CDMA CODE ASSIGNMENT IN WIRELESS AD HOC NETWORKS

XIANG-YANG LI∗ WEN-ZHAN SONG ∗

Abstract. Many CDMA code assignment algorithms have been developed and studied for cellular wireless networks, however, little is known about the ad hoc wireless networks. In this chapter, we first briefly introduce several Orthogonal Constant Spreading Factor (OCSF) CDMA code assignment schemes based on distance-2 vertex coloring algorithm. We then present several distributed OVSF-CDMA code assignment algorithms for wireless ad hoc networks modelled by unit disk graph. Orthogonal Variable Spreading Factor (OVSF) CDMA code has the ability to support higher and variable data rates with a single code using one transceiver. We first study how to assign OVSF-CDMA code such that the total throughput achieved is within a constant factor of the optimum. Then we give a method such that the minimum rate achieved is within a constant factor of the minimum rate of any valid code assignment. A method that can approximate both the minimum rate and total throughput is also presented. Finally, we present a post processing method to further improve these code assignments. All those methods use only \(O(n)\) total messages (each with \(O(\log n)\) bits) for an ad hoc wireless network of \(n\) devices modelled by UDG.

Key words. CDMA, code assignment, OCSF, OVSF, vertex coloring, throughput, bottleneck, interference graphs, wireless ad hoc networks.

1. Introduction. To increase the capacity of wireless networks, frequency spectrum has to be reused as it is one of the scarce resources available. Several multiple access methods are used in wireless networks, e.g., conventional FDMA (frequency division multiple access) and TDMA (time division multiple access). In the next generation of wireless systems, it is expected that a majority of the traffic will carry bursty data, which is drastically different from the voice traffic carried in the existing second-generation wireless systems. To access a mixture of multimedia applications, the system must support variable transmission rates for different users. International Mobile Telephony 2000 (IMT-2000) system aims to support differentiated quality-of-service (QoS) guarantees for emerging multimedia applications. To fulfill such applications’ requirements, the Universal Mobile Telecommunication System (UMTS) proposes employing the wideband code-division multiple-access (W-CDMA) technology. CDMA (code division multiple access) provides higher capacity, flexibility, scalability, reliability and security than conventional FDMA and TDMA. In a CDMA system, the communication channels are defined by the pseudo-random codewords, which are carefully designed to cancel each other out as far as possible. Every bit of data is multiplied by the codeword used by the communication channel. The number of duplicates, which is equal to the length of the codeword, is known as the spreading factor. For example, the Walsh code, used by the cdmaOne cellular system, consists of 64 codewords, each 64-bits long. The inverse to the length of the codeword is known as the rate of the codeword. There is a trade-off on the length of the codewords. On one hand, longer codewords can increase the number of channels and the robustness of the communications. On the other hand, since the raw rate seen by the user is inverse to the codeword length, longer codewords would result in low data rate of the communication channels.

In a direct-sequence code-division multiple-access (DS-CDMA) system, each multiple-access user is assigned a unique signature code sequence. In the forward (base-to-mobile) link, the assigned code are mutually orthogonal. In a second generation wireless CDMA system such as IS-95, each mobile user is assigned a single orthogonal

∗Department of Computer Science, Illinois Institute of Technology, 10 W. 31st Street, Chicago, IL 60616. Email: xli@cs.iit.edu, songwen@iit.edu
constant-spreading-factor (OCSF) code. It is possible to support higher data rates in DS-CDMA system by assigning a multiple of OCSF codes to a call. This mode of operation is called multicode CDMA (MC-CDMA). In an alternative CDMA scheme known as OVSF-CDMA, each user is assigned a single orthogonal variable-spreading-factor (OVSF) code. In this case, a higher data rate access is possible by using a lower spreading factor. Both MC-CDMA and OVSF-CDMA have been proposed in UMTS/IMT-2000 for supporting variable data rates. MC-CDMA requires multiple transceiver units to support higher data rates, thus resulting in increased hardware complexity. On the other hand, using OVSF-CDMA, only a single transceiver unit is required per user. Therefore, in terms of hardware complexity for mobile handsets, OVSF-CDMA is preferred over MC-CDMA for higher data rate transmission. However, some important issues must be resolved in a CDMA system based only on OVSF codes. An OVSF-CDMA system has several constraints such as code blocking, coarsely quantized data rates and a limitation on the maximum data rate (due to minimum spreading factor requirement). In particular, code blocking in OVSF-CDMA leads to a higher call blocking rate for higher data rate users. As a result, an OVSF-CDMA system with code blocking may have a lower spectral efficiency than a MC-CDMA system.

Many CDMA code assignment algorithms have been developed and studied for cellular wireless networks, however, little is known about the ad hoc wireless networks. In wireless ad hoc networks, there are no wired infrastructures. Multi-hop communication (carried out by the relaying of intermediate nodes) is required when the receiver node is not within the sender’s transmission range. Nodes need to know which code to use in transmitting or receiving a packet. Code assignment problem is trivial in small networks, but assigning a unique code to each transmitter or receiver in a large or growing network is inefficient. Spatial reuse of codes becomes increasingly important when CDMA is extended to a large multihop Packet Radio Network (PRN) or so called multihop ad hoc networks.

In this chapter, we first briefly introduce several Orthogonal Constant Spreading Factor (OCSF) CDMA code assignment schemes based on distance-2 vertex coloring algorithm in Section 3. We then introduce several distributed OVSF-CDMA code assignment algorithms for wireless ad hoc networks modelled by unit disk graph in Section 4. Orthogonal Variable Spreading Factor (OVSF) CDMA code has the ability to support higher and variable data rates with a single code using one transceiver. We first study how to assign OVSF-CDMA code such that the total throughput achieved is within a constant factor of the optimum. Then we give a method such that the minimum rate achieved is within a constant factor of the minimum rate of any valid code assignment. A method that can approximate both the minimum rate and total throughput is also presented. Finally, we present a post processing method to further improve these code assignments. All our methods use only $O(n)$ total messages (each with $O(\log n)$ bits) for an ad hoc wireless network of $n$ devices modelled by UDG. We conclude this chapter in Section 5 with a discussion of possible future works.

2. Preliminaries. Problems: What is a good network structure for CDMA protocols? How many orthogonal codes are needed in different schemes? How can one assign codes in a dynamic topology?

There are no absolute answers, but the topology of a wireless ad hoc network is determined by transmitting powers and topographic features and we can derive a guideline: adjust each terminal’s transmitting power to limit the number of in-range neighbors so as to maximize network throughput. Recently, topology control has
drawn significant research interest and several energy efficient topology [17, 32, 27] has been proposed to guarantee degree-bounded by $\Delta$. In CDMA system, the advantage of degree-bounded network structure is that number of codes needed are known as a priori. Worth to mention that, energy conservation and network performance are two most critical issues in wireless ad hoc network, besides node degree bounded, the topology proposed in [32, 27] are also power spanner and planar, which hence support energy efficient routing.

A wireless ad hoc network is presented by a graph $G = (V, E)$, where $V$ is the set of nodes and $E$ set of logical links. A logical link is bilateral. We assume:

1. A logical link $(u, v)$ means that node $u$ considers node $v$ as a valid neighbor for packet forwarding and adjusts transmitting power according to their distance, and vice versa.
2. Single transceiver per node, so communications is half-duplex.
3. Omni-directional antennas are used.
4. There is an existing routing protocol with up-to-date neighbor table.

For spread spectrum code-division without loss of generality a particular set of codes with two properties are assumed:

1. Each signal has low auto-correlation except for $\tau = 0$ and can therefore be easily distinguished from a time-shifted version of itself.
2. Each signal can easily be distinguished from every other signal on the set.

In ad hoc networks, same code can not be assigned to two wireless devices if they cause interferences. Two kinds of interference could occur:

1. **Primary Interference**: Primary Interference happens when two transmissions in the same code arrive at a receiver simultaneously.
2. **Secondary Interference**: Secondary interference (or called hidden terminal problem) happens if a third device is within the transmission regions of two nodes using the same code.

The *interference graph* is the graph over all wireless nodes and has an edge $uv$ if $u$ and $v$ will generate interference when they are assigned the same code.

### 3. Code Assignment in OCSF-CDMA based Wireless Ad hoc Networks

Assigning frequency channel efficiently in unit disk graphs has been well-studied [12, 21] but little is known about assigning codes for CDMA wireless ad hoc networks while achieving some global quality such as the total throughput or the bottleneck of the networks. In CDMA wireless ad hoc networks, four code assignment schemes were proposed:

1. **Common Code Assignment (CCA)** All terminals use common spreading code to transmit all the packets.
2. **Receiver-based Code Assignment (RCA)** Each terminal is assigned a receiving code such that no two logical neighbors of any node will have the same code. Using TCA scheme, we need find a spreading code for each node to use in receiving packets, with the constraint that all logical neighbors of a give transmitting node have different receiving codes. In this scheme, a receiver only has to listen to one code, but the **primary interference** may occur.
3. **Transmitter-based Code Assignment (TCA)**[19] All neighbor nodes of a given node have different codes for transmitting. Using TCA scheme, we need find a spreading code for each node to use in transmitting packets with the constraint that all logical neighbors of a given receiving node have different transmitting codes.
4. **Pairwise Code Assignment (PCA)**[15] A new approach to assign codes to
transmitter-receiver pair (edge) such that no two adjacent edges in the logical topology have the same code. This scheme can result in a smaller number of codes than RCA or TCA scheme in a sparse topology.

In RCA and TCA it is possible for two adjacent nodes to have the same code without violating constraints. RCA and TCA are both two-hop coloring problems and they have the same constraints, so they are equivalent problems and only TCA is discussed. PCA corresponds to the graph edge-coloring problem, i.e., all links to a node have different codes. The same code is used for both transmission and reception over a link. PCA schemes retain the same interference avoidance properties of TCA ones, while requiring a more expensive hardware though (in some cases) a smaller number of codes, and yielding a slightly worse performance[15]. In the other hand, in a sparsely connected network, a PCA system provides a comparable throughput with much fewer codes than a TCA system.

For calculating the number of codes needed for TCA, $G_{TCA}$ is constructed as shown in Figure 3.1(a). The importance of $G_{TCA}$ is that legal coloring of graph $G_{TCA}$ implies a legal TCA solution to graph $G$. Denote the number of codes needed for TCA, $\#(TCA)$, according to the famous Brook and Vizing theorem [3, 14]: $\#(TCA) \leq \min\{\Delta(\Delta - 1) + 1, |V|\}$ where $\Delta$ is maximum node degree. Simple greedy algorithm: sequentially assigning a code randomly from the possible code set requires $\Delta(\Delta - 1)+1$ codes. the complexity for this is $O(\Delta^2 |V|)$, or $O(\Delta |E|)$ for a network topology with maximum degree $\Delta$.

![Fig. 3.1. Different Code Assignment Schemes (a) Original Communication Graph G (b) Interference Graph G_{TCA} based on TCA Scheme (c) Interference Graph G_{PCA} based on PCA Scheme](image-url)

A PCA problem can be solved by a TCA algorithm through a transformation shown in Figure 3.1(c). Solving the coloring problem on $G_{PCA}$ solves the corresponding PCA problem on G. The maximum degree in $G_{PCA}$ is at most $2(\Delta - 1)$ when there is an edge with maximum-degree nodes at both ends. Greedy algorithm for PCA typically needs fewer codes than greedy TCA. For example when $\Delta = 6$, codes required by greedy TCA and PCA are 31 and 11, respectively. Hence, the code assignment problem actually becomes a minimum vertex coloring problem in the secondary interference graph described above.

The minimum (proper) vertex coloring of the interference graph has been studied in the context of channel assignment in wireless ad hoc networks channelized by FDMA, TDMA or OCSF-CDMA [7, 8, 9, 11, 23, 24, 28, 26]. The majority of these works simply presented networking protocols to obtain a proper coloring without addressing the computational complexity and the optimization. Sen and Huson [25] gave a proof of the NP-hardness of the vertex coloring in interference graph even when all nodes are located in a plane and have the same transmission radii. A problem related to the vertex coloring of the interference graphs is the distance-2
vertex coloring of a graph [16]. A distance-2 vertex coloring of a graph $H$ is a coloring of the vertices such that any two vertices separated by at most two hops receive different colors. In other words, it is a proper vertex coloring of $H^2$, the square graph of $H$, which is the graph obtained by creating an edge between each pair of vertices of $H$ separated by at most two hops in $H$. When all nodes have equal transmission radii, their interference graph happens to be the square of unit-disk graph over these nodes, and hence in this case, the vertex coloring of the interference graph is the same as a distance-2 vertex coloring of a unit-disk graph.

Graf et al. [12] discussed four classes of disk graphs and studied the relations between their chromatic number and the clique number. They proved that, for all such four classes of graphs, their chromatic number is within a constant factor of the clique number. They considered the unit disk graphs, intersection disk graphs (corresponding to disk graph models here), containment disk graphs (corresponding to conflict graph models here), and double disk graphs. Here double disk graph is defined over a set of nodes. Each node $u$ defines two disks $D(u,r_u)$ and $D(u,R_u)$. The double disk graph has an edge $uv$ iff $D(u,r_u)$ intersects $D(v,R_v)$ or $D(v,r_v)$ intersects $D(u,R_u)$.

Let $\delta(G)$ denote the largest $d$ such that $G$ contains a subgraph $H$ in which each vertex has degree at least $d$. It was proved by Szekeres and Wilf [29] that, every graph $G$ can be colored in $\delta(G) + 1$ colors. Then Hochbaum [13] presented a method to find the value of $\delta(G)$ and gave an efficient method to color $G$ using $\delta(G) + 1$ colors with only $O(|V| + |E|)$ time. For the completeness of presentation, we review the algorithm here. To evaluate $\delta(G)$, it dismantles $G$ by successive removals of vertices of minimum degree and all incident edges. Let $v_i$ denote the $i$th vertex removed from $G$ and $G_{i+1}$ be the graph after $v_i$ is removed (set $G_1 = G$). The degree of $v_i$ in graph $G_i$ is called its valid degree. Set $\delta(G)$ as the maximum valid degree of all nodes. Let $v_j$ be the node with the maximum valid degree $\delta(G)$. Then, $v_j$ has $\delta(G)$ neighbors among the vertices $v_{j+1}, v_{j+2}, \ldots, v_n$. In addition, all nodes $v_i$ with $i > j$ have degree at least $\delta(G)$ in graph $G_j$. To color $G$ in no more than $\delta(G) + 1$ colors, it scans the sequences of $v_i$’s from $v_n$ to $v_1$ and assigns to each $v_i$ the smallest positive integer not yet assigned to any of its neighbors. Marathe et al. [20] also used this method to color the disk graphs and show an approximation factor of 6.

**Theorem 3.1.** The above coloring method achieves a constant approximation ratio for the interference graph models introduced in this chapter. **Proof.** It was already known that, for any graph $G$, $G$ can be colored by $\delta(G) + 1$ colors. Let $H$ be a subgraph such that all nodes have at least $\delta(G)$ degree in $H$. Let $u$ be the node of $H$ with the smallest radius. Let $N_H(u)$ be all neighbors of $u$ in $H$. Then $|N_H(u)| \geq \delta(G)$. Consider the induced coloring on $N(u)$ by any coloring of $G$. The node in $N(u)$ with the same color form an independent set. It was proved in [18] that, if $u$ has a radius less than all its neighbors $N_H(u)$, then the maximum independent set in $N_H(u)$ has size at most 40 for the interference graph model. Let $OPT$ be the chromatic number of the corresponding graph. There are only $OPT - 1$ colors for nodes in $N(H)$. Then $|N(H)| \leq 40(OPT - 1)$ for the interference graph. Thus, the colors used by the above method is at most $40 \cdot OPT - 39$ for the interference graph. 

4. **Code Assignment in OVSF-CDMA based Wireless Ad hoc Networks.** Motivated by the support of variable rate data service at low hardware cost, a variable-length code, known as orthogonal variable-spreading-factor (OVSF) code, was developed [1] in 1997. The idea of the OVSF code is to allow the codewords in
the code to have variable lengths, and a higher-rate request is assigned by a single shorter codeword. The generation of OVSF code can be depicted by the code-tree structure as shown in Figure 4.1 (a). The code-tree is a balanced binary tree, whose vertices represent the codewords. The root, which is at the level 0, is associated with the codeword 1. Recursively, if a vertex has codeword $C$, then its two children have codewords $CC$ and $C\overline{C}$ respectively, where $\overline{C}$ is the complement of $C$. Thus, at level $\ell$ there are $2^\ell$ codewords, each $2^\ell$ bits long. Notice, not all codewords in an OVSF code are orthogonal to each other. Two OVSF codewords are orthogonal to each other if and only if neither is an ancestor, or equivalently, a prefix of the other. In CDMA system, two nodes possibly interfering each other should use two codes that are orthogonal.

As always, it is convenient to represent the channels by colors. For the channelization by OVSF code, a representation of the channels or the codewords by positive binary colors is given in Figure 4.1 (b). Two binary colors are said to be prefix-free if neither is a prefix of the other. Then, two binary colors are prefix-free if and only if the corresponding codewords are orthogonal. Additionally, we associate with each binary color with a rate attribute, which is equal to the rate of the corresponding codeword. Thus, the rate of an $\ell$-bit binary color is equal to the $2^{-\ell+1}$. We also say that an $\ell$-bit color is in the $\ell$-th layer of the OVSF-CDMA code tree structure. The root has layer 1.

Most prior studies of conflict-free OVSF-CDMA code assignment have been restricted to complete graphs in the context of channel assignment to nodes in a single cell of an OVSF-CDMA cellular networks [2, 6, 10, 22]. The OVSF-CDMA code assignment of complete graphs is fairly easy. Indeed, since each node must receive a unique code different from others, a OVSF-CDMA code assignment can thus be represented by a binary tree with one-to-one correspondence between the nodes (or their colors) and the leaves. Every binary tree with $n$ leaves leads to a valid OVSF-CDMA code assignment. If the binary tree is full, then the corresponding code assignment achieves the maximum throughput one. If the binary tree is full and balanced, the corresponding coloring achieves both maximum throughput and maximum bottleneck. Furthermore, if each node specifies a demand equal to a power of $1/2$, then as an immediate application of Kraft’s inequality, all demands can be satisfied if and if the total demands is at most one. The dynamic reassignment of colors to meet a new demand is addressed in [22], where they discussed an optimal dynamic code assignment (DCA) scheme using OVSF code. It is optimal in the sense that it minimizes the number of OVSF codes that must be reassigned to support a new call. By admitting calls that would normally be blocked without code assignments, the spectral efficiency of the system is also maximized.
We consider a wireless ad hoc network consisting of a set \( V \) of nodes distributed in a two-dimensional plane, assuming that the nodes are static or can be viewed as static during a reasonable period of time and every node has the same transmission range. Obviously, if there is a node \( w \) inside the common transmission region of two nodes \( u \) and \( v \), then \( w \) is a hidden terminal. We only consider the code assignment based on TCA scheme, since PCA scheme can be solved similarly, we will let \( G \) denote the interference graph modelling the secondary interference. We will study how to assign OVSF-CDMA code to each wireless nodes in a distributed manner, i.e., each ad hoc device will determine its own CDMA code based on its neighbors’ information. In a CDMA system, each device is assigned a code such that two possibly interfering nodes are assigned orthogonal codes.

In an OVSF-CDMA wireless ad hoc network, a channel assignment must be conflict-free, i.e., any pair of neighboring nodes in the interference graph must receive orthogonal codewords. With the representation of the codewords by the binary colors, an interference-free channel assignment is equivalent to a vertex coloring of the interference graph by positive binary colors such that adjacent nodes in the interference graph receive prefix-free colors. Notice that, the OVSF-CDMA code assignment problem possesses several unique features that makes itself different from the distance-2 vertex coloring. We will study various optimization problems on prefix-free vertex coloring of the interference graphs. Specifically, we will address how to maximize the total throughput, the minimum rate, and the both at the same time.

Given a conflict-free OVSF-CDMA code assignment \( \{c_v : v \in V\} \) of the interference graph \( G \), its throughput and bottleneck are defined as \( \sum_{v \in V} 2^{-|c_v|+1} \) and \( \min_{v \in V} 2^{-|c_v|+1} \) respectively, where \( |c_v| \) denotes the number of bits of the color \( c_v \). In other words, the throughput of a conflict-free OVSF-CDMA code assignment is the sum of the rates of the assigned codes, and its bottleneck is the minimum of the rates of the assigned codes. The throughput of an interference graph \( G \), denoted by \( \tau (G) \), is then the maximum of the throughput over all possible conflict-free OVSF-CDMA code assignment of \( G \). Similarly, the bottleneck of an interference graph \( G \), denoted by \( \beta (G) \), is then the maximum of the bottleneck over all possible conflict-free OVSF-CDMA code assignment of \( G \).

We now describe the localized methods to assign conflict-free OVSF-CDMA code to the wireless nodes. Remember that our objective is to maximize the minimum rate assigned to all nodes or maximize the summation of rates assigned to all nodes, or both. The total communication cost of those methods is at most \( O(n \log n) \) bits for a wireless network of \( n \) nodes, which is also within a constant factor of the optimum since any method needs at least \( n \log n \) bits to announce the existence of all wireless devices. Hereafter, let \( N_k(u) \) be the set of all wireless nodes that are at most \( k \) hops away from node \( u \) in the original unit disk graph. Notice that letting every node broadcast its identity to its one-hop neighbors enables all node to find their one-hop neighbors using only total \( n \) communications. If every node knows its exact geometry location, a communication efficient protocol [4] is known to find all two hop neighbors of all nodes using at most \( O(n) \) communications. As analyzed before, to generate a conflict-free CDMA code assignment based on TCA, we only need consider secondary interference (hidden terminal) with each node \( u \). Obviously, \( H_2(u) = N_2(u) - N_1(u) \) is the set of all possible interference nodes. Consequently, we are only interested in \( H_2(u) \), let \( d_2(u) \) be the cardinality of \( H_2(u) \).

4.1. Maximize Network Throughput. Intuitively, we would like to have as many wireless devices as possible to be assigned with short OVSF-CDMA code so
as to maximize the throughput. Note that in any conflict-free OVSF-CDMA code assignment all nodes receive the same OVSF-CDMA code form an independent set in the interference graph. Greedy algorithms have been used and proved to be efficient in many problems and we found that greedy OVSF-CDMA code assignment method also generates a code assignment that is almost as good as the optimum. First-fit coloring is a class of greedy algorithms for vertex coloring. Assume that there is an ordering of all wireless nodes, we then assign code to the wireless devices sequentially according to the associated ordering by assigning each device the least possible OVSF-CDMA code. In particular, in any first-fit coloring, all nodes receiving the same smallest OVSF-CDMA code must form a maximal independent set. Such maximal independent set is desirable to be a small constant approximation of a maximum independent set to maximize the total throughput intuitively.

It is not hard to see that the performance of a first-fit code assignment depends on the associated nodes ordering. Indeed, there always exists a node ordering in which the first-fit coloring generates an optimal OVSF-CDMA code assignment. However, such node ordering is unlikely to be found in polynomial time due to the expected NP-hardness of the max-throughput OVSF-CDMA code assignment. So we seek some nodes ordering that produces a OVSF-CDMA code assignment approximates well the optimal assignment in terms of the maximum throughput and the node ordering can be generated efficiently. We propose to use several different nodes ordering for OVSF-CDMA code assignment. We show that all of them can produce a code assignment with total throughput $O(\tau(G))$ and use total communications $O(n)$. Hereafter, we assume that each message has $O(\log n)$ bits. Notice that all node orderings used in this chapter are just partial ordering computed locally. To compute such ordering, we assume that a synchronous communication is used.

The code assignment methods in this chapter compute a partial ordering based on the node degree. The algorithms first construct the interference graph and then construct a maximal independent set based on node degree. Node in the computed maximal independent set are assigned the shortest code 10. For the remaining nodes, we assign code using the first fit heuristics based on the partial ordering from the degree.

Compute Maximal Independent Set. The nodes compute a maximal independent set as follows. Originally, all nodes are unmarked. A node $u$ marks itself InIS if $u$ has the smallest degree among all its unmarked neighbors in the interference graph $G$. Node $u$ then informs its neighbors in the interference graph of its mark InIS using a communication efficient protocol described later. If a node $v$ receives a mark InIS from its neighbor in the interference graph, then $v$ marks itself NotInIS and informs its neighbors in the interference graph of its mark NotInIS. All nodes with mark InIS form a maximal independent set and it is known [18, 31] that its size is within a constant factor of the maximum independent set.

Algorithm 1. Max-Throughput Assignment
1. All nodes broadcast its ID to nodes within its transmission range. If secondary interference is concerned, all nodes together compute the $H_2(u)$ for every node $u$ using a communication efficient protocol in [4]. Each node $u$ computes its degree $d(u)$ in the interference graph $G$ and informs its neighbors in $G$ about its degree $d(u)$ using a communication efficient protocol.
2. All nodes together compute a maximal independent set as describe above: node with smaller degree has priority and ties are broken by smaller ID. In other words, the maximal independent set is computed in the increasing order of
node degree.

3. Node \( u \) assigns a CDMA/OVSF code represented by binary 10 (see Figure 4.1 (b) for illustration) if \( u \) is in the maximal independent set. \( u \) then informs its neighbors in the interference graph about its CDMA/OVSF code.

4. If a node \( v \) receives a CDMA/OVSF code from its neighbor in the interference graph, \( v \) marks the corresponding code used in the CDMA/OVSF tree structure stored locally.

5. We then assign code to the remaining nodes. If a node \( u \) has the smallest degree among all its neighboring nodes in the interference graph without CDMA/OVSF code, then node \( u \) finds the smallest layer \( h > 0 \) in the CDMA/OVSF tree structure stored locally such that layer \( h \) has at least 2 free codes\(^1\) not used by its neighbors in \( G \). \( u \) then picks the first unused code in layer \( h \) and informs its neighbors in the interference graph \( G \) about its CDMA/OVSF code. The picked code is called the first fit code for node \( u \).

Obviously the above method generates a conflict-free OVSF-CDMA code assignment since, for each pair of neighboring nodes \( u \) and \( v \) in the interference graph, the node with smaller degree can only assign code after it gets the code of the other node.

The total communication cost is \( O(n) \) since we use communication efficient protocol to collect \( H_2(u) \) for all nodes and to inform the assigned OVSF-CDMA code to its neighbors in the interference graph.

Notice that our method basically first computes a maximal independent set and then assigns the shortest OVSF-CDMA code to the nodes in this independent set. For the remaining nodes, we assign codes in the order of decreasing node degree. Notice that the throughput generated by such method depends on the number of nodes in the computed independent and also the number of nodes assigned before each node.

We then show that this method indeed approximate the optimum throughput \( \tau(G) \). To analyze the approximation ratio of different methods on the throughput, we first study the structure of some optimum CDMA code assignment, called canonical coloring. We [30] define the canonical coloring as follows. Given a graph \( G = (V, E) \), partition the vertex set \( V \) into independent sets \( V_1, V_2, \cdots, V_k \) with

\[ V_1 \geq V_2 \geq \cdots \geq V_k. \]

Let \( G_0 = G \) and \( G_i \) be the graph of removing the vertices \( V_i \) and the incident edges from graph \( G_{i-1} \), for \( 1 \leq i \leq k \). Vertex set \( V_i \) is a maximum independent set of graph \( G_{i-1} \). For \( 1 \leq i \leq k - 1 \), all nodes in \( V_i \) receive the code \( 1^0 \), and all nodes in \( V_k \) receive the code \( 1^k \). Obviously, the throughput of such canonical coloring is

\[ \sum_{i=1}^{k-1} \frac{|V_i|}{2^i} + \frac{|V_k|}{2^{k-1}}. \]

Notice that, If there are multiple maximum independent set \( V_i \), we have to choose the one that produces the largest maximum independent set \( V_2 \). Similarly, the selection of the first \( i \) maximum independent sets \( V_1, V_2, \cdots, V_i \) produces the largest maximum independent \( V_{i+1} \), for \( 1 \leq i < k \). Call such sequence of maximum independent set as canonical maximum independent set decomposition and the corresponding coloring canonical coloring.

\(^1\)Using a code such that there are at least 2 unused codes in that layer guarantees that there is always code available for unassigned neighboring nodes later.
Theorem 4.1. [30] The canonical coloring maximizes the throughput. **Proof.** Let \( S \) be the set of all colors used by a coloring, each of which represents a distinctive CDMA code. For each item \( x \) in \( S \), let \( \ell(x) \) denote the code length of the color \( x \). Thus, its rate is \( 2^{-\ell(x)} \).

Consider any optimum coloring that maximize the throughput. For each item \( x \) in \( S \), let \( \omega(x) \) denote the number of mobile hosts receiving the corresponding CDMA code \( x \). Consequently, the total rate (throughput) of such coloring is \( \sum_{x \in S} \omega(x) \cdot 2^{-\ell(x)} \). Since the coloring must be prefix-free, the colors used by any valid coloring can be represented by a binary tree \( T \) with \( |S| \) leaves representing all used colors. Obviously, the tree \( T \) of any optimal coloring is always a full binary tree: if there is one leave node is used and its sibling node is not used, we can use its parent node instead, which improves the throughput.

It is easy to prove that for every optimal coloring, the least frequent used color \( x \), i.e., \( \omega(x) \) is minimum among all used colors in \( S \), has the longest code \( \ell(x) \). This can be proved by a simple contradiction. Assume that \( \omega(x) \) is the smallest and there is another code \( y \in S \) with larger code length, i.e., \( \ell(y) > \ell(x) \), but \( \omega(x) < \omega(y) \). By swapping the code \( x \) and \( y \), the throughput is improved by \( \omega(x) \cdot 2^{-\ell(y)} + \omega(y) \cdot 2^{-\ell(x)} - \omega(x) \cdot 2^{-\ell(x)} - \omega(y) \cdot 2^{-\ell(y)} \). This is equal to \( (\omega(y) - \omega(x)) \cdot (2^{-\ell(x)} - 2^{-\ell(y)}) > 0 \).

For the two longest sibling code \( x \) and \( y \), if we merge them to its parent node \( z \) by setting \( \omega(z) = (\omega(x) + \omega(y))/2 \), and removing codes \( x \) and \( y \), the total throughput does not change. Additionally, since \( x \) and \( y \) have the lowest weights (because they have the longest codeword), node \( z \) has the smallest weight in the new tree. It implies that node \( z \) has the largest height in the new tree. This implies that the imbalanced full binary tree shown in the following Figure 4.2 is an optimal. It is easy to show that the number of nodes using the code in level \( i \) must be \( |V_i| \). This finishes the proof.

![Figure 4.2. Canonical coloring is optimum.](image)

This theorem implies that the maximum throughput of any code assignment is at most the independence number \( \alpha(G) \) of the interference graph \( G \). Based on this observation, we can assign the code as follows. First, compute a maximal independent set that approximates the maximum independent set (with approximation ratio \( \rho \)). Then assign the node in the maximal independent set a code 10 (its rate is 1/2). For the remaining nodes, we can recursively finding the maximal independent set and assign code 10 for the maximal independent set retrieved in the \( i \)th iteration but the messages of this approach could be very large. Actually, for the remaining nodes, any conflict-free OVSF-CDMA code assignment algorithm works here. Obviously, the throughput generated by assigning nodes in maximal independent set a code 10 is at least \( \rho \cdot \alpha(G)/2 \). In other words, a \( \rho \)-approximation algorithm for the maximum independent set assignment problem works here. Obviously, the throughput generated by assigning nodes in maximal independent set a code 10 is at least \( \rho \cdot \alpha(G)/2 \). In other words, a \( \rho \)-approximation algorithm for the maximum throughput CDMA code assignment algorithm.
Theorem 4.2. Algorithm 1 generates a conflict-free OVSF-CDMA code assignment whose total throughput is within \( \varrho/2 \) of the optimum, where \( \varrho \) is the approximation ratio of the maximum independent set algorithm.

To reduce communication cost for announcing node degree at the initial stage, we may construct a maximal independent set based on node ID. However, simulation shows the performance is not as good as the algorithm described above, we take it as a tradeoff. This could because Algorithm 1 computes a maximal independent set using the nodes with smaller degree, which in turn produces a larger maximal independent set. When every node knows its position, we can further improve the throughput of the assigned OVSF-CDMA codes as follows. We still construct a maximal independent set first, but instead of using the node ID or the degree as selection criterion, we select a node \( u \) to the maximal independent set if all unassigned neighboring nodes are inside one half of the disk centered at \( u \). Notice that such node \( u \) always exists since the most left undecided node trivially satisfies this condition. Then the following steps are also similar with Algorithm 1 except that all other nodes assign the first fit code in increasing ordering of x-coordinates instead of degree. The better throughput is achieved because of larger maximal independent set constructed.

4.2. Maximize Network Bottleneck. In previous sections, we showed how to assign OVSF-CDMA codes to wireless nodes such that the total throughput of the network is maximized. We continue to study how to assign OVSF-CDMA codes such that the minimum rate of all nodes is maximized. Intuitively, to maximize the throughput, from the canonical code assignment discussion, we know that the used code should be imbalanced. However, to maximize the minimum rate of the network, the assigned codes should be as balance as possible. Clearly, the previous greedy method does not generate a balanced code assignment. In this section, we present a novel distributed method to assign a balanced OVSF-CDMA code.

Our method is based on the following observation. Consider a node \( u \) and all its neighbors in the interference graph \( G \). If all such neighbors and \( u \) form a clique, then the minimum rate is approximately \( 1/d \), where \( d \) is the size of the clique. This is achieved when all nodes use the code in level \( \log d \). In other words, to maximize the minimum rate assigned, node \( u \) cannot choose the first fit code; it has to use a code in level close to \( \log d \). Putting in other way, node \( u \) cannot be too greedy and it has to leave good codes for its neighbors. We then present our algorithm as follows.

Algorithm 2. Max-Bottleneck Assignment
1. All nodes together compute the interference graph using a communication efficient protocol. Assume that each node \( u \) knows its degree \( d(u) \) in the interference graph. Each node \( u \) informs its neighbors in the interference graph \( G \) about its degree \( d(u) \). Thus, we assume that each node \( u \) knows the degree \( d(v) \) for every neighboring node \( v \) in \( G \).
2. If node \( u \) has the largest degree \( d(u) \) among all neighboring nodes in the interference graph without OVSF-CDMA code, node \( u \) picks the first unmarked code in the code tree stored locally from layer \( \ell \), where
   \[
   2^{\ell-2} < d(u) + 1 \leq 2^{\ell-1}.
   \]
   Ties are broken by smaller ID. Node \( u \) informs all its neighbors in the interference graph the selected OVSF-CDMA code using a communication efficient protocol.
3. If a node \( v \) receives a OVSF-CDMA code from its neighbor in the interference graph, \( v \) marks the corresponding code used, and marks all prefix-codes of
this code conflicted in the OVSF-CDMA tree structure stored locally.

Here we say a code is marked if it is either marked as used or conflicted. Later, we will describe how to compress the code to improve the throughput and bottleneck. That method actually requires that, for each used code, the node \( v \) will remember how many times this code is used by its neighbors. We first show that every node \( u \) can find an unmarked code in layer \( \ell \). Notice that the number of total OVSF-CDMA codes in layer \( \ell \) is \( 2^{\ell - 1} \).

**Theorem 4.3.** Algorithm 2 generates a conflict-free OVSF-CDMA code assignment. **Proof.** It is obvious that if node \( u \) can find a OVSF-CDMA code, then the found code does not conflict with the code assigned to any other neighboring node. It remains to show that \( u \) can find an unmarked OVSF-CDMA code in layer \( \ell \). Notice that when we assign code to node \( u \), node \( u \) has the largest degree \( d(u) \) among all neighboring nodes without CDMA code in the interference graph. This implies that all neighbor nodes with assigned code must have degree at least \( d(u) \). Thus, the codes already used by its neighboring nodes, at the moment of assigning code for \( u \), are on or below layer \( \ell \). Since there is only one code at layer \( \ell \) that is the prefix of a code on or below layer \( \ell \), the number of codes at layer \( \ell \) that are marked and thus cannot be used by \( u \) is at most \( d(u) \). Notice that at layer \( \ell \), there are at least \( d(u) + 1 \) codes. Clearly, there is still one unused OVSF-CDMA code when processing node \( u \). This finishes the proof.

We then show that the above Algorithm 2 generates a conflict-free OVSF-CDMA code assignment whose minimum rate is within a constant factor of the optimum.

**Theorem 4.4.** Algorithm 2 generates a code assignment whose minimum rate is within \( 2^{-\left\lfloor \log_2 c_2 \right\rfloor} \) factor of any conflict-free OVSF-CDMA code assignment where \( c_2 = 1/13 \). **Proof.** Consider a node \( u \) with the largest degree \( d(u) \) in the interference graph. We partition the disk \( D(u, 2) \) into 13 equal-sized sectors. Obviously all neighboring nodes of \( u \) inside one sector form a complete subgraph in \( G \). Using the pigeonhole principle, it is easy to show that among the neighbors of \( u \) in the interference graph and \( u \), the minimum clique size is at least \( c_2 d(u) + 1 \), where \( c_2 = 1/13 \) for the secondary interference graph we adopted. For a clique of size \( q \), the minimum rate of nodes in the clique is obviously at most \( 2^{-\left\lfloor \log_2 q \right\rfloor} \). Thus, for any assignment, the minimum rate among neighboring nodes of node \( u \) and \( u \) is at most \( 2^{-\left\lfloor \log_2 (c_2 d(u) + 1) \right\rfloor} \). Obviously, the rate by our approach is \( 2^{-\left\lfloor \log_2 (d(u) + 1) \right\rfloor} \). It is easy to show that

\[
2^{-\left\lfloor \log_2 (d(u) + 1) \right\rfloor} \geq 2^{-\left\lfloor \log_2 c_2 \right\rfloor} \cdot 2^{-\left\lfloor \log_2 (c_2 d(u) + 1) \right\rfloor}
\]

In other words, the minimum rate achieved by Algorithm 2 is at least \( 1/16 \) of optimum. This finishes the proof.

Notice that, the assigned codes can be further improved. For example, if all neighboring nodes of \( u \) in the interference graph already has a OVSF-CDMA code assigned, node \( u \) can pick the first fit code from the lowest layer. We will discuss in detail how to further compact the assigned OVSF-CDMA code to improve the performance later.

### 4.3. Maximize Network Throughput and Bottleneck Simultaneously.

In previous two subsections, we have described several methods to assign OVSF-CDMA code to wireless nodes in a distributed manner to maximize either the total throughput of the network or maximize the bottleneck rate of the network, but not both. As we discussed before, to maximize the throughput, the assigned codes should
be as imbalance as possible, while to maximize the bottleneck rate, the assigned codes should be as balanced as possible. It seems impossible to have a OVSF-CDMA code assignment that approximates both the total throughput and the bottleneck rate. In this subsection, we show that by retreating little bit of both requirement, we can achieve this. Our method is almost a straightforward combination of previous methods. We first assign the shortest code to the nodes in a maximal independent set. For the remaining nodes, we assign a balanced code.

**Algorithm 3. Max-Throughput and Max-Bottleneck Code Assignment**

1. All nodes together compute the interference graph using a communication efficient protocol. Assume that each node $u$ knows its degree $d(u)$ in the interference graph. Each node $u$ informs its neighbors in the interference graph $G$ about its degree $d(u)$. Thus, we assume that each node $u$ knows the degree $d(v)$ for every neighboring node $v$ in $G$.

2. All nodes together compute a maximal independent set using the degree as selecting criterion: node with smaller degree has priority and ties are broken by smaller ID. In other words, the maximal independent set is computed in the increasing order of node degree. Node ID or $x$-coordinate can also be used as the selecting criterion for maximal independent set. A node assigns OVSF-CDMA code 10 if it is in the computed maximal independent set. The remaining steps will assign code for the other nodes not in the maximal independent set.

3. If node $u$ is not assigned and has the largest degree $d(u)$ among all neighboring nodes in the interference graph without CDMA code, node $u$ picks the first unmarked code from layer $\ell$, where

\[2^{\ell-3} < d(u) \leq 2^{\ell-2}.

Node $u$ informs all its neighbors in the interference graph the selected code using a communication efficient protocol.

4. If a node $v$ receives a OVSF-CDMA code from its neighbor in the interference graph, $v$ marks the corresponding code used, and marks all prefix-codes of this code conflicted in the OVSF-CDMA tree structure stored locally.

Similar to Theorem 4.3, Algorithm 3 also generates a conflict-free OVSF-CDMA code assignment. A subtle difference is that, in Algorithm 3, node $u$ chooses code from layer that has $2^{\ell-1} \geq 2d(u)$ codes, while in Algorithm 2, node $u$ chooses code from layer that has at least $d(u) + 1$ codes. This is because each node $u$ not in the maximal independent set is connected to some node, say $v$, in the maximal independent set, and node $v$ already uses OVSF-CDMA code 10. Thus, node $u$ can only use the bottom half codes in Figure 4.1 (b), i.e., all OVSF-CDMA codes starting with 11. In other words, we push the code to one layer below. Since for neighboring nodes of $u$, at least one node is already assigned a OVSF-CDMA code 10, the number of nodes that need balanced codes is thus at most $d(u)$, including $u$ itself, instead of $d(u) + 1$ for Algorithm 2.

It is not difficult to prove the following theorem about the quality of assigned OVSF-CDMA codes.

**Theorem 4.5.** Algorithm 3 generates a conflict-free OVSF-CDMA code assignment whose total throughput is within $g/2$ of the optimum, and whose minimum rate is within $2^{-\left\lceil \log_2 c_2 \right\rceil - 1}$ factor of the optimum, where $g$ is the approximation ratio of the maximum independent set algorithm, and $c_2 = 1/13$. 

13
4.4. Post Processing. Although we proved that all our algorithms generate a conflict-free OVSF-CDMA code assignment either whose total throughput or whose minimum rate, or both, is within a constant factor of the optimum, the assigned code can still be improved. We then present a communication efficient method to further improve the code assignment generated by previous algorithms. We assume that originally every node has a mark to indicate whether it has performed the improvement. We also assume that, for each code $\mathcal{C}$, each node $u$ stores the number of times $t(\mathcal{C})$ the code $\mathcal{C}$ is used by its neighboring nodes when assign OVSF-CDMA code, and stores the number, denoted by $p(\mathcal{C})$, of children codes that are used by its neighbors. Obviously, if either $t(\mathcal{C}) > 0$ or $p(\mathcal{C}) > 0$ this code $\mathcal{C}$ cannot be used by node $u$ since it causes interference with neighboring nodes in $G$. When $p(\mathcal{C}) > 0$, we say the code conflicted and when $t(\mathcal{C}) > 0$ we say the code used. When $p(\mathcal{C}) = 0$ and $t(\mathcal{C}) = 0$, we say the code is unmarked.

Algorithm 4. Compress Assigned OVSF-CDMA Code
1. If node $u$ is not marked as improved and has the smallest degree $d(u)$ in $G$ among all neighboring nodes of $u$ without improving the OVSF-CDMA code in the interference graph $G$, node $u$ picks the first unmarked code from the smallest layer that has at least one unmarked code.
2. Node $u$ informs all its neighbors in the interference graph about the new selected code $\mathcal{C}_n$ and the old code $\mathcal{C}_o$ using a communication efficient protocol. Node $u$ also marks itself improved.
3. When a node $v$ receives a pair of the new code $\mathcal{C}_n$ and old code $\mathcal{C}_o$ from its neighbor $u$, $v$ increases the used times $t(\mathcal{C}_n)$ by 1 and decreases the used times $t(\mathcal{C}_o)$ by 1. Node $v$ also increases the value $p(\mathcal{C})$ of all prefix-code $\mathcal{C}$ of the new code $\mathcal{C}_n$ by 1 and decreases the value $p(\mathcal{C})$ of all prefix-code $\mathcal{C}$ of the old code $\mathcal{C}_o$ by 1.

Algorithm 4 will be used to improve the OVSF-CDMA code assignment generated by the three algorithms presented in the previous subsections. Notice that, when we improve the OVSF-CDMA code assignment, we start from the node with the smallest degree. The reason is as follows. Assume a node with the smallest degree improves the assigned code to some upper layer, this node cannot further improve the code after some of its neighbors improve their assigned OVSF-CDMA codes since the codes in upper layer cannot be freed by its neighbors. This implies that the code generated by Algorithm 4 is locally optimum. Our simulations show that Algorithm 4 improves the performance by a factor or almost 2.

5. Conclusion. In this chapter, we first reviewed the CDMA code assignment schemes for wireless ad hoc networks and introduced distance-2 vertex coloring algorithm for OCSF-CDMA ad hoc networks. Then several efficient distributed OVSF-CDMA code assignment algorithms are discussed in detail. We first studied how to assign OVSF-CDMA code such that the total throughput achieved is within a constant factor of the optimum. Then we introduced a method such that the minimum rate achieved is within a constant factor of the minimum rate of any valid code assignment. A method that can approximate both the minimum rate and total throughput was also presented. Finally, we presented a post processing method to further improve the code assignment. The simulations showed that the methods introduced here performs even much better than the theoretical analysis.

Notice that our methods can also be used to generate conflict-free OVSF-CDMA code assignment for wireless ad hoc networks that are not modelled by unit disk graphs. However, it is unclear how the number of messages in the protocol could be
bounded by $O(n)$: how to collect $H_2(u)$ efficiently and how to inform the assigned OVSF-CDMA code to the neighbors of interference graph efficiently. If the network is not dense enough, a straightforward method by letting all nodes inside the transmission range of $u$ to relay the message may also be good enough practically. When the network is dense, such flooding could be very expensive. Selective forwarding [5] could be one way to save the messages. We leave it as future work to design communication efficient protocol to assign OVSF-CDMA code for wireless ad hoc networks modelled other than UDG.

Notice that the throughput of the network is related to the node degree: if the node degrees in the interference graph increase, then the OVSF-CDMA codes become longer and thus the throughput decreases. The throughput of a network is proved to be within a constant factor of the independence number $\alpha(G)$. However, this independence number is somewhat a global property and not easy to capture. We conjecture that the throughput is also within a constant factor of the following number $\sum_{u \in V} d(u)$, where $d(u)$ is the degree of node $u$ in the interference graph. We leave it as an open question.

REFERENCES


