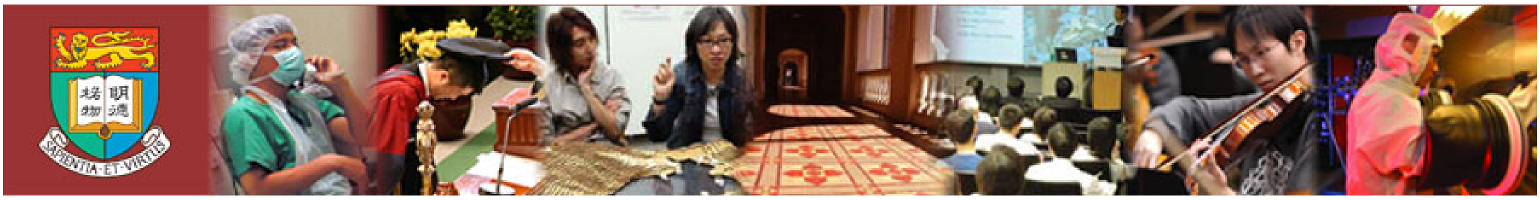


Title	Topology-Transparent Scheduling in Mobile Ad Hoc Networks Supporting Heterogeneous Quality of Service Guarantees
Author(s)	Liu, Y; Zhang, L; Li, VOK; Leung, KC; Zhang, W
Citation	The 46th Annual Conference on Information Sciences and Systems (CISS 2012), Princeton, United States, 21-23 March 2012. In Proceedings of the Annual Conference on Information Sciences and Systems, 2012, p. 1-6, article no. 6310759
Issued Date	2012
URL	http://hdl.handle.net/10722/153070
Rights	Proceedings of the Annual Conference on Information Sciences and Systems. Copyright © I E E E.

Topology-Transparent Scheduling in Mobile Ad Hoc Networks Supporting Heterogeneous Quality of Service Guarantees

Yiming Liu*, Lin Zhang*, Victor O. K. Li†, Ka-Cheong Leung†, and Wenzhu Zhang*
*Department of Electronic Engineering
Tsinghua University, Beijing, 100084, China
Email: liu-ym05@mails.tsinghua.edu.cn, {linzhang, zhwz}@tsinghua.edu.cn
†Department of Electrical and Electronic Engineering, University of Hong Kong, Hong Kong, China
Email: {vli, kcleung}@eee.hku.hk

***Abstract*—Transmission scheduling plays a critical role in mobile ad hoc networks. Many transmission scheduling algorithms have been proposed to maximize the spatial reuse and minimize the time-division multiple-access (TDMA) frame length. Most algorithms require information on the network topology and cannot adapt to the dynamic topology in mobile scenarios. To overcome this limitation, topology-transparent scheduling algorithms have been proposed. Most of them, based on Galois field theory, Latin square, and block design theory, assign time slots to users and guarantee that there is at least one collision-free slot in each frame for each user. To the best of our knowledge, none of these topology-transparent algorithms support multiple quality of service (QoS) requirements. In this paper, we exploit the variable-weight optical orthogonal codes (VW-OOC) to design a topology-transparent scheduling algorithm in wireless ad hoc networks with multiple QoS levels. We study the performance, in terms of minimum guaranteed throughput and average throughput, of our proposed algorithm analytically and by extensive simulations.**

I. Introduction

Scheduling medium access in mobile wireless ad hoc networks is challenging because of the node mobility, and the limited and varying wireless bandwidth. In conventional time division multiple access (TDMA) networks, each node is assigned a unique time slot to transmit. However, TDMA does not scale well in mobile ad hoc networks, because the total number of nodes is typically much larger than that in the interference range. Therefore, the system performance can be greatly improved by spatial reuse. Previous approaches in topology-dependent scheduling require each node to maintain accurate network topology information. This is impractical in wireless ad hoc networks and thus such approaches are not adaptive to dynamic topology changes.

To overcome the aforementioned disadvantages of topology-dependent scheduling approaches, topology-transparent scheduling algorithms have been proposed. Chlamtac and Farago [2] developed a topology-transparent algorithm that guarantees at least one collision-free time slot in each frame, but the performance is even worse than the conventional TDMA in some cases. Ju and Li [6] proposed another algorithm to maximize the minimum guaranteed throughput.

As ad hoc networks evolve, multiple applications with different quality of service (QoS) guarantees must be supported. However, as far as we know, none of the existing topology-transparent scheduling methods supports different levels of QoS requirements. A simple but inefficient solution is to modify the conventional TDMA methods by assigning different numbers of unique time slots to nodes with different QoS requirements. However, in the topology-transparent methods, assigning multiple codes or transmission slot allocation functions (TSAFs) [6] to a node is not feasible, since it may introduce much more collisions and fail to support throughput guarantees.

In this paper, we propose to exploit variable-weight optical orthogonal codes (VW-OOCs) to design a novel topology-transparent scheduling algorithm supporting different levels of QoS requirements. Optical orthogonal codes are first introduced in [8] and have been widely used in optical code division multiple access (OCDMA). By adapting VW-OOC to wireless networks, our proposed topology-transparent algorithm guarantees heterogeneous, node-oriented minimum throughput and outperforms the modified conventional TDMA, in terms of both guaranteed throughput and average throughput.

The remainder of this paper is organized as follows. The definition and properties of VW-OOCs are presented in Section II. Due to the code construction difficulty, we propose a greedy heuristic algorithm to construct VW-OOCs in Section III. Then, Section IV introduces our system model and exploits the use of VW-OOCs to design a topology-transparent scheduling algorithm supporting multiple QoS requirements. In Section V, we study the performance of the proposed algorithm by extensive simulations. We summarize and conclude the paper in Section VI.

II. Variable-Weight Optical Orthogonal Codes

Optical orthogonal codes (OOC) were introduced [8] as a means of realizing code division multiple access (CDMA) among asynchronous users on optical fiber networks. A set of OOCs is a family of binary sequences with good auto- and cross- correlation properties. VW-OOC, as a specific type of OOC, relaxes the equivalent weight assumption, and allows the code weight to be different over the codewords. The properties and combinatorial construction methods of VW-OOC were studied in [4], [9].

A set C of VW-OOCs, denoted by (n, W, L, λ_c, Q), is a set of binary n-tuples, where $W = \{w_1, w_2, \cdots, w_p\}$ is the set of codeword weights, $L = \left\{\lambda_a^1, \lambda_a^2, \cdots, \lambda_a^p\right\}$ represents the auto-correlation bound, λ_c denotes the cross-correlation bound, and $Q = \{q_1, q_2, \cdots, q_p\}$ denotes a set of the numbers of codewords with various weights, respectively. Therefore, the total number of codewords is $\Sigma_{i=1}^{p} q_i$.

The following three properties hold for a VW-OOC.

1) Weight Distribution Property: Each n-tuple in C has a Hamming weight contained in the set W and there are q_i codewords of weight w_i in C.
2) Bounded Auto-correlation Property: For any n-tuple $\mathbf{x} = \{x_0, x_1, \cdots, x_{n-1}\} \in C$ with Hamming weight $w_i \in W$ and any integer τ such that $0 < \tau < n$,
$$\sum_{t=0}^{n-1} x_t x_{t \oplus \tau} \leq \lambda_a^i. \tag{1}$$
3) Bounded Cross-correlation Property: For any $\mathbf{x} = \{x_0, x_1, \cdots, x_{n-1}\} \in C$ and any $\mathbf{y} = \{y_0, y_1, \cdots, y_{n-1}\} \in C$ such that $\mathbf{x} \neq \mathbf{y}$ and any integer τ such that $0 \leq \tau < n$,
$$\sum_{t=0}^{n-1} x_t y_{t \oplus \tau} \leq \lambda_c. \tag{2}$$

Note that $\oplus$ is the modulo-n sum.

Since each VW-OOC codeword is an n-bit binary sequence, an alternative way to represent it is by the position of pulses, i.e., the bit '1'. Let $\mathbf{x} = \{x_0, x_1, \cdots, x_{n-1}\} \in C$ be a binary n-tuple of weight w_i. It can also be represented by its *adjacent relative delay vector*, denoted by $\mathbf{t_x} = [t_1, t_2, \cdots, t_{w_i}]$, where t_i corresponds to the relative cyclic delay between the ith pulse and $(i+1)$th pulse for $i = 1, 2, \cdots, w_i$. The cyclic delay corresponds to the relative delay between two pulses in the periodic sequence $\mathbf{x}$.

Another important property of a VW-OOC is its *relative delay vector* R_x, which is a $w_i(w_i - 1)$-dimensional vector to denote a set of the relative cyclic delays between any pair of pulses in a codeword. For a codeword $\mathbf{x} = \{x_0, x_1, \cdots, x_{n-1}\} \in C$ of weight w_i, the ith element of R_x, r_i, is given by:

$$r_i = \sum_{m=1}^{\left\lfloor \frac{i-1}{w_i} \right\rfloor + 1} t_{\phi+1}, \tag{3}$$

where $\phi = t_{[m-1+(i-1) \bmod w_i] \bmod w_i}$.

Two important lemmas that reveal the relations between the properties of VW-OOCs and their relative delay vectors are listed below.

Lemma 1: Let $\mathbf{x} = \{x_0, x_1, \cdots, x_{n-1}\} \in C$ be a binary n-tuple of weight w_i and R_x be its relative delay vector. Then, the inequality
$$\sum_{t=0}^{n-1} x_t x_{t \oplus \tau} \leq \lambda_a^i \tag{4}$$
holds for all $0 < \tau < n$ if and only if no element in R_x appears more than λ_a^i times.

Lemma 2: Let $\mathbf{x} = \{x_0, x_1, \cdots, x_{n-1}\} \in C$ and $\mathbf{y} = \{y_0, y_1, \cdots, y_{n-1}\} \in C$ be two distinct binary n-tuples. Let R_x and R_y be relative delay vectors of $\mathbf{x}$ and $\mathbf{y}$, respectively. Then, for $\lambda_c = 1$, the inequality
$$\sum_{t=0}^{n-1} x_t y_{t \oplus \tau} \leq 1 \tag{5}$$
holds for all $0 \leq \tau < n$ if and only if R_x and R_y are disjoint.

The proof of the two lemmas are straightforward and omitted due to space limitations. Interested readers can refer to [9] for a detailed proof.

III. Construction of Variable-Weight OOCs

Various previous code construction approaches for OOC and VW-OOC are based on combinatorial design theory, and several different design methods of VW-OOC are devised in [4] and [9]. The main disadvantage of the combinatorial-based construction is that we can only design a VW-OOC based on given parameters, but not with an arbitrary setting of parameters. Another method is to incorporate an exhaustive search through all the $\binom{n}{w_i}$ possible codewords to find a code of weight w_i, resulting in an unacceptable time complexity.

Using Lemmas 1 and 2, we propose a greedy algorithm. The input parameters are the weight distribution W and the set of numbers of codes with different weights, Q. Besides, we use the maximum number of iterations performed to constrain the computational time. The output is the length of the codes, n. We use the proposed algorithm to construct $(n, W, \mathbf{1}, 1, Q)$ VW-OOCs. The details of the algorithm is omitted here due to space limitations. Interested readers can refer to [7] for the details.

The key advantage of our proposed heuristic algorithm is the capability of generating VW-OOCs with an arbitrary setting of parameters under an acceptable loss of performance. To express the efficiency of our algorithm, we define the efficiency ratio, μ, as follows:

$$\mu = \frac{n_{opt}}{n} = \frac{\sum_{i=1}^{|W|} w_i q_i + 1}{n}, \tag{6}$$

where $n_{opt} = \sum_{i=1}^{|W|} w_i q_i + 1$ is a lower bound of the code length of $(n, W, \mathbf{1}, 1, Q)$ in [9].

The efficiency ratios of the codes constructed are presented in Fig. 1. It is shown that the efficiency ratio on the constructed codes decreases when the weight of the codes increases. In our application scenario where the weight is typically low, the

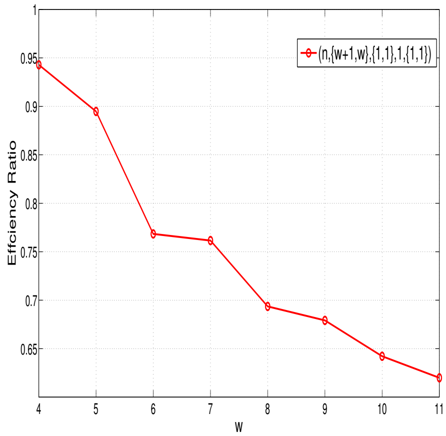


Fig. 1. Efficiency ratio of $(n, \{w+1, w\}, \{1,1\}, 1, \{1,1\})$ against w.

performance of the heuristic code construction algorithm is acceptable. Designing VW-OOCs with high efficiency ratio is an open research problem and left for future work.

IV. Modeling, Algorithm, and Analysis

A. System Model

A mobile ad hoc network can be represented by a graph $G(V, E)$, where V is the set of the network nodes and E is the set of links connecting pairs of mutually interfering nodes. The node set V can be divided into disjoint subsets, V_i, based on different QoS requirements. Without loss of generality, we assume that the network offers two different types of QoS requirements, corresponding to node subsets V_1 and V_2. The QoS requirement for nodes in V_1 is that each node can have at least one conflict-free transmission slot per frame, while for the nodes in V_2, each node is guaranteed to have at least two successful transmission slots per frame. It can be easily generalized to the scenarios with more levels of QoS requirements.

The degree of a node v, $D(v)$, defines the number of nodes in v's interference range. Due to the characteristics of mobile ad hoc networks, the maximum degree $D_{\max}$ is much smaller than the total number of nodes N. $D_{\max}$ is assumed to remain constant while the network topology changes [5].

Assume that time is divided into equal-sized, synchronized transmission slots, which are grouped into frames. Each slot is designed to accommodate the transmission of one equal-sized packet and a guard time, and a frame preamble during which the nodes are synchronized. When nodes communicate, they may suffer two types of conflicts, namely, the primary conflict and the secondary conflict [3]. The primary conflict refers to the situation that a node cannot send nor receive a packet in the same time slot, whereas the secondary conflict refers to the situation that a node cannot receive more than one packet in a time slot. We assume that a reception failure is only due to transmission collision. A transmission from Node w to Node v succeeds when 1) Node v is not transmitting and 2) other nodes in v's interference range are not transmitting.

B. Proposed Algorithm

Using the number of nodes in the network N, the percentage of nodes in V_2, α (where $\alpha = \frac{|V_2|}{N}$), and the maximum node degree D_{max} as the design parameters, we exploit VW-OOCs to design a distributed topology-transparent scheduling algorithm, which can support heterogeneous guarantees for transmissions from different nodes.

Considering the auto-correlation and cross-correlation properties of a set C of $(n, W, \mathbf{1}, 1, Q)$ VW-OOCs, we have the following theorem,

Theorem 1: For a set C of $(n, W, \mathbf{1}, 1, Q)$ VW-OOCs, we can construct a new code set C' of code length n and weight distribution W, which consists of nq_i codewords of weight w_i. For any $\mathbf{x}, \mathbf{y} \in C'$ and $\mathbf{x} \neq \mathbf{y}$, $\langle \mathbf{x}, \mathbf{y} \rangle \leq 1$.

Proof: Shifting each codeword in a set C of $(n, W, \mathbf{1}, 1, Q)$ VW-OOCs by i elements, where $i = 1, 2, \cdots, n-1$, we can obtain a new code set C' of code length n and weight distribution W, which consists of nq_i codewords of weight w_i.

If $\mathbf{x}$ and $\mathbf{y}$ differ simply by shifting elements, the inner product of $\mathbf{x}$ and $\mathbf{y}$ is obviously no more than one due to the auto-correlation property of C. Suppose that codes $\mathbf{x}$ and $\mathbf{y}$ are degenerated from two different codes $\mathbf{X}$ and $\mathbf{Y}$ in C, respectively. Assume that $\langle \mathbf{x}, \mathbf{y} \rangle > 1$, implying that there is at least one common element in the relative delay vectors of $\mathbf{X}$ and $\mathbf{Y}$. By Lemma 2, there is a violation of the fact that the relative delay vectors of $\mathbf{X}$ and $\mathbf{Y}$ are disjoint, since cyclic shifting does not change the relative delay vectors of the original codes. Thus, For any $\mathbf{x}, \mathbf{y} \in C'$ and $\mathbf{x} \neq \mathbf{y}$, $\langle \mathbf{x}, \mathbf{y} \rangle \leq 1$. ■

Thus, we design our proposed topology-transparent scheduling algorithm as follows:

1) Set $w = D_{max} + 1$.
2) Use the VW-OOC construction method proposed in Section III to generate a set of $(n, \{w+1, w\}, \{1,1\}, 1, \{q_2, q_1\})$ codes, which satisfies both $nq_1 \geq (1-\alpha)N$ and $nq_2 \geq \alpha N$.
3) Shift each codeword of weight w in the constructed $(n, \{w+1, w\}, \{1,1\}, 1, \{q_2, q_1\})$ VW-OOCs by i elements, for $i = 1, 2, \cdots, n-1$. We get a new code C_1 with nq_1 codewords, each of which is of weight w. Shift each codeword of weight $w+1$ in the constructed $(n, \{w+1, w\}, \{1,1\}, 1, \{q_2, q_1\})$ VW-OOCs by i elements, for $i = 1, 2, \cdots, n-1$. We get a new code C_2 with nq_2 codewords, each of which is of weight $w+1$.
4) Each node in V_1 is assigned with a unique codeword in C_1 and each node in V_2 is assigned with a unique codeword in C_2.
5) Each node transmits its data packets only at its assigned slots determined by its codeword.

C. Performance Analysis

1) Minimum guaranteed throughput: Denote the minimum guaranteed throughput of a node in V_1 by T_1^g and V_2 by T_2^g, respectively. By employing the aforementioned algorithm, there are at least one and two collision-free slots per frame for the nodes in V_1 and V_2, respectively. Thus,we obtain:

$$T_1^g = \frac{w - D_{\max}}{n}, \tag{7}$$

$$T_2^g = \frac{w + 1 - D_{\max}}{n}, \tag{8}$$

where n is the frame length, i.e., the length of a constructed $(n, \{w+1, w\}, \{1,1\}, 1, \{q_2, q_1\})$ VW-OOC.

2) Average throughput: Denote the average throughput of a node in V_1 and V_2 by T_1^a and T_2^a, respectively.

We assume that the interfering neighbours of a chosen node are chosen at random. Consider a source-destination pair (S, D). Let C_S be the codeword of S and $C_1, \cdots, C_{D_{\max}-1}$ be the codewords of the other $D_{\max} - 1$ interfering nodes of Node D (where the maximum number of the interfering nodes is $D_{\max}$). Let C_D be the codeword of D.

Consider the situation that $S \in V_2$ (i.e., the codeword C_S is of weight $w + 1$). Given a codeword C_S of weight $w+1$, let N_{w+1}^k denote the number of ways to select D_{max} other codewords, where the union of these D_{max} codewords intersects C_S in exactly k specific positions. In this case, there are $w + 1 - k$ collision-free slots per frame. There are $\binom{w+1}{k}$ ways to choose the k intersected elements. Since the frame length is n, the average throughput for a node in V_2 can be expressed as follows:

$$T_2^a \geq \sum_{k=0}^{w-1} (w + 1 - k) \frac{\binom{w+1}{k} N_{w+1}^k}{\binom{n(q_1+q_2)-1}{D_{max}}} \frac{1}{n}. \tag{9}$$

Similarly, consider the situation that $S \in V_1$ (i.e., the codeword C_S is of weight w). Given a codeword C_S of weight w, let N_w^k denote the number of ways to select D_{max} other codewords, where the union of these $D_{\max}$ codewords intersects C_S in exactly k specific positions. Thus, the average throughput for a node in V_1 can be expressed as follows:

$$T_1^a \geq \sum_{k=0}^{w-1} (w - k) \frac{\binom{w}{k} N_w^k}{\binom{n(q_1+q_2)-1}{D_{max}}} \frac{1}{n}. \tag{10}$$

Now, we are going to calculate N_w^k and N_{w+1}^k for the codewords of weights w and $w + 1$, respectively. We have the following theorem.

Theorem 2: Let C' be a code set generated by a set C of $(n, \{w+1, w\}, \{1,1\}, 1, \{q_2, q_1\})$ VW-OOCs. For the codewords of weight φ (where φ is either w or $w+1$),

$$N_\varphi^k = \binom{\Phi_t - (\varphi - k)\Phi}{D_{max}} - \sum_{m=1}^{k} (-1)^{m-1} \binom{k}{m} \binom{\Phi_t - (\varphi - k + m)\Phi}{D_{max}}, \tag{11}$$

and

$$N_\varphi^0 = \binom{\Phi_t - \varphi\Phi}{D_{max}}, \tag{12}$$

where $\Phi = (w+1)q_2 + wq_1 - 1$, $\Phi_t = n(q_1 + q_2) - 1$, and $k = 1, 2, \cdots, D_{\max}$.

Proof: Consider a specific codeword CW of weight φ. We categorize the remaining codewords in the code set C' into $\varphi + 1$ different subsets according to their intersections with CW. There are some properties about CW and C',

1) The cardinality of the code set C' is $n(q_1 + q_2)$.
2) Each of the remaining $\Phi_t = n(q_1 + q_2) - 1$ codewords intersects Codeword CW in exactly zero or one element, since the inner product of any two different codewords is zero or one.
3) The number of codewords which intersect Codeword CW in Element i, where $i = 1, 2, \cdots, \varphi$, is $\Phi = (w + 1)q_2 + wq_1 - 1$. This is because we obtain $w+1$ and w codewords intersecting Codeword CW in Element i by shifting each of the q_2 codewords of weight $w+1$ and q_1 codewords of weight w in C', respectively. Discarding Codeword CW itself, we get $\Phi = (w+1)q_2 + wq_1 - 1$.
4) The number of codewords, which do not intersect Codeword CW, is $\Phi_t - \varphi\Phi$.

Denote the subset of codewords which intersects CW in Element i by SC_i, where $i = 1, 2, \cdots, \varphi$. The cardinality of SC_i is $\Phi = (w + 1)q_2 + wq_1 - 1$. Let SC_0 be the subset of codewords which does not intersect CW in any element. The cardinality of SC_0 is $\Phi_t - \varphi\Phi$. Consider k specific elements of CW, namely, $p_1, p_2, \cdots, p_k$. We classify the codewords other than CW into three different groups:

- *Group 1:* Codewords which intersect CW in one of these k specific elements.
- *Group 2:* Codewords which intersect CW in one of the other $\varphi - k$ elements.
- *Group 3:* Codewords which do not intersect CW.

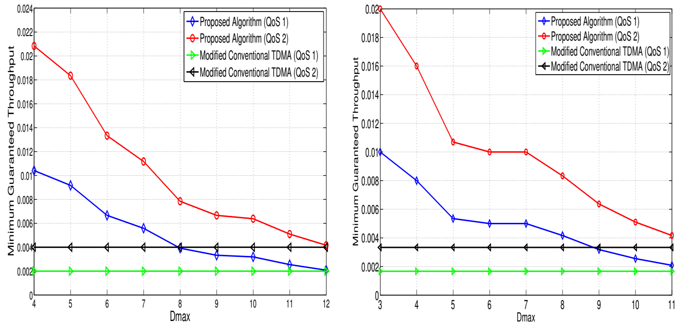
Let $A_{p_i}^\varphi$ (where $i = 1, 2, \cdots, k$) be the set of events that none of the chosen D_{max} codewords from Groups 1 and 3 intersects Codeword CW at Element p_i. Note that the number of codewords intersecting Codeword CW at Element p_i (where $i = 1, 2, \cdots, k$) is Φ and we choose $D_{\max}$ codewords from Groups 1 and 3. Thus, the cardinality of the intersection of any m sets from $A_{p_i}^\varphi$, where $i = 1, 2, \cdots, k$, is $\binom{\Phi_t - (\varphi - k + m)\Phi}{D_{max}}$. N_φ^k is equal to the cardinality of the complementary set of $\bigcup_{i=1}^{k} A_{p_i}^\varphi$, denoted by $\complement$. Note that there are $\binom{\Phi_t - (\varphi - k)\Phi}{D_{max}}$ ways to select D_{max} codewords from Groups 1 and 3. Thus,

$$N_\varphi^k = |\complement| = \binom{\Phi_t - (\varphi - k)\Phi}{D_{max}} - \left| \bigcup_{i=1}^{k} A_{p_i}^\varphi \right|.$$

Applying the Inclusion-Exclusion Principle, we obtain N_φ^k as follows:

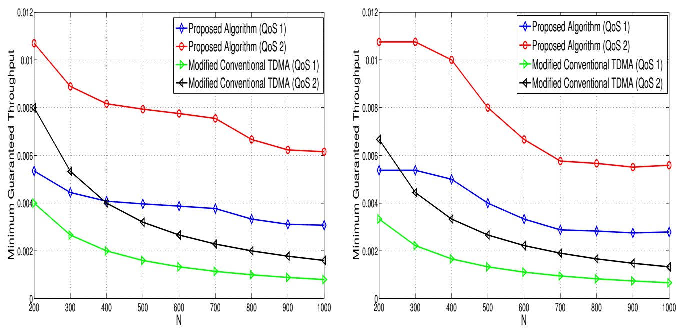
$$N_\varphi^k = \binom{\Phi_t - (\varphi - k)\Phi}{D_{max}} - \sum_{m=1}^{k} (-1)^{m-1} \binom{k}{m} \binom{\Phi_t - (\varphi - k + m)\Phi}{D_{max}}. \tag{13}$$

$N_\varphi^0 = \binom{\Phi_t - \varphi\Phi}{D_{max}}$ follows. ■



(a) $\alpha = 0.25$. (b) $\alpha = 0.5$.

Fig. 2. Effect of $D_{\max}$ on minimum guaranteed throughput.



(a) $\alpha = 0.25$. (b) $\alpha = 0.5$.

Fig. 3. Effect of N on minimum guaranteed throughput.

V. PERFORMANCE EVALUATION

In this section, we study the minimum guaranteed throughput and the average throughput as the performance metrics and quantitatively compare our proposed scheduling algorithm with the modified conventional TDMA fixed assignment scheme. In the modified conventional TDMA, we refer to the scheme that assigns exactly one distinct time slot for nodes with a less stringent QoS requirement (QoS 1), and assigns exactly two distinct time slots for nodes with a more stringent QoS requirement (QoS 2).

A. Simulation Setup

We adopt the Gauss-Markov mobility model [1], which has been shown to be more realistic than the widely used Random Waypoint model. All nodes are initially distributed uniformly in a region of 1000 m $\times$ 1000 m. The tuning parameter β, which is used to present different levels of randomness of node movement, is set to 0.5. Brownian motion is obtained by setting β to zero and the linear motion is obtained by setting β to one. The speed follows a Gaussian distribution. The mean and standard deviation of the speed are set to 0.9 ms^{-1} and 0.5 ms^{-1} [10].

We apply the construction method proposed in Section III to generate VW-OOCs for different parameters. We run each simulation for 500 times.

B. Minimum Guaranteed Throughput

To compare the proposed algorithm with the modified conventional TDMA scheme, we study the effect of $D_{\max}$ and N on the minimum guaranteed throughput in this subsection. Without loss of generality, we investigate the performance with $\alpha = 0.25$ (i.e., one quarter of the N nodes are of more stringent QoS requirement), and $\alpha = 0.5$ (i.e., half of the N nodes are of more stringent QoS requirement).

1. Effect of $D_{\max}$ on Minimum Guaranteed Throughput

Given $N = 400$, we investigate the minimum guaranteed throughput of our proposed algorithm with nine settings of D_{max} from three to 11. A larger D_{max} indicates that the network is denser and there are more possible conflicts. As shown in Fig. 2(a) (with $\alpha = 0.25$) and Fig. 2(b) (with $\alpha = 0.5$), we can see that our algorithm outperforms the modified conventional TDMA, especially when $D_{\max}$ is small. For the case $\alpha = 0.25$, there are 100 nodes requiring more stringent QoS requirement and 300 nodes requiring less stringent QoS requirement. Thus, VW-OOCs with longer code length n are sometimes needed to satisfy $nq_1 \geq 300$ and $nq_2 \geq 100$ so as to have at least 300 codewords for those nodes with less stringent QoS requirement. However, for the case $\alpha = 0.5$, we can select a smaller n to satisfy both $nq_1 \geq 200$ and $nq_2 \geq 200$ simultaneously. That is why the minimum guaranteed throughput for the case $\alpha = 0.5$ is sometimes larger than that for the case $\alpha = 0.25$, that is counterintuitive.

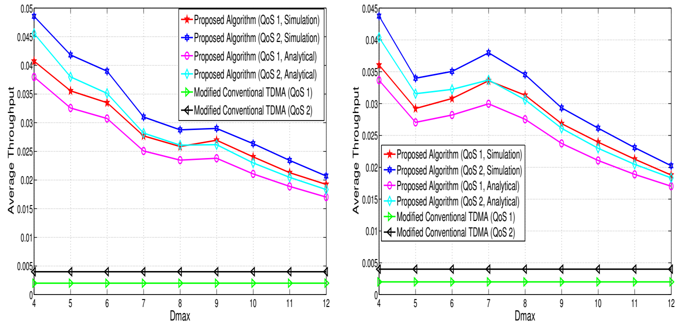
2. Effect of N on Minimum Guaranteed Throughput

Given $D_{\max} = 7$ and varing N from 200 to 1000, we evaluate the minimum guaranteed throughput of our proposed algorithm, which is compared with the modified conventional TDMA. As shown in Fig. 3(a) (with $\alpha = 0.25$) and Fig. 3(b) (with $\alpha = 0.5$), our proposed algorithm performs much better than the modified conventional TDMA. Besides, we can observe that the performance of our proposed algorithm deteriorates slowly with increasing N, implying that the minimum guaranteed throughput of our algorithm is rather insensitive to the number of nodes in the network.

C. Average Throughput

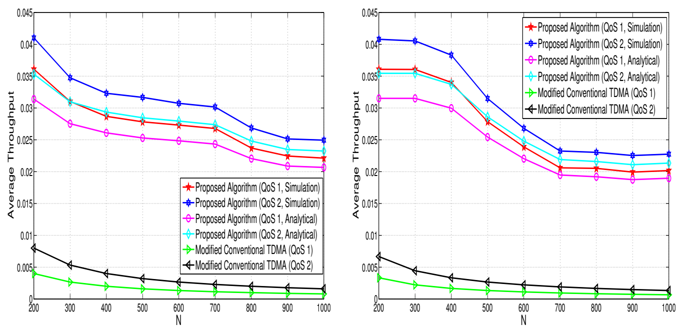
1. Effect of $D_{\max}$ on Average Throughput

Given $N = 400$, we investigate the average throughput of our proposed algorithm with 11 settings of D_{max} from four to 12. In Fig. 4(a) (with $\alpha = 0.25$) and Fig. 4(b) (with $\alpha = 0.5$), both analytical and simulation results show that our proposed algorithm outperforms the modified conventional TDMA. The simulation results match closely with our analytical results. With increasing $D_{\max}$, the average throughput of our proposed algorithm generally decreases. As shown in Fig. 2(b), the length of the codes increases very slowly when $D_{\max}$ is between five and seven (i.e., the minimum guaranteed throughput drops very slowly when $D_{\max}$ is between five and seven), whereas the number of successful transmission slots increases much faster than the length of codes. That is why the average throughput increases when $5 \leq D_{\max} \leq 7$ as shown in Fig. 4(b). Besides, we notice that the difference between



(a) $\alpha = 0.25$.   (b) $\alpha = 0.5$.

Fig. 4. Effect of $D_{\max}$ on average throughput.



(a) $\alpha = 0.25$.   (b) $\alpha = 0.5$.

Fig. 5. Effect of N on average throughput.

the average throughput of a node with QoS 2 and that with QoS 1 drops with increasing $D_{\max}$.

2. Effect of N on Average Throughput

Given $D_{\max} = 7$ and N is configured with nine different settings from 200 to 1000, we evaluate the average throughput of our proposed algorithm, and compare it with the modified conventional TDMA. In Fig. 5, both analytical and simulation results show that our proposed algorithm performs much better than the modified conventional TDMA. Besides, we can observe that the average throughput of our proposed algorithm deteriorates slowly with increasing N, implying that the average throughput of our algorithm is rather insensitive to the number of nodes in the network.

As shown in Figs. 4-5, the average throughput obtained by simulation is slightly higher than that obtained analytically. This is because the number of neighbours for any node is no more than $D_{\max}$. Hence, the average throughput is lower-bounded by its analytical results.

VI. CONCLUSION

In this paper, we propose a method to construct variable-weight optical orthogonal codes and exploit it to design a novel topology-transparent scheduling algorithm supporting heterogenous quality of service (QoS) guarantees. Though we only study two different QoS levels in this paper, our proposed algorithm can be easily generalized to support multiple levels of QoS requirements. To the best of our knowledge, this is the first topology-transparent scheduling algorithm supporting heterogeneous QoS requirements. We show that our proposed algorithm outperforms the modified conventional TDMA scheme, in terms of the minimum guaranteed throughput and average throughput, both analytically and by extensive simulations.

ACKNOWLEDGEMENT

This research is supported in part by the Research Grants Council of Hong Kong, under Grant No. HKU 714310E.

REFERENCES

[1] J. Ariyakhajorn, P. Wannawilai, and C. Sathitwiriyawong, *"A Comparative Study of Random Waypoint and Gauss-Markov Mobility Models in the Performance Evaluation of MANET,"* in Proc. IEEE ISCIT'06, pp. 894-899, Sep. 2006.

[2] I. Chalamatac and A. Farago, *"Making Transmission Schedules Immune to Topology Changes in Multi-hop Packet Radio Networks,"* IEEE/ACM Trans. on Networking, Vol. 2, No. 2, pp. 23-29, Feb. 1994.

[3] A. Ephremides and T. V. Truong, *"Scheduling Broadcast in Multihop Radio Networks,"* IEEE Trans. on Comm., Vol. 38, No. 4, pp. 456-460, Apr. 1990.

[4] F.-R. Gu and J. Wu, *"Construction and Performance Analysis of Variable Weight Optical Orthogonal Codes for Asynchronous Optical CDMA Systems,"* J. Lightwave Technol., Vol. 23, No. 2, pp. 740-748, Feb. 2005.

[5] T. Hou and V. O. K. Li, *"Transmission Range Control in Multihop Packet Radio Networks,"* IEEE Trans. on Comm., Vol. 34, No. 1, pp. 38-44, Jan. 1986.

[6] J. H. Ju and V. O. K. Li, *"An Optimal Topology-Transparent Scheduling Method in Multhop Packet Radio Networks,"* IEEE/ACM Trans. on Networking, Vol. 6, No. 3, pp. 298-306, Jun. 1998.

[7] Y. Liu, L. Zhang, V. O. K. Li, and K.-C. Leung, *"Topology-Transparent Scheduling in Mobile Ad Hoc Networks Supporting Heterogeneous Quality of Service Guarantees,"* Technical Report, TR-2012-001, Department of Electrical and Electronic Engineering, The University of Hong Kong, Feb. 2012.

[8] J. A. Salehi, *"Code Division Multiple-Access Techniques in Optical Fiber Networks—Part I: Fundamental Principles,"* IEEE Trans. on Comm., Vol. 37, No. 8, pp. 824-833, Aug. 1989.

[9] G.-C. Yang, *"Variable-Weight Optical Orthogonal Codes for CDMA Networks with Multiple Performance Requirements,"* IEEE Trans. on Comm., Vol. 44, No. 1, pp. 47-55, Jan. 1996.

[10] E. Zola and F. Barcelo-Arroyo, *"Impact of Mobility Models on the Cell Residence Time in WLAN Networks,"* in Proc. IEEE SARNOFF'09, pp. 1-5, April, 2009.