Resource Optimization in CDMA based Wireless Ad Hoc Networks

Saurabh Srivastava†, Sachee Tripathi‡, Dheeraj Sanghi†, Ajit Kumar Chaturvedi‡

Abstract— **The introduction of the concept of ad hoc grouping of portable terminals to form a wireless network spurred a large amount of interest in the field of wireless data communications. CDMA had been shown to outperform other multiaccess schemes and IEEE specified the 802.11 set of standards for wireless LANs using a CDMA based physical layer. But the lack of proper provisions in 802.11 to provide a communication network without the existence of an access point resulted in extremely low data rates for truely ad hoc networks. In this project, we propose an MC-CDMA based physical layer for wireless ad hoc networks and present techniques to efficiently utilize network resources in a distributed manner.**

The network utilization in a CDMA system depends heavily on the codes being used. Using a graph theoretic approach to the problem we model the code allocation problem as a variant of graph coloring. When variable length orthogonal codes are employed for spreading, the allocation problem induces an interesting version of coloring which we call the *range sum* of a graph. We show the NP-Completeness of range sum and present an approximation algorithm for the same. We then propose a distributed version of the approximation algorithm for use in ad hoc networks and evaluate its performance by extensive simulations.

I. Motivation

THE IEEE standards for wireless communication networks, the IEEE802.11 series, have provisions for the use of Code Division Multiple Access for allowing multiple users to communicate simultaneously. But a closer inspection reveals that the standards do not fully exploit the advantages offered by a code division physical layer. They have given preference to ease of implementation over efficient use of network resources. Since it has been observed that in practice a wireless data network employing the currently available hardware, based on the IEEE standards, provides low data rates, their use is limited. In fact, the only factor that currently inhibits the employment of wireless networks for basic communication is the low data rates offered by them in comparison to wireline networks.

In a wireless Ad Hoc network the problems of low throughput are further exaggerated due to the distributed nature of control required. The standard makes heavy assumptions about the presence of an access point infrastructure and its protocols are in part customized for such a network. The facility for 'ad-hoc' operation is provided for completeness. Thus, a study of the the physical layer for alternative modulation techniques providing better bit-error-rates was required and protocols were required that would efficiently exploit the lower layer to provide better network throughput. In this project, we aimed to exploit the distributed nature of ad hoc networks as an advantage since it allows spatial reuse of resources and obtain better network utilization.

The authors are at the Department of Computer Science and Engineering† and at the Department of Electrical Engineering‡, Indian Institute of Technology, Kanpur.

E-mail: {srivs,sachee,dheeraj,akc}@iitk.ac.in

II. Introduction

A SET OF mobile terminals equipped with transceivers constitute a *mobile ad hoc* network. Considerations of conserving battery power necessitate the formation of a *multihop* network where the terminals limit their transmission powers and hence communicate with only a small set of in-range terminals, which are termed its *one-hop neighbors.* Packets are stored and retransmitted for destinations not in this set. Multihop networks also facilitate the reuse of resources at terminals which are spatially separated.

A. Physical Layer Design

For multiple transmitters which are unavoidably in range of each other, multiaccess schemes are employed which allow multiple transmitters to share the spectrum at the same time. Fixed allocation schemes such as TDMA (Time Division Multiple Access) and FDMA(Frquency Division Multiple Access) are impractical for implementation in mobile ad hoc networks because of the absence of any coordinating entity and because they lead to resource wastage under low loads. Code Division Multiple Access (CDMA) on the other hand allows all terminals to use the entire bandwidth at the same time and provides flexibility and graceful degradation capabilities to the system. CDMA is proving to be the most promising technique for the high data rate, large user networks currently demanded. CDMA tries to address the issues of robustness, multiplexing and resource sharing by allowing each user to transmit at the entire available bandwidth.

Many variants of CDMA have been proposed in the literature. The major schemes giving good performance are Direct Sequence (DS) CDMA, Multicarrier Direct Sequence (MC-DS) CDMA, Multitone (MT) CDMA and Multicarrier (MC) CDMA. We examine each of these in Section IV and compare their relative performance with respect to each other.

A design issue that greatly influences the performance of any particular multicarrier technique is the combining technique being used at the receiver of the CDMA system. Because of the frequency selective fading channel, at the receiver, signals at all the subcarriers of the multicarrier system have different amplitude levels and different phase shifts. Consequently, there is loss of orthogonality amongst the different users. To compensate for this distortion of orthogonality, in order to fully exploit the frequency diversity, it is important to use an efficient combining strategy. Various schemes proposed in the literature for achieving this combining are Orthogonality Restoring Combining(ORC), Equal Gain Combining(EGC), Maximal Ratio Combining(MRC), Minimum Mean Square Error Combining(MMSEC). We study these techniques in Section V-C and evaluate their performance for the case of a MC-CDMA receiver in Section VI.

B. *Resource management and Throughput maximization in a CDMA system*

In a CDMA system, the set of codes is the resource that is to be distributed. Spreading codes are sequences of pseudo random bits that are used to expand the bandwidth occupied by a narrowband signal which is termed as the spreading operation in a CDMA system. Orthogonal codes have good cross-correlation properties and hence are employed for allowing simultaneous transmissions by various users. Instead of using fixed length codes for the entire network, orthogonal codes of variable length have been proposed in [1]. Since the length of a code also dictates the bit rate designated to a user, using codes of smaller length provides larger data rates and hence smaller length codes should be preferred to achieve greater throughput. But the number of such mutually orthogonal codes decreases with decreasing length of the codes. Hence, there is a need to *spatially reuse* codes, i.e. codes need to be reused at terminals which are not in hearing range of each other. But allocation of the same code to multiple terminals gives rise to the possibility of space and time overlap of two or more packet receptions on the same code, called a *collision or interference*, which results in corrupted bits being received at the receiver.

Collisions increase the number of retransmissions and decrease the throughput. In [2], interference has been described as either being *direct* or *secondary*. Direct interference occurs when two nodes simultaneously initiate transmissions to each other. Such collisions can be avoided by proper scheduling at the MAC layer. Secondary interference occurs at a receiver due to simultaneous transmissions by two transmitters who cannot hear each other. They presented a protocol which allocates a code to each terminal which is different from the codes allocated to its two-hop neighbors. This can be seen to eliminate collisions but requires a large number of codes in the worst case.

The code allocation problem for eliminating secondary collisions has been shown to be NP-Complete[3] and therefore several heuristic solutions have been presented [3], [4]. Due to the known intractability of the problem there has been little work on bounding the worst case behavior of code allocation techniques and none on the increased throughput obtained by using codes of smaller length while doing code allocation.

In fact, the use of variable length codes provides optimal allocation of the radio spectrum allowing the code allocation to be independent of worst case scenarios that might exist locally. If we were to use fixed length codes, the smallest length of the code would be dominated by the minimum required at any place in the network. Suppose some part of the network had an abnormally high density of mobiles. Then the length of the code used here would be extremely large and since the codes are fixed length, the throughput in the whole network (which is related to the length of the codes), would be low. On the other hand, if we were to use variable length codes then we could allocate larger length codes in this part of the network and use shorter lengths in parts where the density was low, thus achieving higher throughput.

We introduce the concept of the *Range Sum* of a graph which is used to provide an analytical basis for design of the code allocation scheme which maximizes throughput in an Ad Hoc network.

C. *Organization of this report*

The rest of the report is composed of two parts discussing the two components of the project. The former part concentrates on the PHY layer aspects of our investigation and the latter on the issue of resource optimization protocols that exploit the CDMA based PHY layer.

In Section III we present the preliminary definitions and assumptions (The channel model, spreading codes and network model are discussed). Section IV compares the various spreading techniques proposed in the literature. Section V explains the components of the MC-CDMA transceiver system, the transmitter, the receiver and outlines the combining strategies employed at the receiver.

The performance of an MC-CDMA system with emphasis on the impact of the various combining strategies on the BER is studied through simulations and the results are presented in Section VI. The four combining schemes ORC, EGC, MRC and MMSEC are studied for different channel conditions and different number of users. Finally based on the simulation results, the design of the PHY layer is presented in Section VII.

In Section VIII we discuss the unique problems that an ad hoc network induces followed by existing solutions. Then the problem of code allocation is introduced in Section IX and a static code allocations scheme is discussed. We formulate the code allocation problem in graph theoretic terms in Section X. The rest of the report is devoted to modelling the problem of code allocation in an MC-CDMA based ad hoc network with a throughput maximization criterion as a new variant of the graph coloring problem and present graph-theoretic analysis of the problem. In Section XI we introduce the concept of the *Range Sum* of a graph and present its NP-Completeness along with an approximation algorithm for the same. In Section XII we present techniques to allocate codes for maximizing aggregate throughput based on the approximation solution to the range sum problem.

Section XIII presents the simulation parameter and results for the second set of simulations which were performed

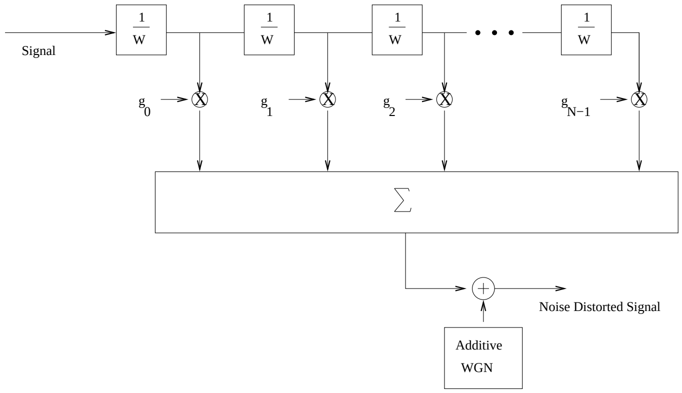

Fig. 1. Tapped Delay Line Model

to evaluate the performance of our dynamic variable length code allocation scheme against the static allocation currently in use and as presented in Section IX-B.

Finally in Section XIV and XV we present the conclusions and future work respectively.

III. Preliminaries and Assumptions

A. Channel Model

For the first part of the investigations where we evaluate the performance of the modulation techniques we take the channel to be frequency-selective and fast Rayleigh fading. We consider the equivalent low-pass impulse response $h(\tau;t)$ which is characterised as a complex-valued random process in the variable t[5]. We assume $h(\tau;t)$ to be wide-sense-stationary, with uncorrelated scattering and having L received paths. As shown in Figure 1 we model the time variant frequency selective channel as a tapped delay line with tap spacing $1/W$ and tap weight coefficients $\{g_l(t)\}$, where $\{g_l(t)\}$ is a set of time varying channel coefficients for the lth path.

$$g_l(t) = \frac{1}{w} c(\frac{l}{w};t) \tag{1}$$

where W is the bandwidth occupied by the real bandpass signal.

The low pass impulse response of the channel is

$$h(\tau;t) = \sum_{l=1}^{L} g_l(t)\delta(\tau - \tau_l) \tag{2}$$

where j is user index, t and τ are the time and the delay, respectively, $\delta(\cdot)$ is the Dirac delta function, τ_l is the propagation delay for the lth path. For the case of Rayleigh fading, as considered here, the amplitudes of the coefficients are Rayleigh distributed and the phases $\phi_l(t)$ are uniformly distributed.

The channel also adds White Gaussian Noise (WGN) to the output of the tapped delay line.

B. Spreading Codes

B.1 Orthogonality of Codes

Two codes are said to be orthogonal when their inner product is zero. The inner product, in the case of codes with elements values $+1$ and -1, is the sum of all the terms we get by multiplying two codes element by element. For example, (1, 1, 1, 1) and (1, 1, -1, -1) are orthogonal: $(1*1)+(1*1)+(1*-1)+(1*-1) = 0$

B.2 Walsh Hadamard Codes

Each user is assigned a signature code from an orthogonal set. In this study the codes are chosen from the Walsh-Hadamard code set defined recursively as

$$H_n = \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix} \tag{3}$$

with the base value

$$H_0 = [1]$$

Each row of the $2^n \times 2^n$ matrix, H_n, gives the 2^n bit code for a user.

B.3 Orthogonal Variable Spreading Factor Codes

CDMA supports different spreading factors therefore it is possible for the channelization codes to have different lengths. Thus, to find a set of codewords (different lengths) that are orthogonal, an algorithm based on the code tree as shown in Figure 2 is used. Codes on different levels of the tree have different lengths. And during spreading, longer codes result in lower bit rates. When two messages with different codewords and spreading factors are transmitted at the same time, the shorter codeword, modulated by its information message, will get repeated a number of times for each transmission of one longer codeword. This would mean that to maintain orthogonality, the longer codeword should not be derivable from the shorter one. Referring to the tree structure, it means that if a codeword is assigned to a certain user, then all the codewords below that assigned codeword cannot be used, as they are not orthogonal to it [1].

A tree structured method for generating such orthogonal codes with different lengths was proposed in [1]. The set of N codes, $C_N(i)'s$, at one level of the tree are generated from the previous level by generating two codes for each code at level $N-1$; i.e. $C_N(2i) = C_{N-1}(i) \oplus \overline{C_{N-1}(i)}$ and $C_N(2i-1) = C_{N-1}(i) \oplus C_{N-1}(i)$, where $\oplus$ denotes the concatenation operation. The codes at level N have a Spreading Factor of N and thus using OVSF codes we can provide variable data rates by allocating codes at different levels. All codes are orthogonal to each other except in the case when one is an ancestor of the other in the code tree. The code tree upto level 3, consisting of a total of 7 codes, is shown in Figure 2.

OVSF codes allow us to allocate variable spread factor codes, resulting in variable data rate allocation to the terminals, while still maintaining orthogonality. A pair of

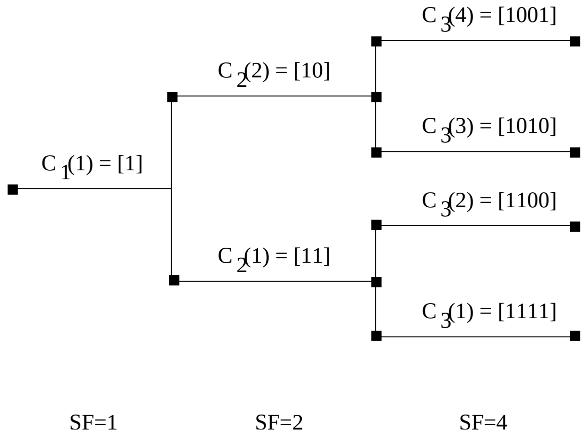

Fig. 2. Orthogonal Variable Spreading Factor Code Tree shown up to level 3.

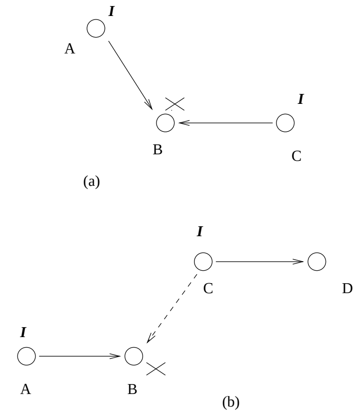

Fig. 3. (a) Interference occurring at the receiver by simultaneous transmission by two senders using the case code I. (b) Interference occurring at a receiver due to overhearing of another transmission at the same code I.

(possibly different length) codes in this set are orthogonal to each other except when one is derived from the other.

C. Network Model

We consider a multihop ad hoc network of mobile terminals for evaluating the performance of code allocation schemes in the second part of the investigation. All terminals communicate through a common wide-band transmission channel using CDMA. A symmetric connectivity channel is assumed, i.e. if i can receive a transmission from j then it is assumed that j can receive any transmission from i. Variable length spreading codes are used to spread the data bit over the entire spectrum and each transmitter-receiver pair mutually agree on a code before initiating communication.

Receiver terminals are equipped with the facility to broadcast the code on which they are receiving their transmission. This is done on a broadcast channel (time slot) during the Medium Access Control layer ACK phase. If the MC-CDMA[6], [7] is being used, then this can be done easily without extra hardware and without encroaching on the data transmission, as shown in [8].

A CDMA system with code reuse is prone to the problem of secondary interference. Two possible situations in which such interference can occur, as observed in [2], are shown in Figure 3. Here, A and C are transmitters transmitting on code I to the receiver shown by their respective arrows and cannot hear each other's transmissions. In both situations a collision occurs at B. In a dynamic code allocation scheme if each transmitter were to observe the codes being used in its neighborhood before deciding its own code then there would have been no collisions in both the cases illustrated above. Thus, for the situations in Figure 3, a broadcast by the receiver B would have allowed the transmitter which started transmitting later (A or C) to avoid the code I.

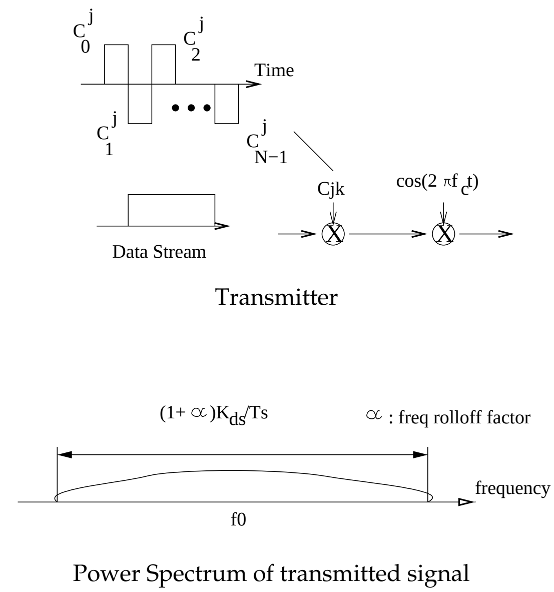

Fig. 4. DS-CDMA Transmitter and transmitted power spectrum

IV. Comparison Of Spreading Techniques

Before going on to describe our physical layer in detail, we will do a cursory examination of other spreading techniques and compare them with our choice: MC-CDMA.

A. DS-CDMA

The DS-CDMA transmitter spreads the original data stream using a given spreading code in the **time domain**. The system model is illustrated in Figure 4

Bandwidth: (main lobe)

$$\begin{cases} B_{DS} = \quad 2K_{DS}/T_S, & for\, rectangular\, pulse\; waveform \\ B_{DS} = \quad (1+\alpha)K_{DS}/T_S, & (0 \leq \alpha \leq 1),\; for\, Nyquist\, waveform\; with\, rolloff\; factor\, \alpha \end{cases}$$

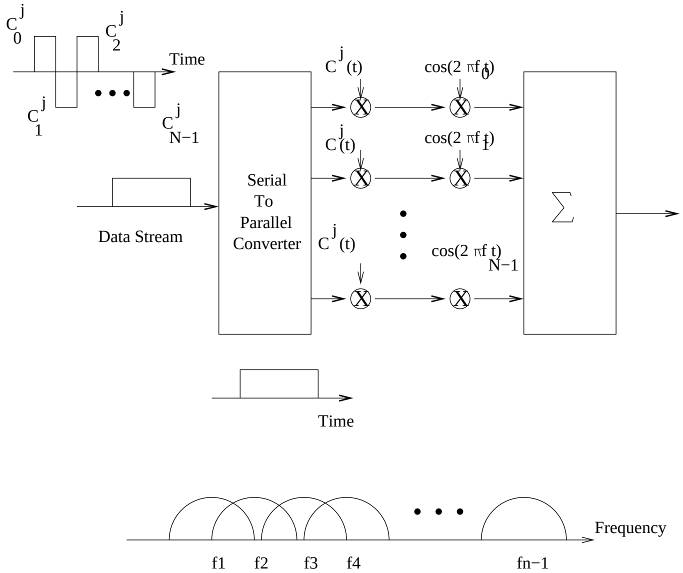

Fig. 5. Transmitter structure and power spectrum for MC-DS-CDMA

B. MultiCarrier-CDMA

The MC-CDMA system is described in Section V.

C. Multicarrier DS-CDMA

This spreads the the S/P converted data streams using a given spreading code in the time domain so that the *resulting spectrum of each subcarrier can satisfy the orthogonality condition* with the minimum frequency separation [9]. The system is shown Figure 5

D. MultiTone-CDMA

The MT-CDMA transmitter spreads the S/P converted data streams using a given spreading code in the time domain so that the spectrum of each subcarrier *prior to spreading operation can satisfy the orthogonality constraint* with the minimum frequency separation [9]. The system is shown in Figure 6.

E. BER performance

The BER performance of the different spreading techniques is shown in Figure. 7.

F. System Performance Comparison

Shown in Table I is the comparison of different techniques wrt. various parameters.

V. MC-CDMA System

MC-CDMA is a digital modulation technique where a single data symbol is transmitted at multiple narrowband subcarriers where each subcarrier is encoded with a phase

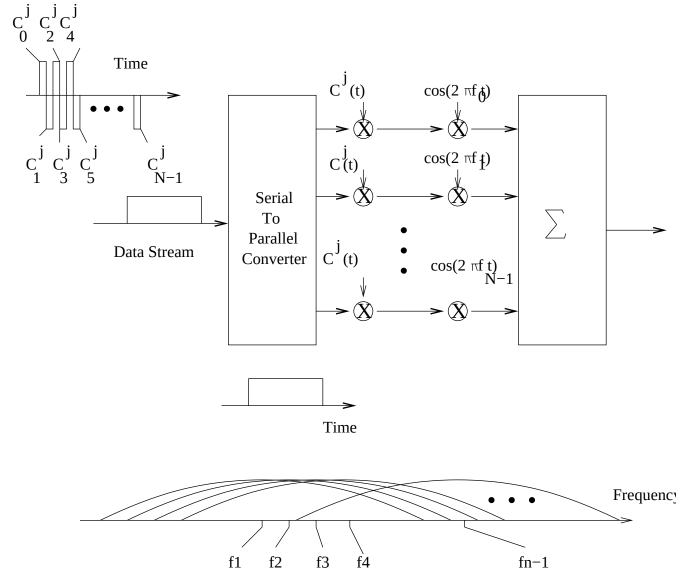

Fig. 6. Transmitter structure and power spectrum for MT-CDMA

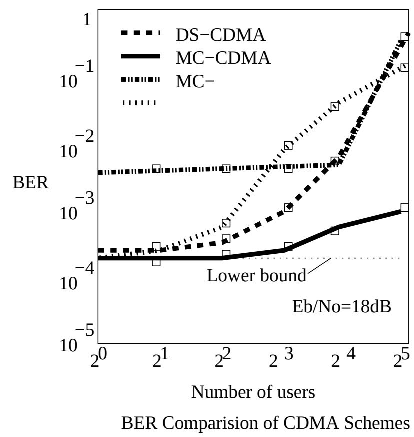

Fig. 7. BER Performance Comparison

offset of 0 or π based on a spreading code, as opposed to a convential OFDM which sends different data bits over different subcarriers. The narrowband subcarriers are separated by a frequency $1/T_b$ at baseband, where T_b is the symbol duration. This results in the subcarriers being orthogonal to each other at baseband. At the receiver, by multiplying with the particular frequency of interest and summing over a symbol duration, we can isolate the symbol component at that subcarrier [10].

MC-CDMA is a combination of DS-CDMA (Direct Sequence CDMA) and the OFDM (Orthogonal Frequency Division Multiaccess) techniques and inherits all the good and bad properties of its parents[9]. MC-CDMA transmits the same symbol over a number of parallel subcarriers, where the number of subcarriers is determined by the processing gain, such that the original baud rate remains unchanged. The frequency diversity thus obtained is used to combat frequency selective fading. Since there is considerable redundancy in the frequency domain the transmitter can afford to lower its power levels at each of the subcarriers. As CDMA systems have evolved from being noise limited

TABLE I

Performance Comparison of different CDMA techniques

	DS-CDMA	MC-CDMA	MC-DS-CDMA	MT-CDMA
Symbol duration at subcarrier	T_S	$N_C T_S / G_{MC}$	$N_C T_S$	$N_C T_S$
Number of subcarriers	(1)	N_C	N_C	N_C
Processing Gain	G_{DS}	$G_{MC} \approx G_{DS}$	$G_{MD} = G_{DS}$	$G_{MT} = G_{DS}$
Chip Duration	T_S / G_{DS}		$N_C T_S / G_{MD}$	$N_C T_S / G_{MT}$
Subcarrier separation		$1/T_S$	$G_{DS}/(N_C T_S)$	$1/(N_C T_S)$
Required Bandwidth	$\frac{G_{DS}}{T_S}^{\star}$	$\frac{N_C+1}{N_C} \cdot \frac{G_{MC}}{T_S}$	$\frac{N_C+1}{N_C} \cdot \frac{G_{MD}}{T_S}$	$\frac{N_C-1+2G_{MT}}{N_C T_S}$

to being interference limited, this lowered energy at each subcarrier allows the system to admit more users than the conventional schemes. To make full use of the received signal energy (scattered in the time domain for DS-CDMA) the DS-CDMA receiver is limited by the number of Rake fingers that can be deployed. However an MC-CDMA receiver can always make full use of the received signal energy since it is scattered in the frequency domain.[9]

A. Transmitter

The MC-CDMA transmitter, as shown in Figure 8(a), replicates d_i^j, the ith bit of the jth user, into N copies, where N is the number of subcarriers. Each copy thus obtained is multiplied by C_i^j, the ith bit of the signature code assigned to the jth user. This operation can be seen as spreading in the frequency domain of the data stream on to the available bandwidth W. These copies, in turn, modulate the N subcarriers, where the subcarrier separation, Δf, is $1/T_b$. These N components are then added together to obtain the transmitted signal, $s^j(t)$, for the jth user.

The transmitted signal for jth user, $s^j(t)$, is given by:

$$s^j(t) = \sum_{i=-\infty}^{+\infty} \sum_{m=0}^{N-1} C_m^j d_i^j e^{j2\pi m \Delta f (t - iT_b)} p_{T_b}(t - iT_b) \qquad (4)$$

where $p_{T_b}(t)$ is the rectangular symbol pulse waveform.

The power spectrum of the transmitted signal, as shown in Fig. 8(b), can be seen to be a combination of N narrowband signals, at frequencies f_o to f_{N-1}, with the constant frequency separation Δf.

A.1 Determination of the Optimal Values for N_C and Δ.

As is intutively explained in Figure 9 there should exist an optimal value in N_C and Δ to minimize the BER [11].

Number of subcarriers: When the transmission rate is given, the transmission performance becomes more sensitive to the time selectivity as the number of subcarriers (N_C) increases because the wider symbol duration is less robust to the random FM noise, whereas it becomes poor as N_C decreases because the wider power spectrum of each subcarrier is less robust to the frequency selectivity.

Guard interval: On the other hand, the transmission performance becomes poor as the length of the guard interval Δ increases because the signal transmission in the guard duration introduces the power loss, whereas it becomes more sensitive to the frequency selectivity as Δ decreases because the shorter guard duration is less robust to the delay spread.

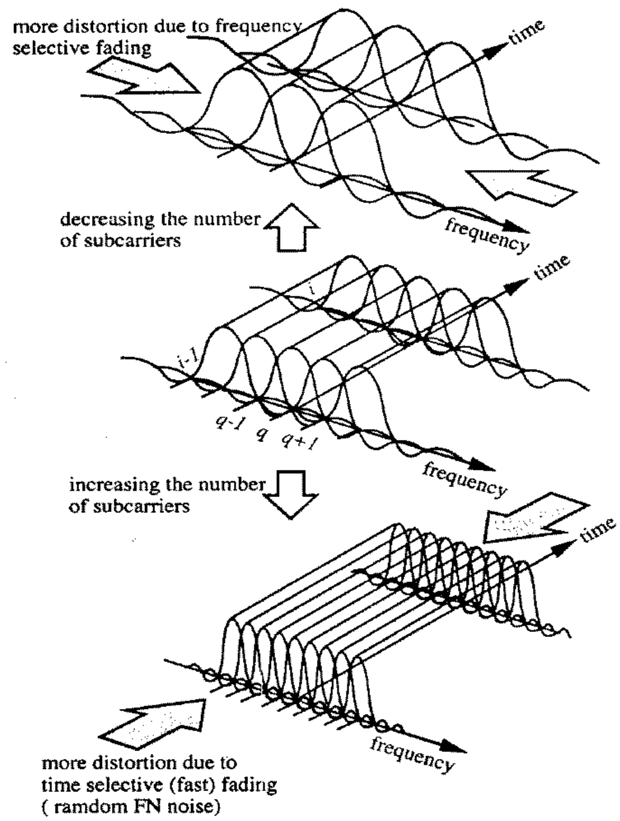

Fig. 9. Determination of optimal values of N_C and Δ.

Therefore, for a given transmission rate, doppler spread and delay spread, there exists an optimal value to minimise the BER in both N_C and Δ .

N_{opt} and Δ_{opt} must maximise the autocorrelation function of the qth subcarrier for the jth user, because it means a measure to show how much the received signal can be distorted in the time-frequency-selective fading channel.

In the case of the channel characteristics being:

- delay spread: 20ns
- doppler spread: 10Hz
- transmission rate: 3Msymbols/s (BPSK format)

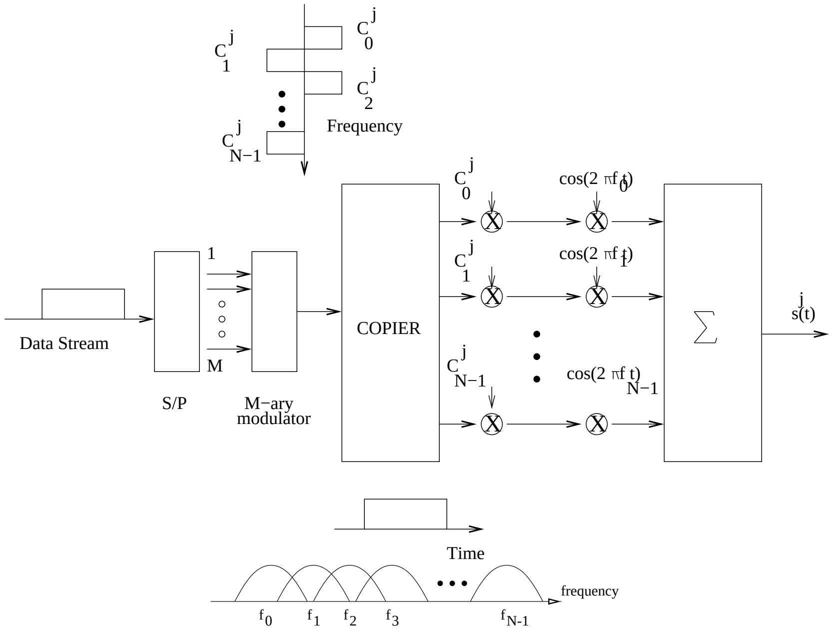

Fig. 8. (a)Transmitter and (b)Power Spectrum of the transmitted signal.

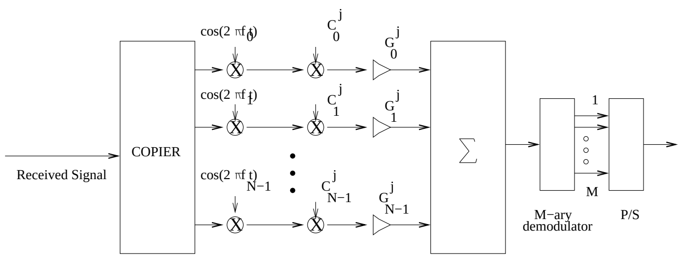

Fig. 10. Receiver Model

• walsh hadamard codes with length=32 for MC-CDMA and gold codes of length 31 for DS-CDMA

The optimal values are [N_{opt} ,Δ/T_S'] = [256 , 0.015], which means that the original data sequence is first converted into eight parallel sequences and then each is mapped onto 32 subcarriers. The bandwidth required for MC and DS are 97.8 and 186 MHz respectively: A factor of 1.9 !!!

B. Receiver

The received signal is [11]

$$\begin{aligned} r(t) &= \sum_{j=1}^{J} \int_{-\infty}^{+\infty} s^j(t-\tau) \otimes h^j(\tau;t) d\tau + n(t) \\ &= \sum_{i=-\infty}^{+\infty} \sum_{m=0}^{N-1} \sum_{j=1}^{J} z_m^j(t) d_i^j C_m^j p_s(t - iT_b) e^{j2\pi m \Delta f t} \\ &\quad + n(t) \end{aligned} \tag{5}$$

where $z_m^j(t)$ is the received complex envelope at the mth subcarrier.

The receiver model is as shown in Figure 10.

The depsreading operation is the inverse of the spreading operation done at the transmitter. The individual components, contributing to the decision variable, can be obtained by demodulating with their respective carrier frequencies, followed by multiplication by the corresponding code bit and the gain at that subcarrier, G_m^j, and integrating over one symbol duration. The gain G_m^j is used to compensate for the distortion, in the amplitude and phase, introduced by the channel at the mth subcarrier.

Finally, the decision variable is given by

$$v^j(t = iT_b) = \sum_{m=0}^{N-1} G_m^j y(m) \tag{6}$$

$$y(m) = \sum_{j=1}^{J} z_m^j(iT_b) d^j c_m^j + n_m(iT_b) \tag{7}$$

where are $y(m)$ and $n_m(iT_b)$ are the complex baseband component of the received signal and and the complex additive gaussian noise at the mth subcarrier at $t = iT_b$, respectively.

C. Combining Schemes

Various combining strategies have been proposed to determine the gain. The underlying rule of these techniques should be to reduce the effect of fading and the interference while not enhancing the effect of noise on the decision. The equalization techniques are not derived with respect to any optimization criterion but are essentially heuristics. [10]They are desirable for their simplicity. In the following, we consider EGC, MRC, ORC and MMSEC. These techniques can be associated with classical diversity theory as they involve multiplying each copy of the signal by some gain factor. Each of these techniques will affect the distribution of noise component differently.

C.1 Equal Gain Combining (EGC)

This technique does not attempt to equalize the effect of the channel in any way. The gain factor is given by [10]

$$G_m^j = c_m^j z_m^{j\star} / \left| z_m^j \right| \tag{8}$$

which implies that the multiplication with the received signal's complex envelope will result in the amplitude being passed as it is.

C.2 Maximum Ratio Combining (MRC)

We expect that the component signals with larger amplitude will have a lesser noise component. Therefore by effectively squaring the amplitude we will increase the influence of these subcarrier components on the decision variable. The gain factor is given by [10]

$$G_m^j = c_m^j z_m^{j\star} \tag{9}$$

C.3 Orthogonality Restoring Combining (ORC)

In this scheme we try to cancel the effect of the channel and restore the orthogonality amongst the users. The gain is given by

$$G_m^j = c_m^j z_m^{j\star} / \left| z_m^j \right|^2 \tag{10}$$

However in this scheme noise amplification takes place at the low level subcarriers which get mulitplied by high gains.

C.4 Minimum Mean Square Error Combining (MMSEC)

The gain in this scheme is given by

$$G_m^j = c_m^j z_m^{j\star} / \left(\sum_{j=1}^{J} \left| z_m^j \right|^2 + \sigma^2 \right) \tag{11}$$

For small$\left| z_m^j \right|$ the gain becomes small to avoid noise amplification and for larger $\left| z_m^j \right|$the gain restores orthogonality by dividing by the subcarrier envelope. This scheme makes use of the complex envelopes of all the users and so is prohibitively complicated.

VI. Simulations and Numerical Results for the PHY Layer Design

A. System Design

For the simulations we have developed a simulator, that models the transceiver system for J users as shown in Figure.11. We take a bit rate of 1Mbps, a processing gain of 128 and 7-path delay profile, commonly found in urban areas [12], [13]. At the receiver end, full knowledge of the channel coefficients is assumed. The Doppler spread is taken to be 10Hz.

The complexity inherent in the parallel system of subcarriers which require the use of a number of filters and modulators, etc., is greatly reduced by eliminating any pulse shaping and by using the IDFT/DFT for modulation/demodulation process [14], [12].

B. BER Performance

B.1 BER comparison at different loads on the system.

At a E_b/N_o value of 10dB and BPSK modulation, the BERs obtained are shown in Figure 12.

It can be seen that MRC performs satisfactorily only for small number of users, thereafter it gives the worst BER of all the schemes. This is due to the fact that at low loads the system is noise limited and by squaring the amplitude, MRC only combats the noise while it destroys the orthogonality. Therefore it is not suitable for interference limited channels.

MMSEC results in the best BERs of all the schemes, because it takes into consideration the noise in the channel and the number of interfering users. However this introduces complexity into the receiver implementation.

EGC and ORC lie between MMSEC and MRC in that order. Since in EGC the interference is not multiplied, and ORC attempts to restore orthogonality they are both suitable for interference limited channel, and might be preferred for their simplicity.

B.2 BER comparison at different channel conditions.

Figure 13 shows the performance at E_b/N_o values in the range of 0 to 20dB with 64 users and BPSK modulation.

Performance of MRC does not show any improvement even as E_b/N_o increases because of the high interference in the channel. ORC is able to successfully restore orthogonality as E_b/N_o increases, hence its BER steadily decreases. EGC also shows a steady improvement in the performance. MMSEC outperforms the other schemes discussed here because in addition to restoring orthogonality at high E_b/N_o, it also takes care of the noise dominant at low E_b/N_o.

B.3 BER comparision for M-PSK modulation.

Increasing the order of modulation does not affect the relative merits of the considered combining schemes as shown in Figure.14 for M-ary PSK, where M is the order of the modulation, taken at an SNR of 15dB, a processing gain of 128 and number of users equal to 64. The BER performance is satisfactory only for low values of M. In these simulations, the bit rate is taken to be constant so that as M increases the transmitted symbol duration increases, making it more susceptible to time selective fading.

The symbol constellation obtained for 8-PSK is shown in Figure 15.

B.4 BER comparison for M-ary Quadrature Amplitude Modulation

The BER graphs obtained by varying the order of modulation is shown in Figure 16. The graph is similar to that

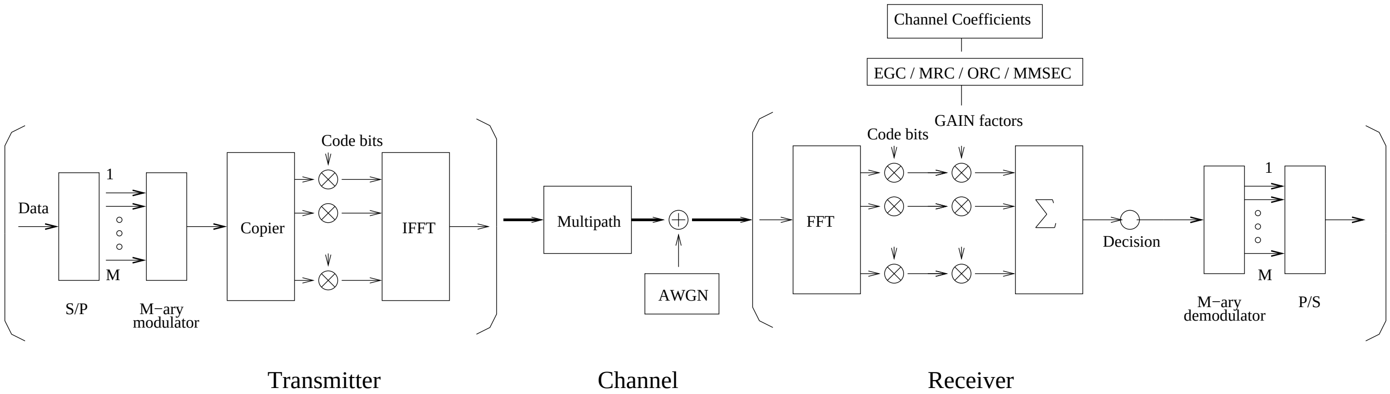

Fig. 11. MC-CDMA System Block Diagram for a single user

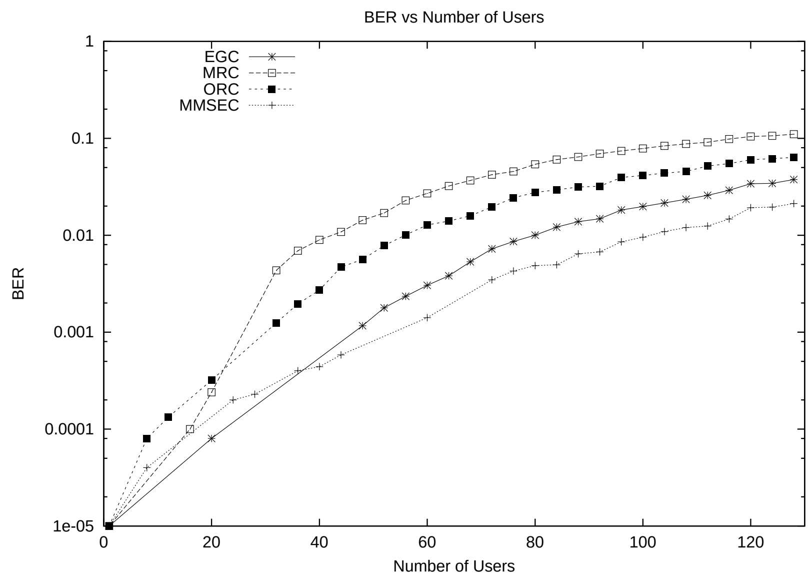

Fig. 12. **Simulation Results:** BER Performance of EGC, MRC, ORC and MMSEC against the number of users.

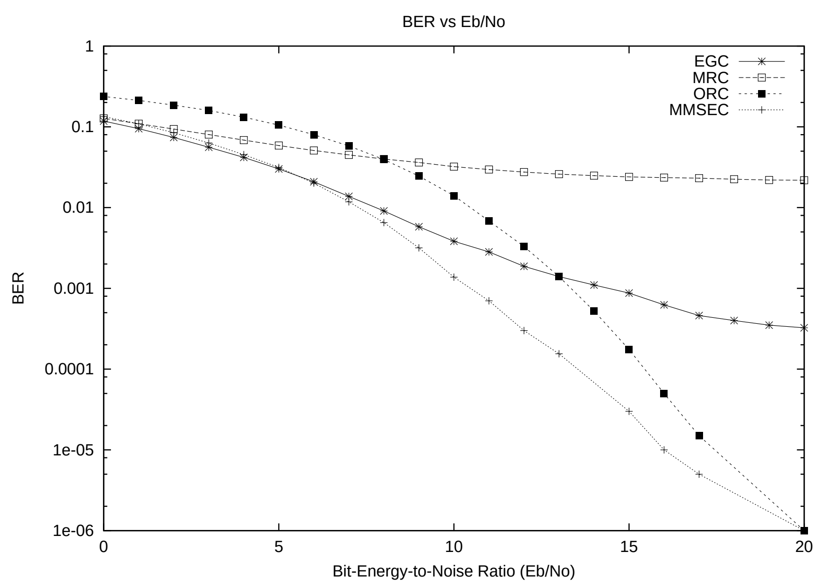

Fig. 13. **Simulation Results:** BER Performance of EGC, MRC, ORC and MMSEC against E_b/N_o

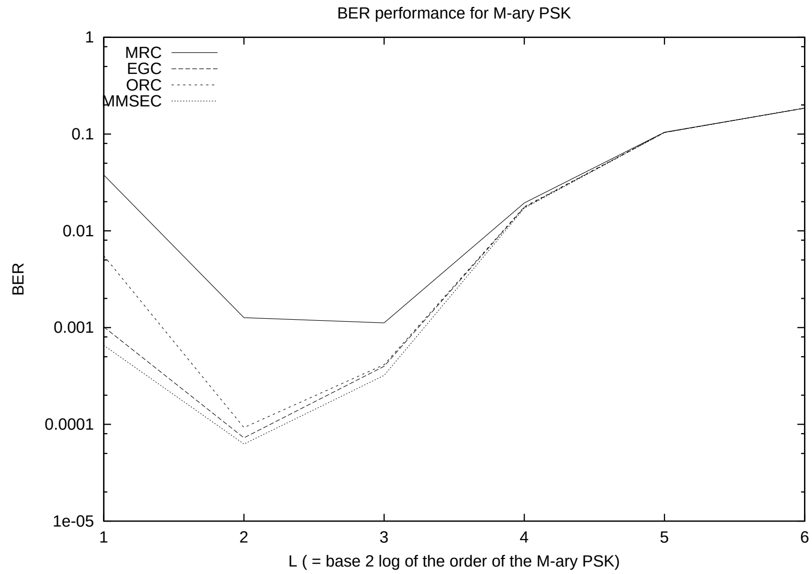

Fig. 14. **Simulations results:** BER performance at M-ary PSK modulation. ($L = \log_2(M)$)

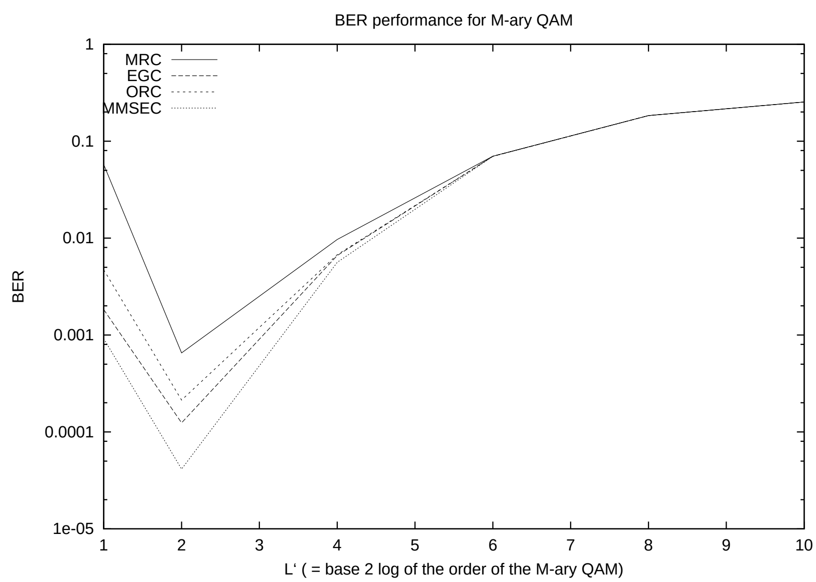

Fig. 16. **Simulation Results:** BER performance at M-ary QAM. ($L = \log_2(M)$)

obtained for MPSK. For very high number of signalling points the probability of error increases and so BER should show an increase as is the case. The dip while going from 2QAM to 4QAM can be explained by the decreased bitrate in 4QAM.

VII. Summary of Physical Layer Design

Based on the simulations done in the project, MC-CDMA was shown to be a most promising technique. Therefore we propose the following physical layer design:

- Spreading Technique: MultiCarrier-Code Division Multiple Access (MC-CDMA).
- Modulation: Quadrature Phase Shift Keying (QPSK).
- Combining Strategy: Minimum Mean Square Error Combining (MMSEC).

MC-CDMA also happens to be one of the contenders for 4G wireless [15]. In addition, the simple implementation of the MC-CDMA transmitter allows us to suggest modifications that will help the MAC layer in its functioning.

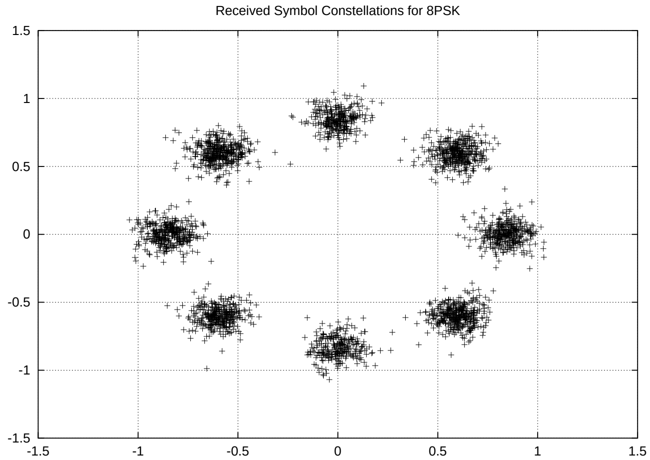

Fig. 15. **Simulation Results:** Received Symbol Constellation for 8PSK

VIII. Resource Management in Wireless Ad Hoc Networks

A major issue in wireless networks is developing efficient medium access protocols that optimise spectral reuse, and hence, maximize aggregate channel utilization. Multiple access based collision avoidance MAC protocols work on the principle that the sender and receiver first 'acquire the floor' before initiating transmission. We are using the term 'floor' in a very generic context to mean the resources available. This could be the bandwidth being used by a transmission in a frequency allocating system, the number of time slots in a time-slotted system or the amount of power being used for a transmission in a power controlled MAC. Due to the lack of a coordinating authority, the nodes involved will have to ensure that they acquire this floor in a consistent manner and the nodes in the vicinity should be aware of all such allocations.

But any kind of floor acquisition will inevitably lead to problems which we shall discuss in Section VIII-A. We then propose our solution to the problem of optimizing resource usage in wireless ad hoc networks by observing and exploiting the advantages offered by using variable spreading factor codes. In Section XI we introduce the **Range Sum** of a graph. If we use variable length orthogonal codes then the problem of code allocation for maximizing throughput induces an interesting version of the graph coloring problem. The code allocation with this maximization criterion will be formulated as an instance of range sum in Section XII. This is used later for designing a distributed greedy algorithm for wireless ad hoc networks.

A. Problems

A.1 Hidden/Exposed node problem

In ad hoc networks nodes have to contend to resolve the hidden and exposed nodes problems. Consider the scenario of three mobile hosts in Figure 17. Hosts A and B are within each other's transmission range and so are B and C. However A and C cannot hear each other. When A is transmitting to B, since C cannot sense A's transmission it may falsely conclude that the medium is free and start transmitting to B causing a collision. The problem that a station cannot detect a potential competitor because the competitor is too far away is called the *hidden terminal problem.*

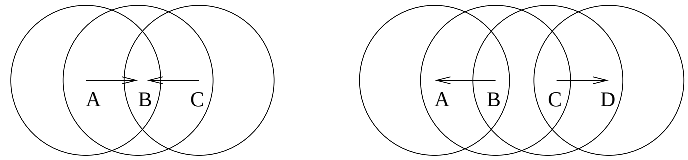

Fig. 17. Hidden and Exposed Node Problems

In Figure 17, when B is transmitting to A, host C can sense the medium and will conclude that it cannot transmit. However if D is the intended recipient then such a transmission can actually be granted. Such inefficiency in channel use is called the *exposed terminal problem.*

A.2 Excess connectivity

In networks with high density of nodes, there can potentially be very high connectivity, ie each node will have a large degree in the graph. This can lead to low channel utlization, since any one node transmitting will shut down all its neighbors (which are are large) in the case of single channel system. Moreover, such large degrees are unwanted in that the network could function suitably even in the presence of lesser number of neighbors for each node, and so the nodes are essentially wasting their power by being connected to such a large number of relatively far away nodes.

B. Existing solutions

Various solutions have been proposed to minimise the above problems:

B.1 RTS/CTS based collision avoidance

To alleviate the problems of hidden terminals disrupting ongoing transmissions the RTS/CTS mechanism was proposed. When a node wishes to transmit to its neighbor, it first transmits a an RTS (Request To Send). The receiver then consents to the communication by replying with a CTS (Clear To Send) packet. Any neighbors in the vicinity will wait for the duration specified on hearing the CTS. This shall prevent node C from disrupting the communication and in essence eliminating the hidden terminal problem. This is used in IEEE802.11 for reducing the possibility of collisions.

B.2 RTS/CTS with busy tones

Although the RTS/CTS mechanism can alleviate some hidden and exposed node problems, when propagation delays are long the CTS packets can easily be destroyed. This shall result in disruption of data packets when traffic load is high. Again consider the Figure . If instead of a transmission from C to D, D wanted to initiate a transmission to C and at the same time A and B's RTS/CTS exchange took place, There is the posibility of C not hearing packets from either of the communications because of long propagation delays. This would allow C to start transmitting later on, in the process disrupting the communication at B. It has been analysed that under heavy traffic loads the probability of collisions can be as high as 60%. To resolve the problem a protocol called DBTMA (*dual busy tone multiple access*) was proposed. The single common channel is split into a data and a control channel. The control channel is used for RTS/CTS and 2 narrowband frequencies are reserved for *transmit busy tones* and *receive busy tones.* The purpose of busy tones is to add capabilities of carrier sensing to transceivers. Hosts make sure that there are no BTs around before initiating transmissions.

IX. Issues in a CDMA based MAC

The use of CDMA as the physical layer adds another dimension to the scenario. In CDMA there is no explicit subdivision of the resources eg. bandwidth. All the users use the entire bandwidth at the same time. This allows us to have multiple simultaneous transmission in the same spatial zone at the same time, and so the concept of floor acquisition cannot be directly translated to the CDMA system.

As we shall see later, the problems mentioned earlier, eg. Hidden nodes, will be automatically taken care of in CDMA since multiple transmissions, overlapping in both time and frequency, are possible. But to do that efficiently we need to properly allocate users with codes.

A. Code Allocation

In the cellular CDMA system the code allocation is centrally monitored, while in mobile ad hoc systems this has to be done in a distributed manner. In a large network, the number of transmission codes is smaller than the number of nodes, and senders and receivers must agree on which transmission code to use in a way that avoids interference as much as possible. In such networks, it is essential that code reuse be done, but in such a way that the same code is used on nodes that are far apart.

Interference due to use of the same code on nodes can be of the following types:
Direct Interference: Two neighbors trying to transmit to each other at the same time.
Secondary Interference: Caused by interference with senders trying to transmit to receivers such that the senders' transmissions interferes at at least one receiver. The two situations in which this can happen have been illustrated in Figure. Two stations unaware of each others' existence can try to transmit to the same receiver; giving rise to the transmitter oriented code assignment problem. The second case of secondary interference occurs when a station is transmitting to a neighbor and a third station's transmission to some other station causes interference at with the first transmission. This leads to the receiver oriented code assignment problem illustrated in Figure As can be seen from this, the code assignment problem requires that no set of stations which are two hops away have the same code.

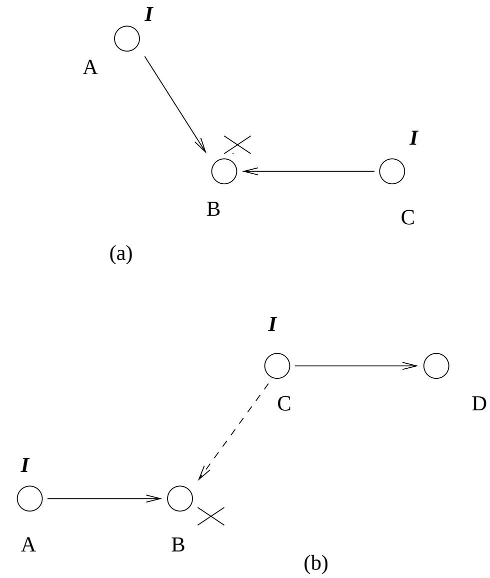

Fig. 18. Secondary Interference

Several approaches have been proposed for channel/code assignment using fixed length codes. We now describe a distributed algorithm, which uses message passing to arrive at a distributed solution to the code allocation problem, given that the number of codes is at least $d(d-1)+2$.[1] The algorithm [2] is designed to be part of the MAC and routing protocols of a multihop packet radio network. It is based on the asynchronous exchange of control messages that are part of the regular MAC and routing messages, and the information generated by any one node propagates up to two hops away from the node. *Code Assignment Messages* (CAM) are used for code assignment and are acknowledged by explicit ACKs.

B. Static Code allocation using Message Exchange

Information maintained at each node:

Priority List: Each node has a unique priority number assigned to it. This allows us to resolve clashes when two nodes discover to another one using the same code and one of them has to leave that code and search for a new one. The priority list consists of priority number of the one-hop and two-hop neighbors and their codes.

Neighbor List: List of immediate neighbors.

Code Assignment Message Retransmission List (CAMRL): One or more retransmission entries, consisting of sequence number of the CAM, retransmission counter, and ACK bitmap of the size of the neighbor list.

Unassigned Code List(UCL): Codes available.

[1]The clustering approach can also be used for this problem, by reducing the task of global code allocation to that of code allocation within the cluster.

Information Exchanged:

The CAMs propagate only from a code to its neighbors and no further. Each CAM contains:
1. Address of CAM sender along with its code.
2. The addresses of the nodes one hop neighbors along with their code.
3. ACKs to earlier CAMs. An ACK entry specifies the source and the sequence number of the CAM being acknowledged.
4. A response list of zero or more nodes which need to send an ACK for this CAM

CAMs are sent when:

1. A new node comes up. Its priority list consists of its own address and its own code. It broadcasts this to all its neighbors.
2. When a node i detects a change of code by any of its one-hop neighbors, i makes the required changes in its priority list and sends a CAM to all its one-hop neighbors including the one that changed the code.
3. When a node i finds that a certain one-hop neighbor j is no longer active, it drops j from its priority list, This information is then conveyed to all its one-hop neighbors.

Updating the priority list:

A node updates its priority list either after detecting a change of code in one of its one-hop neighbors or after receiving a CAM about a two-hop neighbor.

When a node notices a change in code of any of its one-hop neighbors, it makes a change in its priority list and sends a CAM with its own address and the address of its one-hop neighbor along with their codes. Three situations can arise:
1. If the new code of the two hop neighbor is not the same as this node code, then the new code is entered into the priority list.
2. If the new code is the same and the neighbors priority is higher then this node picks up the new code and updates its priority list.
3. If the new code is the same as this nodes code, then a temporary conflict occurs and the new node will be informed by the intervening one-hop neighbor.
In the first two cases, the UCLs have to be updates.

Complexity of Code Assignment

- **Communication Complexity:** In the worst case, a code change might induce a code change in its two hops neighbors code which can recurse and cause a worst case complexity of $O(|V| \cdot d^2)$
- **Computation Complexity:**In the worst case there are $d+1$ entries in the CAM, This includes new entries which did not exist in the priority list earlier. The entries are presumed to be in sorted order. A scan of the priority list is done and new codes added to it. Updating the UCl based on this scan can take $O(d^2)$ which is the complexity.
- **Storage Complexity:**Can be trivially shown to be $O(d^2)$

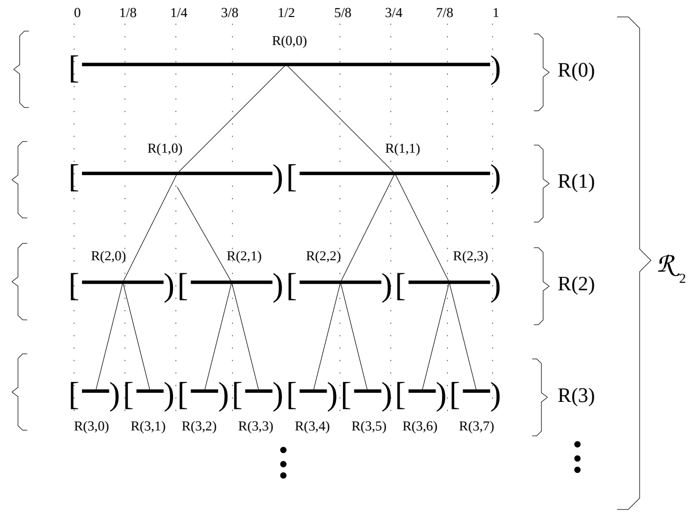

Fig. 19. Graphical illustration of the range set $\Re_2$. Overlapping ranges are shown connected and form a complete binary tree. For simplicity, the subscript 2 has been omitted.

X. Graph Theoretic Formulation of the Code Allocation Problem

A mobile ad hoc network can be modeled as an undirected graph $G_{topo} = (V, E)$, called the *topology graph*, where the set of vertices $V = \{1, \ldots, n\}$ represent the set of mobile terminals with an edge between two vertices iff the corresponding mobiles can hear each others transmissions. In such a case, we call such mobile terminals *adjacent*.

In a mobile ad hoc network, the problem of allocating variable length codes with a throughput maximization criterion can be formulated as follows: Reduce the topology graph of the network, given a set of communicating transmitter-receiver pairs, to a new graph in which each transmitter-receiver pair is represented by a vertex and there is an edge between two vertices in the new graph if they have to be assigned orthogonal codes, i.e. their transmissions will interfere if they were to occur on the same code. The variable length code allocation problem is then reduced to assigning colors(codes) to this new graph with codes from the available OVSF code set such that the aggregate throughput is maximized.

Example Let us now consider the topology graph as shown in Figure 20a. The bold lines connect communicating entities and the dashed lines connect terminals which can hear each other. It can be easily verified that the assignment of orthogonal OVSF codes as shown in the figure will lead to the maximum throughput for this set of communications. If we merge each pair of transmitter-receiver pair as shown in Figure 20b, then the problem reduces to that of assigning codes to these vertices such that throughput is maximized. In Section XII, we shall show how this throughput maximization problem is reducible to the problem of computing a coloring of this new graph with a sum maximization criterion.

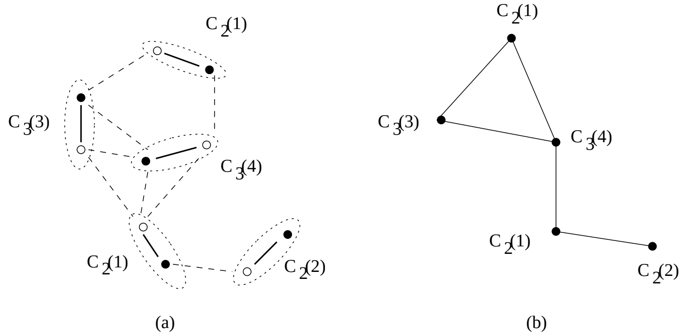

Fig. 20. (a)A set of communication requests and an optimal assignment of OVSF codes which will maximize throughput. The transmitters are shown by filled black nodes and their corresponding receivers are hollow and connected by bold black lines. (b)Reduced graph.

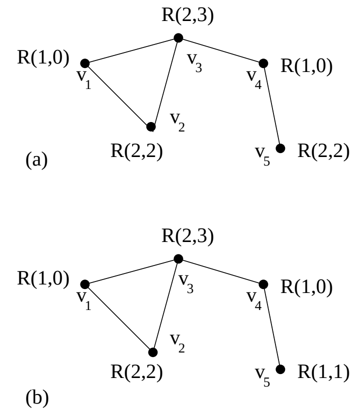

Fig. 21. Two proper range colorings with the Range Sum being achieved with more than the number of colors required to achieve the Chromatic Sum

XI. The Range Sum of a Graph

To facilitate the theoretical analysis of the variable length code allocation problem we first define the Range Sum of a graph.

A. *Graph Theoretic Definitions and Notation*

For a given undirected graph $G = (V, E)$, the order and size of the graph are the number of vertices and edges respectively. The degree of a vertex is the number of edges incident on it and let Δ and $\bar{d}$ denote the maximum and average degree of the graph.

A coloring of a graph is an assignment of colors from a specified set to the vertices of a graph. A coloring is called proper if no two adjacent vertices share the same color. The *chromatic number* of a graph is defined to be be the minimum number of colors which are required to achieve a proper coloring of the graph from the set of natural numbers. The *chromatic sum*[16] of a graph is the minimum sum of the colors of the vertices over all colorings of the graph with natural numbers.

B. *The Range Sum*

We now introduce a new variant of the chromatic sum of a graph. Instead of coloring the nodes with colors from the set of natural numbers, we instead color the vertices with ranges which are obtained by equally subdividing the line segment $[0, 1)$ recursively into σ equal parts such that no two adjacent nodes are colored with ranges having a non zero intersection. More formally:

Definition Define the sets $R_\sigma(k)$ with parameters σ, k recursively as:

- $R_\sigma(0) = \{(0, 1]\}$
- $R_\sigma(k+1) = \left\{ \left[l(r) + (t-1) * \frac{|r|}{\sigma},\ l(r) + t * \frac{|r|}{\sigma} \right) \middle| t \leq \sigma,\ \forall r \in R_\sigma(k) \right\}$

where $l(r)$ denotes the left end of the range r.

For notational convenience we shall refer to the ith element of $R_\sigma(k)$ as $R_\sigma(k, i)$ and denote the set of $R_\sigma(k)$'s as $\Re_\sigma$ i.e. $\Re_\sigma = \lim_{k\to\infty} \bigcup_{i=0}^{k} R_\sigma(k)$.

Example Figure 19 shows the hierarchical organization of the ranges in $\Re_2$. The ith level in this *range tree* corresponds to the set $R_2(i)$. Moreover the parameter $\sigma(=2$ in this case) can be visualized as the degree of branching of this range tree.

Definition Let $G = (V, E)$ be a graph with vertex set V and edge set E. A proper range coloring is a coloring of the nodes with adjacent nodes being colored with non-intersecting ranges from $\Re_\sigma$; i.e. $r : V \to \Re_\sigma$ such that $r(u) \cap r(v) = \phi$ whenever $(u, v) \in E$.

We define the *Range Sum* of the graph G, $\Gamma_\sigma(G)$, to be the maximum sum of the lengths of ranges over all proper range colorings of G.

Example Consider the problem of range coloring the graph shown in Figure 21 with $\Re_2$. For $\Re_2$, the length of a range in $R(1, -)$ is 0.5 and in $R(2, -)$ is 0.25.

This example illustrates the difference of the range sum problem from the chromatic sum and chromatic number problem. The chromatic number of this graph is easily seen to be 3 which is also the number of colors required to achieve the chromatic sum of $9(=1+2+3+1+2)$. As can be seen, both colorings shown in the figure are proper range colorings but the one in Figure 21b uses 4 colors(ranges) while the one in Figure 21a uses 3 colors(ranges) $(=$ the chromatic number and also the number of colors for chromatic sum coloring).

The range coloring in Figure 21b achieves the Range Sum for this graph which is equal to 2.0 $(= \sum |r(i)| = 0.25 \times 2 + 0.5 \times 3)$, even though it uses more ranges than the minimum required for proper coloring. Any range coloring using less than four colors cannot achieve a sum greater than 1.75, one of which is shown in Figure 21b.

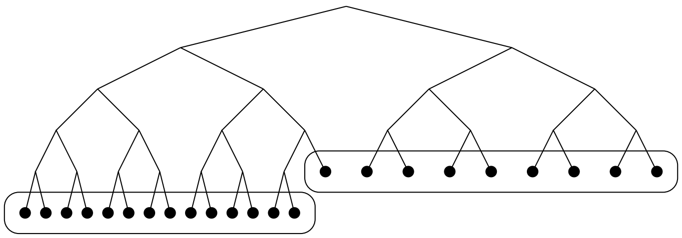

Fig. 22. An optimal range coloring for the clique K_{23}. Here $r = 4$ and 9 ranges are chosen from $R(4)$ and 14 from $R(5)$ summing up to 1.0.

C. NP-Completeness of Range Sum Problem

Consider the following decision problem Π:

Instance: Graph $G = (V, E)$ and positive integers k, σ

Question: Is there a proper range coloring, r, of G, from ranges in the set $\Re_\sigma$ such that $\sum_{v \in V} |r(v)| \geq k$.

Theorem XI.1: The decision problem Π is NP-Complete.

Proof: It is easy to see that $\Pi \in NP$, since given an instance of Π, a nondeterministic turing machine need only guess an assignment of ranges to the vertices of the graph and verify, in polynomial time, whether the assignment is proper and its sum is not less than k.

Let Π' be known NP-Complete decision problem of coloring a graph with no more than k' colors. We consider the special case of the problem Π with $\sigma = 2$ and show its NP-Completeness by reducing an arbitrary instance of Π', given by a graph $G' = (V', E')$ and a positive integer k', to an instance of Π. The proof for any general σ is a simple extension.

We use the same construction as was used for reducing an instance of the chromatic sum problem[16] to an instance of the chromatic number problem. In particular, we construct the new graph $G = G' \times K_{k'}$, the Cartesian product of graph G' with a complete graph $K_{k'}$. That is, if $V' = \{v_1, v_2, \ldots, v_n\}$ then $V = \{v_j^i | 1 \leq i \leq k', 1 \leq j \leq n\}$ and $(v_j^i, v_s^r) \in E$ whenever either $(j = s \,\text{and}\, i \neq r)$ or $(i = r$ and $(v_j, v_s) \in E')$. Also let $k = n$ and $\sigma = 2$. Since $k' \leq n$ this construction can be done in polynomial time.

We know that a clique of size t, ie K_t, can be range colored from $\Re_2$ by considering $2^{r+1} - t$ ranges from $R_2(r)$ and $2t - 2^{r+1}$ ranges from $R_2(r+1)$ where $r = \lfloor \log_2 t \rfloor$ as shown in Figure 22 for the case of K_{23}. Additionally the sum, $\Gamma_2(K_t)$, is always 1. Let $O_t = \{r_1, r_2, \ldots, r_t\}$, be an arbitrary ordering of one such set of t ranges, used for optimally coloring K_t.

Example We envision the graph G, as a k'-exploded version of the original graph and with each vertex in G' being replaced with a k'-clique in the new graph. An example of this construction is shown in Figure 23.

We can obtain a range coloring R for G given a coloring C' for G' by the following method: For each vertex, v_j colored with color $c(v_j)$, color the vertices, v_j^i, of its associated clique with the ith color in the $c(v_j)$-shifted(cyclically) version of $O_{k'}$ i.e $r(v_j^i) = O_{k'}((c(v_j) + i')_{mod\, k'})$. Thus, each clique in the graph contributes a sum of 1.0 and the total sum over all vertices equals n. Therefore G has a range sum of n.

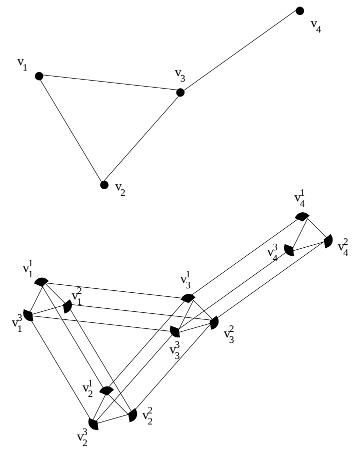

Fig. 23. The construction for showing the NP-completeness of the Range Sum problem.

Conversely, suppose there exists a proper range coloring of G with sum not less than n. Since each clique, $K_{k'}$, corresponding to one vertex in the original graph G' can not be colored with less than k' ranges and each such set of vertices contributes not more than 1.0 to the range sum, at least k' colors are required to color each clique and not more than k' colors will be required. Hence, G is k'-colorable and so is its subgraph G'.

Using the polytime reduction above we can reduce each instance of Π' to an instance of Π. Since Π' is known to be NP-Complete, Π is also NP-Complete. ■

D. Approximating the Range Sum

The range sum problem for arbitrary graphs has been shown to be NP-Complete in the previous section. Therefore, we now look at approximate algorithms for computing the range sum. In this section we consider the family of sparse graphs with average degree $\bar{d}$. The ith range in the lexicographic ordering on the index, (k, t), of the ranges in $\Re_\sigma$ shall be referred to as $Range_\sigma(i)$.

We consider the following greedy algorithm.

Algorithm **Greedy-Range-Color**(G, σ):

Input: A undirected graph $G = (V, E)$ and integer parameter σ.

Output: A proper range coloring $r : V \to \Re_\sigma$ of G.

For each node $v \in V$ maintain:

- *r(v)*, the range(*color*) assigned to v, initialized to ϕ
- *Unused(v)*, the subset of $[0, 1)$ not being used to color any node u adjacent to v; or more formally

$$Unused(v) = [0, 1) - \bigcup_{\forall u; (u,v) \in E} r(u)$$

Algorithm:
Step 1: Determine an ordering of the vertices $v \in V$, $v_1, v_2, \ldots, v_n$.
Step 2: Consider the vertices in the order determined above. Let the current vertex be v_r. Find the least value of t such that

- $Range_\sigma(t) \cap Unused(v_r) = Range_\sigma(t)$ and
- $Unused(u) - Range_\sigma(t) \neq \phi, \quad \forall u$ satisfying $(u, v_r) \in E$ and $r(u) = \phi$

Color v_r with $Range(t)$, i.e $r(v_r) = Range_\sigma(t)$ and update $Unused(u) = Unused(u) - r(v_r) \quad \forall u, (u, v_r) \in E$. Repeat until there is some uncolored vertex.

Example Consider again the graph shown in Figure 21. The lexicographic order of the ranges in $\Re_2$ is R(0,0), R(1,0), R(1,1), R(2,0), R(2,1), R(2,2), R(2,3), R(3,0)…. And consider the ordering of the vertices as shown in the figure itself. It can be easily verified that the range coloring that is produced by Greedy-Range-Color is the same as shown in Figure 21b which is coincidently optimal.

E. Performance of Greedy-Range-Color on Sparse graphs

Theorem XI.2: For each G with average degree $\bar{d}$, algorithm Greedy-Range-Color is a $\sigma^{\frac{\sigma+\bar{d}}{\sigma-1}}$-approximation to $\Gamma_\sigma(G)$.

Proof: Given a specified ordering of the vertices of a graph, we define the *lower degree* l_i to be the number of lower indexed neighbors of vertex i, while the *higher degree*, h_i, is the number of higher indexed neighbors. Of course, this means that $l_i + h_i = d_i$, where d_i is the degree of v_i. Since each edge is counted only once in the sequence l_i (h_i), we have the size of the graph, $e = \sum l_i = \sum h_i = \frac{1}{2}\sum d_i$.

Let $n = |V|$ be the order of the graph G and $v \in V$, $v_1, v_2, \ldots, v_n$ be the vertices of G listed in the order specified by the algorithm in Step 1. At most l_i colors and their induced subranges are forbidden colors for the vertex v_i. Therefore v_i is colored with a range that contributes at least $\frac{1}{\sigma^{\lceil \frac{1+l_i}{\sigma-1} \rceil}}$ if $h_i \neq 0$ and $\frac{1}{\sigma^{\lceil \frac{l_i}{\sigma-1} \rceil}}$ if $h_i = 0$, to the sum. Therefore

$$|r(v_i)| \geq \frac{1}{\sigma^{\lceil \frac{1+l_i}{\sigma-1} \rceil}}$$

Summing over all vertices, the sum of the range coloring produced by Greedy-Range-Color, $\Gamma_\sigma^{greedy}(G)$, is given by

$$\begin{aligned}
\Gamma_\sigma^{greedy}(G) &= \sum_{v \in V} |r(v)| \\
&\geq \sum \frac{1}{\sigma^{\lceil \frac{1+l_i}{\sigma-1} \rceil}} \geq \frac{n}{\sqrt[n]{\prod \sigma^{\lceil \frac{1+l_i}{\sigma-1} \rceil}}} \\
&= \frac{n}{\sqrt[n]{\sigma^{\Sigma \lceil \frac{1+l_i}{\sigma-1} \rceil}}} \geq \frac{n}{\sqrt[n]{\sigma^{n+\frac{n+\Sigma l_i}{\sigma-1}}}} \\
&= \frac{n}{\sigma^{1+\frac{1+\bar{d}}{\sigma-1}}} \geq \frac{\Gamma_\sigma(G)}{\sigma^{1+\frac{1+\bar{d}}{\sigma-1}}} = \frac{\Gamma_\sigma(G)}{\sigma^{\frac{\sigma+\bar{d}}{\sigma-1}}}
\end{aligned}$$

The second step is the geometric mean $\geq$ harmonic mean inequality and the last inequality follows from the trivial $n \geq \Gamma_\sigma(G)$ bound, which is achieved for the null graph. Hence Greedy-Range-Color finds a $\sigma^{\frac{\sigma+\bar{d}}{\sigma-1}}$-approximation to the range sum of G. For the special case of $\sigma = 2$ the ceiling function is removed and we get a better bound of $2^{1+\bar{d}}$. ■

XII. Code Allocation for Maximizing Throughput

As was discussed in Section III, the problem of code allocation such that there are no collisions reduces to coloring (allocating a code) a pair of communicating entities such that their immediate neighbors are not using the same color (code). We model this problem by constructing a new graph, called the *communication graph*, based on the 'snapshot' communications in progress at some particular instant.

A. The Communication Graph

Let us denote by $G_{topo} = (V, E)$, the topology graph. Then the communication graph, $G_{comm} = (V', E')$, for some set of transmitters and receivers, equal in number, is defined as below. We exclude any nodes which are not communicating at that instant.

- $V' = \{v_{(u,v)} \mid$ u and v are a transmitter-receiver pair in the set of communicating entities $\}$
- $E' = \{(v_{(u,v)}, v_{(p,q)}) \mid$ if (u is within communicating range of q) or (p is within communicating range of v) $\}$

Example It can be easily seen that the graph in Figure 21 is the communication graph for the set of communications and topology shown in Figure 20 with the dotted lines being the edges in the new graph and each pair of transmitter and receiver being merged into a single node in the new graph.

For OVSF codes, the following lemma holds:

Lemma XII.1: The problem of maximizing the aggregate throughput by optimal code allocation is equivalent to finding the range sum of the communication graph, G_{comm}.

Proof: Consider the mapping which overlays each element of the complete binary tree $\Re_2$ with the corresponding element in the OVSF code tree i.e. the one-to-one mapping $f(C_N(i)) = R_2(\log N, i-1)$ from the set of OVSF codes to the set of ranges in $\Re_2$. The data rate achieved by a transmission using code $C_N(i)$ is given by $B/length(C_N(i))$, where B is the bit rate for a spreading factor of 1. The aggregate throughput per node, γ, in a network consisting of M nodes, is given by

$$\gamma = \frac{1}{M}\sum \frac{B}{length(C_N(i))} = \frac{1}{M}\sum \frac{B}{N} \qquad (12)$$

For the mapping $f(\cdot)$, this sum is equal to $B/M \times \sum |r|$, since the length of a range in $R(\log N)$ is $1/N$. Hence maximizing γ is the same as finding an allocation of ranges from $\Re_2$ which maximizes $\sum |r|$, which is the same as computing $\Gamma_2(G_{comm})$. ■

B. *Distributed Code Allocation using Greedy Range Coloring (DCA-GRC)*

We present a distributed version of the greedy range coloring algorithm presented in Section XI-D. In this section we shall refer to codes and ranges ($\subseteq [0, 1)$) interchangebly using the one-to-one mapping from the OVSF code tree to the range tree for $\Re_2$ referred to Lemma XII.1.

B.1 Maximizing Throughput by Optimal Code Allocation

A major issue in wireless ad hoc networks is developing efficient medium access protocols that optimize spectral reuse and hence maximize aggregate channel utilization. Several distributed access schemes have been proposed which allocate spectrum or time or code[17] or a combination of these. The ideal CDMA system with perfect synchronization and perfectly orthogonal codes does away with the need to allocate spectrum/time bands to the users and hence any issues in co-ordinating the allocation of such resources. But the problem of perfect code allocation is a simple instance of the graph coloring which is known to be extremely hard. Thus for instance, consider the simpler problem of allocating equal length codes to the mobiles in an ad hoc network. Since a larger number of codes implies a larger spreading factor and hence decreased bit-rate we need a coloring with the minimum number of colors - i.e. computing the chromatic number!

Therefore, most researchers ignores computing the optimal (since it is a known intractable problem). Thus in [2], the authors assume the existence of $\Delta(\Delta - 1)$ codes. Such an allocation can lead to extremely poor performance in the worst case, which occurs when there is a single node with a dominating edge degree. Moreover, there has been no attempt to consider throughput as a parameter when doing code allocation. OVSF codes have been discussed previously for optimizing WCDMA but only in regard to decreasing the *call blocking probability*[18] and algorithms have been presented[19]. In WCDMA, there is a provision to support mutliple user data rates by assigning spreading codes of different lengths. Since they also specify a minimum data rate that shall be alloted, the protocol cannot assign codes greater than some specified depth in the code tree. Thus, codes might be *blocked* since their ancestor is in use. The probability of this happening is the call blocking probability.

With OVSF codes it is possible to conceptually aggregate multiple codes into a code of smaller length, compromising on the number of of orthogonal codes left for use by others, but in the process increasing throughput. This is reasonable when a particular node has small number of neighbors. We now present a distributed code allocation protocol based on the range sum of the communication graph, as discussed in Section XI. By Lemma XII.1, this amounts to finding an allocation which maximizes throughput.

B.2 Protocol Idea and Overview

The main idea of the protocol is to prevent two neighbors from choosing their codes at the same instant as this can lead to inconsistent code allocation. Consider, for example, the situation in Fig3b. If B and C received messages to initiate communications at *exactly* the same time then both could potentially choose non-orthogonal codes, because each will be unaware of each others' code. To prevent such possibilities we introduce a *locking procedure* that is executed by each pair of transmitter-receiver to 'lock' all of their neighbors onto their current code. A **locked** terminal is prohibited from making any changes to the code it is using and does not reply to any control messages thus ensuring that no other nodes can successfully complete a lock on *their* neighbors. In the protocol terminals make a change in the code allocation only when all their neighbors are locked. Thus a locked node does not notice a change in the local allocation of codes except at the terminal which caused the lock.

Except for the states in the locking procedure each terminal is in either of the following states: *Transmitting, Idle, Code-Negotiation*. Whenever the MAC layer of a terminal, in the Idle state, receives a signal to start a communication session, from either the higher protocol layers or from a different terminal, it initiates an attempt to choose a code for this transmission/reception. It always succeeds in finding a code, either due to availability or by sending appropriate messages to neighbors to change the code they are currently using.

The locking phase: On receiving a COMMUNICATE message the MAC sublayer attempts to lock all neighbors onto their codes by sending LOCK messages. If the terminal itself was locked when it received the message, it changes state to *Waiting for Code*, and initiates locking as soon as it is unlocked. The terminal then waits for all neighbors to acknowledge receipt of the LOCK message by an ACK-LOCK. The terminal sends a LOCK-COMPLETED message along with relevant details about the codes being used in its neighborhood to the terminal which is to be communicated with. After the code has been chosen an UNLOCK message is sent to all neighbors. Each locked terminals maintains a priority list of neighbors which have requested a LOCK, and services LOCK-ACKs in that order. This ensures a neighbor receives a LOCK-ACK after a bounded number of retries.

B.3 Formal Description of DCA-GRC.

We formally describe the protocol in terms of the state transition diagram as shown in Figure 24. A node in some state can be either *Locked* or *Unlocked*. The Locked variant of the state is shown by a darkened circle in the diagram. The complex state *Code-Negotiation* is described later. The Start State is the Idle(Unlocked) state. The transition events $E_1 \ldots E_{10}$ are:

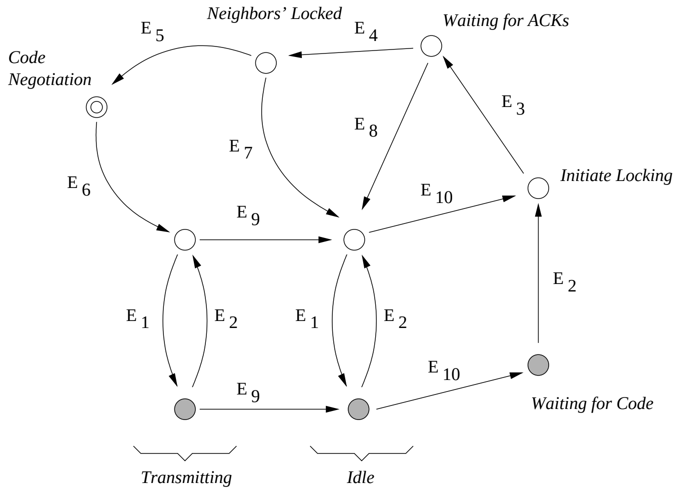

Fig. 24. State Transition diagram for code allocation. (For description of each transition, $E_1 - E_{10}$, see text)

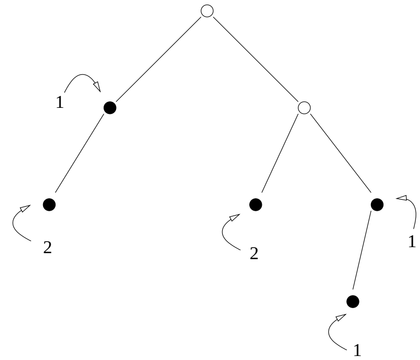

Fig. 25. Construction of T_{code}. The number beside each node denote the number of neighbors using the code corresponding to this node.

E_1: Received LOCK message; Sent LOCK-ACK.
E_2: Received UNLOCK message.
E_3: Broadcasted LOCK message to all neighbors.
E_4: Received LOCK-ACK messages from all neighbors.
E_5: Sent and Received a LOCK-COMPLETED message.
E_6: Broadcasted RESERVE-CODE(C) and UNLOCK message to all neighbors.
E_7: Timed-out waiting for LOCK-COMPLETED.
E_8: Timed-out waiting for ACK message, i.e. at least one neighbor failed to reply with LOCK-ACK to the LOCK message; Sent LOCK-FAILED message.
E_9: Finished transmitting/receiving; Broadcasted a RELEASE-CODE(C) to all neighbors.
E_{10}: Received a COMMUNICATE message from higher protocol layers or from a prospective transmitter.

The *Code-Negotiation* phase shown in the transition diagram is not a simple state and denotes the following algorithm:

Both the terminals which want to initiate a transmission send control packets containing information about the codes that are being used in their vicinity. Given a set of codes being used in the neighborhood, a code for communication is chosen as follows:

Algorithm Code-Negotiation:
1. Construct a tree, T_{code}, of the codes that are being used, i.e. for each code that is being used in the neighborhood construct the appropriate leaf in the tree along with any nodes required in the path to the root code. Label all nodes in T_{code} as *used*
2. Binarize T_{used}, i.e. for each used node in the tree, add *unused* children until all used nodes have two children and all unused nodes are leaves.
3. For each leaf node, l, (which is unused) calculate

$$suitability(l) = \frac{1}{code\_length(l)} - \sum \frac{n}{2 * code\_length(v)} \quad (13)$$

where the sum is taken over all nodes in the tree which are ancestors of l and n is the number of neighbors using the code corresponding to that node in the OVSF code tree.
4. Find the leaf nodes, l_{max}, with largest value of $suitability(l)$. Resolve ties in a left to right manner.
5. In the *conservative* version of the protocol if there exists a neighbor which is *'Waiting for Code'* then choose the code corresponding to one of the descendants of l_{max}.
In the purely greedy version the code corresponding to l_{max} will be chosen. By not choosing the descendent, there exists the possibility that a terminal that is allocated a code later will not have a empty subtree to choose its code from. Therefore it will need to send SPLIT messages to neighbors that are obstructing its path to the root. The terminal will send a SPLIT message to all neighbors which are using codes corresponding to ancestors of l_{max} in the code tree. The new code that the neighbors will use will depend on the direction of split required to vacate the path to the root.

Example Consider the codes in use to be the ones which are shown as dark nodes in the OVSF code tree in Figure 25. The hollow nodes are the ones added to create a path from each used node to the root. Figure 26 shows the tree after leaf nodes have been added. Node 5 is then chosen as l_{max} and the neighbor using the code corresponding to node 2 will start using 4.

B.4 Throughput Performance of the Protocol

We now present a bound on the throughput resulting from a code allocation produced by DCA-GRC. We consider the case of a idle network in which no transmissions are in progress at time $t = 0$ and a set of communication requests arrive at time $t = t_0$. For such a scenario the following lemma holds:

Lemma XII.2: Given a set of communications, algorithm DCA-GRC produces a proper code allocation which is a $2^{2\bar{d}-1}$-approximation to γ_{max}, where $\bar{d}$ is the average degree in the topology graph, G_{topo}, and γ_{max} is the maximum possible aggregate throughput for the given set of communications.

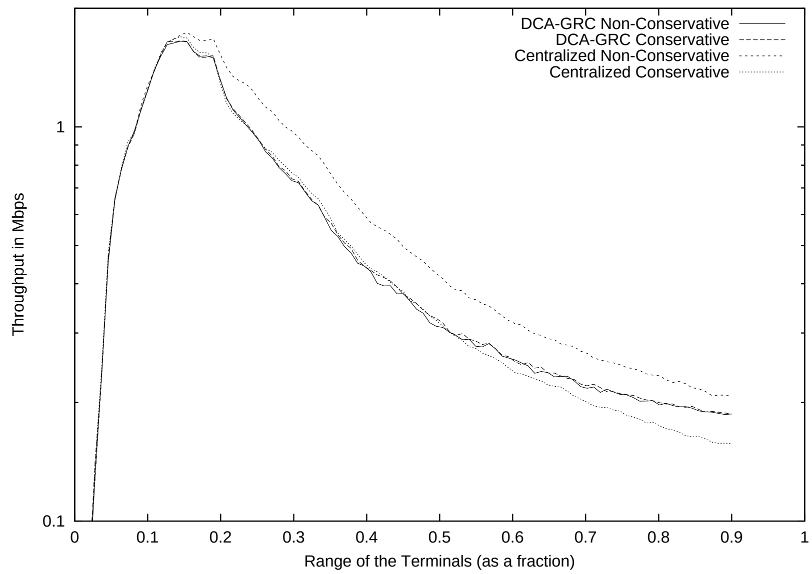

Fig. 27. Performance comparison of centralized and distributed greedy code allocation.

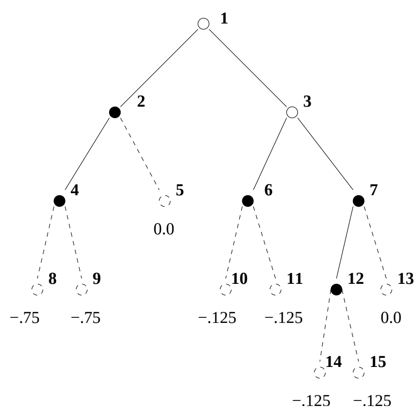

Fig. 26. T_{code} after binarizing. All *unused* leaves have been populated with their *suitability* values.

Proof: Due to the locking protocol, there is a unique ordering in time which can be assigned to two nodes which are neighbors. Since a node will choose a range which has a zero-intersection with all its neighbors, the protocol produces a proper range coloring of the communication graph. Moreover given a set of initial communications, initially, each terminal will be contending for a code and therefore Step 5 of the *Choosing Code* phase will always leave a code subtree for each neighbor not yet having a code.

Since there exists a unique ordering in time between each pair of adjacent vertices in G_{comm}, we can construct a global ordering, O, of the vertices as follows:

- u<v if u and v are adjacent and u chooses before v.
- arbitrarily choose an ordering between p,q which are not adjacent.

We can visualize a run of the protocol as a run of Greedy-Range-Color with the ordering in Step 1 being O. Observing that the degree of each node, $v_{(u,v)}$, in G_{comm} satisfies $deg(v_{(u,v)}) \leq deg(u) + deg(v) - 2$, therefore the average degree in the new graph, $\bar{d}'$ is upper bounded by the $2(\bar{d}-1)$. With Lemma XII.1 and Theorem XI.2 the result follows. ■

DCA-GRC is an incremental protocol in the sense that it does not recompute the code allocation on each new Communication Graph, which will happen when a communication terminates and/or a new one starts. For a snapshot communication graph Lemma XII.2 holds which bounds the throughput. But this code allocation will be different from the one obtained by DCA-GRC which will be an incremental modification to the already existing code allocation.

Since the number of such changes in the network is extremely large, it is reasonable to expect that the throughput obtained by the incremental protocol will not differ significantly from the one obtained by doing the greedy code allocation all over again for each new communication graph. This conjecture was verified through simulations of a network of 50 nodes and a plot of the aggregate throughput per node vs the range of the terminals is plotted in Figure 27. For comparison, we assume that for each new communication graph (ie at each transmission request) an entity with full knowledge of the network state, will recompute an allocation greedily and terminals will start using this new allocation from that instant. This central allocation ensures that, at any instant, the codes are allocated as per Lemma XII.2. Conservative and non-conservative versions of both the protocols have been plotted. It can be observed that the throughput obtained are virtually similar with non-conservative version giving the better values. This is because in the conservative versions each node incurs the overhead of leaving codes for its neighbors. Since these codes do not necessarily coincide, a new node will have multiple free parts in the tree, which cannot be utilized. The non-conservative versions save this overhead at the cost of sending a small control packet to a few neighbors. Moreover, as expected the centralized version outperforms the distributed implementation but the performance degradation is within acceptable limits to be used.

B.5 Practical Implementation of DCA-GRC

Since the probability of two adjacent nodes, from different communication pairs, initiating search for a code at exactly the same moment is extremely small, the locking procedure prevents inconsistent allocation but only in some rare worst case scenarios. Thus, in a practical system we can afford to have *some* inconsistent allocations. Nodes will in such a case omit the locking procedure, exchange their knowledge of the codes being used in their neighborhood and choose a code by a simple traversal of the code tree. This would significantly reduce the number of control packets in the broadcast channel at the cost of a few collisions.

XIII. Simulation Model and Results for the Dynamic Code Allocation Scheme

Simulations were done using random topologies for DCA-GRC and the throughput obtained were compared to the case when an offline static code allocation had been done. For the offline static allocation, we consider the greedy algorithm presented in [4] and presented in Section IX-B and assume that a centralized entity decides the codes which are communicated to the nodes without any overhead. For a static topology, this means that we are ignoring the network setup time; for a network with mobile nodes, this means that the throughputs obtained are an upper bound on the throughput that can actually be achieved.

The simulation were done with the following parameters:

- Packet burst arrival process is Poisson with mean arrival rate λ.
- Packet burst size is uniformly distributed between 1-5 packets, each of size 1Kb.
- The physical layer is assumed to provide a communication channel with a maximum bit rate of 11Mbps.
- The nodes are randomly distributed in a square region of unit area and their range of communication is represented as a fraction of the length of the side.

Using an event driven simulator we simulated the packet transmission for the above parameters. For DCA-GRC each packet burst transmission is preceeded by a code negotation phase in which the nodes execute **Code-Negotiation** to decide the code to use. No such overhead is incurred in the case of static allocation. A packet is assumed to be corrupted and lost if any other transmitter in the range of the receiver transmitted in the duration of the packet using a non-orthogonal code.

A. Throughput Performance of the Code Allocation under varying network density

Figure 28 shows the drastic degradation in the throughput as the number of nodes in the network increases which is obtained when terminals are assigned codes statically. The ‘steps’ in the graph can be explained by observing that as the number of nodes increases, with the range of each node reaining constant, the maximum degree in the network increases causing static code allocation schemes to require an increasing number of codes. Since the number of codes determines the speading factor used by the terminals the throughput decreases by a multiplicative factor as the threshold for a given spread factor is crossed. Moreover the number of codes at some level in the code tree is twice that in the previous level, hence, we see an increase in the step size as more and more nodes are added to the network. On the other hand, our dynamic code allocation scheme is never really effected in its performance since it is dependent on the number of communications in progress which is independent of the maximum degree in the network. The slight downward curve is expected since a larger number of nodes implies larger interference.

A similar result is seen in Figure 29 when we increase the range of the terminals while keeping their number constant. When the range is too small nodes are isolated or have too few neighbors. This causes packet loses and hence we see a rise in aggregate throughput as we increase the range. But thereafter the same effect as observed in the previous plot comes into play and causes the throughputs to decrease; only marginally for DCA-GRC, but drastically for static code allocation schemes.

The same can be seen with better granularity in Figure 30 which has been plotted for a network of 50 nodes. We can see a more pronounced increase followed by a step-like decrease for static allocation and gradual decrease for DCA-GRC. It is interesting to observe that the right part of this plot is almost entirely similar to Figure 28. This is due to the relation between the number of nodes(with constant range) and the range(with constant nodes), since both are related to the average degree of each node.

B. Fairness of the proposed Allocation

Since dynamic code allocation can possibly lead to widely varying date rate allocation, unfair distribution of resources can occur. Hence, in Figure 31 and Figure 32 we plot the variance of the bit rate vs the range and the number of nodes. It can be seen that there is a marked maxima in the plots. Initially when connectivity is low there are lesser number of neighbors to deal with and hence the bit rate allocated is close to the maximum possible and the variance is low. On the other hand as connectivity increases, bit rate allocation moves to the other stable point where all nodes use codes which are larger in length and hence the variance again is low.

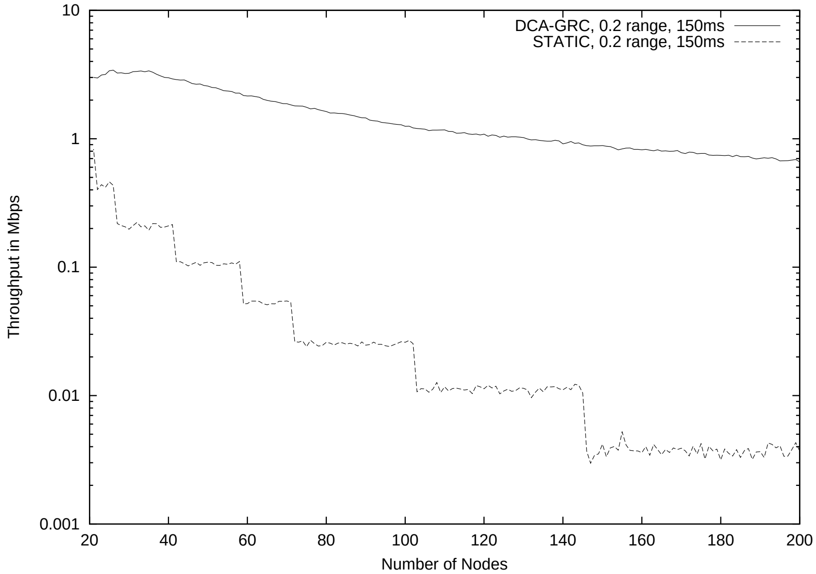

Fig. 28. Effect of network density.

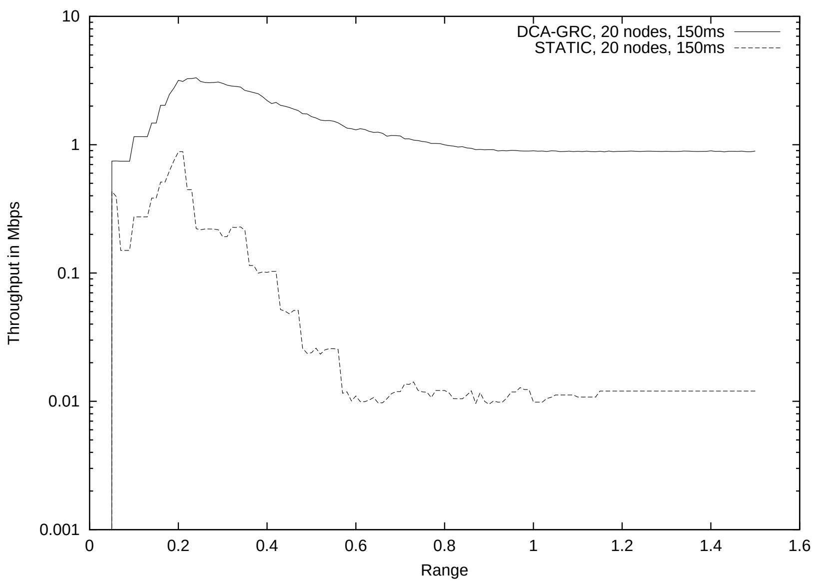

Fig. 29. Effect of range of the nodes.

XIV. Conclusions

In the first part of this investigation, we studied various multicarrier CDMA techniques and evaluated the performance of an MC-CDMA system for various combining techniques. The simulations done for this part of the project illustrated the superior performance of MMSEC since it takes into account the number of users as well as the noise as a parameter. Based on the extensive simulations, we proposed a design of the physical layer based on MC-CDMA.

In the latter part of the investigation we introduced a variant of the graph coloring problem called the **Range Sum** of a graph. We have illustrated its difference from the Chromatic Sum problem and have shown its NP-Completeness. We further gave an approximation algorithm for the range sum problem which was later used for designing a code allocation scheme for CDMA based ad hoc networks. The distributed version of the approximation algorithm was shown to provide a code allocation from the OVSF code set which is within a specified bound of the optimal.

Based on the results of the simulations of the dynamic code allocation, we find that it substantially outperforms static code allocation when the network density is high. The only overhead incurred is the exchange of control information required to negotiate the code. This, we expected, would decrease the performance of our protocol. But simulation results show that the increase in throughput obtained by variable length codes far outweighs the loss of bandwidth due to control message exchange.

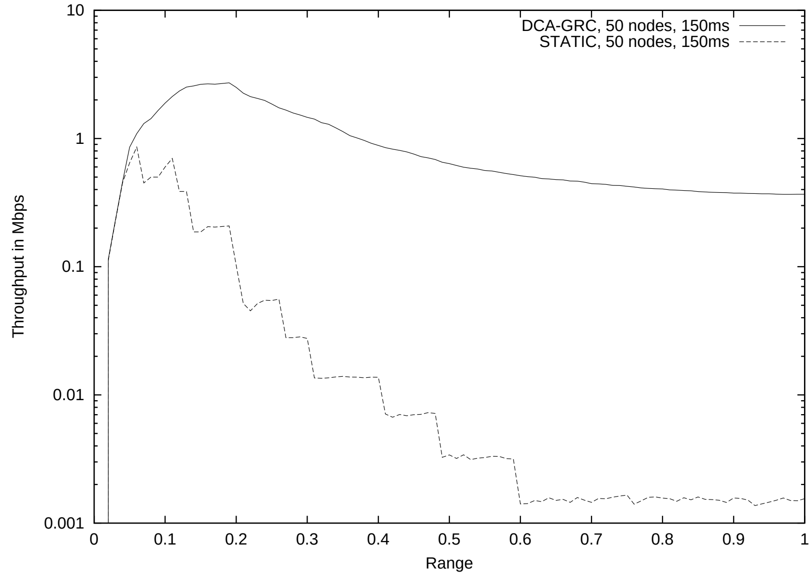

Fig. 30. Effect of range of the nodes.

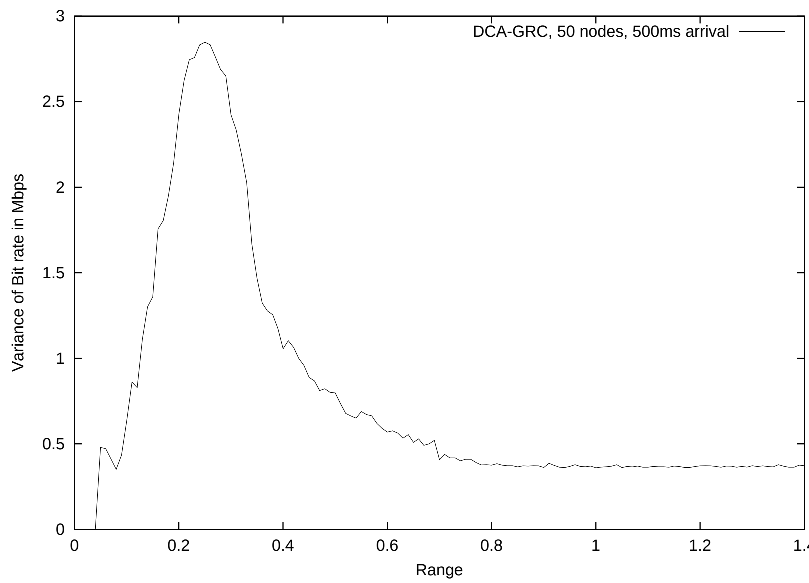

Fig. 31. Variance of the Data Rate allocated.

We would also like to mention that in high mobility scenarios, our scheme as well as static allocation based on minimizing the number of codes used [4] (see Section IX-B), will incur some collisions. Static receiver/transmitter oriented code allocation has also been proposed in the literature where each terminal is assigned a separate code and communication to a terminal occurs on the receiver's and transmitter's code in receiver and transmitter oriented code allocation respectively. Such schemes will never have this drawback but will use $O(n)$ codes and hence the length of each code will be much larger than the aggregate length used in our scheme and hence is expected to achieve drastically low throughput.

XV. Future Work

In the first part of the investigations we assumed synchronization. However, in practical networks synchronization needs to be achieved by some mechanism and therefore further work is required to address this issue.

For the code allocation scheme, the performance has been evaluated for static ad hoc networks while mobility in real senarios will introduce changes in the channel model. Further suppose two nodes, which were assigned the same code due to their spatial separation, might come within wireless range of each other and cause collisions. In retrospect, it will be interesting to see the benefits of this code allocation at the MAC layer and how dynamic changes in the network due to mobility will effect the validity and performance of the incremental solutions generated.

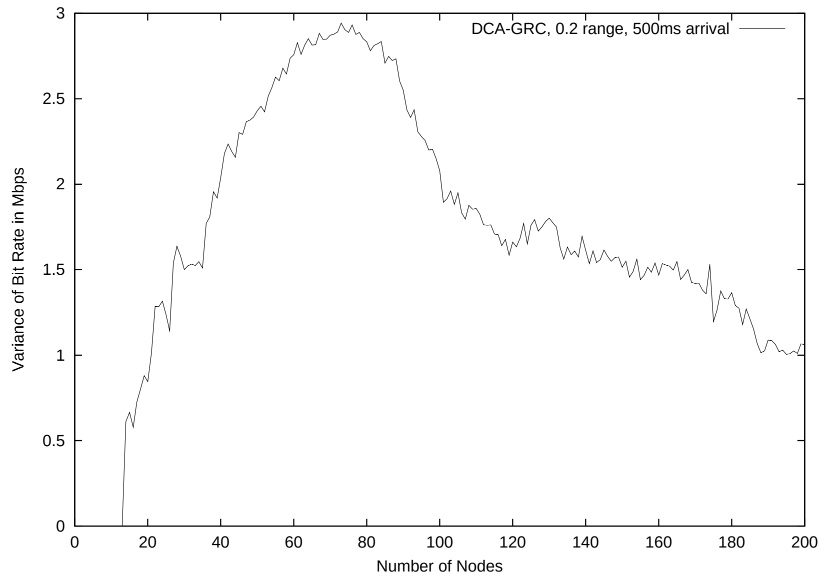

Fig. 32. Variance of the Data Rate allocated.

XVI. Acknowledgements

We would like to thank our guides Dr. A. K. Chaturvedi of the Electrical Engineering Department and Dr. Dheeraj Sanghi of the Computer Science and Engineering department at IIT Kanpur for their precious guidance, valuable criticism and encouragement at all stages of the project. It was their constant motivation that kept the project going with enthusiasm throughout the two semesters. We would also like to thank our senior, Amit Jain, for initiating us into this interdepartmental endeavour in the first place.

References

[1] F. Adachi, M. Sawahashi, and K. Okawa, "Tree structured generation of orthogonal spreading codes with different lengths for the forward link of DS-CDMA mobile radio," *IEE Electronic Letters*, vol. 33, no. 1, pp. 27–28, 1997.

[2] J.J. Garcia-Luna-Aceves and J. Raju, "Distributed assignment of codes for multihop packet-radio networks," in *Proc. IEEE MILCOM*, 1997.

[3] Alan A. Bertossi and Maurizio A. Bonuccelli, "Code assignment for hidden terminal interference avoidance in multihop packet radio networks," *IEEE/ACM Transactions on Networking*, vol. 3, no. 4, Aug 1995.

[4] Limin Hu, "Distributed code assignments for CDMA packet radio networks," *IEEE/ACM Transactions on Networking*, vol. 1, no. 6, Dec 1993.

[5] *Digital Communications*, pp. 758–785, McGraw Hill, 3rd edition, 1995.

[6] Nathan Yee, J.P. Linnartz, and G. Fettweis, "Multicarrier CDMA in an indoor wireless radio network," in *Proc. IEEE PIMRC*, Sep 1993, pp. 109–113.

[7] Shinsuke Hara and Ramjee Prasad, "Design and performance of a Multicarrier CDMA system in frequency selective rayleigh fading channels," *IEEE Trans. on Veh. Tech.*, vol. 48, no. 5, pp. 1584–1595, Sep 1999.

[8] S. Srivastava, S. Tripathi, D. Sanghi, and A.K. Chaturvedi, "Using extraneous subcarriers in MC-CDMA for control data transmission," *in preparation*.

[9] Shinsuke Hara and Ramjee Prasad, "Overview of multicarrier CDMA," *IEEE Communications Magazine*, pp. 126–133, Dec 1997.

[10] Nathan Yee, J.P. Linnartz, and G. Fettweis, "Multicarrier cdma in an indoor wireless radio network," in *Proc. IEEE PIMRC*, Sep 1993, pp. 109–113.

[11] Shinsuke Hara and Ramjee Prasad, "Design and performance of multicarrier cdma system in frequency-selective rayleigh fading channels," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 5, pp. 1584–1595, Sep 1999.

[12] R.V. Nee and R. Prasad, *OFDM for Wireless Multimedia Communications*, Artech House Publishers, 2000.

[13] *Mobile Communications Engineering*, pp. 40–44, McGraw Hill, 1995.

[14] L. J. Jr. Cimini, "Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing," *IEEE Transactions on Communications*, vol. 33, no. 7, pp. 665–675, July 1985.

[15] "Making the leap to 4G wireless," .

[16] Ewa Kubicka and A. J. Schwenk, "An introduction to Chromatic Sums," in *Proc. of ACM Computer Science Conference*, 1989, pp. 39–45.

[17] Yunaan Wu, Qian Zhang, Wenzu Zhu, and S.Y. Kung, "Spreading code assignment in an ad hoc DS-CDMA wireless network," in *Proc. ICC'02*, 2002.

[18] Thit Minn and Kai-Yeung Siu, "Dynamic assignment of orthogonal-variable-spreading-factor codes in W-CDMA," *IEEE Journal on Selected Areas in Communications*, pp. 1429–1440, Aug 2000.

[19] M. Dell' Amico, M.L. Merani, and F. Maffioli, "Efficient algorithms for assignment of OVSF codes in wideband CDMA," in *Proc. ICC'02*, 2002.