Theoretically Good Distributed CDMA/OVSF Code Assignment for Wireless Ad Hoc Networks

Xiang-Yang Li* Peng-Jun Wan*†

Abstract

Orthogonal Variable Spreading Factor (OVSF) CDMA code has the ability to support higher and variable data rates with a single code using one transceiver. A number of 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 paper, we propose several distributed CDMA/OVSF code assignment algorithms for wireless ad hoc networks modelled by unit disk graph (UDG). We first study how to assign CDMA/OVSF code such that the total throughput achieved is within a constant factor of the optimum. Then we give a distributed method such that the minimum rate achieved is within a constant factor of the minimum rate of any valid code assignment. A distributed 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.

Keywords: CDMA, code assignment, vertex coloring, throughput, bottleneck, interference graphs, wireless ad hoc networks.

1 Introduction

Mobile ad hoc wireless networking has received significant attentions over the last few years. To increase the capacity of the network, frequency spectrum has to be reused as it is one of the scarcest 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), and recently developed CDMA (code division multiple access). Same channel is not assigned to two wireless devices if it causes interferences. Here the interferences could be *primary interference* or *secondary interference*. Primary interference occurs if two wireless devices use the same channel and one is inside the transmission region of the other. And the secondary interference (or called hidden terminal problem) occurs if a third device is within the common transmission regions of two nodes using the same frequency channel. The *interference graph* is the graph over all wireless nodes and has an edge uv if two wireless nodes u and v will generate interference when they are assigned the same channel. Assigning frequency channel efficiently in unit disk graphs has been well-studied [12, 17] but little is known about assigning CDMA/OVSF code for wireless ad hoc networks while achieving some global quality such as the total throughput or the bottleneck of the networks. In this paper, we are interested in assigning CDMA/OVSF codes to ad hoc wireless devices such that either the total network throughput or the bottleneck or both are maximized approximately.

CDMA 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 wireless communication channel. The number of duplicates, which is equal to the length of the codeword, is known as the *spreading factor*. 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 lower data rate of the communication channels. For example, the Walsh code, used by the cdmaOne cellular system, consists of 64 codewords, each 64-bits long.

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 in 1997 [1]. The idea of the OVSF code is to allow the codewords in the CDMA code to have variable

*Department of Computer Science, Illinois Institute of Technology, 10 W. 31st Street, Chicago, IL 60616, USA. The research of Xiang-Yang Li and Peng-Jun Wan was supported in part by NSF CCR-0311174. Email: xli@cs.iit.edu, wan@cs.iit.edu

†Department of Computer Science, Hong Kong City University, Hong Kong.

lengths, and a higher-rate request is assigned with a single shorter codeword. The generation of OVSF code can be depicted by the code-tree structure as shown in Figure 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 a codeword $\mathbb{C}$, then its two children will have codewords $\mathbb{C}\mathbb{C}$ and $\mathbb{C}\overline{\mathbb{C}}$ respectively, where $\overline{\mathbb{C}}$ is the complement of $\mathbb{C}$. Thus, at level ℓ there are 2^ℓ codewords, each 2^ℓ 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 a CDMA system, two nodes possibly interfering each other should use two codes that are orthogonal.

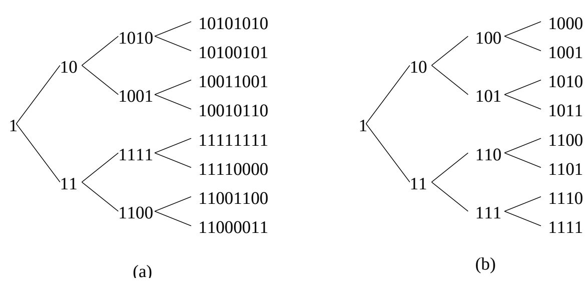

Figure 1: OVSF code: (a) code-tree structure; (b) binary color representation.

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 integers (called colors hereafter) is given in Figure 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 each binary color with a *rate* attribute, which is equal to the rate of the corresponding codeword. Thus, the rate of an ℓ-bit binary color is equal to $2^{-\ell+1}$. We also say that an ℓ-bit color is in the ℓ-th layer of the CDMA/OVSF code tree structure. The root has layer 1.

All prior studies of conflict-free CDMA/OVSF code assignment have been restricted to *complete* graphs in the context of channel assignment to nodes in a single cell of an CDMA/OVSF cellular networks [2, 5, 9, 18]. The CDMA/OVSF code assignment of complete graphs is fairly easy. Indeed, since each node must receive a unique code different from others, a CDMA/OVSF code assignment can thus be represented by a binary tree with one-to-one correspondence between the nodes (or their colors) and the leaves of the tree. Every binary tree with n leaves leads to a valid CDMA/OVSF 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 code assignment achieves both the 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 only if the total demands is at most one. The dynamic reassignment of colors to meet a new demand is addressed in [18].

A proper vertex coloring is to assign each vertex a color such that two adjacent vertices receive different colors. 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 CDMA/OVSF [6, 7, 8, 11, 19, 20, 23, 22]. The majority of these works simply presented networking protocols to obtain a proper vertex coloring without addressing the computational complexity and/or the optimization. Sen and Huson [21] 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. Muqattash and Krunz [16] studied the power control for wireless ad hoc networks using CDMA based MAC. Several CDMA code assignment methods were proposed for wireless ad hoc networks [10, 13], but no theoretical analysis of their performances was given.

A problem related to the vertex coloring of the interference graphs is the *distance-2 vertex coloring* of a graph [14]. A *distance-2 vertex coloring* of a graph H 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. However, the CDMA/OVSF code assignment problem possesses several unique features that makes itself different from the distance-2 vertex coloring. The colors assigned to two adjacent nodes in H^2 should only be different for a vertex coloring problem, while these two colors should further be prefix-free for CDMA/OVSF code assignment.

The main contributions of this paper are as follows. We propose several efficient distributed CDMA/OVSF code assignment algorithms for wireless ad hoc networks modelled by unit disk graph. We first study how to assign CDMA/OVSF 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 simultaneously is also presented. Finally, we present a post processing method to further improve the performance of 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 also conducted extensive simulations to study the practical performances of our methods. Our methods not only have theoretically proven performance bounds but also perform close to optimum practically.

It will be seen that the correctness of the protocols presented in this paper does not require that the wireless networks are modelled by UDG. Out methods apply to all wireless networks when the communication channels are varying with distance, with time, and with obstacles. The UDG network model only enables us to prove that our methods have theoretical performance guarantees. The correctness of our methods also do not depend on the node positions. The usage of the node positions enables us to show that the total messages needed by each of our method is $O(n)$. The position error will not affect our methods as long as the position error will not change the topology of the network, i.e., the UDG topology derived from the perceived nodes' positions is the same as the actual physical network topology.

This paper is not intended to solve all critical issues in CDMA based wireless ad hoc networks. In addition to the code assignment problem, there are several other important issues that should be addressed so the CDMA/OVSF code can be used practically for wireless ad hoc networks. The first issue is about how the communication of code assignment methods is performed before a CDMA/OVSF code is assigned to nodes (sort of chicken and egg problem here). For this, we assume that there is already a separated control channel available for communication when the wireless network is deployed. Another issue is the mobility of wireless nodes. When wireless nodes move around and in consequence of the movement the interference graph is changed, we should re-assign the CDMA/OVSF codes to wireless nodes. The algorithms proposed in this paper mostly use the information local to each node to select its CDMA/OVSF code. Consequently, when nodes are mobile, we could update the codes fairly quickly. The moving node will check if movement causes its code to be invalid. If so, it will run our methods to find the new code and inform its neighbors about this new code. Here, instead of letting the ID be the rank in assigning code, we will use the updating time as the rank or the moving speed of a node as the rank (slow moving node will have chance to get higher rate code). To maintain a good performance, we may need to re-assign the codes periodically for all nodes after the codes are updated locally for a while.

The third issue is the time synchronization among the mobile wireless nodes. In MANET, it is impossible to have a common time reference for all the transmissions that arrive at an intended receiver, since signals originate from different transmitters. Equipping the mobile wireless nodes with GPS receivers can reduce some asynchronizations but cannot eliminate it. In addition, these transmissions suffer different time delays since they propagate through different paths, which introduces another domain of difficulty of synchronization. In an asynchronous system, it is thus impossible to design spreading codes that are orthogonal for *all* time offsets [16]. The synchronization of wireless ad hoc networks is thus not an easy task, which should be addressed by further research. A possible direction is to design code assignment methods to assign codes that can tolerate the asynchronization to some extent while it still achieves some sort of performance bounds.

The remainder of the paper is organized as follows. In Section 2, we present several efficient distributed CDMA/OVSF code assignment methods. We conclude our paper in Section 3 with a discussion of possible future works.

2 Distributed Code Assignment

We consider a wireless ad hoc network consisting of a set V of n nodes distributed in a two-dimensional plane. The nodes are assumed to be static or can be viewed as static during a reasonable period of time. We assume that the omnidirectional antenna is used by the wireless devices. The transmission range of a node is thus often modelled as a disk centered at this node. Assume all nodes have the same transmission radius r, thus, wireless ad hoc networks are modelled by unit disk graphs (UDG), in which two nodes are connected iff their Euclidean distance is no more than r. Obviously, if there is a node w inside the common transmission region of two nodes u and v, then w is a hidden terminal. From now on, we will let G denote the interference graph, which models the primary interference or both the primary interference and the secondary interference.

In a CDMA/OVSF 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. A CDMA/OVSF code assignment is said to be *valid* if its is conflict-free. With the representation of the codewords by the binary colors, a conflict-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. We propose to 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 both at the same time.

Given a conflict-free CDMA/OVSF code assignment

$$\{c_v \mid v \in V, \forall\, uv \in G,\ c_u \text{ and } c_v \text{ are orthogonal}\}$$

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 CDMA/OVSF 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 CDMA/OVSF code assignments of G. Similarly, the *bottleneck of an interference graph* G, denoted by $\beta(G)$, is then the maximum of the bottleneck over all conflict-free CDMA/OVSF code assignments of G.

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, $d_k(u)$ be the cardinality of $N_k(u)$. Obviously, nodes that can have primary interference with a node u are $N_1(u)$ only; the nodes that can have either primary interference or secondary interference (hidden terminal) with u are $N_2(u)$ only. Consequently, we are interested in $N_k(u)$ for $k = 1$ if primary interference is concerned, and $k = 2$ if secondary interference is concerned. Letting every node broadcast its identity to its one-hop neighbors enables all nodes to find their one-hop neighbors using only total n communications. If every node knows its exact geometry location, a communication efficient protocol [3] is known to find all two-hop neighbors of all nodes using at most $O(n)$ communications. Thus, we assume that each node knows $N_k(u)$ and then $d_k(u)$ for $k = 1, 2$.

2.1 Maximize Throughput $\tau(G)$

Greedy algorithms have been used and proved to be efficient in many problems and we found that greedy CDMA/OVSF 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 shortest possible CDMA/OVSF code. In particular, in any first-fit coloring, all nodes receiving the same smallest CDMA/OVSF 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.

Clearly, the performance of a first-fit code assignment depends on the node ordering used. Indeed, there always exists a node ordering in which the first-fit coloring generates an optimal CDMA/OVSF code assignment. However, such node ordering is unlikely to be found in polynomial time due to the expected NP-hardness of the max-throughput CDMA/OVSF code assignment. So we seek some node ordering that produces a CDMA/OVSF code assignment approximating well the optimal assignment in terms of the maximum throughput; such node ordering should be generated efficiently. We propose several different node orderings for CDMA/OVSF code assignment. We show that all of them 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 paper are just partial ordering computed locally. To compute such ordering, we assume that a synchronous communication is used.

Our code assignment methods compute a partial ordering based on the node ID, or degree, or the node position. Here we assume that every node has a distinctive ID and knows its position if communication efficient protocol is needed. The algorithms first construct the interference graph and then construct a maximal independent set based on ID, or degree, or the node's geometry position. Nodes 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 ID, or degree, or node position. Algorithm 1 presents our method (run by every node u) of assigning CDMA/OVSF code based on ordering by ID to maximize the throughput.

Finding a maximal independent set (MIS) for a wireless ad hoc network modelled by UDG is well-studied. For completeness of presentation, we review the general distributed method that computes a MIS based on a rank. Algorithm 2 reviews the distributed method run by every node u. We define three marks for nodes: *White*, *InIS* and *NotInIS*. Initially, all nodes are marked as *White*. A node already in the computed maximal independent set is marked *InIS* and the node that is ruled out of the maximal independent set is marked *NotInIS*. All nodes with mark *InIS* form a maximal independent set and it is known [15, 25] that its size is within a constant factor of the maximum independent set when G is UDG.

Obviously Algorithm 1 generates a conflict-free CDMA/OVSF code assignment since, for each pair of neighboring nodes u and v in the interference graph, the node with larger ID 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 $N_2(u)$ for all nodes and to inform the assigned CDMA/OVSF code to its neighbors in the interference graph.

Notice that our method basically first computes a

Algorithm 1 Max-Throughput Using ID-ordering by u

1: Node u sends a message to tell its ID to all nodes inside its transmission region. If secondary interference is not permitted, node u finds $N_2(u)$ using a communication efficient method [3].

2: All nodes collectively find a maximal independent set (MIS) based on ID using Algorithm 2.

3: Node u assigns a CDMA/OVSF code represented by binary 10 (see Figure 1 (b) for illustration) if u is in the maximal independent set. Node u then informs its neighbors in the interference graph about its CDMA/OVSF code.

4: If node u receives a CDMA/OVSF code from its neighbor in the interference graph, u marks the corresponding code *used* in the CDMA/OVSF tree structure stored locally.

5: We then assign code to the remaining nodes. If node u has the smallest ID 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. Node 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.

Algorithm 2 Compute a MIS based on a $rank$ by a node u

1: $mark(u) \leftarrow$ *White*;

2: Node u sends $rank(u)$ to its one-hop neighbors in G.

3: **while** node u received a message from its neighbors **do**

4:     **if** message is $rank(v)$ **then**

5:         Node u stores $rank(v)$; $mark(u) \leftarrow$ ***InIS*** if u has the *smallest* rank among all its *White* neighbors in G.

6:         Node u sends a message $InIS(u)$ to its neighbors in G using a communication efficient protocol [3].

7:     **else if** message is $InIS(v)$ **then**

8:         $mark(u) \leftarrow$ ***NotInIS***; Node u then sends a message $NotInIS(u)$ to its neighbors in G using a communication efficient protocol [3].

9:     **else if** message is $NotInIS(v)$ **then**

10:         Node u updates the mark of v stored locally.

maximal independent set and then assigns the shortest CDMA/OVSF code to the nodes in this independent set. For the remaining nodes, we assign codes in the order of increasing node ID. Notice that the throughput generated by such method would be larger if the size of the computed independent set is larger or the number of nodes assigned before each node is smaller. Intuitively, we could further improve the throughput if we use an ordering based on the node degree in the interference graph: using the increasing order of the node degree for computing a maximal independent set and assigning codes to the remaining nodes. The algorithm is described as follows.

Algorithm 3 Max-Throughput Assignment by Degree

1: 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 [3].

2: All nodes together compute a maximal independent set using the degree as selecting criterion: node with smaller degree has a higher priority and ties are broken by smaller ID.

3: A node assigns CDMA/OVSF code 10 if it is in the maximal independent set.

4: All other nodes assign the *first fit* code in the *increasing* order of degree using method similar to the last step of Algorithm 1.

We then show that the above methods 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*. In [24], we defined 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^i0, 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 sets V_1, 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*.

Theorem 1 *[24] The canonical coloring maximizes the throughput.*

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 ϱ). Then assign the nodes in the maximal independent set a code 10 (its rate is $1/2$). For the remaining nodes, we can recursively find the maximal independent set and assign code 1^i0 for the maximal independent set retrieved in the ith iteration but the messages of this approach could be very large. To optimize the message complexity, Algorithms 1 and 3 used a different approach for the remaining nodes (actually any conflict-free CDMA/OVSF code assignment for the remaining nodes works here). Obviously, the throughput generated by assigning nodes in maximal independent set a code 10 is at least $\varrho \cdot \alpha(G)/2$. In other words, a ϱ-approximation algorithm for the maximum independent implies a $\varrho/2$ approximation algorithm for the maximum throughput CDMA code assignment algorithm. This implies the following theorem (see appendix for the proof).

Theorem 2 *Algorithm 1 and 3 generate a code assignment whose throughput is at least $\varrho/2$, where $\varrho = 1/5$ if only primary interference is concerned and $\varrho = 1/13$ if secondary interference is also concerned.*

When every node knows its position, we can further improve the theoretical lower bounds on the throughput of the assigned CDMA/OVSF 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.

Theorem 3 *Algorithm 4 generates a code assignment whose throughput is at lease $\varrho/2$, where $\varrho = 1/3$ if only primary interference is concerned and $\varrho = 1/7$ if secondary interference is also concerned.*

The proof of this theorem is similar to the proof of Theorem 2 and thus is omitted. The approximation ratio could be further improved to be better than $\varrho/2$, which is analyzed as follows. The new approach will compute a maximal independent V_1', and then compute a maximal independent V_2' for the remaining nodes. Clearly, the total communication cost is still $O(n)$. The nodes in V_1' will receive a code 10 and the nodes in V_2' will receive a code 110. We then assign codes to other nodes using a method that is similar to the last step of Algorithm 1.

Algorithm 4 Max-Throughput Using Position-ordering by u

1: Every node finds its neighbors in the interference graph using a communication efficient protocol in [3].
2: All nodes together compute a MIS based on rank $(x(u), y(u), ID(u))$ using Algorithm 2, where $x(u)$, and $y(u)$ are the x-coordinate and y-coordinate of a node u.
3: Node u gets code 10 if it is in the computed MIS.
4: All nodes not in MIS get the *first fit* code in an *increasing* ordering of $(x(u), y(u), ID(u))$ using method similar to the last step of Algorithm 1.

Theorem 4 *An ϱ-approximation algorithm for the maximum independent set gives a $\frac{5}{8}\varrho$-approximation algorithm for the maximum throughput CDMA code assignment.*

See appendix for the proof. We then summarize our results in the following main theorem.

Theorem 5 *If node position is known, we can produce a CDMA/OVSF code assignment, using $O(n)$ total messages, whose total throughput is at least $5/24$ of the optimum when only primary interference is concerned, and $5/56$ of the optimum when secondary interference is concerned.*

Notice that our theoretical analysis is pessimistic and our simulations show that the practical performances of our methods are much better than these pessimistic analysis.

2.2 Maximize Bottleneck $\beta(G)$

In previous sections, we showed how to assign CDMA/OVSF codes to wireless nodes such that the total throughput of the network is maximized. We continue to study how to assign CDMA/OVSF codes such that the minimum rate of all nodes is maximized. Intuitively, to maximize the throughput, from the canonical code assignment discussion, the assigned codes should be imbalanced. However, to maximize the minimum rate of the network, the assigned codes should be as balanced as possible. Clearly, the previous greedy methods do not generate a balanced code assignment. In this section, we present a novel distributed method to assign a balanced CDMA/OVSF 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. The following Algorithm 5 details our method.

Algorithm 5 Max-Bottleneck by Degree-ordering by u

1: All nodes together compute the interference graph G using a communication efficient protocol. Assume that each node u knows its degree $d(u)$ in G. Each node u informs its neighbors in G its degree $d(u)$.

2: Node u constructs a local binary code tree T.

3: If node u has the *largest* degree $d(u)$ among all neighbor nodes in G without CDMA/OVSF code, where ties are broken by smaller ID, node u picks the first unmarked code in the code tree T stored locally from layer ℓ, where

$$2^{\ell-2} < d(u) + 1 \leq 2^{\ell-1}.$$

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

4: If a node u receives a CDMA/OVSF code from its neighbor in G, u marks the corresponding code *used* in T, and marks all prefix-codes of this code *conflicted* in T.

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 ℓ. Notice that the number of total CDMA/OVSF codes in layer ℓ is $2^{\ell-1}$.

Theorem 6 *Algorithm 5 generates a conflict-free CDMA/OVSF code assignment.*

PROOF. It is obvious that if node u can find a CDMA/OVSF code, then the found code does not conflict with the code assigned to any other neighbor. It remains to show that u can find an unmarked CDMA/OVSF code in layer ℓ. Notice that when we assign code to node u, node u has the largest degree $d(u)$ among all neighbors 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 neighbors, at the moment of assigning code for u, are on or below layer ℓ. Since there is only one code at layer ℓ that is the prefix of a code on or below layer ℓ, the number of codes at layer ℓ that are marked and thus cannot be used by u is at most $d(u)$. Notice that at layer ℓ, there are at least $d(u) + 1$ codes by the selection of layer ℓ. Clearly, there is still one unused CDMA/OVSF code when processing node u. This finishes the proof. □

We then show that the above Algorithm 5 generates a conflict-free CDMA/OVSF code assignment whose minimum rate is within a constant factor of the optimum.

Theorem 7 *Algorithm 5 generates a code assignment whose minimum rate is within a constant factor of any conflict-free CDMA/OVSF code assignment.*

PROOF. Consider a node u with the largest degree $d(u)$ in the interference graph. If primary interference is concerned, we partition the disk $D(u, 1)$ into 6 equal-sized sectors. If secondary interference is also concerned, we partition the disk $D(u, 2)$ into 13 equal-sized sectors. We already showed that all neighbors 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 $c \cdot d(u) + 1$, where $c = 1/6$ for primary interference graph, and $c = 1/13$ for secondary interference graph. For a clique of size q, the minimum rate of nodes in the clique is obviously at most $2^{-\lceil \log_2 q \rceil}$. Thus, for any assignment, the minimum rate among neighbors of u and node u is at most $2^{-\lceil \log_2 (c \cdot d(u)+1) \rceil}$. Obviously, the rate by our approach is $2^{-\lceil \log_2 (d(u)+1) \rceil}$. It is easy to show that

$$2^{-\lceil \log_2 (d(u)+1) \rceil} \geq 2^{-\lceil \log_2 c \rceil} \cdot 2^{-\lceil \log_2 (c \cdot d(u)+1) \rceil}$$

In other words, the minimum rate achieved by Algorithm 5 is at least $1/8$ of the optimum if only the primary interference is concerned and $1/16$ of the optimum if the secondary interference is also concerned. This finishes the proof. □

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

2.3 Maximize $\tau(G)$ and $\beta(G)$

In previous two subsections, we have described several methods to assign CDMA/OVSF code to wireless nodes in a distributed manner to maximize either the total throughput or the bottleneck rate of the network, but not both. As we discussed before, to maximize the throughput, the assigned codes should be as imbalanced as possible, while to maximize the bottleneck rate, the assigned codes should be as balanced as possible. It seems impossible to have a CDMA/OVSF code assignment that approximates both the total throughput and the bottleneck rate. In this subsection, we show that by retreating little bit of both requirements, 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 6 Max-Throughput and Bottleneck by a node u

1. All nodes together compute the interference graph G. Each node u computes its degree $d(u)$ in G and informs its neighbors in G about its degree $d(u)$.
2. All nodes together compute a MIS based on the rank by degree using Algorithm 2. Node ID or $(x(u), y(u), ID(u))$ can also be used as the rank criterion. Node u gets CDMA/OVSF code 10 if it is in the computed MIS. The remaining steps will assign code for other nodes.
3. Each node u constructs a binary code tree T.
4. If node u is not assigned and has the largest degree $d(u)$ among all its neighbors in G without a CDMA code, node u picks the first unmarked code from layer ℓ in T, where $2^{\ell-3} < d(u) \leq 2^{\ell-2}$. Node u informs all its neighbors in G the selected code using a communication efficient protocol.
5. If node u receives a message from its neighbor v informing the CDMA/OVSF code of v, u marks this code *used*, and marks all prefix-codes of this code *conflicted* in tree T.

Similar to Theorem 6, Algorithm 6 also generates a conflict-free CDMA/OVSF code assignment. A subtle difference is that, in Algorithm 6, node u chooses code from layer ℓ that has $2^{\ell-1} \geq 2d(u)$ codes, while in Algorithm 5, node u chooses code from layer ℓ 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 CDMA/OVSF code 10. Thus, node u can only use the bottom half codes in Figure 1 (b), i.e., all CDMA/OVSF 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 CDMA/OVSF 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 5. It is not difficult to prove the following theorem about the quality of assigned CDMA/OVSF codes.

Theorem 8 *Algorithm 6 generates a conflict-free code assignment whose total throughput is within $\varrho/2$ of the optimum, and whose minimum rate is within $2^{-\lceil \log_2 c \rceil - 1}$ factor of the optimum, where ϱ is the approximation ratio of the maximum independent set algorithm, $c = 1/6$ for primary interference and $c = 1/13$ for secondary interference.*

Remember that, using the geometry information, the maximum independent set of the interference graph can be approximated within $1/3$ for primary interference graph and $1/7$ for secondary interference graph. If node position is unknown, the approximation ratio becomes $1/5$ and $1/13$ respectively.

2.4 Post Processing

Although we proved that all our algorithms generate a conflict-free CDMA/OVSF 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 $\mathbb{C}$, each node u stores the number of times $t(\mathbb{C})$ the code $\mathbb{C}$ is used by its neighboring nodes when assign CDMA/OVSF code, and stores the number, denoted by $p(\mathbb{C})$, of children codes that are used by its neighbors. Obviously, if either $t(\mathbb{C}) > 0$ or $p(\mathbb{C}) > 0$ this code $\mathbb{C}$ cannot be used by node u since it causes interference with neighboring nodes in G. When $p(\mathbb{C}) > 0$, we say the code *conflicted* and when $t(\mathbb{C}) > 0$ we say the code *used*. When $t(\mathbb{C}) = 0$ and $p(\mathbb{C}) = 0$, we say the code is *unmarked*. Our code compressing method basically will move the code to the upper layer as much as possible.

Algorithm 7 will be used for the CDMA/OVSF code improvement for the algorithms presented in the previous subsections. Notice that, when we improve the CDMA/OVSF 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 code of upper layer. This node cannot further improve the code after some of its neighbors improve their assigned CDMA/OVSF codes since the codes in upper layer cannot be freed by its neighbors. This implies that the code generated by Algorithm 7 is locally optimum. Our simulations show that Algorithm 7 improves the performance by a factor of almost 2.

Algorithm 7 Compress Assigned CDMA/OVSF Code by u

1: Each node u has a binary code tree T. For each code $\mathbb{C}$, it already has correct $t(\mathbb{C})$ and $p(\mathbb{C})$. All node are initially unmarked.
2: **while** u is unmarked and $d(u) \leq d(v)$ (ties are broken by ID) for all its unmarked neighbors v **do**
3: From T, node u picks the first unmarked code, say $\mathbb{C}_n$, from the smallest layer that has at least one unmarked code.
4: Node u informs its neighbors in the interference graph its new code $\mathbb{C}_n$ and its old code $\mathbb{C}_o$ using a communication efficient protocol. Node u also marks itself *improved*.
5: When node u receives a pair of the new code $\mathbb{C}_n$ and old code $\mathbb{C}_o$ from its neighbor v, u increases $t(\mathbb{C}_n)$ by 1 and decreases $t(\mathbb{C}_o)$ by 1. Node u also increases $p(\mathbb{C})$ of all prefix-code $\mathbb{C}$ of code $\mathbb{C}_n$ by 1 and decreases $p(\mathbb{C})$ of all prefix-code $\mathbb{C}$ of code $\mathbb{C}_o$ by 1.

2.5 Build Interference Graph Efficiently

So far we always assumed that all nodes can collect the neighbors in the interference graph G efficiently in a distributed manner and can inform its assigned CDMA/OVSF code to all its neighbors in G efficiently. Here, we will describe in detail how to perform these using total $O(n)$ number of messages if the wireless ad hoc networks are modelled by unit disk graphs and each node knows its geometry position.

When only the primary interference is concern, then the interference graph is the unit disk graph itself. Obviously, a node can inform all its neighbors in the unit disk graph about its assigned CDMA/OVSF code by using only one message to all nodes inside its transmission range.

We will thus concentrate on how to do so if the secondary interference is concerned. Clearly, the interference graph has exactly all links uv, where $v \in N_2(u)$. We briefly review the communication efficient method presented in [3] to collect $N_2(u)$ for every node u. Assume a maximal independent set is computed. Each node uses its adjacent node(s) in the MIS to broadcast over a larger area relevant information. Listening to the information about other nodes broadcast by the MIS nodes enables a node to compute its 2-hop neighborhood. We start from the moment the virtual backbone is already constructed by an efficient method such as [25], and every node knows the ID and the position of its neighbors. The responsibility for announcing the ID and position of a node v is taken by the MIS nodes adjacent to v. Each such MIS node assembles a packet with the ID and position of v and a variable *counter* being set to 2. The MIS node then broadcasts the packet. A connector node is used to establish a link in between several pairs of virtually-adjacent MIS nodes, and will not retransmit packets which do not travel in between these pairs of MIS nodes. The connector node will rebroadcast packets with nonzero counter originated by one of the nodes in a pair of virtually-adjacent MIS nodes, thus making sure the packet advances toward the other MIS node in the pair. When an MIS node receives a packet it checks whether this is the first message with this ID and $counter > 0$, and if yes, it decreases the counter variable and rebroadcasts the packet. A node listens to the packets broadcast by all the adjacent MIS nodes, and, using its internal list of 1-hop neighbors, checks if the node announced in the packet is a 2-hop neighbor or not - thus constructing the list of 2-hop neighbors.

The above approach can also be used by each node to inform the assigned CDMA/OVSF code to its neighbors in the interference graph.

3 Conclusion

We presented several efficient distributed CDMA/OVSF code assignment algorithms for wireless ad hoc networks modelled by unit disk graph. We first studied how to assign CDMA/OVSF code such that the total throughput achieved is within a constant factor of the optimum. Then we gave 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. If every node knows its position, we showed how to produce a CDMA/OVSF code assignment in a distributed manner, using $O(n)$ total messages, whose total throughput is at least $5/24$ of the optimum when only primary interference is concerned, and $5/56$ of the optimum when secondary interference is concerned.

Notice that our methods can also be used to generate conflict-free CDMA/OVSF 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 $N_2(u)$ efficiently and how to inform the assigned CDMA/OVSF 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 be good enough practically. When the network is dense, such flooding could be very expensive. Selective forwarding [4] could be one way to save the messages but it cannot guarantee linear number of messages. We leave it as a future work to design communication efficient protocol to assign CDMA/OVSF code for heterogeneous wireless ad hoc networks, in which different nodes may have different transmission ranges.

References

[1] ADACHI, F., SAWAHASHI, M., AND OKAWA, K. Tree-structured generation of orthogonal spreading codes with different lengths for the forward link of ds-cdma mobile radio. *IEE Electronics Letters 33*, 1 (January 1997), 27–28.

[2] AMICO, M. D., MERANI, M., AND MAFFIOLI, F. Efficient algorithms for assignment of ovsf codes in wideband CDMA. In *Proc. ICC'02* (2002).

[3] CĂLINESCU, G. Computing 2-hop neighborhoods in ad hoc wireless networks, 2003. Accepted for publication, AdHocNow.

[4] CĂLINESCU, G., MĂNDOIU, I., WAN, P.-J., AND ZELIKOVSKY, A. Selecting forwarding neighbors in wireless ad hoc networks. In *ACM DialM* (2001).

[5] CHENG, R. G., AND LIN, P. OVSF code channel assignment for IMT-2000. In *IEEE Vehicular Technology Conference* (2000), vol. 3, pp. 2188–2192.

[6] CHLAMTAC, I., AND FARAGO, A. Making transmission schedules immune to topology changes in multi-hop packet radio networks. *IEEE/ACM Transactions on Networking 2*, 1 (1994), 23–29.

[7] CHLAMTAC, I., AND KUTTEN, S. A spatial reuse TDMA/FDMA for mobile multihop radio nertworks. In *IEEE INFOCOM* (1985), pp. 389–394.

[8] EPHREMIDES, A., AND TRUONG, T. Scheduling broadcasts in multihop radio networks. *IEEE Transactions on Communications 38*, 4 (April 1990), 456–460.

[9] FANTACCI, R., AND NANNICINI, S. Multiple access protocol for integration of variable bit rate multimedia traffic in UMTS/IMT-2000 based on wideband CDMA. *IEEE Journal on Selected Areas in Communications 18*, 8 (August 2000), 1441–1454.

[10] GARCIA-LUNA-ACEVES, J. AND RAJU, J. Distributed assignment of codes for multihop packet-radio networks. *Proceedins of the IEEE MILCOM Conference*, volume 1, pages 450-454, 1997.

[11] GOLDBERG, A., AND RAO, S. Flows in undirected unit capacity networks. Tech. Rep. 97-103, NEC Research Institute, Inc, 1997.

[12] GRAF, A., STUMPF, M., AND WEISENFELS, G. On coloring unit disk graphs. *Algorithmica 20*, 3 (1998), 277–293.

[13] HU, L. Distributed code assignments for CDMA packet radio networks. *IEEE/ACM Transactions on Networking*,1:668, Dec. 1993.

[14] KRUMKE, S., MARATHE, M., AND RAVI, S. Models and approximation algorithms for channel assignment in radio networks. *Wireless Networks 7* (2001), 567–574.

[15] LI, X.-Y., AND WANG, Y. Simple heuristics and PTASs for intersection graphs in wireless ad hoc networks. In *ACM DialM* (2002).

[16] MUQATTASH, A. AND KRUNZ, M. CDMA-based MAC protocol for wireless ad hoc networks. *International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '03)* (2003), 153-164.

[17] MCDIARMID, C., AND REED, B. Colouring proximity graphs in the plane. *Discrete Mathematics 199* (1999), 123–137.

[18] MINN, T., AND SIU, K.-Y. Dynamic assignment of orthogonal variable spreading factor codes in W-CDMA. *IEEE Journal on Selected Areas in Communications 18*, 8 (August 2000), 1429–1440.

[19] NELSON, R., AND KLEINROCK, L. Spatial-TDMA: A collision-free multihop channel access protocol. *IEEE Transactions on Communications 33*, 9 (1985), 934–944.

[20] RAMANATHAN, S., AND LLOYD, E. Scheduling algorithms for multi-hop radio networks. *IEEE/ACM Transactions on Networking 1* (April 1993), 166–172.

[21] SEN, A., AND HUSON, M. L. A new model for scheduling packet radio networks. *ACM/Baltzer Journal Wireless Networks 3* (1997), 71–82.

[22] SEN, A., AND MALESINSKA, E. Approximation algorithms for radio network scheduling. In *Proceedings of 35th Allerton Coneference on Communication, Control and Computing* (1997), pp. 573–582.

[23] STEVENS, D., AND AMMAR, M. Evaluation of slot allocation strategies for TDMA protocols in packet radio networks. In *IEEE Military Communications Conference* (1990), pp. 835–839.

[24] WAN, P.-J., LI, X.-Y. AND FRIDER, O. OVSF-CDMA code assignment for wireless ad hoc networks. Accepted for publication, ACM DialM, 2004.

[25] WANG, Y., AND LI, X.-Y. Geometric spanners for wireless ad hoc networks. In *Proc. of 22nd IEEE International Conference on Distributed Computing Systems (ICDCS)* (2002).

4 Appendix

This section contains the proofs of several theorems presented in this paper.

Theorem 1 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 maximizes 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 2 is an optimal. It is easy to show that

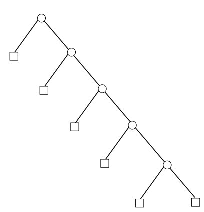

Figure 2: Canonical coloring is optimum.

the number of nodes using the code in level i must be $|V_i|$. This finishes the proof. □

Theorem 2 Algorithm 1 and 3 generate a code assignment whose throughput is at lease $\varrho/2$, where $\varrho = 1/5$ if only primary interference is concerned and $\varrho = 1/13$ if secondary interference is also concerned.

PROOF. Let's consider all nodes, denoted by V_1, that receive code 10. Clearly, V_1 is an independent set. We will show that V_1 is within ϱ factor of the maximum independent set.

If only primary interference is concerned, the interference graph is the original unit disk graph and it is well-known that each presented greedy method generates a maximal independent set whose size is at least $1/5$ of the maximum independent set. Obviously, the total throughput generated by our approach is at least $|V_1|/2$ and the optimum throughput is at most $\alpha(G) \le 5|V_1|$.

If the secondary interference is also concerned, we will prove that V_1 has size at least $1/13$ of the maximum independent set of the interference graph by showing that, for every node $u \in V_1$, there are at most 13 independent nodes in the interference graph. Let $D(x, r)$ be the disk centered at a point x with radius r hereafter. Consider a disk $D(u, 2)$ centered at node u with radius 2. Then all its neighbors $N_2(u)$ are inside the disk $D(u, 2)$. Partition this disk into 13 equal-sized sectors, each with angle $2\pi/13$. It is easy to show that the chord ab defined by the sector $\measuredangle aub$ has length $4\sin(\pi/13) < 1$. We will show that all neighboring nodes in one sector are connected. Consider any two nodes x and y from $N_2(u)$. We actually will prove a stronger result: any two neighbors of u in the sector $\measuredangle aub$ with $\|ab\| = 1$ are connected in the interference graph.

If x and y are inside $D(u, 1)$, then obviously $\|xy\| < 1$. Thus, x and y are connected in the interference graph.

If y is inside $D(u, 1)$ but x is not, then there exists a node w connected to both x and u. Clearly, y and w are all inside $D(u, 1)$ now, thus, edge yw exists in the original unit disk graph. Thus, node w is inside the common transmission range of nodes y and x. It implies that x, y are connected in the interference graph (concerning the secondary interference).

Finally, we consider the case when both x and y are not inside the disk $D(u, 1)$. Assume that node w is connected to both x and u, and node v is connected to both y and u. See Figure 3 for an illustration.

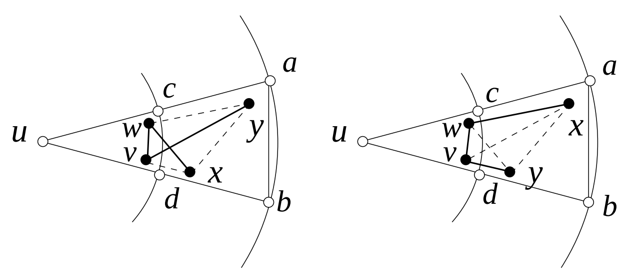

Figure 3: All neighbors in the sector conflict with each other. Here $\|uc\| = \|ud\| = \|ab\| = 1$ and $\|ua\| = \|ub\| = 2$. Left: wx and vy intersect. Right: wx and vy do not intersect.

We will then show that either $\|yx\| \le 1$, or $\|yw\| \le 1$, or $\|vx\| \le 1$. Notice that, if any one is true, then x, y are connected in the interference graph. For the sake of contradiction, assume that $\|yx\| > 1$, $\|yw\| > 1$, and $\|vx\| > 1$. We partition the region $\measuredangle aub - \measuredangle cud$ into 6 regions. Figure 4 illustrates such six partitions. Here segments ca, db, ab, eb, am have length 1.

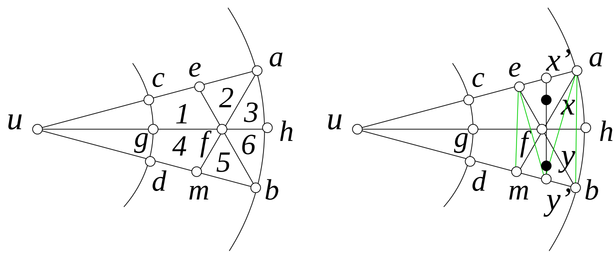

Figure 4: Left: Six possible regions we can place node x or y. Right: node two independent nodes can be in regions 2 and 5.

We then prove that any two nodes in region $efa \cup mfb$ have distance at most 1 and any two nodes in region $efbha$ have distance at most 1. Consider any two nodes x and y in the region $efa \cup mfb$. If both are in the same triangle, then clearly $\|xy\| < 1$ since the edges of the triangles have length less than 1. Otherwise, let x' and y' be the intersection point of line xy with segment ea and segment mb respectively. Figure 4 illustrates the proof that follows. Obviously, $\|xy\| \leq \|x'y'\| \leq \min(\|ey'\|, \|ay'\|)$. Note that $\|ey'\| \leq \min(\|em\|, \|eb\|) < 1$ and similarly $\|ay'\| \leq \min(\|am\|, \|ab\|) = 1$. Thus, $\|xy\| \leq 1$. Similar proof will reveal that any two nodes in region $efbha$ have distance at most 1.

If node x is in region 2, then node y cannot be in region 3, 5, and 6 since we can show that otherwise $\|xy\| \leq 1$. In other words, node y must be in region 1 or 4 in this case. Similarly, if node x is in region 3, 5, or 6, node y must be in region 1 or 4 in this case. Thus, we assume that either node x or y (say x w.l.o.g) is in region 1 by symmetry. Obviously, node y cannot be inside the disk $D(x, 1)$ since we assume that $xy \notin G$. Thus, we have to place node v inside the sector $\measuredangle cud$ but not inside the disk $D(x, 1)$ and place y inside region $efmbha$ but not inside the disk $D(x, 1)$ while still maintain $\|yv\| \leq 1$. We then show that this is impossible.

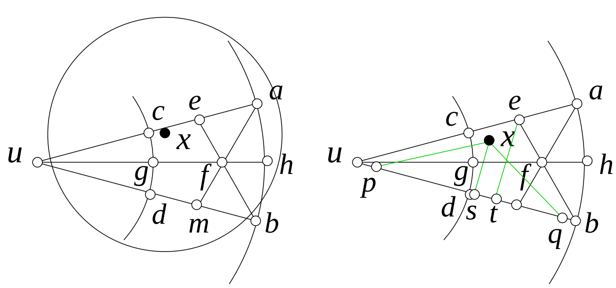

Figure 5: No two independent nodes can be placed in region 1.

If the disk $D(x, 1)$ contains the region $cgdbha = \measuredangle aub - \measuredangle cud$, then clearly node y is inside the disk $D(x, 1)$. It implies that xy is an edge in the interference graph. Let p and q be the points on line ub such that $\|xp\| = \|xq\| = 1$. Let s be the point on ub such that xs is perpendicular to segment pq and t be the point on ub such that et is perpendicular to segment pq. Clearly, $\|xs\| \leq \|et\|$ since x is inside the triangle $\triangle eub$. It is not difficult to show that $\|ce\| = \|ea\| = 1/2$. Then, $\|et\| = \|ue\| \cdot \sin(\measuredangle aub) < \frac{3}{2}\sin(\frac{\pi}{6}) = 3/4 < \sqrt{3}/2$. It implies that $\measuredangle xqp = \arcsin(\|xt\|/\|xq\|) < \frac{\pi}{3}$. Thus, edge pq is the longest in triangle $\triangle xpq$. Consequently, $\|pq\| > 1$. It is easy to show that, for any two-hop neighbor y of u connected through node v, $\|yv\| \geq \|pq\|$ if both v and y are not inside the disk $D(x, 1)$. This is a contradiction to $\|yv\| \leq 1$.

This finishes the proof of the theorem. □

Theorem 4 An ϱ-approximation algorithm for the maximum independent set gives a $\frac{5}{8}\varrho$-approximation algorithm for the maximum throughput CDMA code assignment.

PROOF. Consider a canonical maximum independent decomposition $V_1, V_2, \cdots, V_k$ of all nodes V. Here $|V_1'| \geq \varrho \cdot |V_1|$. Let $t_{i,j} = \frac{|V_i' \cap V_j|}{|V_j|}$, i.e., the portion of V_j is used in V_i'. After V_1' is generated, we know that the maximum independent set in the remaining graph has size at least $\max((1 - t_{1,1}) \cdot |V_1|, (1 - t_{1,2}) \cdot |V_2|)$, since $V_1 - V_1' \cap V_1$ and $V_2 - V_1' \cap V_2$ are still independent sets. Notice that $t_{1,1} \cdot |V_1| + t_{1,2} \cdot |V_2| \leq V_1$. Then $(1 - t_{1,1}) \cdot |V_1| + (1 - t_{1,2}) \cdot |V_2| \geq |V_2|$. It implies that V_2' has size at least $\varrho \cdot |V_2|/2$. Consequently, the throughput τ' generated by partition $V_1', V_2', \cdots, V_k', \cdots, V_{k_2}'$ is at least $\varrho \cdot (\frac{|V_1|}{2} + \frac{|V_2|}{2 \cdot 2^2})$. Remember that the canonical coloring has throughput τ at most $\frac{|V_1|}{2} + 2 \cdot \frac{|V_2|}{2^2}$ using fact $|V_i| \leq |V_2|$. From $|V_2| \leq |V_1|$, it is easy to show that $\tau' \geq \frac{5}{8}\varrho \cdot \tau$. This finishes the proof. □