hopping-slot dual FH/MFSK system with two addresses as an example.

Probability of error: In deriving the probability of error, the difference between the dual FH/MFSK system and the conventional system is the probability of insertion in each cell of time-frequency matrix of the received signal. The probability of insertion over $M$ possibilities, the probability of insertion due to other user interference of the conventional FH/MFSK system is given by

$$p = \left(1 - \frac{1}{M} \right)^{L_q} \left(1 - p_m \right)$$

where $K$ is the number of users and $p_m$ is the miss probability for each cell of time-frequency matrix, i.e., the probability that the receiver can miss a transmitted signal from the detection matrix. For a dual FH/MFSK system, assuming that an address is selected at random with equal probability over possibilities $M'$ and that a hopping frequency is selected at random with equal probability over possibilities $Q$, the probability of insertion due to other user interference is given by

$$p = \left(1 - \frac{1}{M'} \right)^{L_q} \left(1 - p_m \right)$$

A full detailed derivation of the probability of error can be obtained from [1].

Numerical results and discussions: The bit error probability of an FH/MFSK system for the number of users is shown in Fig. 4. The numerical results show the bit error probability of the 10-chip, 256-level ($q = 8$) conventional FH/MFSK system and that of the various $L'$-chip, $M'$-level ($q'$ bits), $Q$-frequency-hopping-slot dual FH/MFSK systems with $H$ addresses. It is shown that the dual FH/MFSK system allows more users than does the conventional system. At $10^{-3}$ bit error probability, the dual FH/MFSK system can support up to 50 users more than the conventional one.

In this Letter, the dual FH/MFSK system is proposed, which utilises the same total system bandwidth as the conventional FH/MFSK system. The proposed dual FH/MFSK system can support
more users than can the conventional system at the same bit error probability. Regarding complexity, although an additional frequency synthesiser is required for frequency hopping in the proposed system, the number of envelope detectors is reduced by a factor of $1/Q$.

© IEE 1996

Electronics Letters Online No: 19961273

Chunham Park and Jae Hong Lee (School of Electrical Engineering, Seoul National University, Shillim-dong, Gwanak-gu, Seoul 151-742, Korea)
e-mail: cpark@dosa.snu.ac.kr

References


Input queuing multicast ATM packet switch with two priority classes using a priority scheme with a window policy

W. Warde and P.A. Ivey

Indexing terms: Queueing theory, Asynchronous transfer mode, Packet switching

To achieve a high performance multicast switch with multiple priority classes, we have proposed a new $N \times N$ input queuing technique which uses a novel priority scheme and a high-performance arbitration algorithm. The scheme uses a priority without iteration (POI) method and a pipeline arbitration. Simulation results show that this scheme improves the performance of delay critical services and it can give almost the same performance as that which uses a priority with iteration (PWI) method.

Introduction: The main uses of ATM switching systems are in high-speed data communications and multimedia communications including voice, video, and data. To support integrated services, an ATM switch must be capable of handling the requirements of different services [1]. One possible way to meet the quality of service (QOS) requirements for different types of traffic is to assign two input queues for each input port, one for the high priority class, and the other for the low priority class [2, 3] and to assign a higher service priority to real-time traffic over non-real-time traffic.

In this Letter we propose a new priority scheme with a window policy which improves the performance of delay critical services. The technique is called a priority scheme with iteration (PWI) method. To avoid the iteration of the contention process and to alleviate the processing time constraint, we have developed a priority scheme without iteration (POI) method and with pipeline arbitration. A bufferless routing (crosspoint matrix architecture) with input buffering structure is proposed for the cell switch fabric. Each input port maintains an input buffer memory and a single queue (with window of $W$) for all outputs as shown in Fig. 1. Cells are of fixed length and the switch operation is slotted. Each cell carries a multicast pattern that is a bitmap of all output ports. To implement the POI algorithm, all bits in the $W$ cells, which indicate if the cell is to be accepted at the same associated output port, are grouped together by an OR gate. So for an $N$-input switch with window size $W$, each input buffer queue will serve all its $W$ cells as one cell with $N$ destination bits. The outputs of the OR gates represent these destination bits. To solve the out of sequence problem, we propose an input controller, for each input buffer queue, with $N$ parallel acknowledgement circuits (one for each destination) based on the concept of preventative control.

Operation: To ensure fairness among the input ports, input buffer queues are served in a cyclical fashion [4]. During each time slot, the input buffer queue which has the highest priority level sends its head of line (HOL) cell (all its copies regardless of its type) without a search. In all the other ($N-1$) input buffer queues, which have lower priority levels, two successive search phases will be performed in the range $(N-1) \times W$. The active cells in the first phase are the delay-sensitive cells. The active cells in the second phase are the delay-insensitive cells. In each phase, the $(N-1)$ queues are served on a queue-by-queue basis according to their priority levels. Each one of the $(N-1)$ input buffer queues sends at most one unicast cell per time slot, but not necessarily the first cell in the queue and no more than one unicast cell or one copy of a multicast cell per time slot. In the POI algorithm, the following process is repeated in the same way for each one of the $(N-1)$ queues. The free outputs are matched in ascending order against destinations (requests) of the input buffer queue. If there is no active cell within the window, $W$, with a copy destined to the first free output, a chance is given to the next free output in ascending order, $i < k \leq N$, a chance is given to the next free output in ascending order, $i < k \leq N$. This matching process is continued until either a free output is matched or all the free outputs are estimated. If all the free outputs are unoccupied, the input buffer queue is reported as being blocked in this phase. If a free output is matched, this input buffer queue is granted and the free output is occupied. If there is more than one active cell in the granted input buffer queue, within the window, $W$, that can occupy the free output, then the first active cell wins. According to the POI algorithm, the output ports can be matched against input buffer queues in a pipeline way. Hence, the first and the second phases can also be performed in a pipeline way (pipeline arbitration). The difference between the PWI algorithm and the POI algorithm is in the search mechanism, in each phase, inside each one of the $N-1$ queues.

Fig. 1 Block diagram of input buffer queue with its input controller for $2 \times 2$ switch

Fig. 2 Mean delay against average offered load for POI and PWI algorithms

(*) delay-sensitive class
(ii) delay-insensitive class
(iii) POI algorithm
(iv) PWI algorithm