# Efficient Link Scheduling Solutions for the Internet of Things Under Rayleigh Fading

Kan Yu, Jiguo Yu, *Senior Member, IEEE*, *Member, ACM*, Xiuzhen Cheng, *Fellow, IEEE*, *Member, ACM*, Dongxiao Yu, *Senior Member, IEEE*, and Anming Dong, *Member, IEEE*

***Abstract*—Link scheduling is an appealing solution for ensuring the reliability and latency requirements of Internet of Things (IoT). Most existing results on the link scheduling problem were based on the graph or SINR (Signal-to-Interference-plus-Noise-Ratio) models, which ignored the impact of the random fading gain of the signals strength. In this paper, we address the link scheduling problem under the Rayleigh fading model. Both Shortest Link Scheduling (SLS) and Maximum Link Scheduling (MLS) problems are studied. In particular, we show that a set of links can be activated simultaneously under Rayleigh fading model if all link SINR constraints are satisfied. Based on the analysis of previous Link Diversity Partition (LDP) algorithm, we propose an Improved LDP (ILDP) algorithm and a centralized algorithm by localizing the global interference (denoted by CLT), building on which we design a distributed CLT algorithm (denoted by RCRDCLT) that converges to a constant approximation factor of the optimum with the time complexity of $O(\ln n)$, where $n$ is the number of links. Furthermore, executing repeatedly RCRDCLT can solve the SLS with an approximation factor of $\Theta(\ln n)$. Extensive simulations indicate that CLT is more effective than previous six popular link scheduling algorithms, and RCRDCLT has the lowest time complexity while only losses a constant fraction of the optimum schedule.**

***Index Terms*—Link scheduling, Internet of Things, distributed algorithm, Rayleigh fading, SINR.**

## I. Introduction

FOR an Internet of Things (IoT), interference, connectivity, capacity, throughput, delay, security and so on are most important indices [1]. Wireless networks play a crucial role in IoT and are a fundamental architecture to gather data of IoT devices. Designing the efficient link scheduling algorithms in wireless networks is one of the effective ways that can help an IoT to realize the connectivity, increase capacity and throughput, and decrease delay. Besides, the interference among signals must be considered when we design link scheduling algorithms. Also, the interference model selection has significant effects on the performance and complexity of link scheduling algorithms.

Generally, the goal of link scheduling problem is two-fold. One is Maximum Link Scheduling (MLS), i.e., maximizing the number of links that can be scheduled concurrently in a single slot. Another is Shortest Link Scheduling (SLS), which aims at scheduling a set of links in the minimum number of slots. Initially, the link scheduling problem is studied from the beginning of the graph model, but the practicability of algorithms designed under this model is limited by its duality. Subsequently, extensive research on the problem mainly considered the deterministic Signal-to-Interference-plus-Noise-Ratio (SINR) model which defines the signal strength fades in a fixed speed. In fact, the strength of a signal cannot be deterministic due to the fading characteristic in signal propagation. The Rayleigh fading model is one of more realistic models, and can describe the real-world environments [2]. However, the model leads to the design and analysis of link scheduling algorithms more challenging and difficult in return. Consequently, there are few works focusing on link scheduling subject to Rayleigh fading constraint (e.g., [3]–[6]).

The success probability of a link cannot achieve 1 under the Rayleigh fading model since the channel fading gain makes the strength of received signal uncertain [4]. Nonetheless, it is reasonable to regard a link as being scheduled successfully if its success probability is $1-\epsilon$ at least, where $\epsilon$ is an acceptable failure probability. Similar assumption was also applied in the previous works (e.g. [3], [5], [6]). However, their works were unsatisfactory to provide a better scheduling performance which refers to scheduling more concurrent links in a single slot for the MLS, and using less slots to schedule all links for the SLS.

In the current paper, to enhance the scheduling performance further, we present a series of novel and efficient algorithms with lower time complexity. The main contributions of the paper are summarized as follows.

- Based on the successive interference cancellation (SIC) technique, we improve Link Diversity Partition algorithm (ILDP) under Rayleigh fading model, a centralized algorithm for the MLS with low scheduling performance under the deterministic SINR model, which builds the relationship between two models. Specifically, ILDP gives a method to localize the global interference, which presents a significant insight into distributed algorithm design.

Manuscript received February 8, 2020; revised April 17, 2021; accepted June 16, 2021; approved by IEEE/ACM Transactions on Networking Editor J. Shin. Date of publication July 15, 2021; date of current version December 17, 2021. This work was supported by the National Natural Science Foundation of China under Grant 61771289, Grant 61832012, Grant 61672321, and Grant 61701269. *(Corresponding author: Jiguo Yu.)*

Kan Yu is with the School of Computer Science, Qufu Normal University, Rizhao 276826, China (e-mail: kanyu@qfnu.edu.cn).

Jiguo Yu is with the School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, China, and also with the Shandong Laboratory of Computer Networks, Jinan 250014, China (e-mail: jiguoyu@sina.com).

Xiuzhen Cheng and Dongxiao Yu are with the School of Computer Science and Technology, Shandong University, Qingdao 266510, China (e-mail: xzcheng@sdu.edu.cn; dxyu@sdu.edu.cn).

Anming Dong is with School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250014, China (e-mail: anmingdong@qlu.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TNET.2021.3093306, provided by the authors.

Digital Object Identifier 10.1109/TNET.2021.3093306

- By utilizing interference localization, we then propose a Centralized and Localized Traversal CLT) algorithm for the MLS, by proving that the interference beyond a certain range is a constant. Compared with the optimal schedule, CLT gives a constant approximation factor. Combining with random contention resolution, we present a distributed implementation of CLT RCRDCLT) with $O(\ln n)$ rounds, where $n$ is the number of links, which improves the performance of our prior works for MLS in [5] and [6].
- Executing RCRDCLT repeatedly gives an effective solution for the SLS, and an approximation factor of $\Theta(\ln n)$ is proposed, which shows that the best performance guarantee for the SLS is logarithmic.

The rest of this paper is organized as follows. Section II provides the related work. In Section III, we present network model and notations. In Section IV, we present our centralized and distributed link scheduling algorithms with rigorous analyses. Extensive experiments are implemented to evaluate the performances of our designed algorithms in Section V. Finally, we conclude the paper in Section VI.

## II. Related Work

The link scheduling problem has been studied extensively in previous literatures. Initially, the problem was considered under the graph model (e.g., [7]–[14]). Afterwards, it was noted that the deterministic SINR model can offer a more realistic representation of interference between signals. Moscibroda and Wattenhofer first studied the link scheduling problem under the deterministic SINR model [15]. Then, Goussevskaia *et al.* gave a NP-hard proof for the link scheduling problem under the SINR model [16]. Subsequently, attentions for designing effective link scheduling algorithms shifted to a more realistic model using SINR, such as centralized (e.g., [15], [17]–[21]) and distributed algorithms (e.g., [22]–[28]).

In [7], Sharma *et al.* modeled the interference by $k$-hop interference models, in which two links within $k$-hop distance interfere with each other and no two links within $k$-hops can successfully transmit at the same time. Although these models were too simplistic, they offered some significant insights into interference localization and distributed algorithm design under the deterministic SINR model.

To localize the global interference under the deterministic SINR model, one of effective methods is partitioning the network area into some cells with enough size. By adopting this method, Goussevskaia *et al.* classified link classes according to link length, and constructed a feasible schedule for each link class using a greedy strategy [16]. Similar idea was also applied in [18], [22], [29]–[31]. Based on the fact that a short link can tolerate more interference while a long link tolerates less, Huang *et al.* proposed a novel approximation algorithm for the SLS [30]. Their motivation is that if two links are far away from each other, the interference of one link on the other should be small. By partitioning the links into disjoint local link sets with a certain distance away from each other, such that independent scheduling inside each local link set is possible. Moreover, hexagon partition based link scheduling algorithms were proposed by Yu *et al.* [18]. Simulations demonstrated that hexagon-partition based link scheduling was more efficient than the square-partition employed by [16] and [31].

For distributed algorithm design under the deterministic SINR model, Dinitz first proposed a regret-learning strategy for the MLS [32]. Asgeirsson and Mitra applied the regret-learning to design an $O(1)$-approximation algorithm for uniform power assignment [33]. However, these game-theoretic algorithms take time polynomial of $n$ to converge. In addition, random contention resolution can be implemented to solve the link scheduling problem in a distributed fashion [34]. The idea is that each agent accesses the limited resource in any slot with a certain probability $q$ until its first success. In case of a collision, none of the involved agents is successful in this slot. This idea is easily applicable in wireless networks by letting each IoT device transmit its packet in each slot with probability $q$ until the first success. For the SLS, Kesselheim and Vöcking gave a distributed $O(\log^2 n)$-approximation algorithm with sublinear power assignment [34], and later resulting in a $O(\log n)$-approximation algorithm by Halldórsson and Mitra [35]. Similar conclusions were also obtained in works [24] and [36]. In [36], Goussevskaia *et al.* proposed a centralized algorithm (GHW) for the MLS with constant approximation ratio guarantee, and solved the SLS by executing GHW repeatedly with the time complexity of $O(n \log n)$.

Note that the deterministic SINR model ignores the impact of random fading on the strength of received signal. In realistic environment, a successful link under the deterministic SINR model does not means that the link can be successfully scheduled under the Rayleigh fading model. More important, to enable distributed algorithm designs, predictable interference control, an open question is whether it is possible to develop a model of measuring interference, which has the locality of the protocol model and high fidelity under the Rayleigh fading.

In [4], Dams *et al.* gave a scheme adapting link scheduling algorithms in the deterministic SINR model to the Rayleigh-fading model, while losing only a factor of $O(\log^* n)$[1] in the approximation guarantee, which was reduced to a constant by Halldórsson and Tonoyan [37]. Significantly, Dams *et al.* showed that success probability of a successful link under the deterministic SINR model is at least $1/e$ under Rayleigh fading model. Following this work, by utilizing the idea of localizing global interference under Rayleigh fading model, some centralized and distributed algorithms were proposed for the MLS and SLS (e.g., [3], [5], [6]).

In [3], Qiu and Shen proposed centralized and distributed algorithms (LDP and RLE) with $O(g(L))$ and $O(1)$ performance guarantees, respectively, where $g(L)$ (called length diversity) is the number of magnitudes of transmission link lengths. In our previous work [5], we solved the MLS by designing a distributed algorithm Syn_DLS to increase reliability and latency requirements of IoT. In our another previous work [6], we derived the minimum distance constraint $d_{\min}$ between the interferers and the receiver under the deterministic SINR model, and designed more effective distributed algorithms (DLS and DDLS) based on random contention resolution, by ignoring interference safely outside $d_{\min}$ as a constant and showing that those interference cannot influence whether a link is successful or not.

In summary, the MLS under the Rayleigh fading model can be simplified as the following one: given a guarantee requirement of success probability $1-\epsilon$, in the deterministic SINR model, removing all interfering links within some

[1] $\log^*$ denotes the iterated-logarithm function, and $O(\log^* n)$ is essentially "almost constant."

distance (a function of $1-\epsilon$) for each selected link, and this link has at least $1-\epsilon$ success probability under the Rayleigh fading model. The next step that we need to do is to maximize the number of those links. Consequently, it enables us to pay attention to algorithm design and analysis with the consideration of the deterministic SINR model, and apply them to the Rayleigh fading model. To sum up, the difference between [3], [5], [6], [36], and this paper is given as follows.

- Compared with [3], [5], and [6], we consider the interference-limited network (i.e., without noise), which simplifies the expression of success probability of a link, and we observe the relationship of distance among links clearly. In addition, SIC is adopted to remove the impact of stronger interfering signal, which can increase success probability of a link.
- Compared with [36], Rayleigh fading model is further considered, under which distributed algorithms were proposed in [5] and [6] based on idea of interference localization. Compared with above scheduling algorithms, by using SIC technique and plane partition, we present a more effective way of interference localization, and the designed distributed algorithm can schedule more successful links in a time slot and use more less number of time slots to ensure that each link has been scheduled at least once.

## III. Network Model and Definition

Given an IoT network consisting of $n$ communication links $L=\{l_1,\ldots,l_n\}$, where $l_i$ represents a transmission from a sender $s_i$ to a receiver $r_i$. The set of senders is denoted by $S=\{s_1,\ldots,s_n\}$. The distance from a sender $s_i$ to a receiver $r_j$ is denoted by $d_{s_i r_j}$, and $d_{ii}$ for short if $i=j$.

The wireless propagation is described by attenuation and fading. The Rayleigh fading means that the instantaneous envelope of the received signal follows Rayleigh distribution and its power is an exponentially distributed random variable. It is viewed as a reasonable model for tropospheric and ionospheric signal propagation as well as the effect of heavily built-up urban environments on radio signals. Under the above assumptions, if a signal is transmitted by sender $s_i$, it will be received by receiver $r_i$ with the strength of $P_i h_{ii} d_{ii}^{-\alpha}$, where $P_i$ is the transmit power and assumed to be the same for all senders, $\alpha$ is the path-loss exponent, $h_{ii}$ is the channel fading gain and follows an exponential distribution with unit mean [3]–[5]. Compared to noise, inter-link interference is the dominating source of limitation on the success probability. Therefore, it is interesting to focus the analysis in the interference-limited regime, which means that the interference dominates the noise since the node density is quite high in large wireless network, and the noise can be neglected [38], [39]. When the received SIR at receiver $r_i$ is greater than or equal to $\gamma_{th}$, $r_i$ can correctly decode the signal transmitted by $s_i$. Mathematically,

$$\gamma_i=\frac{h_{ii}\cdot d_{ii}^{-\alpha}}{\sum\limits_{s_j\in S, j\neq i} h_{ji}\cdot d_{s_j r_i}^{-\alpha}}\geq\gamma_{th},\tag{1}$$

where $\gamma_{th}$ is the decoding threshold [5], [36].

Due to the existence of channel fading gain, a link cannot be successfully scheduled with probability 1 under the Rayleigh fading model [3]–[5]. Therefore, we tolerate an acceptable failure probability $\epsilon$ for the transmission. That is, the link $l_i$ can be successfully scheduled under Rayleigh fading model if the probability of $\gamma_i\geq\gamma_{th}$ is greater than or equal to $1-\epsilon$. Similar assumption was also adopted in [3]–[5].

*Lemma 1 ([3], [5]):* Given a link $l_i$ and a sender set $\mathcal{P}\subset S$, the success transmission probability of $l_i$ is

$$p_i=\prod_{j\neq i, s_j\in\mathcal{P}}\frac{1}{1+\left(\frac{d_{ii}}{d_{s_j r_i}}\right)^{\alpha}\gamma_{th}}.\tag{2}$$

A schedule $\mathcal{P}$ is feasible if all senders in $\mathcal{P}$ can successfully transmit the message to their intended receivers with probability of at least $1-\epsilon$.

*Definition 1 (Interference Factor [3]):* Define the interference factor of $s_i$ on $r_j$ as

$$f_{s_i,r_j}=\begin{cases}\ln\left(1+\left(\dfrac{d_{jj}}{d_{s_i r_j}}\right)^{\alpha}\gamma_{th}\right) & \text{if } i\neq j;\\ 0 & \text{if } i=j.\end{cases}$$

Accordingly, we use $f_{\mathcal{P},r_j}$ to denote the interference factor of set $\mathcal{P}$ on receiver $r_j$, where $f_{\mathcal{P},r_j}=\sum\limits_{s_i\in\mathcal{P}} f_{s_i,r_j}$.

Based on the definition of interference factor, a link $l_i$ can be regarded as successful under the Rayleigh fading model iff $f_{\mathcal{P},r_j}\leq\gamma_\epsilon$ holds, where $\gamma_\epsilon=\ln\left(\frac{1}{1-\epsilon}\right)$.

### A. Successive Interference Cancellation (SIC)

According to the decreasing order of received power, the receiver with SIC technique can decode those stronger signals than its intended signal. If an interferer is stronger enough for the receiver, then it can be decoded first and canceled off by this receiver before decoding its intended signal. Finally, the expected signal can be decoded successfully.

In $k$-SIC, at most $k$ stronger signals can be decoded. The necessary and sufficient conditions for $k$-SIC at $r_m$ is

$$\frac{S_{im}}{\sum_{j=i+1}^{k} S_{jm}+\sum_{s>k} S_{sm}}\geq\gamma_{th},\quad 1\leq i\leq k,\tag{3}$$

where $S_{im}$ is the $i^{th}$ strongest power received at $r_m$ [40], [41].

Table I shows some important notations.

## IV. Centralized and Distributed Algorithms for MLS Problem

In this section, we first propose an Improved LDP (ILDP) based on 1-SIC by analyzing classic link scheduling algorithm, namely Link Diversity Partition (LDP) [16]. Then utilizing the idea of interference localization, we propose a centralized link scheduling algorithm (denoted by CLT) with a constant approximation factor, and obtain its distributed implementation (denoted by RCRDCLT) with logarithmic rounds.

### A. An Improved LDP Algorithm

LDP is designed for link scheduling under the deterministic SINR [16]. The key idea of LDP is that it first classifies the links with similar lengths to one class. For each link class, LDP partitions the whole network area into squares with same size and set neighboring squares to different colors. Such color setting ensures that the links in the same-color squares always have a far enough distance from each other. Then, all the selected links in the same-color squares form a feasible schedule.

TABLE I
NOTATIONS

| Term | Description | Page |
|---|---|---|
| $n$ | the number of communication links | 3 |
| $L$ | the set of $n$ communication links | 3 |
| $s_i$ | the sender of link $l_i$ | 3 |
| $r_i$ | the receiver of link $l_i$ | 3 |
| $S$ | the set of all senders | 3 |
| $d_{s_i r_j}$ | the distance between node $s_i$ and node $r_j$ | 3 |
| $h_{ij}$ | channel fading gain between node $s_i$ and $r_j$ | 3 |
| $\alpha$ | path-loss exponent | 3 |
| $\gamma_{th}$ | decoding threshold | 3 |
| $\epsilon$ | acceptable failure probability | 3 |
| $\gamma_\epsilon$ | $\gamma_\epsilon = \ln\left(\frac{1}{1-\epsilon}\right)$ | 3 |
| $\mathcal{P}$ | a feasible schedule | 3 |
| $f_{s_i, r_j}$ | the interference factor of $s_i$ on $r_j$ | 3 |
| $f_{\mathcal{P}, r_j}$ | the interference factor of set $\mathcal{P}$ on $r_j$ | 3 |
| $g(L)$ | the link length diversity, a small constant | 4 |
| $L_k$ | the link subset for link class $k$ | 4 |
| $l_{\min}$ | the minimum link length | 4 |
| $\beta_k$ | the size of square for link class $k$, $\beta_k = 2^{h_k+1} l_{\min}\beta$ | 4 |
| $\beta$ | a constant, and $\beta = \left(\frac{16}{\alpha-2}\cdot\frac{\gamma_{th}}{\gamma_\varepsilon}\cdot\frac{\alpha-1}{\alpha-2}\right)^{\frac{1}{\alpha}} + 1$ | 4 |
| $\rho$ | a small constant, and $\rho \le \left[\frac{\gamma_\epsilon}{\gamma_{th}}\cdot\frac{1}{1+\frac{16}{(\beta-1)^\alpha}\cdot\frac{\alpha-1}{\alpha-2}}\right]^{\frac{1}{\alpha}}$ | 4 |
| $\psi$ | the optimal solution for the SLS | 6 |
| $d_{\min}$ | the minimum distance between interferer and receiver | 6 |

However, it is difficult to apply LDP under the Rayleigh fading model directly. The reason is that, on the one hand, a successful link of deterministic SINR model may fail to be scheduled under the Rayleigh fading model. On the other hand, for a given link class, the size of all squares is fixed under the deterministic SINR model, while difficult to be estimated once the Rayleigh fading model is considered. In particular, given a successful link in the deterministic SINR model, above failure probability can be upper-bounded. Also, combing with upper bound of interference factor (considering failure probability constraint $\epsilon$), we can calculate a fixed size of each square, which ensures the success probability of each scheduled link in same-color squares is at least $1-\epsilon$ under the Rayleigh fading model.

*Definition 2:* ( [3], [16]) Given a set of links, the link length diversity, denoted by $g(L)$, is defined as

$$g(L) = \{h \mid \exists l_i \in L : \lfloor \log_2(d_{ii}/l_{\min}) \rfloor = h\}. \tag{4}$$

Namely the diversity $g(L)$ is the number of magnitudes of distances. In real applications, $g(L)$ is usually a small constant [3], [16].

We only consider 1-SIC protocol for analysis simplicity, i.e., at most 1 strong interfering signal can be eliminated. Next, we introduce ILDP based on 1-SIC in detail. This algorithm starts to run by building $g(L)$ disjoint link classes $L_1, \ldots, L_{g(L)}$ from $L$, s.t.,

$$L_k = \{l_i \in L \mid 2^{h_k} l_{\min} \le d_{ii} < 2^{h_k+1} l_{\min}\}.$$

Note that $h_k$ is independent of $n$ and can be upper-bounded by a constant if the maximum link length is pre-known.

Then, for link class $L_k$, ILDP partitions the entire network region into squares with size length $\beta_k = 2^{h_k+1} l_{\min}\beta$, where

$$\beta = \left(\frac{16}{\alpha-2}\cdot\frac{\gamma_{th}}{\gamma_\varepsilon}\cdot\frac{\alpha-1}{\alpha-2}\right)^{\frac{1}{\alpha}} + 1, \tag{5}$$

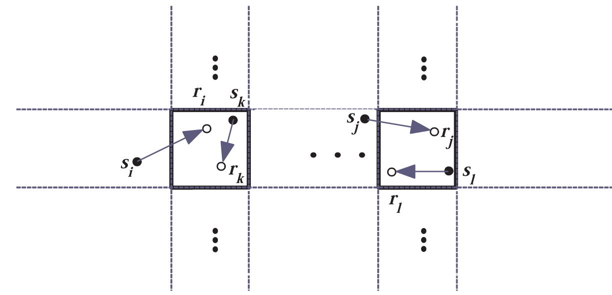



Fig. 1. Partition and two links exist in a same square by using 1-SIC.

and set neighboring squares to different colors. And thus, all the selected links from the same-color squares form a feasible schedule [16], denoted by $\mathcal{P}$. By using 1-SIC, each receiver of a successful link in $\mathcal{P}$ can tolerate at most one strongest interferer, when the distance between them satisfies the constraint in the following Lemma 2. For instance, assume that $l_k$ and $l_i$ belong to the same link class. As shown in Fig. 1, $r_i$ allows to the existence of $s_k$ if $d_{s_k r_i} \le \rho d_{ii}$. Moreover, $r_k$ can also remove the impact of $s_i$ in the same-color square by 1-SIC and joins into scheduling set, if $d_{s_i r_k} \le \rho d_{kk}$. In this way, compared with LDP, the size of scheduling set $\mathcal{P}_{ildp}$ obtained by ILDP may be doubled in the best case.

*Lemma 2:* If two successful links are allowed to exist in the same square, the parameter $\rho$ must satisfy

$$\rho \le \left[\frac{\gamma_\epsilon}{\gamma_{th}}\cdot\frac{1}{1+\frac{16}{(\beta-1)^\alpha}\cdot\frac{\alpha-1}{\alpha-2}}\right]^{\frac{1}{\alpha}}.$$

The proof of **Lemma** 2 is shown in Appendix A.

The pseudo-code of ILDP is shown in Algorithm 1.

**Algorithm 1** 1-SIC Based Improved LDP (ILDP)

**Initialization:** Scheduling set $\mathcal{P}_{ildp}^{(k,j)} = \emptyset$ for link class $L_k$ and color $j$
**Input:** $g(L)$ link classes $\{L_1, \ldots, L_{g(L)}\}$
**for** $k = 1$ to $g(L)$ **do**
    Partition the network region into squares of size $\beta_k \times \beta_k$ for link class $L_k$
    Color the squares with $\{1, 2, 3, 4\}$, s.t. no two adjacent squares have the same-color
    **for** $j = 1$ to $4$ **do**
        The sender $s_i$ joins into $\mathcal{P}_{ildp}^{(k,j)}$
        Find link $l_k$ in link class $L_k$ and $d_{kk} \le d_{ii}$
        **if** $d_{s_k r_i} \le \rho d_{ii}$ and $d_{s_i r_k} \le \rho d_{kk}$ **then**
            The sender $s_k$ joints into $\mathcal{P}_{ildp}^{(k,j)}$
        **else**
            Remove link $l_k$ from $L_k$
        **end if**
    **end for**
**end for**
return $\mathcal{P}_{ildp} = \arg\max\left\{\mathcal{P}_{ildp}^{(k,j)}, \forall k, j\right\}$

We show that $\mathcal{P}_{ildp}$ is feasible with the consideration of the Rayleigh fading model in the proof of Theorem 1.

*Theorem 1:* ILDP provides a feasible schedule.

The proof of **Theorem** 1 is shown in Appendix B.

*Theorem 2:* The approximation factor of ILDP is $O(1)$.

*Proof:* Assume that there are two links in a same square satisfying the distance constraints in lines 8 and 9 of Algorithm 1. Thus, $|\mathcal{P}_{ildp}^k| = 2|\mathcal{P}_{ldp}^k|$. Let $\mathcal{P}_{opt}^k$ be the optimal schedule in a link class $L_k$, $\mathcal{P}_{opt} = \arg\max\{\mathcal{P}_{opt}^k, \forall k\}$, and $\mathcal{P}_{ildp}^k = \arg\max\{\mathcal{P}_{ildp}^{(k,j)}, \forall j\}$. According to conclusion in [3], we know $|\mathcal{P}_{ldp}^k| \geq \frac{|\mathcal{P}_{opt}^k|}{4}$, then

$$\frac{|\mathcal{P}_{ildp}|}{2} \geq \frac{\sum_{k=1}^{g(L)} |\mathcal{P}_{ldp}^k|}{4g(L)} \geq \frac{\sum_{k=1}^{g(L)} |\mathcal{P}_{opt}^k|}{8g(L)} \geq \frac{|\mathcal{P}_{opt}|}{8g(L)}, \tag{6}$$

where $\mathcal{P}_{ildp} = \arg\max\{\mathcal{P}_{ildp}^k, \forall k\}$. Therefore, $\frac{|\mathcal{P}_{opt}|}{|\mathcal{P}_{ildp}|} \leq 8g(L)$. ∎

*Theorem 3:* For the SLS, executing ILDP repeatedly yields an $\Theta(\ln n)$ approximation factor in the worst case.

*Proof:* Assume that $\psi$ is the minimum number of slots in the optimal solution, and then $\psi = \frac{n}{\mathcal{P}_{opt}}$. Combining with result in Theorem 2, the maximum number of slots needed by ILDP, denoted by $\psi_{ildp}$, satisfies $\psi_{ildp} \leq 8g(L)\psi$.

The size of scheduling set obtained by ILDP is at least $\frac{n}{\psi_{ildp}}$. Then, the number of remaining unscheduled links is at most $n\left(1 - \frac{1}{\psi_{ildp}}\right)$, and is reduced to $n\left(1 - \frac{1}{\psi_{ildp}}\right)^s < ne^{-s/\psi_{ildp}}$ after $s$ slots. Thus, when $s \geq \psi_{ildp} \ln n$, all links have been scheduled successfully, and $\frac{s}{\psi} \leq 8g(L)\ln n$. Moreover, $\frac{s}{\psi} \geq \ln n$ since $\psi_{ildp} \geq \psi$, which gives an $\Theta(\ln n)$ approximation factor. ∎

So far, we have proposed one centralized algorithm for link scheduling problem. Centralized scheduling algorithms require a central coordinator to allocate the link schedules. For a large IoT network, centralized algorithms are vulnerable to single point failure, if the central controller is down, there is no one else to coordinate the resource allocation, and distributed link scheduling algorithms are urgently designed, which only needs the coordination among the nodes in a local neighborhood. Moreover, it is difficult to handle global interference defined by the deterministic SINR model using local distributed algorithms, since only considering local interference can lead to incorrect conclusions. From the proof in Theorem 1, we can see that if a link is successfully scheduled, those neighbouring links need to be removed. In other words, whether a link schedule is successful or not only depends on the localization information, which is better for a distributed implementation.

### *B. Interference Localization-Based Distributed Algorithm*

In this subsection, to design distributed algorithm, we first propose a centralized and localized link scheduling algorithm for the MLS. In each traversal, the algorithm first greedily selects the unselected sender with the shortest link length, since the strength is stronger at close range. Define $d_{\min}$ as the minimum distance between a non-intended sender and the receiver. And $d_{\min}$ satisfies

$$d_{\min} = (\beta - 1)d_{ii} \tag{7}$$

for link $l_i$. That is, the distance between link $l_i$ and successful links scheduled before and after it is at least $d_{\min}$. The pseudo-code is given in Algorithm 2.

**Algorithm 2** Centralized and Localized Traversal (CLT)

**Initialization:** Scheduling set $\mathcal{P}_{clt} = \emptyset$
**Input:** Sender set $S = \{s_1, \ldots, s_n\}$
**Output:** $\mathcal{P}_{clt}$
**while** $S$ is not empty **do**
  Pick up the sender $s_i$ in $S$ that has the shortest link length and add it to $\mathcal{P}_{clt}$
  Remove each sender $s_j$ from $S$, i.e. $d_{s_j r_i} < d_{\min}$
**end while**
Return $\mathcal{P}_{clt}$

For link $l_i$, we construct the disc centered at receiver $r_i$ with the radius of $d_{\min}$. After executing CLT, all links whose senders, apart from $s_i$, are in the disc are removed. This process is repeated until all links have been either scheduled or deleted. The time complexity is $O(n \log n)$.

In the following, we prove that the schedule $\mathcal{P}_{clt}$ is both feasible (Theorem 4) and efficient, i.e., only a constant factor away from the optimum (Theorem 5).

*Theorem 4:* CLT can output a feasible schedule.

*Proof:* The distance between the senders in $\mathcal{P}_{clt} \setminus \{s_i\}$ and $r_i$ is at least $d_{\min}$, the interference factor of $\mathcal{P}_{clt}$ on $r_i$ is at most

$$f_{\mathcal{P}_{clt}, r_i} = \sum_{s_j \in \mathcal{P}_{clt} \setminus \{s_i\}} \ln\left(1 + \frac{d_{ii}^\alpha}{d_{s_j r_i}^\alpha}\gamma_{th}\right) \leq \sum_{\mu=1}^{+\infty} \frac{1}{(\mu(\beta-1))^\alpha}\gamma_{th} < \frac{\alpha(\alpha-2)^2}{16(\alpha-1)^2}\gamma_\epsilon < \gamma_\epsilon,$$

based on the following Formula [6], [36]

$$\sum_{\mu \geq d_{\min}} \frac{1}{\mu^\alpha} < \alpha \int_{d_{\min}}^{+\infty} \frac{1}{\mu^\alpha} dx,$$

which means that link $l_i$ can be successfully scheduled. ∎

*Lemma 3:* Let $s_i \in \mathcal{P}_{clt}$. The number of senders in set $\mathcal{P}_{clt} \setminus \{s_i\}$ with distance $kd_{ii}$ away from $s_i$ is at most

$$\frac{\gamma_\epsilon e^{\gamma_\epsilon}}{\gamma_{th}}(1+k)^\alpha. \tag{8}$$

*Proof:* From definition 1, we know that interference factor of each sender $s_j \in \mathcal{P}_{clt} \setminus \{s_i\}$ on receiver $r_i$ cannot be greater than $\gamma_\epsilon$. Accordingly, $\frac{d_{ii}^\alpha}{d_{s_j r_i}^\alpha} \leq \frac{1}{e^{\gamma_\epsilon} - 1}$. Moreover, $\ln(1+x) \geq \frac{1}{e^{\gamma_\epsilon}}x$ holds when $x \in [0, e^{\gamma_\epsilon} - 1]$, and $d_{s_j r_i} \leq d_{s_j s_i} + d_{ii} = (1+k)d_{ii}$, $f_{s_j, r_i}$ is lower bounded by

$$f_{s_j, r_i} = \ln\left(1 + \frac{d_{ii}^\alpha}{d_{s_j r_i}^\alpha}\gamma_{th}\right) \geq \frac{1}{e^{\gamma_\epsilon}} \frac{d_{ii}^\alpha}{d_{s_j r_i}^\alpha}\gamma_{th} \geq \frac{1}{e^{\gamma_\epsilon}} \frac{1}{(1+k)^\alpha}\gamma_{th}. \tag{9}$$

Since the interference factor of $\mathcal{P}_{clt} \setminus \{s_i\}$ on $r_i$ cannot exceed $\gamma_\epsilon$, and there are at most $\frac{\gamma_\epsilon e^{\gamma_\epsilon}}{\gamma_{th}}(1+k)^\alpha$ such senders. Thus, the lemma holds. ∎

*Theorem 5:* The cardinality of the difference between $\mathcal{P}_{clt}$ and $\mathcal{P}_{opt}$ is bounded by a constant, i.e., $\frac{|\mathcal{P}_{opt}|}{|\mathcal{P}_{clt}|} \leq 5\frac{\gamma_\epsilon e^{\gamma_\epsilon}}{\gamma_{th}}(2+\delta)^\alpha + 1$, where $\delta = \beta - 1$.

*Proof:* By contradiction. Suppose that

$$|\mathcal{P}_{opt} \setminus \mathcal{P}_{clt}| > 5e^{\gamma_\epsilon}(2+\delta)^\alpha \frac{\gamma_\epsilon}{\gamma_{th}}|\mathcal{P}_{clt}|. \tag{10}$$

We label the set of senders in $\mathcal{P}_{opt} \setminus \mathcal{P}_{clt}$ by blue (i.e., $\mathcal{N}_b = \mathcal{P}_{opt} \setminus \mathcal{P}_{clt}$) and those in $\mathcal{P}_{clt}$ by red ($\mathcal{N}_r = \mathcal{P}_{clt}$). By Lemma 4.3 [36], if $|\mathcal{N}_b| > 5z|\mathcal{N}_r|$, then there is a $z$-blue-dominant point (sender) $s_i \in \mathcal{N}_b$, where

$$z = \frac{\gamma_\epsilon e^{\gamma_\epsilon}}{\gamma_{th}}(2+\delta)^\alpha. \tag{11}$$

We shall argue that the sender $s_i$ would have been picked by CLT, leading to a contradiction.

According to Lemma 4.3 [36], for any red point $s_j \in \mathcal{N}_r$, there exists a subset of blue points $G(s_j)$ such that $|G(s_j)| \geq z$ ($z = \frac{\gamma_\epsilon e^{\gamma_\epsilon}}{\gamma_{th}}(2+\delta)^\alpha$). Next, we prove that $s_i$ can be selected by CLT after $s_j$, i.e., $d_{ii} \geq d_{jj}$. We can derive that $d_{s_i s_j} > (\delta+1)\, d_{ii}$. Otherwise the number of senders within distance $(\delta+1)d_{ii}$ away from $s_i$ would be greater than $\frac{\gamma_\epsilon e^{\gamma_\epsilon}}{\gamma_{th}}(1+\delta+1)^\alpha$, which contradicts with the conclusion in Lemma 3. Using the triangle inequality, we get

$$d_{s_j r_i} \geq d_{s_j s_i} - d_{ii} > \delta d_{ii}$$

and

$$d_{s_i r_j} \geq d_{s_i s_j} - d_{jj} > d_{s_i s_j} - d_{ii} \geq \delta d_{ii} \geq \delta d_{jj},$$

which shows that $s_i$ should not have been deleted by CLT, resulting in a contradiction. ■

Next, we achieve the distributed implementation of CLT based on the random contention resolution. The main idea is that each sender starts to transmit independently, and it will join into the scheduling set, when there is no interferers within $d_{\min}$ starting to transmit or having been selected into scheduling set. The pseudo-code is given in Algorithm 3.

To model the scenario where the senders with the shorter link length have a priority to transmit, each sender whose link belongs to link class $L_k$ selects randomly a value from range $[2^{h_k}, 2^{h_k+1}]$ to transmit first. In this way, the shorter link is, the faster its sender starts to transmit. On the one hand, the senders in different link classes have different transmission times. On the other hand, denote $\mathcal{N}_k = 2^{h_k}$ and the senders whose corresponding links are in $L_k$ as the set $S_k$. The probability that at least two senders in $S_k$ select a same value is at most $\frac{1}{\mathcal{N}_k^2}$.

**Algorithm 3** Random Contention Resolution Based Distributed CLT (RCRDCLT)

1: **Initialization:** Scheduling set $\mathcal{P}_{rcrdclt} = \emptyset$
2: **Input:** Sender set $S = \{s_1, \ldots, s_n\}$
3: **Output:** $\mathcal{P}_{rcrdclt}$
4: Select $t_i$ randomly in range $[2^{h_k}, 2^{h_k+1}]$ for sender $s_i$
5: **if** Sender $s_i$ starts to transmit at time $t_i$ **then**
6:     **if** There is no senders starting to transmit or belonging to $\mathcal{P}_{rcrdclt}$ within $\beta d_{ii}$ **then**
7:         $s_i$ joins into $\mathcal{P}_{rcrdclt}$
8:     **else**
9:         Quit from current schedule
10:     **end if**
11: **end if**
12: **Output:** $\mathcal{P}_{rcrdclt}$

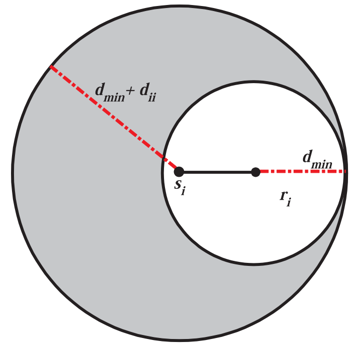



Fig. 2. A link $l_i$ scheduled by RCRDCLT, and those senders which are not successful in RCRDCLT but successful in CLT locate in the gray area.

Based on the following Lemma, the approximation factor of RCRDCLT will be given in Theorem 6.

*Lemma 4:* [42] Given two disks $D_1$ and $D_2$ of radii $R_1$ and $R_2$, respectively, assume that $R_1 > R_2$, we define $\chi^{R_1,R_2}$ to be the smallest number of disks $D_2$ needed to cover the larger disk $D_1$. It holds that

$$\chi^{R_1,R_2} \leq \frac{2\pi}{3\sqrt{3}} \cdot \frac{(R_1+2R_2)^2}{R_2^2}. \tag{12}$$

*Theorem 6:* RCRDCLT only losses a constant fraction of the optimum schedule, i.e., $|\mathcal{P}_{rcrdclt}| \geq \left(5\frac{\gamma_\epsilon e^{\gamma_\epsilon}}{\gamma_{th}}(2+\delta)^\alpha + 1\right) \cdot \left(\frac{2\pi}{3\sqrt{3}}\left(3+\frac{1}{\delta}\right)^2 + 1\right)|\mathcal{P}_{opt}|$.

*Proof:* For any link $l_i$, let $R_1 = d_{\min} + d_{ii}$ and $R_2 = d_{\min}$. From line 6 in Algorithms 2 and 3, there are no interferers within $d_{\min}$ around $r_i$ and within $d_{\min} + d_{ii}$ around $s_i$, as shown in Fig. 2. According to the conclusion in Lemma 4, those links scheduled by CLT cannot be successfully scheduled by RCRDCLT are at most

$$\chi^{R_1,R_2} \leq \frac{2\pi}{3\sqrt{3}}\left(3+\frac{1}{\delta}\right)^2.$$

That is, $|\mathcal{P}_{clt} \setminus \mathcal{P}_{rcrdclt}| \leq \chi^{R_1,R_2}|\mathcal{P}_{rcrdclt}|$.

Combining with conclusion in Theorem 5, we get

$$\frac{|\mathcal{P}_{opt}|}{|\mathcal{P}_{rcrdclt}|} \leq \left(5\frac{\gamma_\epsilon e^{\gamma_\epsilon}}{\gamma_{th}}(2+\delta)^\alpha + 1\right) \cdot (\chi^{R_1,R_2} + 1).$$

■

*Theorem 7:* Algorithm 3 ends after $O(\ln n)$ rounds.

*Proof:* RCRDCLT uses time selector to reduce collisions among different senders and ensure that shorter links have priority to transmit first. It can be clearly seen from the above algorithm that the smaller $h_k$ is, the sooner senders start to transmit. For sender $s_i$ in link class $L_k$, if all senders in $S_k \setminus \{s_i\}$ are away from $s_i$ at least $d_{\min} + d_{ii}$, it transmits at $t_i$ (selecting from $[2^{h_k}, 2^{h_k+1}]$) and joins into the schedule $\mathcal{P}$. Otherwise, $s_i$ should start to transmit first, when $s_i$ selects $t_i = \mathcal{N}_k$, the probability that all the senders in $S_k$ start to transmit after or along with $s_i$ is 1 since they can select any value from $[2^{h_k}, 2^{h_k+1}]$. Similarly, when $s_i$ determines $t_i = \mathcal{N}_k + 1$, above probability is $\left(\frac{\mathcal{N}_k - 1}{\mathcal{N}_k}\right)^{|S_k|}$. In this way, the probability that the sender $s_i$ starts to transmit before other senders in $S_k$ is

$$\begin{aligned} p &= \frac{1}{\mathcal{N}_k} 1^{|S_k|} + \cdots + \frac{1}{\mathcal{N}_k}\left(\frac{1}{\mathcal{N}_k}\right)^{|S_k|} \\ &= \frac{1}{\mathcal{N}_k}\left(\sum_{i=1}^{|\mathcal{N}_k|}\left(\frac{i}{\mathcal{N}_k}\right)^{|S_k|}\right) \\ &\geq \frac{1}{\mathcal{N}_k}\left(\sum_{j=1}^{|\mathcal{N}_k|}\left(\frac{1}{j}\right)^{|S_k|}\right), \quad j \text{ is natural number} \\ &\approx \frac{\zeta(|S_k|)}{\mathcal{N}_k}, \end{aligned}$$

where $\zeta(\cdot)$ is Riemann zeta function and it is a constant for $|S_k| > 1$ [43], which does not means that $s_i$ joins into $\mathcal{P}_{rcrdclt}$, once at least one sender in $\mathcal{P}_{rcrdclt}$ selects a time as same as $t_i$, they will quit the current schedule. Otherwise, $s_i$ joints into $\mathcal{P}_{rcrdclt}$. After consuming $\tau\frac{\mathcal{N}_k}{\zeta(|S_k|)}\ln n$ rounds for a constant $\tau$, $s_i$ either joins into $\mathcal{P}_{rcrdclt}$ or quits from current schedule with the high probability, i.e.,

$$1-\left(1-\frac{\zeta(|S_k|)}{\mathcal{N}_k}\right)^{\tau\frac{\mathcal{N}_k}{\zeta(|S_k|)}\ln n} \geq 1-\frac{1}{n^\tau}.$$

Therefore, the number of rounds for Algorithm 3 is $O(\ln n)$. ■

*Theorem 8:* Executing RCRDCLT repeatedly yields an $\Theta(\ln n)$ approximation factor for the SLS.

*Proof:* From Theorem 6, we know

$$|\mathcal{P}_{rcrdclt}| \geq \frac{1}{\eta_\epsilon}|\mathcal{P}_{opt}|,$$

where $\eta_\epsilon = \left(\frac{2\pi}{3\sqrt{3}}\left(3+\frac{1}{\delta}\right)^2+1\right)\cdot\left(\frac{5\gamma_\epsilon e^{\gamma_\epsilon}}{\gamma_{th}}(2+\delta)^\alpha+1\right)$.

Define $\psi_{rcrdclt}$ as the number of rounds to solve the SLS for RCRDCLT. Besides, similar to Theorem 3, $\psi_{rcrdclt} \leq \eta_\epsilon\psi$. Any subset of $n$ links contains a scheduling set and its size is at least $\frac{n}{\eta_\epsilon\psi}$. After costing at most $\psi\eta_\epsilon\tau\ln n$ rounds, the number of unsuccessful links is

$$n\left(1-\frac{1}{\eta_\epsilon\psi}\right)^{\psi\eta_\epsilon\tau\ln n} < 1,$$

which shows that all the links are successfully scheduled.

Then approximation factor for the SLS can be bounded by

$$\begin{aligned} \frac{\psi_{rcrdclt}}{\psi} &\leq \frac{\psi\eta_\epsilon\tau\ln n}{\psi} \\ &= \eta_\epsilon\tau\ln n. \end{aligned}$$

Similar to Theorem 3, $\frac{\psi_{rcrdclt}}{\psi} \geq \ln n$, resulting in $\Theta(\ln n)$. ■

## V. Evaluations

In this section, considering two network topologies with $500 \times 500$ $m^2$: random and cluster topologies, as shown in Fig. 3 and Fig. 4, the red nodes are senders and the white ones are receivers. We validate the impacts of system parameters on the scheduling performance of designed link scheduling algorithms by MATLAB. In the random topology, the senders and receivers are distributed randomly. In the cluster topology, $n_c$ clusters are selected randomly on the plane, and $n/n_c$ links are positioned inside disks of radius $r_c$ around each of them. The cluster topology aims to simulate a scenario of heterogeneous density distribution. Define the maximum length of a link as $l_{\max}$. The simulation was done over 100 different networks.

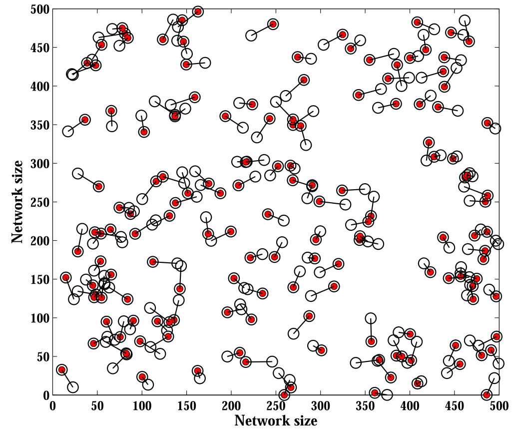



Fig. 3. Random topology.

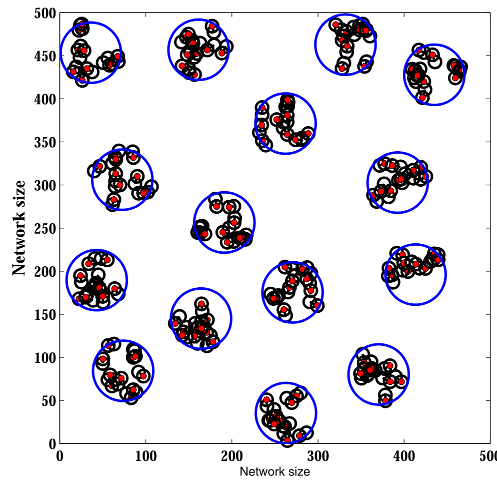



Fig. 4. Cluster topology.

For comparison, we use the centralized one-slot scheduling algorithms proposed by Goussevskaia, Halldórsson and Wattenhofer (GHW) [36], LDP [3], RLE [3], Syn_DLS [5], DLS [6] and DDLS [6] to solve the MLS and SLS. GHW is a simple greedy algorithm, where the links are processed in a non-decreasing order of length, and each link can be scheduled successfully if the affectance of the link, caused by all successful links is less than or equals to the following parameter

$$c = \frac{1}{(2+\max(2,((2^3\cdot 9+1)\gamma_{th}\frac{\alpha-1}{\alpha-2})^{\frac{1}{\alpha}}))}.$$

Related parameters in RLE are set to $c_2 = 0.5$ and

$$c_1 = \sqrt{2}\left(\frac{12\zeta(\alpha-1)\gamma_{th}}{\gamma_\epsilon(1-c_2)}\right)^{\frac{1}{\alpha}}+1,$$

respectively.

Moreover, although GHW is an $O(1)$-approximation algorithm and the time complexity is $O(n\log n)$, a small scheduling set can be obtained due to a lower $c$.

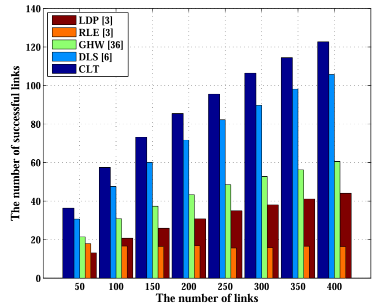



Fig. 5. Impact of $n$ with $\epsilon = 0.1$.

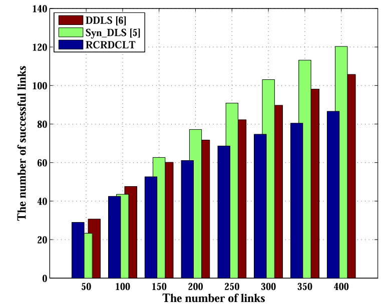



Fig. 6. Impact of $n$ with $\epsilon = 0.1$.

TABLE II

SIMULATED PARAMETERS AND VALUES

| Symbol | Meanings | Value |
|---|---|---|
| $n$ | the number of links | 200 |
| $l_{\min}$ | the minimum link length | 1 |
| $l_{\max}$ | the maximum link length | 20 |
| $\alpha$ | the path-loss exponent | 5 |
| $\gamma_{th}$ | the decoding threshold | 1 |
| $\epsilon$ | the acceptable failure probability | 0.1 |

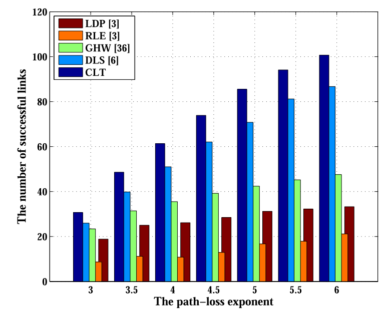



Fig. 7. Impact of $\alpha$ with $\epsilon = 0.1$.

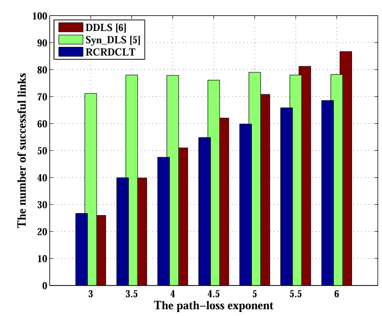



Fig. 8. Impact of $\alpha$ with $\epsilon = 0.1$.

### *A. MLS Validation in Random Networks*

First, in Fig. 5, we evaluate the influence of the number of links on performances of CLT, RCRDCLT and centralized and distributed algorithms in [3], [5], [6], [36]. The simulated parameters were set according to Table II. On the one hand, CLT gives a larger enhancement when $n$ increases, which indicates the parameter $d_{\min}$ in Eq. (7) is in effect: compared with LDP, GHW and DLS, the numbers of successful links in a single slot (i.e., single slot scheduling performance) are improved about $174.2\%$, $90.5\%$, and $18.4\%$, respectively.

As expected, GHW is not able to schedule more links for a larger $n$ due to the constraint of the parameter $c$. Even in sparse topology (100 links), CLT and RCRDCLT still compute 1.93 and 1.21 times the number of successful links obtained by GHW. The performance of LDP is poor since the size of each square is relatively larger and more links can located in a same square, i.e., fewer links are scheduled successfully. Setting $c_2 = 0.5$ in RLE, DDLS achieves a better performance than that of RLE due to low efficiency of the parameter $c_2$ in RLE. Among all the seven algorithms, CLT schedules the most number of successful links in a single slot for different $n$ in random topology.

On the other hand, from Fig. 6 we can see that compared with CLT, the single-slot scheduling performance of RCRDCLT decreases about $39.5\%$, this is because that the distance between the interferer and the receiver in CLT is at least $d_{\min}$, while RCRDCLT will reject a fraction of successful senders, scheduled by CLT, where the distance between them and the receiver changes from $d_{\min}$ to $d_{\min} + 2d_{ii}$, as shown in Fig. 2. But the performance difference between CLT and RCRDCLT can be upper-bounded, as shown in Theorem 6 and Table III. Although, compared with Syn_DLS and DDLS, the performance decrements of RCRDCLT are about $27.9\%$ and $18.9\%$, the time complexity of RCRDCLT decreases from $O(n \ln n)$ (Syn_DLS and DDLS needed) to $O(\ln n)$. Note that if we design distributed CLT based on the idea of DDLS in [6], achieved scheduling performance will be better than those of Syn_DLS and DDLS, resulting in $O(n \ln n)$ time complexity.

In Fig. 7 and Fig. 8, we analyze the influence of the path-loss exponent on the schedule performance for above algorithms. The simulation is done with $n = 200$, other settings keep the same as before. The single slot scheduling performances increase as $\alpha$ increasing. For LDP, the size of each square decreases with $\alpha$ increasing, as shown in Table IV, the probability that only one link exists in a square gets larger, resulting in more successful links. For CLT and RCRDCLT, distance constraint between the interferers and the receiver get smaller over $\alpha$ increasing, more links can join the scheduling set, as shown in Table V. In other words, the impact of $\alpha$ on interfering signal outperforms that of intended signal, a link can be successfully scheduled easily for larger $\alpha$. Similarly, the performances of RLE and Syn_DLS are also increased with the increment of $\alpha$, since $c_1 d_{ii}$ used in RLE and $d_{\min}$ used in DLS (or DDLS) decrease as $\alpha$ increases, more links can be scheduled. For GHW, the impact of $\alpha$ on interfering signal is larger than that of intended signal. Specifically, scheduling performance of Syn_DLS is not influenced by $\alpha$. This is because that message dissemination among nodes is executed if the distance between nodes is less than $2l_{\max}$, changing $\alpha$ cannot change the scheduling performance.

Among all the algorithms, CLT presents the best single slot scheduling performance, and the gap between CLT and RCRDCLT is upper-bounded, as shown in Table III. Compared

TABLE III
THE APPROXIMATION FACTOR IN THEOREM 6

| $\alpha$ and $\epsilon$ / $\gamma_{th}$ | $\alpha=3$ | | $\alpha=4$ | | $\alpha=5$ | | $\alpha=6$ | |
|---|---|---|---|---|---|---|---|---|
| | $\epsilon=0.1$ | $\epsilon=0.05$ | $\epsilon=0.1$ | $\epsilon=0.05$ | $\epsilon=0.1$ | $\epsilon=0.05$ | $\epsilon=0.1$ | $\epsilon=0.05$ |
| $\gamma_{th}=1$, theoretical | 3.2226 | 3.0664 | 3.5547 | 3.3674 | 3.7858 | 3.5983 | 3.9500 | 3.7719 |
| $\gamma_{th}=1$, simulated | 1.1331 | 1.1149 | 1.2606 | 1.2212 | 1.3949 | 1.3477 | 1.4364 | 1.3805 |
| $\gamma_{th}=3$, theoretical | 3.0053 | 2.9178 | 3.2871 | 3.1618 | 3.5140 | 3.3763 | 3.6894 | 3.5510 |
| $\gamma_{th}=3$, simulated | 1.0976 | 1.0848 | 1.2234 | 1.1983 | 1.3112 | 1.3084 | 1.3732 | 1.3662 |
| $\gamma_{th}=5$, theoretical | 2.9399 | 2.8728 | 3.1949 | 3.0906 | 3.4135 | 3.2940 | 3.5889 | 3.4655 |
| $\gamma_{th}=5$, simulated | 1.0744 | 1.0586 | 1.1722 | 1.1628 | 1.3035 | 1.2607 | 1.3628 | 1.3355 |

TABLE IV
THE PARAMETER NEEDED $\beta$ IN EQUATION (5)

| $\alpha$ and $\epsilon$ / $\beta$ | $\alpha=3$ | | $\alpha=4$ | | $\alpha=5$ | | $\alpha=6$ | |
|---|---|---|---|---|---|---|---|---|
| | $\epsilon=0.1$ | $\epsilon=0.05$ | $\epsilon=0.1$ | $\epsilon=0.05$ | $\epsilon=0.1$ | $\epsilon=0.05$ | $\epsilon=0.1$ | $\epsilon=0.05$ |
| $\gamma_{th}=1$ | 7.7219 | 9.5447 | 4.8849 | 5.6509 | 3.8925 | 4.3404 | 3.3973 | 3.7029 |
| $\gamma_{th}=3$ | 10.6946 | 13.3236 | 6.1129 | 7.1209 | 4.6033 | 5.1613 | 3.8790 | 4.2460 |
| $\gamma_{th}=5$ | 12.4943 | 15.6112 | 6.8093 | 7.9547 | 4.9909 | 5.6089 | 4.1349 | 4.5345 |

TABLE V
THE MINIMUM DISTANCE $d_{\min}$ WITH $l_{\max}=8$ IN EQ. (7)

| $\alpha$ and $\epsilon$ / $d_{\min}$ | $\alpha=3$ | | $\alpha=4$ | | $\alpha=5$ | | $\alpha=6$ | |
|---|---|---|---|---|---|---|---|---|
| | $\epsilon=0.1$ | $\epsilon=0.05$ | $\epsilon=0.1$ | $\epsilon=0.05$ | $\epsilon=0.1$ | $\epsilon=0.05$ | $\epsilon=0.1$ | $\epsilon=0.05$ |
| $\gamma_{th}=1$ | 146.38 | 209.79 | 63.93 | 81.26 | 42.81 | 51.25 | 33.82 | 39.05 |
| $\gamma_{th}=3$ | 253.53 | 363.37 | 92.20 | 117.20 | 56.34 | 67.45 | 42.13 | 48.65 |
| $\gamma_{th}=5$ | 327.31 | 469.10 | 109.31 | 138.95 | 64.02 | 76.64 | 46.66 | 53.88 |

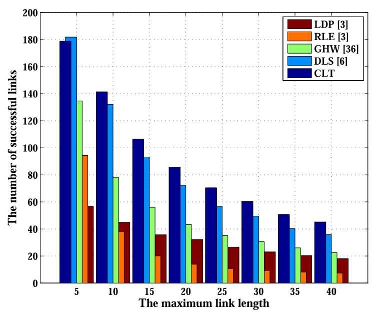



Fig. 9. Impact of $l_{\max}$ with $\epsilon=0.1$.

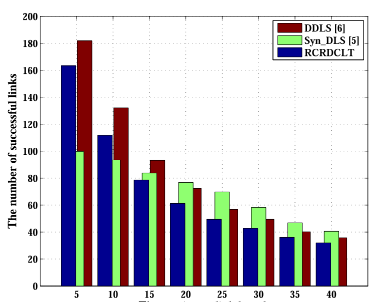



Fig. 10. Impact of $l_{\max}$ with $\epsilon=0.1$.

with LDP, GHW, and DLS, the single slot scheduling performances of CLT are improved about 151.6%, 84.2% and 18.8%, respectively, while performance losses of RCRDCLT are separately about 28.8% and 17.4% for $\alpha \geq 4$ when compared with Sys_DLS and DDLS. Moreover, the performance gaps between Syn_DLS and RCRDCLT gets smaller (for instance, RCRDCLT losses 10 links when $\alpha=6$ while losses 30 for $\alpha=4$), which further validates the effectiveness of $d_{\min}$ in Eq. (7), and RCRDCLT costs more less rounds.

The impact of maximum link length on algorithmic performance is shown in Fig. 9 and Fig. 10, all algorithm performances decrease when $l_{\max}$ increases. This is because that when a link is selected into scheduling set, it will reject more links since $d_{\min}$ becomes larger over $l_{\max}$ increasing for LDP, RLE, DLS (DDLS), and CLT (RCRDCLT). For GHW and Syn_DLS, the strength of received signal weakens as $l_{\max}$ increasing, resulting in less successful links. From Fig. 9 and Fig. 10, we can see that in all the algorithms, CLT shows the best average scheduling performance, and RCRDCLT cannot deal with the scenarios where communication distance between nodes are larger, it shows relatively good scheduling performance in shorter communication distance (such as 5 and 10). Compared with Syn_DLS and DDLS, performance losses of RCRDCLT are about 26% and 16.2% on the average, but it saves a lot of running time.

Fig. 11 and Fig. 12 validate the influence of $\epsilon$ with settings of $n=200$. Note that, on the one hand, increasing $\epsilon$ improves scheduling performances of LDP, RLE, DLS (or DDLS), CLT and RCRDCLT, the reason is that the size of each square (for LDP), $c_1 d_{ii}$ (for RLE), and $d_{\min}$ (for DLS or DDLS and CLT or RCRDCLT) decrease with $\epsilon$ increasing, as shown in Tables IV and V, respectively, resulting in more successful links. On the other hand, scheduling performances of Syn_DLS and GHW are not influenced by $\epsilon$, the reasons are similar to ones in Fig. 8.

### *B. MLS Validation in Cluster Networks*

In this subsection, using the simulated parameters in Table VI, we validate the scheduling performance for above algorithms in the cluster networks, as shown in Fig. 13-Fig. 20.

In Fig. 13 and 14, the scheduling performances of LDP, GHW, DLS (DDLS), CLT and RCRDCLT increase with the increment of $N_c$. Generally, the more the number of links in a cluster is, the more interference each link will suffer, which makes it harder to schedule links in same clusters concurrently. Different from all link scheduling algorithms

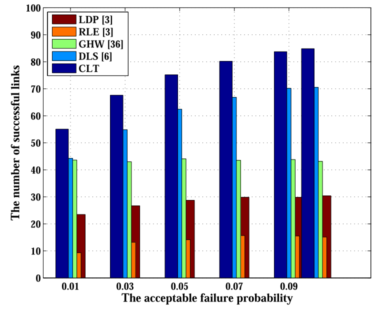



Fig. 11. Impact of $\epsilon$ with $n = 200$.

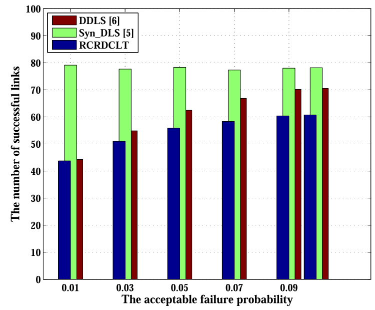



Fig. 12. Impact of $\epsilon$ with $n = 200$.

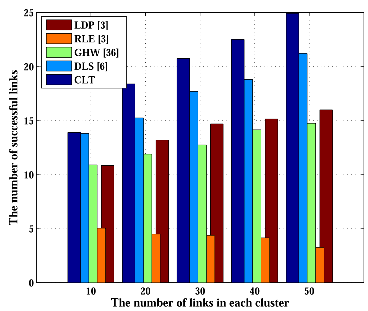



Fig. 13. Impact of $N_c$ with $\epsilon = 0.1$.

TABLE VI

SIMULATED PARAMETERS AND VALUES

| Symbol | Meanings | Value |
|---|---|---|
| $n_c$ | the number of clusters | 10 |
| $r_c$ | the radius of clusters | 20 |
| $N_c$ | the number of links in each cluster | 20 |
| $l_{\min}$ | the minimum link length | 5 |
| $l_{\max}$ | the maximum link length | 30 |
| $\alpha$ | the path-loss exponent | 5 |
| $\gamma_{th}$ | the decoding threshold | 1 |
| $\epsilon$ | the acceptable failure probability | 0.1 |

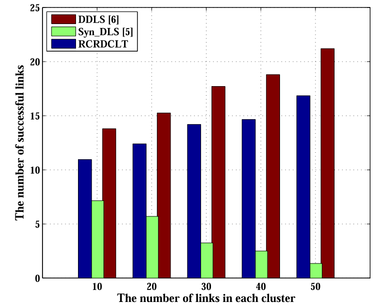



Fig. 14. Impact of $N_c$ with $\epsilon = 0.1$.

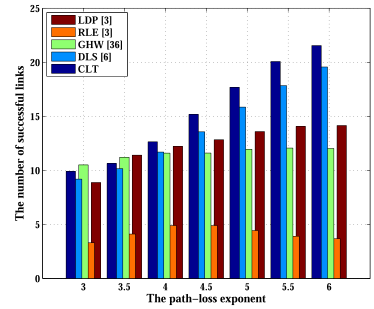



Fig. 15. Impact of $\alpha$ with $\epsilon = 0.1$.

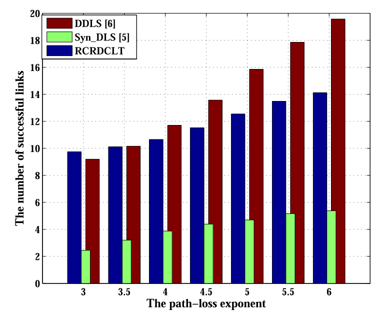



Fig. 16. Impact of $\alpha$ with $\epsilon = 0.1$.

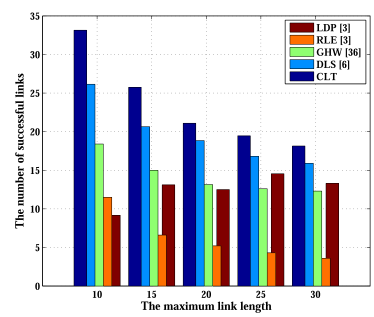



Fig. 17. Impact of $l_{\max}$ with $\epsilon = 0.1$.

cited in this paper, CLT can find at least two successful links in the same cluster on the average when $N_c \geq 30$, which shows that it can deal with dense cluster networks. Although RCRDCLT presents a relatively poor performance compared with DDLS, it costs more less time complexity and still schedules at least one successful link in each cluster averagely, since cluster radius of $r_c = 20$ separates clusters, RCRDCLT and DDLS can be able to take advantage of this property well, while Sys_DLS cannot, since Syn_DLS will reject those links within $2l_{\max} = 60$, the radius of cluster is 20. That is, the more links are in the cluster, the more links are rejected. Moreover, compared with DDLS, RCRDCLT only losses about 6 successful links at most in each schedule, while Syn_DLS cannot present a better scheduling performance in cluster networks.

In Fig. 21 and Fig. 22, we analyze the influence of the cluster radius with $n = 200$ and $l_{\max} = 10$, other parameters keep the same as before. In topologies with larger clusters, scheduling performance of CLT increases significantly than other algorithms. Generally, larger cluster radius separates links in the same cluster more easily, which makes it easier to schedule links in the same cluster concurrently. However, LDP, RLE, GHW and DLS cannot be able to take advantage of this property well. Additionally, in the dense networks (for

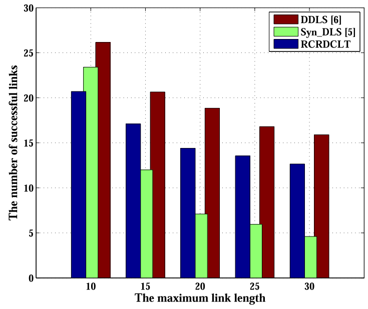



Fig. 18. Impact of $l_{\max}$ with $\epsilon = 0.1$.

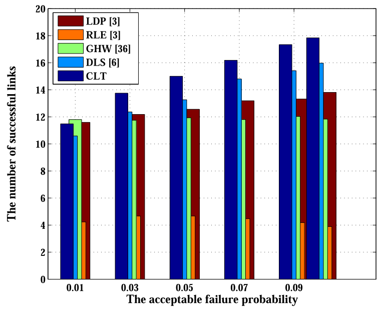



Fig. 19. Impact of $\epsilon$.

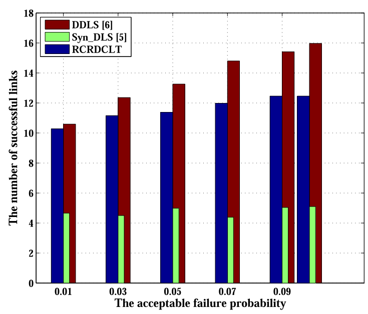



Fig. 20. Impact of $\epsilon$.

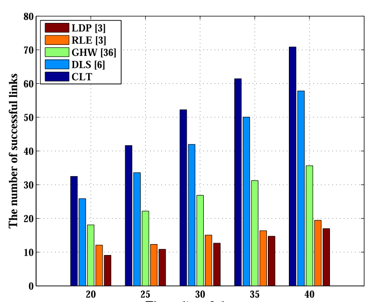



Fig. 21. Impact of $r_c$ with $\epsilon = 0.1$.

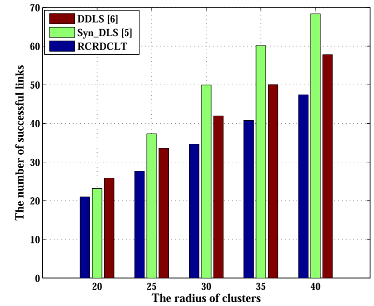



Fig. 22. Impact of $r_c$ with $\epsilon = 0.1$.

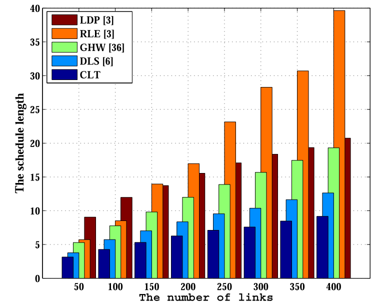



Fig. 23. Impact of $n$ with $\epsilon = 0.1$.

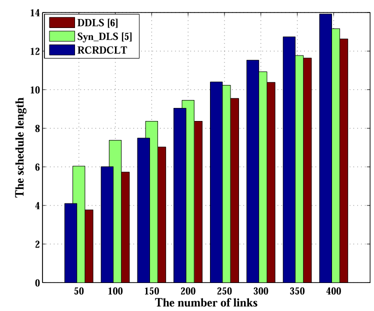



Fig. 24. Impact of $n$ with $\epsilon = 0.1$.

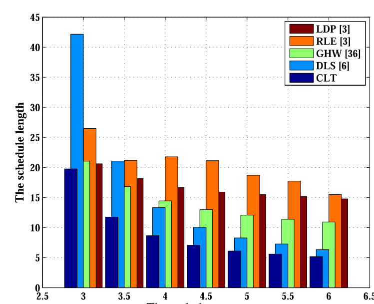



Fig. 25. Impact of $\alpha$ with $\epsilon = 0.1$.

example $r_c = 20$), three algorithms in Fig. 22 present similar performance, Syn_DLS outperforms than other two ones over $r_c$ increasing. The reason is as the same as described as above, i.e., larger cluster radius makes the distance among links in the same cluster larger.

### C. SLS Validation in Random Networks

In Fig. 23-Fig. 30, simulated parameters are set according to Table II, we consider the impact of network parameters on the schedule length in random networks.

In Fig. 23 and Fig. 24, over $n$ increasing, the needed schedule length increases for all algorithms. As could be expected, LDP, RLE, GHW and DLS cannot compare with CLT, while the average performance of RCRDCLT is still better than that of Syn_DLS, and only increases 7% than that of DDLS. On the average, LDP, GHW and DLS compute 2.8, 2.1 and 1.7 times longer schedule lengths than CLT, since it can compute a larger scheduling set, as shown in Fig. 5 and Fig. 6.

As shown in Fig. 25 and Fig. 26, over $\alpha$ increasing, all algorithms present more better performance besides Syn_DLS. The reasons are similar to those in Fig. 7 and Fig. 8, namely a larger $\alpha$ results in a smaller distance between the interferers and the receiver, more links can be scheduled in a time slot and then scheduling all given links needs less slots. When

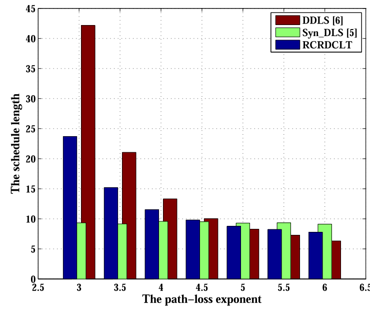



Fig. 26. Impact of $\alpha$ with $\epsilon = 0.1$.

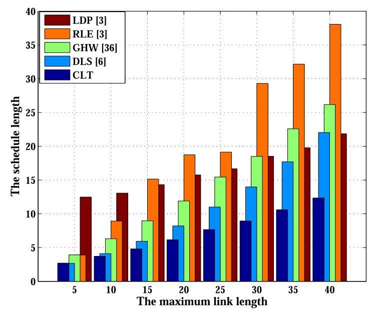



Fig. 27. Impact of $l_{\max}$ with $\epsilon = 0.1$.

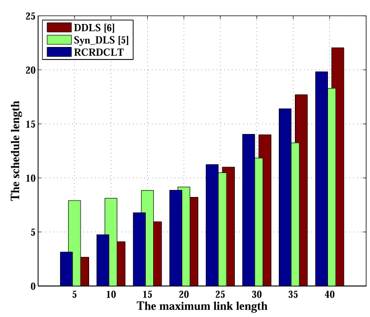



Fig. 28. Impact of $l_{\max}$ with $\epsilon = 0.1$.

$\alpha \geq 4$, on the average, there's hardly no difference between RCRDCLT and DDLS (differ by 1%), and they need a shorter schedule length than Syn_DLS.

The impacts of the maximum link length on schedule length are given in Fig. 27 and Fig. 28. Compared with CLT, averages of 2.5, 2.1 and 1.6 times schedule length are given by LDP, GHW and DLS for different $l_{\max}$, respectively, since they have lower scheduling performance, as shown in Fig. 9 and Fig. 10. Moreover, Syn_DLS, DDLS and RCRDCLT are particularly affected by $l_{\max}$ compared with other parameters, and RCRDCLT presents the best scheduling performance from the perspective of the average.

In Fig. 30, RCRDCLT and Syn_DLS show a similar averaged performance in the schedule length (differ by 8%), and obtain a shorter schedule times than DDLS. In addition, CLT is the fastest algorithm to schedule all given links, as shown in Fig. 29.

### *D. SLS in Cluster Networks*

In this subsection, simulated parameters are set to those in Table VI, from Fig. 22, Fig. 31 and Fig. 32, we can see that CLT obtains the shortest schedule length among all algorithms, and Syn_DLS cannot deal with cluster networks well, since it schedules more successful links in a time slot

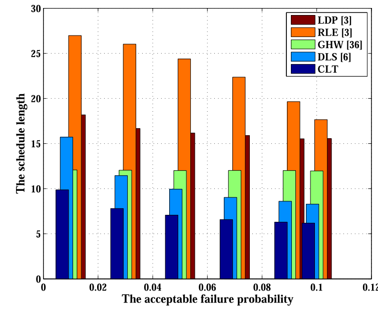



Fig. 29. Impact of $\epsilon$.

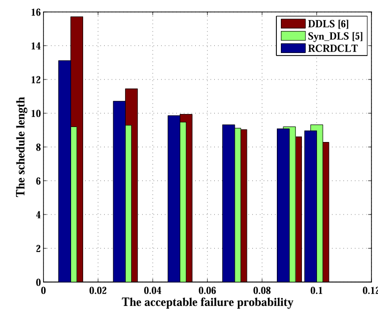



Fig. 30. Impact of $\epsilon$.

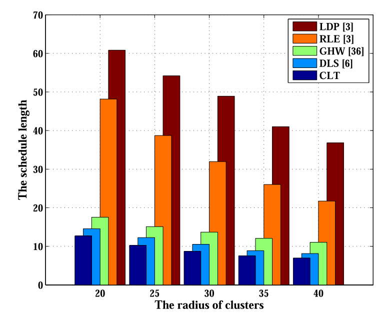



Fig. 31. Impact of $r_c$ with $\epsilon = 0.1$.

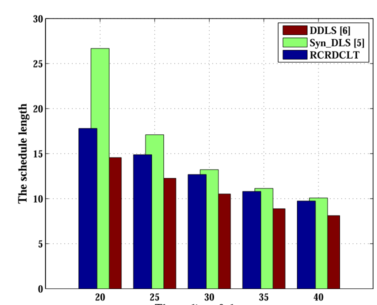



Fig. 32. Impact of $r_c$ with $\epsilon = 0.1$.

than DDLS and RCRDCLT, but it needs more time slots to schedule all links. Additionally, performance difference in schedule length between RCRDCLT and DDLS gets smaller as $r_c$ increases. On the average, RCRDCLT computes only 1.27 times schedule length than DDLS. The impacts of other parameters on the SLS in cluster networks are similar to those in random networks.

### *E. Comparison of LDP and ILDP*

In the end of this section, we compared LDP and ILDP in cluster networks for different $N_c$ and $\alpha$, respectively, as shown in Fig. 33 and Fig. 34. We can see that scheduling

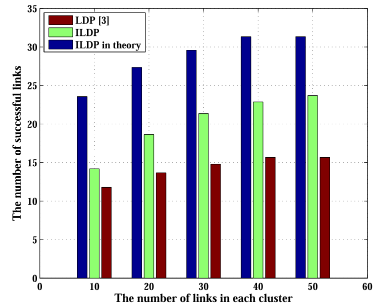



Fig. 33. Impact of $N_c$ with $\epsilon = 0.1$.

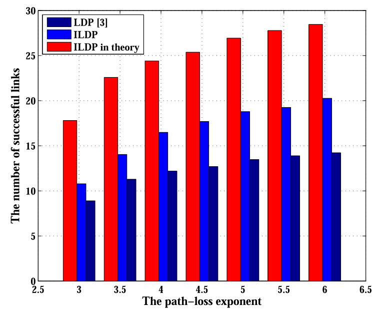



Fig. 34. Impact of $\alpha$ with $\epsilon = 0.1$.

performances of LDP and ILDP increase as $N_c$ (or $\alpha$) increasing, and ILDP can obtain 35.4% and 43.4% enhancements for $N_c$ and $\alpha$, respectively. The reasons are similar to those in Fig. 13 and Fig. 15.

To sum up, the simulations demonstrate that CLT is the best scheduling performance for the MLS and SLS in the random and cluster networks. RCRDCLT, besides having lowest time complexity than Syn_DLS and DDLS, only losses a constant approximation factor compared with the optimal schedule. Moreover, compared with Syn_DLS and DDLS from the perspective of the average, RCRDCLT presents advantages in some specific scenarios, such as high path-loss and longer communication distance. If we apply the idea of DDLS to design distributed CLT, the scheduling performance obtained will be better than Syn_DLS and DDLS, but the time complexity is also increased from $O(\ln n)$ to $O(n \ln n)$.

## VI. Conclusion

In this paper, we proposed two efficient algorithms for the problems of MLS and SLS. To begin with, by constructing the relationships between the acceptable failure probability and the needed SINR constraint in the Rayleigh fading model, we can design link scheduling algorithms under the deterministic SINR model rather than in the Rayleigh fading model directly. Then, we utilized the idea of interference localization to design centralized and distributed algorithms for the MLS, and the proposed algorithms not only localize the global interference, but also costs a logarithmic time complexity. Executing them repeatedly obtained a logarithmic approximation guarantee for the SLS with rigorous analysis. Finally, compared with current popular link scheduling algorithms, simulations validated the efficiency of our algorithms. Only uniform power assignment was considered, hence one future direction is to consider other methods of power control. Another promising research direction is the dynamic network scenarios where nodes move randomly to broadcast or collect data.

## Acknowledgment

The authors would like to thank the anonymous reviewers for their extremely useful comments and suggestions.

## References

[1] Y. Xiao, Y. Jia, C. Liu, X. Cheng, J. Yu, and W. Lv, “Edge computing security: State of the art and challenges,” *Proc. IEEE*, vol. 107, no. 8, pp. 1608–1631, Aug. 2019.

[2] W. Ye and A. M. Haimovich, “Performance of cellular CDMA with cell site antenna arrays, Rayleigh fading, and power control error,” *IEEE Trans. Commun.*, vol. 48, no. 7, pp. 1151–1159, Jul. 2000.

[3] C. Qiu and H. Shen, “Fading-resistant link scheduling in wireless networks,” in *Proc. ICPP*, 2017, pp. 312–321.

[4] J. Dams, M. Hoefer, and T. Kesselheim, “Scheduling in wireless networks with Rayleigh-fading interference,” *IEEE Trans. Mobile Comput.*, vol. 14, no. 7, pp. 1503–1514, Jul. 2015.

[5] K. Yu, Y. Wang, J. Yu, D. Yu, X. Cheng, and Z. Shan, “Localized and distributed link scheduling algorithms in IoT under Rayleigh fading,” *Comput. Netw.*, vol. 151, pp. 232–244, Mar. 2019.

[6] J. Yu *et al.*, “Efficient link scheduling in wireless networks under Rayleigh-fading and multiuser interference,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5621–5634, Aug. 2020.

[7] G. Sharma, N. Shroff, and R. Mazumdar, “On the complexity of scheduling in wireless networks,” in *Proc. ACM Mobicom*, 2006, pp. 227–238.

[8] X. Lin and N. B. Shroff, “The impact of imperfect scheduling on cross-layer congestion control in wireless networks,” *IEEE/ACM Trans. Netw.*, vol. 14, no. 2, pp. 302–315, Apr. 2006.

[9] W. Wang, Y. Wang, X. Li, W. Song, and O. Frieder, “Efficient interference aware tdma link scheduling for static wireless networks,” in *Proc. ACM Mobicom*, 2006, pp. 262–273.

[10] A. Krifa, C. Barakat, and T. Spyropoulos, “An optimal joint scheduling and drop policy for delay tolerant networks,” in *Proc. IEEE WoWMoM*, Jun. 2008, pp. 1–6.

[11] C. Joo, X. Lin, and N. B. Shroff, “Understanding the capacity region of the greedy maximal scheduling algorithm in multihop wireless networks,” *IEEE/ACM Trans. Netw.*, vol. 17, no. 4, pp. 1132–1145, Aug. 2009.

[12] L. Jiang, D. Shah, J. Shin, and J. Walrand, “Distributed random access algorithm: Scheduling and congestion control,” *IEEE Trans. Inf. Theory*, vol. 56, no. 12, pp. 6182–6207, Dec. 2010.

[13] K. Kar, S. Sarkar, A. Ghavami, and X. Luo, “Delay guarantees for throughput-optimal wireless link scheduling,” *IEEE Trans. Autom. Control*, vol. 57, no. 11, pp. 2906–2911, Nov. 2012.

[14] X. Wang, Z. Li, and J. Wu, “Joint TCP congestion control and CSMA scheduling without message passing,” *IEEE Trans. Wireless Commun.*, vol. 12, no. 12, pp. 6194–6204, Dec. 2013.

[15] T. Moscibroda and R. Wattenhofer, “The complexity of connectivity in wireless networks,” in *Proc. IEEE INFOCOM*, Apr. 2006, pp. 1–13.

[16] O. Goussevskaia, Y. Osfwald, and R. Wattenhofer, “Complexity in geometric SINR,” in *Proc. ACM Mobihoc*, 2007, pp. 1872–1880.

[17] B. Huang, J. Yu, D. Yu, and C. Ma, “SINR based maximum link scheduling with uniform power in wireless sensor networks,” *KSII Trans. Internet Inf. Syst.*, vol. 8, no. 11, pp. 4050–4067, 2014.

[18] J. Yu, B. Huang, X. Cheng, and M. Atiquzzaman, “Shortest link scheduling algorithms in wireless networks under the SINR model,” *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2643–2657, Mar. 2017.

[19] M. Halldörsson, “Wireless scheduling with power control,” in *Proc. ESA*, 2009, pp. 361–372.

[20] A. Fanghänel, T. Kesselheim, H. Räcke, and B. Vöcking, “An improved spine-based infrastructure for routing in ad hoc networks,” in *Proc. ACM PODC*, 2009, pp. 220–229.

[21] A. Fanghänel, T. Kesselheim, and B. Vöcking, “Improved algorithms for latency minimization in wireless networks,” in *Proc. ICALP*, 2009, pp. 447–458.

[22] Y. Zhou, X.-Y. Li, M. Liu, X. Mao, S. Tang, and Z. Li, “Throughput optimizing localized link scheduling for multihop wireless networks under physical interference model,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 10, pp. 2708–2720, Oct. 2014.

[23] L. Le, E. Modiano, C. Joo, and N. Shroff, “Longest-queue-first scheduling under SINR interference model,” in *Proc. ACM MobiCom*, 2010, pp. 41–50.

[24] L. Qu, J. He, and C. Assi, “Distributed link scheduling in wireless networks with interference cancellation capabilities,” in *Proc. WoWMoM*, Jun. 2014, pp. 1–7.

[25] D. Qian, D. Zheng, J. Zhang, N. Shroff, and C. Joo, "Distributed CSMA algorithms for link scheduling in multihop MIMO networks under SINR model," *IEEE/ACM Transmiss. Netw.*, vol. 21, no. 3, pp. 1503–1514, Aug. 2013.

[26] P. Chaporkar, K. Kar, X. Luo, and S. Sarkar, "Throughput and fairness guarantees through maximal scheduling in wireless networks," *IEEE Trans. Inf. Theory*, vol. 54, no. 2, pp. 572–594, Feb. 2008.

[27] S. P. Weber and J. G. Andrews, "Transmission capacity of wireless networks," *Found. Trends Netw.*, vol. 5, nos. 2–3, pp. 109–281, 2012.

[28] A. Fanghänel, S. Geulen, M. Hoefer, and B. Vöcking, "Online capacity maximization in wireless networks," *J. Scheduling*, vol. 16, no. 1, pp. 81–91, Feb. 2013.

[29] X. Xu, S. Tang, and P. Wan, "Maximum independent set of links under physical interference model," in *Proc. WASA*, 2010, pp. 68–74.

[30] B. Huang, J. Yu, X. Cheng, H. Chen, and H. Liu, "Scheduling in wireless networks with rayleigh-fading interference," *J. Netw. Comput. Appl.*, vol. 77, pp. 64–72, Jan. 2017.

[31] D. M. Blough, G. Resta, and P. Santi, "Approximation algorithms for wireless link scheduling with SINR-based interference," *IEEE/ACM Trans. Netw.*, vol. 18, no. 6, pp. 1701–1712, Dec. 2010.

[32] M. Dinitz, "Distributed algorithms for approximating wireless network capacity," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[33] E. I. Asgeirsson and P. Mitra, "On a game theoretic approach to capacity maximization in wireless networks," in *Proc. IEEE INFOCOM*, Apr. 2011, pp. 3029–3037.

[34] T. Kesselheim and B. Vöcking, "Distributed contention resolution in wireless networks," in *Proc. DISC*, 2010, pp. 163–178.

[35] M. M. Halldórsson and P. Mitra, "Nearly optimal bounds for distributed wireless scheduling in the SINR model," *Distrib. Comput.*, vol. 29, no. 2, pp. 77–88, Apr. 2016.

[36] O. Goussevskaia, M. Halldörsson, and R. Wattenhofer, "Algorithms for wireless capacity," *IEEE/ACM Trans. Netw.*, vol. 22, no. 3, pp. 745–755, May 2014.

[37] M. M. Halldórsson and T. Tonoyan, "Wireless link capacity under shadowing and fading," in *Proc. ACM MobiHoc*, Jul. 2017, pp. 1–10.

[38] H. Ding, F. Zhao, J. Tian, and H. Zhang, "Performance analysis of MISINR user association in 3-D heterogeneous cellular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4119–4129, Apr. 2020.

[39] O. Goussevskaia, R. Wattenhofer, M. M. Halldorsson, and E. Welzl, "Capacity of arbitrary wireless networks," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 1872–1880.

[40] A. Sankararaman and F. Baccelli, "CSMA k-SIC—A class of distributed MAC protocols and their performance evaluation," in *Proc. IEEE INFOCOM*, Apr. 2015, pp. 2002–2010.

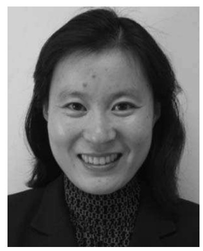
[41] S. Lv, W. Zhuang, X. Wang, and X. Zhou, "Scheduling in wireless ad hoc networks with successive interference cancellation," in *Proc. IEEE INFOCOM*, Apr. 2011, pp. 1287–1295.

[42] O. Goussevskaia, T. Moscibroda, and R. Wattenhofer, "Local broadcasting in the physical interference model," in *Proc. ACM DIALM-POMC*, 2008, pp. 35–44.

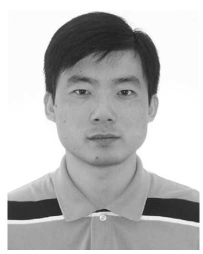
[43] I. Gradshteyn and I. Ryzhik, *Table of Integrals, Series, and Products*, A. Jeffrey and D. Zwillinger, Eds. Amsterdam, The Netherlands: Elsevier, 2007.


**Kan Yu** received the M.S. degree from the School of Information Science and Engineering, Qufu Normal University, Rizhao, China, in 2016, and the Ph.D. degree from the College of Computer Science and Engineering, Shandong University of Science and Technology, in 2019. He is currently a Lecturer with the School of Computer Science, Qufu Normal University. His main research interests include wireless networks, the IoT security, and distributed algorithm design and analysis.

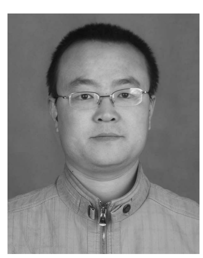
**Jiguo Yu** (Senior Member, IEEE) received the Ph.D. degree from the School of Mathematics, Shandong University, Jinan, China, in 2004. He became a Full Professor with the School of Computer Science, Qufu Normal University, Jining, China, in 2007. He is also a Full Professor with the Qilu University of Technology (Shandong Academy of Sciences), Jinan, and the Shandong Computer Science Center (National Supercomputer Center), Jinan. He is interested in designing and analyzing algorithms for many computationally hard problems in networks. His current research interests include privacy-aware computing, wireless networking, distributed algorithms, peer-to-peer computing, and graph theory. He is a member of the ACM and a Senior Member of the China Computer Federation (CCF).



**Xiuzhen Cheng** (Fellow, IEEE) received the M.S. and Ph.D. degrees in computer science from the University of Minnesota–Twin Cities in 2000 and 2002, respectively. She is currently a Professor with the School of Computer Science and Technology, Shandong University. Her current research interests include cyber physical systems, wireless and mobile computing, sensor networking, wireless and mobile security, and algorithm design and analysis. She is a member of the ACM. She received the NSF CAREER Award in 2004. She has served on the editorial boards of several technical journals and the technical program committees of various professional conferences/workshops. She has also chaired several international conferences. She worked as the Program Director of the U.S. National Science Foundation (NSF) from April 2006 to October 2006 (full time) and from April 2008 to May 2010 (part time).



**Dongxiao Yu** (Senior Member, IEEE) received the B.Sc. degree from the School of Mathematics, Shandong University, Jinan, China, in 2006, and the Ph.D. degree from the Department of Computer Science, The University of Hong Kong, Hong Kong, in 2014. He is currently a Professor with the School of Computer Science and Technology, Shandong University. His research interests include wireless networks and distributed computing.



**Anming Dong** (Member, IEEE) received the B.E. degree in electronic information science and technology from Liaocheng University, Liaocheng, China, in 2004, the M.E. degree in communications and information systems from Lanzhou University, Lanzhou, China, in 2007, and the Ph.D. degree in communications and information systems from Shandong University, Jinan, China, in 2016. He is currently an Associate Professor with the School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan. His research interests include MIMO techniques, wireless communications, wireless networks, and the application of optimization techniques for signal processing and wireless communications. He was a recipient of the Excellent Doctoral Dissertation Awards of Shandong Province, in 2017.