Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2013, Article ID 982713, 9 pages
http://dx.doi.org/10.1155/2013/982713

Research Article

Hypergraph Modeling and Approximation Algorithms for the Minimum Length Link Scheduling in Multiuser MIMO Networks

Hu Shen, Shaohe Lv, Xuan Dong, Junquan Deng, Xiaodong Wang, and Xingming Zhou

Science and Technology on Parallel and Distributed Processing Laboratory, College of Computer, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Hu Shen; shenhu@nudt.edu.cn

Received 25 April 2013; Revised 16 September 2013; Accepted 2 October 2013

Academic Editor: Ching-Jong Liao

This paper investigates the problem of the minimum length link scheduling (MLLS) in multiuser MIMO (MU-MIMO) networks. Generally, in the networks with MU-MIMO capability, the number of concurrent transmissions can be as large as that of antenna elements at the receiver. As a result, link interference is no longer binary but demonstrates a strong correlation among multiple links, which cannot be captured by the conventional conflict graph interference model. Hence, we propose a novel hypergraph interference model, which can accurately and efficiently characterize the relationship of multiple interferences induced by concurrent transmissions, and provide a tractable formalization of the minimum length link scheduling in MU-MIMO networks (MU-MIMO MLLS). Afterwards, we prove that the MU-MIMO MLLS problem is NP-hard and introduce two approximation algorithms to find the near-optimal feasible schedule. Finally, extensive simulation experiments are presented.

1. Introduction

Link scheduling in wireless networks is one of the key and classical research topics for the communication and network communities. In wireless networks, nodes communicate with each other through a shared common channel. On one hand, to shorten the communication latency, node pairs are required to communicate concurrently as much as possible; on the other hand, concurrent transmissions induce the cochannel interference, which results in severe deterioration or even interruption of an ongoing transmission. To control the cochannel interference, link scheduling is required to coordinate the transmission of different links. A good schedule not only avoids communication failures by silencing the interferers of every receive node in each time slot but also minimizes the number of time slots and hence the communication latency. To sum up, link scheduling in wireless networks usually requires the minimum length conflict-free assignment of time slots in which each communication link is activated at least once [1, 2] and the link scheduling problem is difficult because many subsets of nonconflicting nodes are candidates for each time slot, and the subset selected for transmission in one slot affects the number of concurrent transmissions in the next time slot.

The interference model, which characterizes the interference relationship of communication links, has a major impact on the complexity of wireless link scheduling. The conflictgraph based interference model, as a simple and powerful modeling tool, has been widely employed [3, 4]. In [3], Hajek and Sasaki consider the MLLS problem and propose an optimal scheduling scheme with polynomial time complexity. However, they consider only the primary interference; that is, two links are interfered with each other when they are neighboring. In practice, the transmission of a link can interrupt that of the other even when they are not neighboring. At this time, link scheduling is shown as an NP-hard problem. Just recently, constant factor approximation bounds have emerged for link scheduling under the conflictgraph interference model with k-hop interference [4].

The conventional conflict graph interference model cannot capture the interference when multiple-input multiple-output (MIMO) is deployed. As a revolutionary technology from an information theoretic perspective and physical layer

communications, MIMO has been introduced as a de facto component of wireless standards [5, 6]. In the networks with multiuser MIMO capability (MU-MIMO networks), a node with multiple antenna elements can decode a composite signal from different transmitters. Therefore, more communication links are permitted to transmit concurrently. In general, the number of concurrent transmissions can be as large as that of antenna elements at the receiver. For example, given a network consisting of a receiver R_1 with two antenna elements, and three senders S_1, S_2, and S_3 with single antenna, respectively, consider the interference of S_1 on S_2: when S_3 is silent, there is no interference; however, when S_3 is active, S_1 and S_2 cannot transmit simultaneously. That is, the interference relationship between S_1 and S_2 has strong correlation with the activity of S_3. Such dependence has not been characterized in the binary conflict graph model.

Recently there is a few existing works in the literature trying to explore the optimal link scheduling problem in MU-MIMO networks. Chu and Wang [7] presented an integer linear programming (ILP) formulation of the optimal link scheduling problem in MU-MIMO networks, and proposed an opportunistic scheduling algorithm that can adaptively select different transmission strategies. Blough et al. [8] considered the same problem in MU-MIMO networks with interference suppression capabilities, and formulized the one-shot optimal link scheduling problem as an integer linear programming. However, all of them cannot be applied directly to the MLLS with MU-MIMO capability: on one hand, the goal of [7, 8] is maximizing throughput while the objective of MU-MIMO MLLS is minimizing the schedule length; on the other hand, both work in [7, 8] model the optimal link scheduling problem in MU-MIMO networks as an ILP formulation, which is well-defined in mathematics while difficult to find approximation algorithms providing a guaranteed performance.

In this paper, we address the issue of the minimum length link scheduling in MU-MIMO networks, aiming for providing at least one transmission time slot for every communication link with a minimum schedule length. Firstly, we propose a novel interference model based on hypergraph to characterize multiple interferences in MU-MIMO networks. The concept of hypergraph is extended from graph, and is a nature way for representing multiple interferences among multiple links. Afterwards, two algorithms for the problem of MU-MIMO MLLS are proposed: the first one is a performance guaranteed approximation algorithm whose approximation ratio is at most $2n/\log_2(n)$; the second one is a time efficient heuristic algorithm based on degrees of interference (DOI) greedy. Extensive simulations are presented to test how well the proposed algorithms based hypergraph model may be applied in practice.

The remainder of this paper is organized as follows. Section 2 presents the integer programming formalization and the proposed hypergraph interference model of MU-MIMO MLLS. Section 3 introduces two algorithms to approximate the optimal solution. Section 4 shows the extensive simulation results. Finally, in Section 5, conclusions are given.

2. System Model

2.1. Problem Statement. Consider a wireless network G of n communication links, and N stationary nodes, each of which is equipped with M antenna elements. A link with transmit node i and receive node j is denoted by l_{ij}. We assume that: (i) nodes work in the half-duplex mode, that is a node can either transmit or receive, but not at the same time; (ii) every link is of a single data stream as to enhance as many concurrent transmissions as possible.

The use of a MIMO antenna array is typically modeled as degree of freedom (DOF) model [9]. For simplicity, a receive node with M antenna elements has up to M DOFs, which can be used for multiplexing A desired data streams and suppressing $(M\text{-}A)$ interferences from other neighboring concurrent transmission links. And if and only if the number of the concurrent transmission link in vicinity of receive node j, including link l_{ij}, does not exceed M, then receive node j can successfully decode the data stream carried by link l_{ij}.

2.2. Integer Programming Formulization. We formulize the problem of MU-MIMO MLLS as an integer programming (IP) by referencing [7, 8].

Firstly, notations used in the IP formulation are specified as follows:

$T = \{t_1, t_2, \ldots, t_m\}$: set of transmission time slots for link scheduling;

$L = \{l_1, l_2, \ldots, l_n\}$: set of communication links to be scheduled;

$V = \{v_1, v_2, \ldots, v_N\}$: set of wireless nodes;

$L(v)$: set of links incident into node v;

t_i, r_i: transmit node and receive node of link l_i, respectively;

$IL(i)$: set of links interfered by t_i;

x_i^t: an integer variable, equals to 1 iff link l_i is active at the tth time slot, otherwise equals 0.

The proposed IP formulation for MU-MIMO MLLS is as follows

$$\text{minimize} \quad m \tag{1}$$

$$\text{subject to:} \quad \sum_{t=1}^{m} x_i^t \geq 1 \quad i = 1, \ldots, n \tag{2}$$

$$\sum_{j \in L(r_i)} x_j^t \leq M + M_i \cdot \left(1 - x_i^t\right) \quad i = 1, \ldots, n; \; t = 1, \ldots, m \tag{3}$$

$$x_i^t \in \{0, 1\} \quad i = 1, \ldots, n; \; t = 1, \ldots, m. \tag{4}$$

The objective function (1) states that we aim to minimize the length of scheduling time slots. Constrain (2) ensures that each of communication links is assigned at least one transmission time slot in a schedule period. The next constraint (3)

provides that every active communication link does not suffer too much cochannel interference. And to make constrain (3) always feasible when $x_i^t = 0$, the constant parameter M_i needs to be set large enough. A sufficiently large value is $M_i = n - M$. From constraint (4), we know that the size of search space is $2^{O(n)}$, rising at an exponential rate with the network size n.

The disadvantage of the IP formulation is that it is difficult to find approximation algorithms, but the IP formulation provides a way to calculate the optimal solution through exhaustive searching in cases of networks with a small size.

2.3. Hypergraph Modeling MU-MIMO MLLS. Typically, a wireless network can be modeled as a *directional graph* $G = (V, E)$. The set V includes all the nodes in the network, and E is a link set, of which for a pair of nodes $u, v \in V$, a directional edge $e = (u, v)$ is a member of E if node u's signals can be decoded successfully at node v in the absence of interference. Further, the interference relationship between transmission links is modeled as a *conflict graph* $G_c = (V_c, E_c)$, where V_c is the set of links (in fact, V_c is the set E in the graph G) and E_c is the set of conflicts; that is, $(i, j) \in E_c$ means that the concurrent transmission of link l_i and l_j is invalid. However, the conflictgraph interference model is restrictive and cannot descript the interference relationship among three links or more. For example, even when $(i, j) \notin E_c$, $(i, k) \notin E_c$, and $(j, k) \notin E_c$ are given, we cannot conclude whether the concurrent transmission of link l_i, l_j, and l_k is valid or not. While generally the interference relationship in MU-MIMO networks involves multiple communication links. Thus, the conflict graph interference model is not appropriate to handle the case of MU-MIMO networks.

2.3.1. Hypergraph Model

Construct. Here, we present a new graph model, *hypergraph* [10, 11], to characterize the interference relationship in MU-MIMO networks. Let $G_H = (V_H, E_H)$ denote a hypergraph, where V_H is the set of links, the same as V_c and E, and $E_H = \{S_1, S_2, \ldots, S_q\}$ is also the set of conflicts, similar to E_c. The difference between E_H and E_c is that the element of E_c is limited to 2-tuple of E, while the element of E_H, named as hyperedge, is extended to a subset of E. And for any link set $S \subseteq V_H$, $S \in E_H$ if and only if satisfies three rules as follows.

R1: (Invalidity Rule). When all the links of S are assigned in a same transmission time slot, there is at least one link in S failing to transmit data successfully.

R2: (Minimality Rule). If any link in S is removed, no failure will occur if only the remaining links in S are scheduled.

R3: (Integrality Rule). All link set S, satisfying conditions R1 and R2, must be contained in E_H.

2.3.2. Hypergraph Model

Problem Formalization. Given a hypergraph model $G_H = (V_H, E_H)$, the problem of MU-MIMO MLLS can be transformed to the problem of finding the optimal coloring of the hypergraph G_H: give all node in V_H a color, and for every hyperedge $S \in E_H$, having at least two nodes as endpoints, the nodes of this hyperedge that connect S are not all of the same color. Denote the hypergraph coloring solution by $C = \{C_1, C_2, \ldots, C_m\}$, where C_i is the link set with the same color i. Then, the coloring solution with the least number of colors is the optimal solution to the problem of MU-MIMO MLLS. And the problem of MU-MIMO MLLS could be formulized as follows:

$$\begin{aligned} &\text{minimize} \quad m, \text{where } C = C_1 \cup C_2 \cup \cdots \cup C_m \\ &\text{subject to:} \quad \text{colors}(S) \geq 2 \quad \forall S \in E_H, \end{aligned} \tag{5}$$

where colors(S) means the number of colors appeared in the hyperedge S.

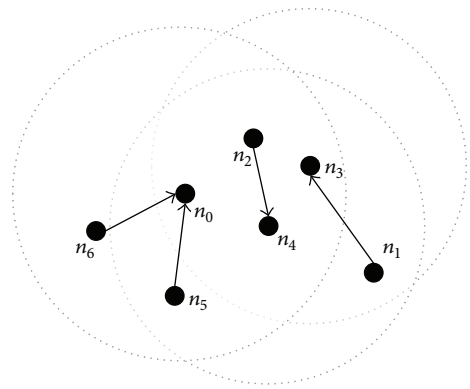


Figure 1: A Multi-user MIMO Network.

Example 1. Consider the network in Figure 1, where there are four communication links and seven nodes, all of which are equipped with two antenna elements. According to the proposed hypergraph model, we can model the network as hypergraph $G_H = (V_H, E_H)$ showed in Figure 2, in which $V_H = \{l_{13}, l_{24}, l_{50}, l_{60}\}$ and $E_H = \{\{l_{13}, l_{24}, l_{50}\}, \{l_{24}, l_{50}, l_{60}\}\}$. And we can easily find a coloring solution $C = \{\{l_{13}, l_{24}\}, \{l_{50}, l_{60}\}\}$; thus, all the four communication links are assigned in two transmission time slots.

2.3.3. Hypergraph Model

Analysis

Lemma 2. *Given a MU-MIMO hypergraph $G_H = (V_H, E_H)$, then, for any $S \in E_H$, the cardinality of S is $M + 1$; that is G_H is $M + 1$-uniform hypergraph.*

Proof. We adopt the counterexample method: (1) if there is any hyperedge $S \in E_H$, satisfying card(S) $\leq M$ (card(S) denotes the cardinality of S), then, according to the receive condition in the DOF model, all links in S could concurrently transmit together; that is, the link set S is a feasible link set, which evidently conflicts with the Invalidity Rule R1; (2) if there is any hyperedge $S \in E_H$, satisfying card(S) $\geq M + 2$, then after removing one link l in S, card($S \setminus \{l\}$) equals $M + 1$ and $S \setminus \{l\}$ is still an infeasible link set, which directly

Procedure B(G_H, k): if $G_H = (V_H, E_H)$ is k-colorable, return a independent link set of size $h_k(|V_H|)$; otherwise, an arbitrary subset of size $h_k(|V_H|)$.
Input: An integer $k \geq 2$ and a hypergraph $G_H = (V_H, E_H)$.
Output: A subset U of V_H with the size $|U| = h_k(|V_H|)$.

```
n = |V_H|; h = h_k(n);
l = ⌊n/hk⌋;
Partition V_H into sets V_1, ..., V_l where |V| = ··· = |V_{l-1}| = hk and hk ≤ |V_l| < 2hk;
for i = 1 to l
    for each subset U of V_i of size |U| = h
        if U is independent in G_H
            return (U);
        endif
    endfor
endfor
return an arbitrary subset of V_H of size h;
```

Procedure 1

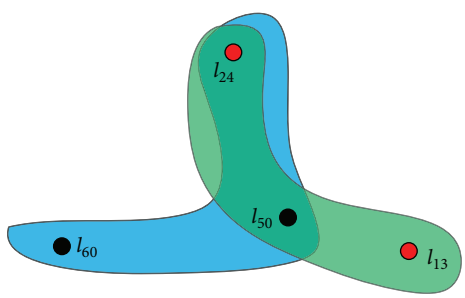


Figure 2: Coloring on Hypergraph.

conflicts with the Minimality Rule R2. Considering (1) and (2) together, we can conclude that, for any $S \in E_H$, card$(S) = M + 1$. □

Lemma 3. *The MU-MIMO MLLS problem is NP-hard.*

Proof. We reduce the problem of finding the chromatic number of a $M + 1$-uniform hypergraph to the MU-MIMO MLLS problem. The chromatic number of a hypergraph G_H is the smallest number k such that G_H is k-colorable and G_H is k-colorable if its vertices can be colored using k different colors in such a way that for every hyperedge in G_H, the nodes of this hyperedge are not all of the same color. According to the hypergraph modeling above mentioned and Lemma 2, we know that every link scheduling solution corresponds to a coloring solution of hypergraph G_H. Meanwhile, due to the three construction rules, every hypergraph coloring is also a link scheduling solution. Thus, the minimum scheduling length is equal to the chromatic number of the $M+1$-uniform hypergraph G_H. And several results [12, 13] show that it is NP-complete to optimally color r-uniform hypergraphs, for various values of r ($r \geq 3$). Specific statements can be found in the corresponding papers. □

3. Algorithms

In this section, two algorithms based on the uniform hypergraph coloring for the problem of MU-MIMO MLLS are proposed. Here, we firstly give some definitions: given a r-uniform hypergraph $G_H = (V_H, E_H)$, a link set $U \subseteq V_H$ is called *independent* if U does not contain any hyperedge in E_H. An m-coloring of G_H is a mapping C: $V_H \rightarrow \{C_1, \ldots, C_m\}$ such that no hyperedge of G_H with one single color. Equivalently, an m-coloring of G_H is a partition of the vertex set V_H into m independent sets. The *chromatic number* of G_H, denoted by $\chi(G_H)$, is the minimal m, for which G_H admits an m-coloring.

3.1. Performance-Guaranteed (P-G) Approximation Algorithm. Similar to Wigderson's paper [14], we find an independent set with the following algorithm, named as performance-guaranteed (P-G) algorithm, whose approximation ratio is $2n/\log_2(n)$. It is worth noting that the idea of partitioning the vertex set of a k-colorable hypergraph G_H on n into groups of size $k\log_k n$ and performing an exhaustive search for an independent set of size $\log_k n$ in each group is due to Berger and Rompel's paper [15].

Define

$$h_k(n) = \log_k n = \frac{\log n}{\log k}. \tag{6}$$

Lemma 4. *For a k-colorable hypergraph G_H with n vertices, Procedure 1 outputs an independent set of size $h_k(n)$, in time polynomial in n (see, Procedure 1).*

Proof. If G_H is k-colorable, it contains an independent set I of size $|I| \geq n/k$. Then, for some $1 \leq i \leq k$, we have $|I \cap V_i| \geq |I|/l \geq n/kl \geq h_k(n)$. Checking all subsets of V_i of size $h_k(n)$ will reveal an independent set of size $h_k(n)$. The number of subsets of size $h_k(n)$ to be checked by the algorithm does not exceed $l\binom{2h_k(n)k}{h_k(n)} = (O(1)k)^{h_k(n)} = n^{O(1)}$. □

The idea of our proposed Algorithm P-G is simple: as long as there are some uncolored vertices (their union is denoted by W), call Procedure 1 for finding an independent set U in the subhypergraph spanned by W, denoted by G_H^W, give the output U a fresh color, and update W.

Algorithm P-G($\mathbf{G_H}$): Providing a coloring solution for hypergraph $G_H = (V_H, E_H)$.
Input: A r-uniform hypergraph $G_H = (V_H, E_H)$.
Output: A coloring solution of G_H.
(1) $k = 2$;
(2) $i = 1$; $W = V_H$;
(3) **while** $W \neq \emptyset$
(4) $U = B(G_H^W, k)$;
(5) **if** U is not independent
(6) $k = k + 1$;
(7) **else**
(8) $C_i = U$; % color U by color i %
(9) $i = i + 1$; $k = 2$;
(10) $W = W \setminus U$;
(11) **endif**
(12) **endwhile**;
(13) **return** a coloring $C = \{C_1, C_2, \ldots, C_{i-1}\}$;

Algorithm 1

Lemma 5. *Algorithm P-G outputs a coloring solution in time polynomial in n (see, Algorithm 1).*

Proof. Inevitably, any hypergraph $G_H^W = (V_H^W, E_H^W)$ is $|V_H^W|$-colorable; thus, Procedure 1 is called at most $\sum_{|V_H^W|=n}^{1} \sum_{k=2}^{|V_H^W|} 1 = n(n-2)/2 = O(n^2)$ times, and according to Lemma 4, the time complexity of Procedure 1 is $n^{O(1)}$; hence, the time complexity of Algorithm *P-G* is polynomial in n. □

Lemma 6. *The approximation ratio of Algorithm P-G is at most $2n/\log_2(n)$ (see, Algorithm 1).*

Proof. For a hypergraph $G_H = (V_H, E_H)$, we denote the chromatic number of G_H by χ; that is, G_H is χ-colorable. Furthermore, the optimal coloring of G_H is the solution of coloring G_H by χ colors. In addition, for any subhypergraph of G_H, which is spanned by the subset W and denoted by $G_H^W = (V_H^W, E_H^W)$ as above mentioned, due to $V_H^W \subseteq V_H$, it is obvious that G_H^W is also χ-colorable. Through Procedure 1, the size of the independent set U picked by Algorithm *P-G* is at least $h_\chi(|G_H^W|)$. And we know that $h_\chi(n)$ is a positive, nondecreasing function. Then, Algorithm *P-G* produces a coloring with at most $f_\chi(n) = \sum_{i=1}^{n}(1/h_\chi(i))$ colors, since each link in V_H contributes at most $1/h_\chi(n')$ colors, where n' is the number of links remaining in the subhypergraph G_H^W at the time when the link was assigned a color. It is proofed that when $h_\chi(n)$ grows no faster than $n^{1/t}$ $(t > 1)$, $f_\chi(n)$ is at most $t/(t-1) \cdot n/h_\chi(n)$ [16]. Specifically in our proposed Algorithm *P-G*, $h_\chi(n) = \log n/\log \chi$ grows no faster than $n^{3/4}$. Thereby, Algorithm *P-G* colors the χ-colorable hypergraph G_H on n vertices in at most $4n \cdot \log \chi/\log n$, then the approximation ratio of Algorithm *P-G* is at most $g(\chi) = 4n \cdot \log \chi/(\chi \cdot \log n)$. Finally, $g(\chi)$ $(\chi \geq 2)$ is decreasing function that is $\max(g(\chi)) = 2n/\log_2 n$. □

3.2. DOI-Based Greedy (D-G) Heuristic Algorithm. Although Algorithm *P-G* can provide a guaranteed performance, its time complexity is $n^{O(1)}$. The distinguishing time complexity promotes us to design another time efficient heuristic coloring algorithm.

Recall that the goal of link scheduling is to keep the cochannel interference at a proper level. Here, the concept of *degree of interference*, denoted by DOI (l), is defined to measure the interferences induced by a link l and presents the impact of the link l brought to the other links. DOI (l) is equal to the amount of communication links affected by the link l when link l is in activation. The main idea of Algorithm *D-G* is to assign the link with maximum degrees-of-interference as early as possible; hence, the interferences in the whole network would reduce drastically and then a relative larger independent link set might be picked in next scheduling time slots. To improve the performance, we refresh the DOI information every time after picking an independent link set.

Lemma 7. *The time complexity of Algorithm D-G is $O(n^2 \log n)$ (see, Algorithm 2).*

Proof. The bottleneck of Algorithm *D-G* is the procedure of QuickSort (G_H^W), which is called at most n times, and the expected running time of QuickSort (G_H^W) is $O(n \log n)$. □

4. Simulation and Discussion

We evaluate the performance of our proposed Algorithms *P-G* and *D-G* via extensive simulations. The network settings include wireless nodes that are distributed in a square area of 1250 m × 1250 m and form an ad hoc network with random or grid topology; each node is equipped with M antennas and has a communication range of 250 m and an interference range of 400 m (Figure 3). A simulation result is obtained by averaging over 50 runs of simulations, and all simulations run on a 2.8 GHz Intel Core Duo machine with 4 GB of RAM.

Algorithm D-G($\mathbf{G_H}$): Providing a coloring solution for hypergraph $G_H = (V_H, E_H)$
Input: An r-uniform hypergraph $G_H = (V_H, E_H)$.
Output: A coloring of G_H.

```
i = 1; W = V_H;
while W ≠ ∅
    QuickSort (G_H^W);  % sort G_H^W by the DOI item in a decreasing order %
    U = ∅
    while (j ≤ |W|)
      if U ∪ {W(j)} is an independent set
          U = U ∪ {W(j)}; % W(j) is the jth element of W %
      else
          break;
      endif
    endwhile
    C_i = U;      % color U by color i %
    i = i + 1;
    W = W \ U;
endwhile;
return a coloring C = {C_1, C_2, ..., C_{i-1}};
```

Algorithm 2

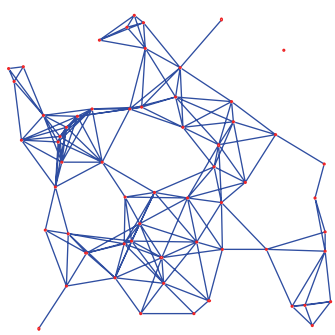


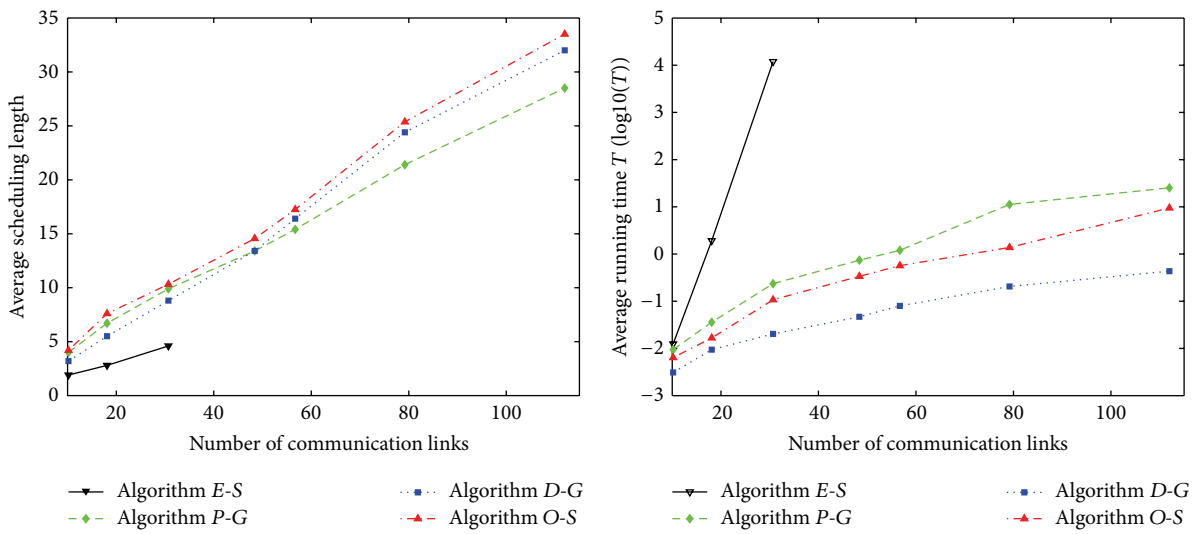
Figure 3: A network with random topology, 60 nodes and 250 communication links.

Four algorithms are implemented and compared in our simulations: (1) the algorithm of exhaustive searching (E-A) to compute the optimal solution of MU-MIMO MLLS; (2) the centralized algorithm of opportunistic scheduling (O-S) modified from [7]; (3) the approximation algorithm of performance guaranteed (P-G) illustrated in Section 3.1; and (4) the heuristic algorithm of DOI-based greedy (D-G) in Section 3.2. Algorithm E-A and Algorithm O-S use the IP formulation presented in Section 2.2, while Algorithm P-G and Algorithm D-G are based on the proposed hypergraph model in Section 2.3. And the later three algorithms only produce suboptimal solutions. To be noted here, our simulation experiments were only partially successful. In certain cases, where the networks are of a small size (refer to the number of links), Algorithm E-S converged quickly to the optimal solution. While in other cases, convergence was much slower and in some cases memory usage rose dramatically preventing the optimal solution from being found. In addition, slow convergence also occurred in the cases with a large network size when executing the approximation or heuristic algorithms. Hence, we set the network size to network cases with a small or medium size.

Here, two performance metrics are considered for the evaluation of the scheduling algorithms: (a) *average scheduling length* (ASL) is defined as the average number of time slots required for assigning all the communication links at least once in a schedule period; (b) *average running time* (ART) is defined as the average time spendt in executing the simulation one time. We investigate three impact factors, namely, network size, network topology, and degree of freedom. For each factor, both the average scheduling length and the average running time are compared. If not otherwise specified, wireless networks are with random topology, and the node's degree of freedom are 2, that is, each wireless node is equipped with two antenna elements.

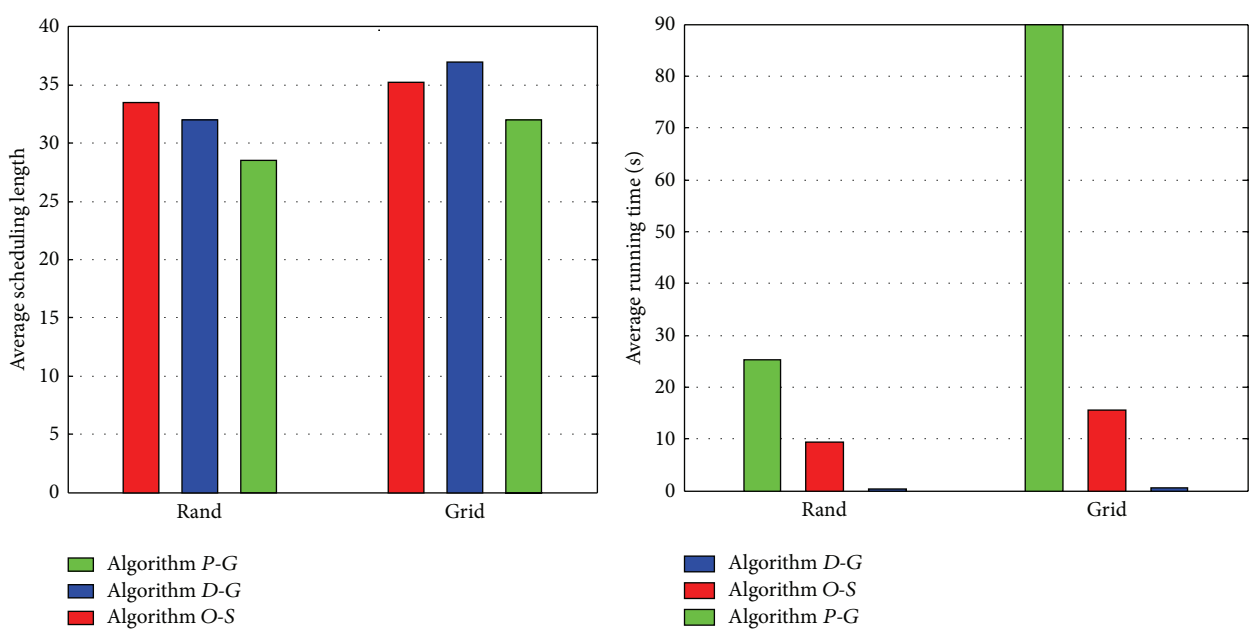
Next, we discuss the results from our simulation experiments.

4.1. Impact of Network Size. In this section, we represent the network size as the number of communication links. The impact of network size is shown in Figures 4(a) and 4(b). As above mentioned, the calculation of the exact optimal solution for the large or even medium scale network is infeasible with our machine, while the proposed approximation Algorithm P-G and the heuristic Algorithm D-G can scale to quite large number of communication links and produce suboptimal solutions that are not too bad. Irrespective of the network size, our proposed two algorithms perform better than the reference algorithm (Algorithm O-S) in terms of average scheduling length, and the performance gain is between 7% and 15%. Besides, Algorithm P-G has the closest performance to that of the optimal solution when the network is of medium or large number of communication links (the ASL performance curves of Algorithm P-G and D-G intersect when the number of links is around 50). The main reason for the performance improvement is that in every schedule time slot, the concurrent link set selected by Algorithm P-G is guaranteed to be of a certain size, and the size increases along



(a) Average scheduling length (b) Average running time

FIGURE 4: Impact of network size.



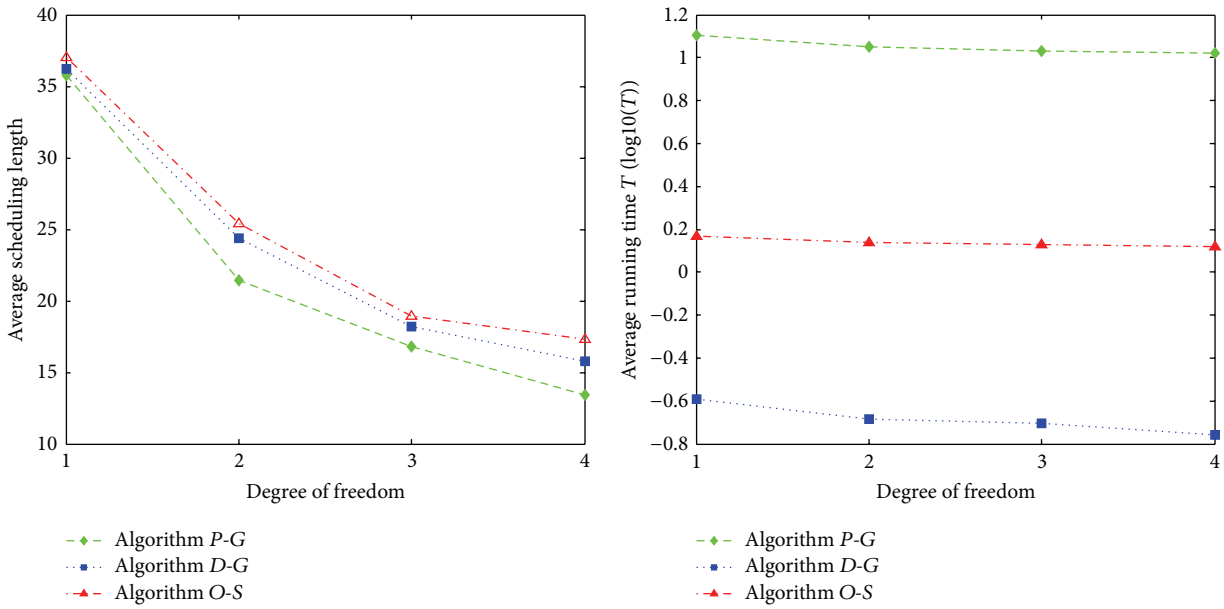
(a) Average scheduling length (b) Average running time

FIGURE 5: Impact of network topology.

with the network size. Although Algorithm *P-G* has ASL performance advantage, the performance difference between Algorithm *P-G* and Algorithm *D-G* is less than 10% when the network size is controlled below 115 communication links.

The comparison of the ART performance of three algorithms is shown in Figure 4(b). To represent the performance difference more effectively, we adopt the denary logarithm value of average running times. The results verify that the running time of Algorithm *E-S* is rising at an exponential rate with the network size; thus, Algorithm *E-S* can be only executed in the case of very small networks; Algorithm *P-G*, Algorithm *D-G*, and Algorithm *O-S* are of polynomial time complexity, while Algorithm *D-G* is of orders-of-magnitudes advantage over Algorithm *P-G* and Algorithm *O-S* in terms of ART performance.

4.2. Impact of Network Topology. We perform extensive simulations with two types of network topology: a randomly generated topology and a grid. In both two networks, the number of communication links to be scheduled is set to the same value, $n = 112$. And the compared algorithms



(a) Average scheduling length (b) Average running time

Figure 6: Impact of degree of freedom.

are Algorithm *P-G*, Algorithm *D-G*, and Algorithm *O-S*. From Figure 5, we see that the efficiency of all the three algorithms is higher with random topology: with random topology, fewer transmission time slots and little running time are required. The reason is that in a grid topology, each wireless node is equidistant from its neighbors; hence, the distribution of communication links is more homogeneous and the signals of neighbors arrive at a receiver node with similar powers; this limits the choices for the scheduling policies, especially for Algorithm *D-G*. Besides, we find that in both networks, Algorithm *D-G* is of a huge ART performance benefits, compared with Algorithm *P-G* and Algorithm *O-S*.

4.3. Impact of Degree of Freedom. Finally, we turn to perform simulations in networks with varied degree of freedom. The networks are with a rand generated topology and 112 communication links to be scheduled. As shown in Figure 6, the value of degree of freedom impacts the ASL performance significantly while it has relative little impact to the ART performance. That is because as the number of antenna elements (degree of freedom) is increased, the receive node is more effective and, thus, more communication links can be grouped in a same schedule time slot.

5. Conclusions

In this paper, we study the issue of the minimum length link scheduling in a wireless network with the MU-MIMO capability. Firstly, we formulate the MU-MIMO MLLS problem as an integer programming problem. Secondly, we introduce a novel and straight hypergraph method to model the multiple interferences, which cannot be captured by the traditional conflict graph model, and then we reformulate the MU-MIMO MLLS problem based on the new hypergraph model. And, we show that, the scheduling problem is NP-hard. Thirdly, we propose two efficient approximation solutions. For the first algorithm, named as Algorithm *P-G*, we show that the approximation ratio is at most $2n/\log_2(n)$. To reduce the time complexity, we also present a time efficient heuristic algorithm based on degree of interference greedy (Algorithm *D-G*). Finally, extensive simulation results show that (1) both Algorithm *P-G* and Algorithm *D-G* can scale to quite a large number of links and produce satisfying suboptimal solutions; (2) generally, Algorithm *P-G* has a better ASL performance than Algorithm *D-G*, while the time complexity of Algorithm *D-G* is largely reduced compared with that of Algorithm *P-G*.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant no. 61070203, No. 61202484, the Ph.D. Program Foundation of Ministry of Education of China under Grant no. 20124307120033, and the Excellent Graduate Innovation Foundation of NUDT under Grant No. B120608.

References

[1] R. Nelson and L. Kleinrock, "Spatial TDMA: a collision-free multihop channel access protocol," *IEEE Transactions on Communications*, vol. 33, no. 9, pp. 934–944, 1985.

[2] S. C. Ergen and P. Varaiya, "TDMA scheduling algorithms for wireless sensor networks," *Wireless Networks*, vol. 16, no. 4, pp. 985–997, 2010.

[3] B. Hajek and G. Sasaki, "Link scheduling in polynomial time," *IEEE Transactions on Information Theory*, vol. 34, no. 5, part 1, pp. 910–917, 1988.

[4] G. Sharma, R. R. Mazumdar, and N. B. Shroff, "On the complexity of scheduling in wireless networks," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM '06)*, pp. 227–238, September 2006.

[5] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 5: Enhancements for Higher Throughput," IEEE P802.11n Standard, 2009.

[6] "Proposed TGac Draft to Wireless LAN Media Access Control (MAC) and Physical Layer (PHY) Specifications: Enhancements for Higher Throughput," IEEE P802.11ac/D0, 2011.

[7] S. Chu and X. Wang, "Adaptive and distributed scheduling in heterogeneous MIMO-based ad hoc networks," in *Proceedings of the IEEE 6th International Conference on Mobile Adhoc and Sensor Systems (MASS '09)*, pp. 217–226, October 2009.

[8] D. M. Blough, G. Resta, P. Santi, R. Srinivasan, and L. M. Cortés-Pena, "Optimal one-shot scheduling for MIMO networks," in *Proceedings of the 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '11)*, pp. 404–412, June 2011.

[9] L. Zheng and D. N. C. Tse, "Diversity and multiplexing: a fundamental tradeoff in multiple-antenna channels," *IEEE Transactions on Information Theory*, vol. 49, no. 5, pp. 1073–1096, 2003.

[10] C. Berge, *Graphs and Hypergraphs*, Elservier, New York, NY, USA, 1973.

[11] C. Berge, *Hypergraphs: The Theory of Finite Set*, vol. 45, North-Holland Publishing, Amsterdam, The Netherlands, 1989.

[12] I. Dinur, O. Regev, and C. Smyth, "The hardness of 3-uniform hypergraph coloring," in *Proceedings of the 34rd Annual IEEE Symposium on Foundations of Computer Science*, pp. 33–40, November 2002.

[13] S. Khot, "Hardness results for approximate hypergraph coloring," in *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pp. 351–359, May 2002.

[14] A. Wigderson, "Improving the performance guarantee for approximate graph coloring," *Journal of the Association for Computing Machinery*, vol. 30, no. 4, pp. 729–735, 1983.

[15] B. Berger and J. Rompel, "A better performance guarantee for approximate graph coloring," *Algorithmica*, vol. 5, no. 4, pp. 459–466, 1990.

[16] M. M. Halldórsson, "A still better performance guarantee for approximate graph coloring," *Information Processing Letters*, vol. 45, no. 1, pp. 19–23, 1993.