Shortest Uplink Scheduling for NOMA-Based Industrial Wireless Networks

Chaonong Xu, Mianze Wu, Yida Xu, and Yongjun Xu

Abstract—**The power-domain nonorthogonal multiple access (NOMA) based on successive interference cancellation (SIC) provides opportunities for fast media access in industrial wireless networks. In this article, given the traffic loads of wireless sensors, we study the shortest uplink scheduling (SUS) problem by joint power allocation and wireless sensor (WS) scheduling. A key term named maximum decoded level (MDL), which models the transmitting characteristics of WSs under SIC, and thus, lays the foundations for revealing a sufficient and necessary condition for successive transmissions under SIC, is presented in the first step. Then, guided by the theoretical condition that decouples WS scheduling from power allocation, we present a two-step greedy algorithm for the SUS problem in the case of continuous transmit powers. We also prove that the proposed algorithm is optimal for two regular cases. One is for any traffic loads under 2-SIC, the other is for unit traffic load under k-SIC. Furthermore, in the case of discrete transmit powers, we further propose an optimal algorithm under 2-SIC and an approximation algorithm under k-SIC by adapting the above-mentioned greedy algorithm for the case of discrete transmit powers. Experimental evaluations reveal the effectiveness of the three algorithms.**

Index Terms—**Nonorthogonal multiple access (NOMA), successive interference cancellation, uplink, schedule, media access.**

I. Introduction

In RECENT years, wireless networks are playing more and more important roles in industries. Distinct from the cellular networks such as long term evolution-advanced, where downlinks carry more traffics than uplinks, in industrial wireless networks (IWNs), a sink usually collects sensory data from wireless sensors (WSs), thus, the performances of uplinks are vital for IWNs [1], [2].

Since the real-time performance with guaranteed delay is often required in IWNs [3], the problem of the shortest uplink scheduling (SUS), i.e., how to minimize the length of the uplink frame with given traffic loads, has to be tackled. Relative to the random media access, the classic time division multiple access (TDMA) technique has the advantage of bounded access time, which is the product of the slot span and the frame length. However, the uplink frame length could be terribly large especially for the high-density heavy-load IWNs, which will be common with further development of Internet of Things. Therefore, better solutions have to be sought.

Power-domain nonorthogonal multiple access (NOMA) is one of the candidate solutions to the next-generation IWNs. Successive interference cancellation (SIC), which is the one important implementation method of the power-domain NOMA.[1] Nowadays, supports parallel transmissions from multiple transmitters by multiplexing in the power domains [4] and, thus, has great potentialities for low-delay applications. Therefore, the problem of finding the SUS strategy if SIC-based sink is adapted in IWNs, has to be solved.

We solve the problem by joint power allocation and WS scheduling. On one hand, WS scheduling determines how to group the WSs, since WSs in a group will transmit simultaneously, and thus, they will interfere with each other. On the other hand, the power allocation sets reasonable transmit powers for WSs, so that the transmitted symbols from the WSs in a group can be decoded by an SIC-based sink without errors.

We first investigate the SUS problem,[2] when transmit powers of WSs are continuously adjustable. First, a key term named maximum decoded level (MDL), which models the transmitting characteristics of WSs under SIC, is defined. Based on MDL, an important characteristic, i.e., the so-called power exclusiveness, is revealed, which lays theoretical foundations for a sufficient and necessary condition for successful transmissions under SIC. The sufficient and necessary condition directly results in the decoupling between WS scheduling and power allocation, which is the key outcome of the first step. Based on the outcome, we present a two-step greedy algorithm for the SUS problem. We also prove that the algorithm is optimal for two most regular cases as follows. One is for any traffic loads under 2-SIC, the other is for unit traffic load under k-SIC. Besides, an explicit analytic expression of the optimal solution is also presented for the both cases, respectively.

In view of the above-mentioned results obtained, we further investigate the same problem, however, with discrete transmit

Manuscript received August 20, 2019; revised December 17, 2019; accepted January 25, 2020. Date of publication February 26, 2020; date of current version November 24, 2020. This work was supported by the National Natural Science Foundation of China under Grant 61702487. *(Corresponding author: Chaonong Xu.)*

C. Xu and M. Wu are with the Beijing Key Lab of Petroleum Data Mining, China University of Petroleum, Beijing 102249, China (e-mail: xuchaonong@cup.edu.cn; w1369834141@qq.com).

Y. Xu and Y. Xu are with the Institute of Computing Technology, China Academy of Sciences, Beijing 100190, China (e-mail: angel_adam@sina.com; xyj@ict.ac.cn).

Digital Object Identifier 10.1109/JSYST.2020.2971499

[1] An inherited flaw of power-domain NOMA is its high-power consumption, and therefore, it is suitable for some industrial applications, which requires low access delays and no stringent power constraints [3]. Besides, extra decoding delay due to the successive decoding process is also inevitable.

[2] Although the scheduling-based policies are beneficial for enhancing the network performance in general cases, it cannot provide real-time processing for burst events, such as the retransmissions.

powers.[3] We also propose an optimal algorithm under 2-SIC, and a heuristic algorithm under k-SIC, respectively, based on the greedy algorithm proposed in the section of continuous transmit powers.

Our major contributions are summarized as follows.

1) As to the SUS problem for SIC-based IWNs with given traffic loads, we formulate it by joint power allocation and WS scheduling.
2) We define MDL and then reveal a sufficient and necessary condition for successful parallel transmissions under SIC, which decouples the WS scheduling from the power allocation. What is more, since the so-called power exclusiveness perfectly depicts the decoding feature of SIC, we believe that it can also be utilized in other SIC-related problems.
3) For general cases, a greedy algorithm is proposed. However, it is also optimal for two regular cases.
4) For the case of discrete transmit powers, we also propose an optimal algorithm under 2-SIC and a heuristic algorithm under k-SIC.

The remainder of this article is organized as follows. Section II reviews the related works, and Section III introduces the system models. Problem formulation and solutions are introduced and analyzed in Section IV. Based on the conclusions drawn in Section IV, the same problem with discrete transmit powers is considered in Section V. Performance evaluations are in Section VI, and Section VII concludes this article.

II. Related Works

NOMA schemes, which are categorized into power-domain NOMA and code-domain NOMA, can be used in the scenarios of single-antenna and multiple-input–multiple-output [5], [6]. Scheduling for performances enhancement is a classic topic in network research works. Nowadays, Scheduling for performances enhancement based on NOMA has attracted great attention in both industry and academia. For example, the power-domain NOMA, which is based on SIC receivers, is now under full consideration for industrial applications or heterogeneous cellular networks [7]. The classic maximum weight schedule has been proven to be the maximum throughput schedule under the primary interference model [8]. As for the minimum length schedule problem, related works can be differentiated from three aspects. The first is the underlying interference models including the protocol interference model and the physical interference model. The second is the network scenario including the single-hop and the ad hoc networks, and the third is the transmit rate and power models adopted, including the signal-interference-plus-noise ratio (SINR) based and the fixed-value-based. For example, the work in[9] is under the physical interference model, for the ad-hoc networks, using the SINR-based rate model and the continuously adjustable transmit power model. Similar

[3]Generally, the transmit powers are not continuously adjustable. Take TI CC1000 transceiver for example. The transmit power is programmable, and there are 30 programmable settings for output levels in the step of 1 dBm. We have a research on the cases of continuous transmit powers because it lays theoretic foundations for the case of discrete transmit powers.

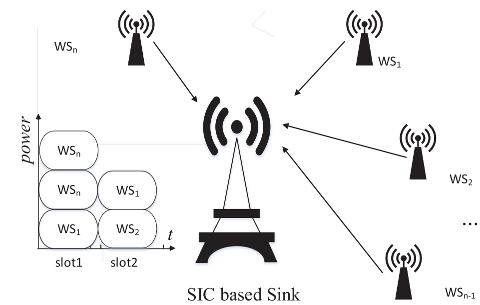


Fig. 1. Uplink transmissions with SIC-based sink.

works on the shortest length schedule include [10]–[12], and etc. We recommend [13] for an overview of the shortest length schedule.

As to the shortest scheduling for SIC, the works in [14] and [15] consider the SUS problem for multiple iterative-SIC based receivers. In [16], the authors also study the shortest scheduling problem given the traffic demands in a SIC-based single-hop wireless network. They modeled the transmission rate using the classic Shannon formula, and formulated the problem as a linear programming problem. They further have a study of the similar problem for the ad-hoc networks [17]. Relatively, in this article, we, under k-SIC, consider a typical single-hop scenario of IWNs with a fixed transmit rate model. Besides, some fast algorithms for computing the optimal scheduling strategy are also proposed in this article.

The complexity of the SUS problem under SIC has to be found because it provides a scientific guide for designing fast algorithms for SUS. In [18], the authors have proven that the SUS problem for the network with multiple SIC-based sinks is NP-hard. We further proved that even if the multiple sinks can exchange decoded information with each other, the SUS problem in the scenario is still NP-hard [15]. We confess that although the NP hardness of the SUS problem is not proven in this article, the SUS problems for two regular cases are proven to be polynomial in this article.

III. System Model

As shown in Fig. 1, we consider a single-hop, single-channel wireless network[4] consisting of n single-antenna WSs,[5] and a single-antenna sink. The WSs set is denoted by $\{WS_1, WS_2, \ldots, WS_n\}$. The sink is equipped with a k-SIC receiver. As a typical uncoded multiple-user detector [19], a k-SIC receiver can decode at most k signals at one time, provided that the SINR of every signal after interference cancellations is beyond the decoding threshold of the receiver [15]. By collecting

[4]For a multihop network, any of its multihop delivery consists of multiple single-hop forwarding processes. The challenge to the single-hop forwarding is the wireless collisions, which may be caused by the unplanned transmissions from its one-hop neighbors. Logic link control (LLC) protocols are designed for dealing with the problem. In other words, multiple single-hop networks have to be involved into the multihop delivery. In this article, we focus on LLC protocols, so a single-hop network model suffices our research intention.

[5]In this article, WS, user, and transmitter are used interchangeably, and receiver is equivalent to sink.

TABLE I
NOTATIONS

WS_i	wireless sensor i
p_{ij}	transmit power of WS_i in slot j
p_i^{max}	the maximal transmit power of WS_i
G_i	channel gain of WS_i
γ	decoding threshold
n_0	power of noise
n_i	the number of type-i WSs
$(\widehat{X}_1, \widehat{X}_2, \cdots, \widehat{X}_r)$	power threshold sequence for r-SIC
tp_i	discrete transmit power of WS_i
$\overline{TP} = \{\overline{tp}_m, \overline{tp}_{m-1}, \cdots, \overline{tp}_1\}$	feasible discrete transmit powers set
$[\![x]\!]$	$\underset{y \in \overline{TP} \bigcap (y \geq x)}{\arg\min} \quad (y-x), (x \geq 0)$

the traffic loads information of all WSs,[6] the sink determines the scheduling strategy and controls the scheduling process.

In the considered network, time is divided into frames, and a frame time is divided into multiple time slots. Without loss of generality (w.l.o.g.), the time span of a slot is set for delivering one data packet. In fact, the above-mentioned assumptions are tenable in reality. The traffic loads of WSs are denoted by $L_1, L_2, \ldots, L_n$, respectively, where L_i is the number of data packets to be transmitted by WS_i in the upcoming frame. In other words, WS_i will transmit L_i times in the next upcoming frame.

We use a parameter G_i to capture the loss of signal power, as the signal propagates through wireless channel from WS_i to the sink.[7] Besides, we also assume that channel gains of all WSs keep constant during a frame time, which is realistic for IWNs since WSs in IWNs are generally stationary.

The received power of WS_i at the sink is modeled as the product of its channel gain and its transmit power. The signal of WS_i can be decoded correctly at the sink only if its SINR $\geq \gamma$, where γ is the decoding threshold and $\gamma > 1$.[8] We require that $\frac{G_i P_i^{\max}}{n_0} \geq \gamma$ holds for all $i \in [1, n]$, where $P_i^{\max}$ is maximal transmit power of WS_i and n_0 is noise power, such that every WS can communicate directly with the sink using its maximal transmit power if there is no interference.

We assume that the perfect k-SIC is used, i.e., the residual error after interference cancellation is zero, which has been widely adopted. For clarity, we also explain the meaning of the decoding phase here. For the multiple WSs that are decoded in the same slot at the sink, if some WS, say WS_i, is decoded in the last, it is said to be decoded in the decoding phase 1. The WS, which is decoded just before WS_i, is said to be decoded in the decoding phase 2, and so on and so forth. Obviously, for k-SIC decoder, the highest decoding phase is k.

[6]At the beginning of a frame, these WSs that have transmission tasks will report their traffic loads to the sink via control channel.

[7]In this article, all scheduling strategies are computed with the given information of channel gains, which can be estimated by regular channel estimation algorithms. Although the value of channel gains are different for wireless signals with different frequencies, the design of the scheduling algorithms is not affected by the value of channel gain.

[8]γ depends upon which type of SIC receiver, modulation, and coding scheme.

IV. SUS FOR k-SIC

The problem of SUS in SIC-based IWNs is defined as follows.

Definition 1 (SUS in k-SIC (SUS-kSIC) Problem): Given a k-SIC sink and n WSs $\{WS_1, WS_2, \ldots, WS_n\}$ with their channel gains to the sink being $G_1, G_2, \ldots, G_n$,[9] and their traffic loads being $L_1, L_2, \ldots, L_n$, respectively, configure transmit power of every WS in every slot, such that the frame length is minimized under the following constraints.

1) All traffic loads of every WS are delivered successfully in the frame.
2) Every WS is allowed to be scheduled at most once in a slot, and multiple times in a frame.
3) SINR for decoding every packet is above the decoding threshold γ.
4) The transmit power of any WS must be no larger than its maximal power bound.

The problem is, thus, formulated as follows:

$$\min_{\{p_{ij}, N_{ij}\}} t \tag{1a}$$

$$\text{s.t.} \quad 0 \leq \sum_{i=1}^{n} N_{ij} \leq k \quad \forall j \in [1, t] \tag{1b}$$

$$\frac{G_i p_{ij}}{I_{ij} + n_0} \geq \gamma N_{ij} \quad \forall i \in [1, n] \quad \forall j \in [1, t] \tag{1c}$$

$$\sum_{j=1}^{t} N_{ij} = L_i \quad \forall i \in [1, n] \tag{1d}$$

$$N_{ij} = \begin{cases} 1, & \text{if } \mathrm{WS}_i \text{ is scheduled in } j\text{th slot} \\ 0, & \text{else} \end{cases} \tag{1e}$$

$$0 \leq p_{ij} \leq p_i^{\max} \quad \forall i \in [1, n] \quad \forall j \in [1, t] \tag{1f}$$

where I_{ij} is the power of interference when the signal of WS_i is decoded in the jth slot. t is the frame length, and N_{ij} is the WS scheduling strategy. Equation (1b) is to reveal that there are at most k decoding phases in a slot. Equation (1c) is for guaranteeing the success of decoding. Equation (1d) is for fulfilling traffic loads requirement, and (1f) reveals that the transmit power should be no larger than the power ceiling.

From formulation (1), SUS-kSIC is obviously a joint optimization of power allocation and WS scheduling. Actually, for SUS-kSIC problem, the power allocation and the WS scheduling are independent, which means that SUS-kSIC is virtually a two-stage instead of a joint optimization problem, and thus, low-complexity algorithms can be expected. To show the independence, we first reveal a sufficient and necessary condition for successful parallel transmissions under SIC,[10] and then show that there is always an eligible power allocation strategy for any WS scheduling strategy satisfying the condition. In other

[9]w.l.o.g., we assume $G_1 \leq G_2 \leq \cdots \leq G_n$.

[10]Which is in fact Lemma 2 in this section.

words, to find the optimal strategy, we only need to find the WS scheduling strategy, which achieves the shortest length without taking power allocation into considerations.

A. Maximum Decodable Level

Intuitively, for a WS that has smaller channel gain and smaller maximal transmit power ceiling, its scheduling flexibility is obviously weaker in SIC decoding schemes. To find the optimal solution to SUS-kSIC, the scheduling flexibility of WS under SIC has to be modeled mathematically. Evidently, under SIC, the transmit power ceiling, the channel gain, and the decoding threshold jointly affect the scheduling flexibility of WSs. The term MDL is set up to model the scheduling flexibility of WSs under SIC.

Definition 2: Power Threshold Sequence for r-SIC (PTS-r) is a sequence $\widehat{X} = (\widehat{X}_1, \widehat{X}_2, \ldots, \widehat{X}_r)$, which satisfies the following equality group:

$$\begin{cases} \dfrac{\widehat{X}_j}{\sum_{i=1}^{j-1} \widehat{X}_i + n_0} = \gamma \quad \forall\, j \in [2, r] \\ \dfrac{\widehat{X}_1}{n_0} = \gamma \end{cases}$$

where $\widehat{X}_j > 0$ for all $j \geq 1$ and $\gamma > 1$.

Obviously, PTS-r is a geometric sequence. An explicit formula for PTS-r is as follows: $\widehat{X}_1 = \gamma n_0, \widehat{X}_{i+1} = (\gamma+1)\widehat{X}_i$ for $\forall i \in [1, r-1]$. PTS-r is in fact the minimum received powers required for r signals if the r signals are to be successfully decoded by a k-SIC receiver where $k \geq r$.

Theorem 1: For the following inequality group:

$$\begin{cases} \dfrac{x_l}{\sum_{i=1}^{l-1} x_i + n_0} \geq \gamma \quad \forall\, l \in [2, r] \\ \dfrac{x_1}{n_0} \geq \gamma \end{cases} \tag{2}$$

any of its solution $(\widetilde{X}_1, \widetilde{X}_2, \ldots, \widetilde{X}_r)^T$ satisfies $\widetilde{X}_i \geq \widehat{X}_i$ for $\forall i \in [1, r]$, where $\widehat{X} = (\widehat{X}_1, \widehat{X}_2, \ldots, \widehat{X}_r)^T$ is PTS-r.

Proof: It is easy to prove using mathematical induction. Please refer to Appendix A. ■

Definition 3 (Maximum Decodable Level): For WS_i, whose channel gain to the sink is G_i and its transmit power ceiling is $p_i^{\max}$, if there exists an integer l, such that $\widehat{X}_l \leq p_i^{\max} G_i \leq \widehat{X}_{l+1}$, the MDL of WS_i is l.

Intuitively, MDL models the interference tolerance capability of WSs. For example, for the WS with MDL=1, it can only be decoded in the first decoding phase, because it has very weak immunity from interferences. Obviously, the larger is its MDL of a WS, the more decoding phases it can choose to be decoded in.

Lemma 1 (Power Exclusiveness of Parallel WSs for k-SIC): Given PTS-k being $(\widehat{X}_1, \widehat{X}_2, \ldots, \widehat{X}_k)$, provided that the following two prerequisites are satisfied, the number of parallel WSs, i.e., WSs which transmits simultaneously, is at most l.

1) Transmissions from different WSs can be decoded simultaneously by the k-SIC based sink.
2) The maximum of all received powers lies in $[\widehat{X}_l, \widehat{X}_{l+1}]$, where $l \leq k$.

Proof: Please refer to Appendix B. ■

Lemma 1 is for revealing the following two characteristics of the optimal solution from the perspective of MDL.

Lemma 2 (Condition for Successful Parallel Transmissions Under SIC): Assume that w packets from w WSs transmit simultaneously where $w \leq k$. All of these w packets can be decoded correctly by a k-SIC-based sink, if and only if the MDL of the WS, which is decoded in decoding phase i, must be no less than i for any $i \in [1, w]$.

Proof: Please refer to Appendix C. ■

B. Greedy Algorithm for SUS-kSIC

Based on the above-mentioned conclusion, we only need to focus on the WS scheduling strategy that achieves the minimum scheduling length. Algorithm 1 is a greedy algorithm based on Lemma 2.

The WS scheduling strategy is generated slot by slot. In determining the scheduling strategy for every slot, there are three stages including the anchoring, the upper phase allocation and the lower phase allocation. In the anchoring stage, the WS with the heaviest loads is selected and assigned to the anchoring phase equal to its MDL. In the upper phase allocation stage, some WSs are selected and assigned to the upper phases. Similar process is done for the lower phases in the lower phase allocation stage.

Lines 1 and 3 are for initialization. Lines 4–8 is the so-called anchoring process, where we compare the traffic load of the unscheduled type-1 WS ensemble with that of every other WS, and choose one WS, i.e., $usr1$ in the algorithm, based on the criterion that the WS with the heaviest traffic load is preferred.[11] In line 7, a decoding phase, i.e., phs_init in the algorithm, is reserved for the chosen WS, i.e., $usr1$, based on its MDL value. The so-called upper phases assignment is from lines 9–11, where we choose the WS, i.e., $usr2$ in the algorithm, for the decoding phase larger than phs_idx. The process goes on until no suitable WS is found. Similar process, i.e., the lower phases assignment, is from lines 12–15, where we choose the WS, i.e., $usr3$ in the algorithm, for the decoding phases less than phs_idx.[12]

After Algorithm 1, we set powers for WSs based on their phases allocated. For u_i, if its phase allocated is j, its transmit power is set as $\widehat{X}_j / G_i$. The correctness of the power allocation strategy is guaranteed by Lemma 2.

The following example under 4-SIC is presented for an overview of the algorithm. There are five WSs, u_1 to u_5, whose MDLs and traffic loads are shown in the left upper corner of Fig. 2. For the first slot, u_2 is selected as $usr1$ and reserved the decoding phase 2 since its MDL is 2. Next, we choose WSs for the phases larger than 2, i.e., phases 3 and 4. For phase 3, u_5 is chosen based on the criterion depicted by line 10. For phase 4, we cannot find an eligible WS because MDLs of the remaining WSs are all less than 4. Furthermore, we choose a WS for phases less than 2, i.e., phase 1. u_3 is, thus, chosen based on

[11]The reason for selecting the WS that has the heaviest unscheduled load is to balance the unscheduled traffic loads among the type-1 WSs ensemble and every other WS. In that way, smaller frame length will be resulted in.

[12]The reason for arranging decoding phases larger than phs_init prior to these less than phs_init is to stuff as many WSs in a slot as possible, because WSs with larger MDL have larger scheduling flexibility.

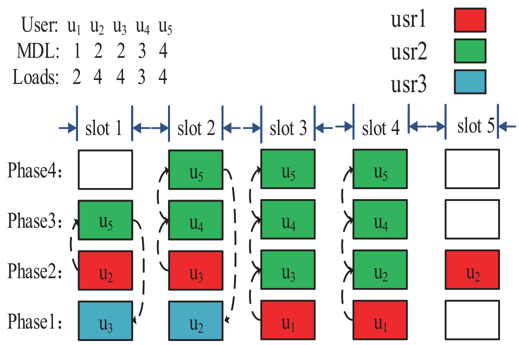


Fig. 2. Example execution process of Algorithm 1.

Algorithm 1: Algorithm for Solving SUS-kSIC Problem:

```
{
// Input: struct {int load; int MDL;} user[n];
// Output: sch   //WS scheduling strategy;
1:  slot_idx=-1; phs_idx=0; phs_init=0; //for each slot
2:  while (there are unscheduled fraffic loads) {
3:    slot_idx++; coll_set=Ø;   //Anchoring begins here
4:    if (unscheduled load of type-1 WSs13 ensemble ≥
      that of every other WS)
5:        usr1=randomly select a type-1 WS;
6:    else usr1 = the non type-1 WS whose load is
      maximal. Besides, if there are multiple eligible WSs,
      choose the one whose MDL is minimal;
7:  sch[slot_idx][usr1.MDL] = usr1;phs_init =
    phs_idx = usr1.MDL;
8:    coll_set = usr1; usr1.load − −;
9:  while(phs_idx ≤ k) {//upper phase allocation
    begins
10:     usr2=the WS which has the heaviest traffic loads
        among WSs whose MDL is no less than phs_idx,
        and not included in coll_set. Besides, if there are
        multiple eligible WSs, the one whose MDL is the
        nearest to phs_idx is chosen.
11:     if(usr2 exists) {sch[slot_idx][phs_idx] = usr2;
      usr2.load − −; phs_idx + +; coll_set = coll_set
      ∪{usr2}; }}
12:     phs_idx = phs_init − 1;
13: while(phs_idx ≥ 1){//lower phase allocation begins
14:       usr3 =the WS which has the heaviest traffic
          loads among WSs whose MDL is no less than
          phs_idx, and not included in coll_set. Besides,
          if there are multiple eligible WSs, the one whose
          MDL is the nearest to phs_idx is chosen.
15:       if (usr3 exists) {
          sch[slot_idx][phs_idx] = usr3;
          usr3.load − −; phs_idx − −;
          coll_set = coll_set ∪ {usr3};}} //end
          while(phase_index ≥ 1)
16:     } //end while(there are unscheduled traffic loads)
```

[13]For convenience, if the MDL of a WS is i, the WS is of i-type.

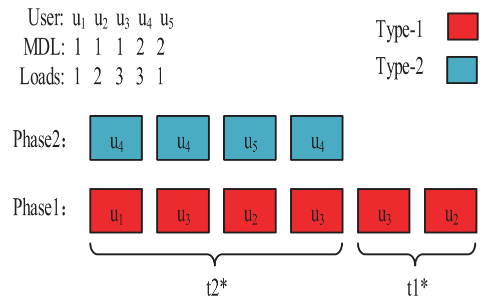


Fig. 3. Example illustrates Proposition 1.

the criterion depicted by line 14. Until now, the WS scheduling strategy for the first slot is settled. With updated traffic loads $\{2, 3, 3, 3, 3\}$, it proceeds to the next slot. The process finishes until the given loads are delivered successfully. Obviously, the shortest scheduling length is 5.

C. Optimality for Two Special Cases

Although Algorithm 1 is not an optimal algorithm for SUS-kSIC,[14] it is proven to be the optimal for two special cases. One is for any traffic loads under 2-SIC, and the other is for unit traffic load under k-SIC. Since 2-SIC is a good tradeoff between implementation complexity and effectiveness, and unit traffic under k-SIC is often the case in many applications involving transmission fairness, we think the two cases are representative for many practical scenarios.

1) Special Case 1: Any Traffic Loads Under 2-SIC: Algorithm 1 is surely suitable for SUS-2SIC. For the special case of 2-SIC, we show that Algorithm 1 outputs an optimal solution to the SUS-2SIC.

For any time slot of the optimal solution to SUS-2SIC, it is either monopolized by a WS or shared by two WSs. For a brief, the single-WS slot is termed as noncompound slot, the slot shared by two WSs as compound slot.

For the WS scheduling strategy output by Algorithm 1, we denote the noncompound slot number by T_1, and that of compound slot by T_2. Therefore, $T_1 + 2T_2 = \sum_{i=1}^{n} L_i$. We notate the number of noncompound slot of the optimal solution by T_1^*, and that of compound slot by T_2^*. Obviously, if Algorithm 1 is the optimal, $T_1 = T_1^*$ and $T_2 = T_2^*$, which will be proven in Theorem 2.

The following lemma reveals a key feature of the WS scheduling strategy constructed by Algorithm 1, which is vital to the proof of the optimality of Algorithm 1 for SUS-2SIC. It reveals that Algorithm 1 always first outputs compound slots, and then the noncompound slots.

Lemma 3: For the WS scheduling strategy output by Algorithm 1, all compound slots are in the slots $1 \sim T_2$, while all noncompound slots are in the slots $T_2 + 1 \sim T_2 + T_1$, where

[14]We take an example for proving that Algorithm 1 is not optimal. There are four WSs, u_1, u_2, u_3, and u_4, their MDLs are 1, 2, 2, 3, and their traffic loads are 6, 3, 3, 6, respectively. The scheduling length output by Algorithm 1 is 8, while there is a feasible sensor scheduling strategy, $\{(u_1, u_2, u_4), (u_1, u_2, u_4), (u_1, u_2, u_4), (u_1, u_3, u_4), (u_1, u_3, u_4), (u_1, u_3, u_4)\}$ whose frame length is 6.

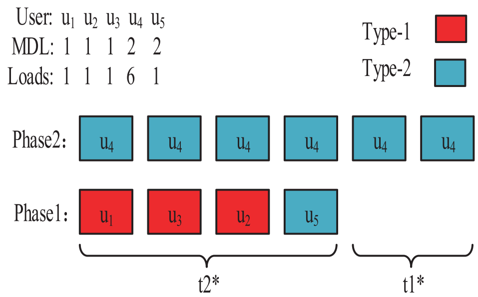


Fig. 4. Example illustrates Proposition 2.

T_1 is the number of noncompound slots and T_2 is the number of compound slots.

Proof: Please refer to Appendix D. ■

Lemma 4: For the WS scheduling strategy output by Algorithm 1, if there are more than one noncompound slots, they must be monopolized either by the type-1 WSs, or by a same WS of type-2.

Proof: Please refer to Appendix E. ■

To prove the optimality of Algorithm 1 in the special case 1, two extra propositions are presented. Note that the two propositions further deepen the conclusion of Lemma 4.

Proposition 1: For the WS scheduling strategy output by Algorithm 1, if there are more than one noncompound slot, and every noncompound slot is monopolized by a type-1 WS, then for any compound slot, it always contains a type-1 WS.[15]

Proof: Please refer to Appendix F. ■

Proposition 2: For the WS scheduling strategy output by Algorithm 1, if there are more than one noncompound slot, and all noncompound slots are monopolized by a same type-2 WS, w.l.o.g., assume the type-2 WS is u_i, then u_i will be contained in every compound slot.[16]

Proof: Please refer to Appendix G. ■

Theorem 2: Algorithm 1 outputs an optimal solution to SUS-2SIC.

Proof: We prove it in two cases using notations in Lemma 3.

Case 1: $T_1 = 1$, i.e., there is only one noncompound slot in the WS scheduling strategy output by Algorithm 1. In this case, the total traffic load, i.e., $(\sum_{i=1}^{n} L_i)$, must be odd, therefore, the minimum frame length in theory is $\lceil \frac{(\sum_{i=1}^{n} L_i)}{2} \rceil$, where $\lceil\rceil$ is for upper rounding. On the other hand, since $T_1 = 1$, the frame length output by Algorithm 1 is also $\lceil \frac{(\sum_{i=1}^{n} L_i)}{2} \rceil$. Thus, the WS scheduling strategy output by Algorithm 1 is the optimal in this case.

Case 2: $T_1 > 1$. Based on the conclusion of Lemma 3, the case could be further put into two subcases as follows.

Subcase 2.1: Every noncompound slot is monopolized by a type-1 WS. In this subcase, based on Proposition 1, the frame length by Algorithm 1 is equal to the load sum of the type-1 WS ensemble. Therefore, the WS scheduling strategy output by Algorithm 1 is the optimal in this subcase.

[15]The proposition can be verified by the example in Fig. 3.
[16]The proposition can be verified by the example in Fig. 4.

Subcase 2.2: All noncompound slots are monopolized by the same type-2 WS, and w.l.o.g., denote the WS by u_i. In this subcase, the minimum frame length is obviously no less than the traffic load of u_i. On the other hand, the length of the WS scheduling strategy output by Algorithm 1 is equal to the traffic load of u_i. Therefore, the WS scheduling strategy output by Algorithm 1 is also optimal in this subcase. ■

In the special case, the time complexity of Algorithm 1 is $O(\sum_{i=1}^{n} L_i)$. It has linear complexity with traffic loads.

2) Special Case 2: Unit Traffic Load Under k-SIC: The special case where every WS can only transmit once and only once in a frame often takes place in IWNs. We prove that Algorithm 1 is also the optimal in this case. Besides, a closed-form expression of the shortest frame length is also presented.

Lemma 5: In the special case where $L_i = 1$ for all $i \in [1, n]$, for the WS scheduling strategy output by Algorithm 1, if the last slot contains a WS, which is scheduled at the jth decoding phase, then in all other slots, there are always j WSs, which will be scheduled from the first to the jth phase, respectively, and besides, all of their MDLs are no larger than j.

Proof: Please refer to Appendix H. ■

Theorem 3: For the special case of SUS-kSIC, where $L_i = 1$ for all $i \in [1, n]$, Algorithm 1 outputs an optimal WS scheduling strategy.

Proof: Please refer to Appendix I. ■

Theorem 4: For the special case of SUS-kSIC, where $L_i = 1$ for all $i \in [1, n]$, the shortest frame length is $\max\{\lceil \frac{n_1}{1} \rceil, \lceil \frac{n_1+n_2}{2} \rceil, \ldots, \lceil \frac{\sum_{i=1}^{k} n_i}{k} \rceil\}$, where n_i denotes the number of type-i WSs.

Proof: Please refer to Appendix J. ■

Theorem 4 can be understood in a more intuitive manner as follows. For the type-1 WSs, they can only be decoded in the decoding phase 1, thus, there are at least n_1 slots in a frame. while for the WSs whose MDLs are no greater than w, they can be assigned to any decoding phase from 1 to w, thus, at least $\lceil \frac{\sum_{i=1}^{w} n_i}{w} \rceil$ slots have to be contained in a frame, and so forth. Therefore, the shortest scheduling length $T_{\min}$ is no less than $\max\{\lceil \frac{n_1}{1} \rceil, \lceil \frac{n_1+n_2}{2} \rceil, \ldots, \lceil \frac{\sum_{i=1}^{k} n_i}{k} \rceil\}$.

V. SUS With Discrete Power for k-SIC

Assume there are m transmit power levels $\overline{tp}_m, \overline{tp}_{m-1}, \ldots, \overline{tp}_1$ where $\overline{tp}_m > \overline{tp}_{m-1} > \cdots > \overline{tp}_1$, and $\frac{\overline{tp}_{i+1}}{\overline{tp}_i} = q$ for $\forall i \in [1, m-1]$. They consist of a feasible power set $\overline{TP} = \{\overline{tp}_m, \overline{tp}_{m-1}, \ldots, \overline{tp}_1\}$. W.l.o.g., assume $p_i^{\max} \in \overline{\text{TP}}$ for all $i \in [1, n]$. The SUS with discrete power for k-SIC (SUSDP-kSIC) problem is formulated as follows:

$$\min_{\{p_{ij}, N_{ij}\}} \quad t \tag{3a}$$

$$\text{s.t.} \quad (1b); \quad (1c); \quad (1\,d); \quad (1e); \quad (1f) \tag{3b}$$

$$p_{ij} \in \overline{\text{TP}} \quad \text{for} \quad \forall i \in [1, n] \quad \forall j \in [1, t]. \tag{3c}$$

Obviously, relative to (1), the continuous power cases, only the extra constraint (3c), which is the constraint of feasible transmit power, is appended.

Algorithm 2: Algorithm for Solving SUSDP-2SIC Problem:

{
$GH = \varnothing$;
for (i=1; $i \leq n$; i++) {
  for (j=1; $j < L_i$; j++) add node V_{ij} to *GH*;}
for (i=1; $i \leq \lceil \frac{n}{2} \rceil$; i++) { for($j = i+1$; $j \leq n$; j++)
  if (u_i and u_j are power compatible) {
    for($s = 1$; $s \leq L_i$; $s++$) {for ($l = 1$; $l \leq L_j$; $l++$)
    connect V_{is} with V_{jl} by an edge;} }}//construct graph
find a maximum match of the graph GH;
for any two matched nodes V_{is} and V_{jl}, a slot is allocated to u_i and u_j. Besides, set their transmit powers based on the decoding phase they are allocated to;
for any unmatched node V_{is}, allocate a slot to u_i, and set its transmit power as $\frac{[\![\widehat{x}_1]\!]}{G_i}$; }

We present an optimal algorithm when $k = 2$, and a greedy algorithm is provided for other cases.

Define $[\![x]\!] = \arg\min_{(y \in \overline{\mathrm{TP}}) \cap (y \geq x)} (y - x)$, $(x > 0)$, in other words, $[\![x]\!]$ is the value that satisfies: 1) It belongs to the set $\overline{\mathrm{TP}}$. 2) It is no less than x and the nearest to x.

A. SUS With Discrete Power Under 2-SIC (SUSDP -2SIC)

For 2-SIC, Algorithm 2 presents an optimal WS scheduling strategy for SUSDP-2SIC.

If there is a feasible power allocation strategy such that the parallel transmissions from two WSs can be decoded correctly, the two WSs are power compatible. For the case of the discrete transmit powers, a practical method is to try all possible combinations of the transmit powers. Therefore, the complexity of finding whether two WSs are power compatible is $O(m^2)$. Of course, simpler algorithms are also possible.

The maximal matching of a graph can be achieved using Edmonds Blossom algorithm, and its complexity is $O((\sum_{i=1}^{n} L_i)^4)$. Since we have traversed all possible combinations of WS scheduling and power allocation, the optimum can be guaranteed. Fortunately, under 2-SIC, Algorithm 2 is still polynomial.

B. SUS With Discrete Power Under k-SIC (SUSDP-kSIC)

For k-SIC where $k > 2$, we present a heuristic algorithm. The heuristic algorithm is analogous to Algorithm 1. It first constructs a discrete powers set, which functions similarly as PTS-r in the continuous transmit powers. The set is utilized to model the scheduling flexibility of WSs under the discrete transmit powers, which is termed as discrete MDL (DMDL). For SUSDP-kSIC, by using DMDL instead of MDL, we employ Algorithm 1 to find a WS scheduling strategy under the discrete transmit powers. The existence of feasible powers for the WS scheduling strategy is revealed by Theorem 5.

Algorithm 3: Algorithm for Solving SUSDP-kSIC Problem:

{ // Input: struct{ int load; int DMDL;} $user[n]$;
1: $G_{\min} = \min\{G_1, G_2, \ldots, G_n\}$;
 $G_{\max} = \max\{G_1, G_2, \ldots, G_n\}$;
2: $tp_1 = [\![\frac{\widehat{x}_1}{G_{\min}}]\!], i = 1$;
3: while$((tp_i < \max(p_i^{\max}, i \in [1, n]))\&\&(i \leq k))\{$
4: i++; $tp_i = [\![tp_{i-1} \frac{G_{\max}}{G_{\min}}(1 + \gamma)]\!]$;}
5: for$(i = 1; i \leq n; i++)$ {
6:   if ($p_i^{\max} < tp_1$) *user*[i].DMDL=1;
7:   else for$(j = 1; j \leq k; j++)$; {
8:    if ($tp_j \leq p_i^{\max} < tp_{j+1}$) $user[i].DMDL = j$; }}
9: using Algorithm 1 to find a WS scheduling strategy;
10: for $(i = 1; i \leq n; i++)\{$ //Allocate power
11: if ($p_i^{\max} < tp_1$) set power of $user[i]$ as $p_i^{\max}$;
12: else set power of $user[i]$ as tp_l where l is the allocated phase;}}

TABLE II
SIMULATION PARAMETER SETTING

Parameter	Value		Parameter	Value
Area size	1*km*×1*km*		Decoding threshold	2
WS number	30		Power spectral density	-169*dBm/Hz*
Bandwidth	200*kHz*		Data packet generation model	$N(\mu, \sigma^2)$
Signal frequency	2.4*GHz*		Max. transmit power	10*dBm*

Line 1–4 of Algorithm 3 aims to construct the discrete powers set $\{tp_1, tp_2, \ldots, tp_k\}$. A vital property of the set is that all parallel transmissions can be decoded successfully if their transmit powers belong to the set and are distinct from each other. The property is proven as follows.

Lemma 6: For a series of positive numbers $[a_n, a_{n-1}, \ldots, a_2, a_1, c]$, if $\frac{a_i}{a_{i-1}} \geq q + 1$ for $i \in [2, n]$, and $\frac{a_1}{c} \geq q$, then $\frac{a_i}{\sum_{j=1}^{j=i-1} a_j + c} \geq q$, for $i \in [2, n]$.

Proof: Please refer to Appendix K. ■

Theorem 5: For the set $\{tp_1, tp_2, \ldots, tp_k\}$ constructed by Algorithm 3, all parallel transmissions can be decoded successfully if their transmit powers belong to the set and are distinct from each other.

Proof: Please refer to Appendix L. ■

VI. PERFORMANCE EVALUATIONS

We conduct a series of simulation experiments to demonstrate the effectiveness of the algorithms presented in this article. Some simulation parameters are listed in Table II. The noise power spectral density is −169 dBm/Hz, and the noise bandwidth is 200 kHz, thus, N_0 is −116 dBm. The following channel gain model is used for performance evaluations: [20]

$$CG = -20\log(f) - 26\log(d) + 19.2$$

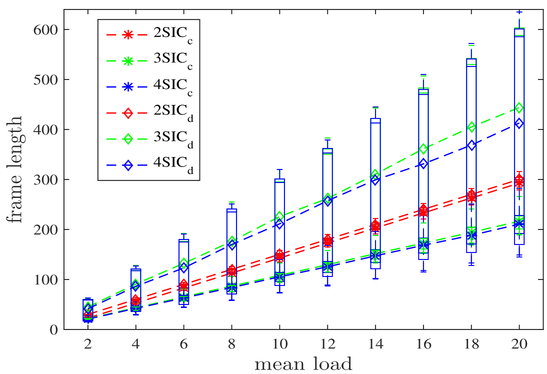


Fig. 5. Frame length w.r.t. traffic loads.

where f is the frequency in megahertz, and d is the Euclidean distance between transmitter and receiver in meters. Using the channel gain model, the channel gain of each WS can be known based on its Euclidean distance with the sink.

A wireless network consists of 30 WSs and one sink, which is at the center of a square with sides of 1 km, and all WSs are placed uniformly in the square. We assume that the maximal transmit power is 10 dBm, and the minimal discrete transmit power $\overline{tp}_1$ is -25 dBm. The decoding threshold γ is 2, which is a typical value nowadays.

We assume that any data packet can be delivered in one slot. Besides, the packet load of every WS is Gaussian $N(\mu, \sigma^2)$, where $\sigma = 0.1\ \mu$ for all experiment cases. In our experiments, μ is set as 2, 4, 6, 8, and 10, respectively, and a slot is 10 ms, which is just for transmitting a data packet. Besides, the max power is 10 dBm if it is not stressed deliberately.

A. Relation Between Shortest Frame Length and Load

First, we aim to reveal how the shortest frame length is influenced by network load with various SIC receivers. For the case of continuous transmit powers, Algorithm 1 is used, and for the case of discrete transmit powers, Algorithm 3 is used instead. For all experiments in this section, for any WS, its data packets are generated and loaded to the WS one time, and then the frame length is got by executing scheduling algorithm until all data packets of all WSs are delivered successfully.

We repeat every experiment $1000\times$ and plot the results using boxplot in Fig. 5, where the subscript d in the legend denotes the case of discrete transmit powers, while c for continuous transmit powers. The x-axis is for μ, which is the mean of traffic loads, and the y-axis is the frame length.

Obviously, the mean frame length almost linearly increases with the traffic loads for all cases. Besides, the variance of frame length is also increasing with the traffic loads, which is doomed because the variance of traffic loads is $0.1\ \mu$.

Comparing the frame lengths of the discrete case with these of the continuous case, we find that the frame lengths of the discrete case are always larger than those of the continuous case for all SIC receivers. The phenomenon obviously coincides with our expectations because relative to the formulation in the continuous case, there is an extra power constraint for the

TABLE III
FRAME LENGTH UNDER 4-SIC RECEIVER

μ \ edge length	125	250	500	1000
2	15.144 (0.1232)	15.144 (0.123)	16.648 (0.388)	20.92 (7.270)
8	60.449 (1.5154)	60.449 (1.515)	61.846 (1.796)	84.157 (105.062)
12	90.464 (3.071)	90.464 (3.071)	91.861 (3.432)	125.814 (263.355)
16	120.313 (4.839)	120.313 (4.839)	121.707 (5.311)	168.000 (473.904)
20	150.246 (8.277)	150.246 (8.277)	151.649 (8.634)	209.604 (712.537)

discrete case. It is the extra constraint that results in a larger frame length.

Intuitively, for the same traffic loads, the frame length will decrease with the capability of SIC receiver, because better scheduling flexibility is provided for stronger SIC receiver. In other words, we believe that the frame length of k-SIC receiver will be larger than that of $(k+1)$-SIC receiver. It indeed the case for the continuous cases in Fig. 5, and however, it is counterintuitive for the discrete cases since the frame length under 2-SIC is the smallest. The reason is that Algorithm 2 is optimal for 2-SIC while Algorithm 3 is not optimal for k-SIC where $k > 2$. The loss is obviously caused by the distinction. Therefore, we confess that Algorithm 3 has still improvement spaces for k-SIC where $k > 2$.

B. Relation Between Frame Length and MDL Distribution

Based on Algorithms 1 and 3, the WS with larger MDL obviously has better flexibility for scheduling, and therefore, results in a shorter frame. In this experiment, we aim to verify the above-mentioned viewpoint by varying the MDL distribution of all WSs.

To generate distinct MDL distributions while keeping other factors unchanged as possible, we still use the network topology in the last experiment, and shrink its edge length from 1 km to 500 m, 250 m and 125 m, respectively. At the same time, the coordinate of every WS shrinks correspondingly. Obviously, based on the definition of MDL, for any WS, its value of MDL will increase with the shrinking of the network area, because of its shorter distance to the sink.

The minimum frame lengths for the four scenarios using 4-SIC receiver are listed in Table III, where every element includes a mean value and a variance.

Take the case $\mu = 2$ as an example, when the edge length shrinks from 1 km to 250 m, smaller frame length will be achieved. The phenomenon is inevitable because shrinking edge length while maintaining relative locations unchanged must result in larger MDLs. As we have talked, larger MDL result in

TABLE IV
FRAME LENGTH UNDER 2-SIC RECEIVER

edge length / μ	125	250	500	1000
2	30.148 (0.150)	30.148 (0.150)	30.148 (0.150)	30.15 (0.154)
8	120.383 (5.308)	120.383 (5.308)	120.383 (5.308)	120.394 (5.487)
12	180.239 (11.436)	180.239 (11.436)	180.239 (11.436)	180.256 (11.516)
16	240.228 (20.476)	240.228 (20.476)	240.228 (20.476)	240.263 (20.884)
20	300.229 (28.047)	300.229 (28.046)	300.229 (28.046)	300.285 (28.400)

TABLE V
SCHEDULING CAPABILITY FOR CONTINUOUS TRAFFIC LOADS

k-SIC / μ	2-SIC	3-SIC	4-SIC	TDMA
2	Z	Z	Z	Z
4	Z	Z	Z	I
6	S	Z	Z	I
8	I	S	Z	I
10	I	I	S	I
12	I	I	I	I

TABLE VI
PERFORMANCE OF THROUGHPUT AND DELAY

(a) Average delay performance w.r.t. traffic loads (slots).

k-SIC / μ	TDMA	2-SIC	3-SIC	4-SIC
2	29.471	14.500	9.732	8.445
8	119.462	59.481	40.475	35.040
16	239.526	119.392	81.684	70.625

(b) Average throught performance w.r.t. area sizes.

k-SIC / edge length	TDMA	2-SIC	3-SIC	4-SIC
250	1	2	3	4
500	1	2	2.968	3.952
1000	1	1.967	2.500	3.057

better scheduling flexibility, and therefore, shorter frame length, just as revealed in Table III.

For the two cases where the edge length is 125 m and 250 m, respectively, their frame lengths are the same. Similar results can also be found in other cases of traffic loads. The phenomenon reveals that the distribution of MDLs instead of the value of MDLs plays a deterministic role in the frame length. In fact, for the above-mentioned four cases of edge lengths, the MDL distributions of WSs, i.e., the proportion of WSs whose MDLs are equal to 1, 2, 3, 4 are (0, 0, 0, 1), (0, 0, 0, 1), (0, 0.023, 0.438, 0.539), and (0.264, 0.422, 0.180, 0.134), respectively. The results also defend that the same MDL distribution results in the same frame length.

To have further verifications, we repeat all experiments under 2-SIC and list results in Table IV. We also take the case $\mu = 2$ in Table IV as an example, the frame lengths when edge length is 125 m, 250 m, and 500 m are all the same, since the MDL distributions in the three cases are the same for 2-SIC receiver. Their distinctions from these in 4-SIC receiver reveal that for the same topology, the MDL distribution of WSs will be more balanced for k-SIC receivers when k is smaller. The similar phenomena can also be found for other traffic load cases. The reason is easy to be understood since for a k-SIC receiver, there are k levels of MDLs. An extreme example where $k = 1$ will be helpful for understanding. The MDLs of all WSs are all 1 in this case, i.e., their MDLs are completely balanced. In one word, the larger the k, the more unbalanced the MDL distributions.

C. Dynamic Scheduling Performance of Algorithm

In all the above-mentioned experiments, for analyzing factors influencing the frame length performance, the traffic load for a WS is allocated in one time, i.e., the traffic loads are given before determining the scheduling strategy. However, in practical scenarios, data are always generated continuously. To have an objective evaluation for practical scenarios, we assume that the buffer of any WS is infinite, and data are generated continuously and stored in buffers until they are scheduled for transmitting.

Based on [8], the network is thought to be balanced if the volume of data buffered in every buffer is finite after infinite time. At the balanced state, the mean rate of data input is thought of as a precise sign of scheduling capability of the scheduling algorithms. The criterion is obviously of theoretical meaning since we cannot wait for infinite time in practice. However, in our experiments, we can know the number of data traffics that have been delivered during the last frame, and that generated during the last frame. When the two numbers are nearly equal, the network can be considered to be balanced.

In Table V, we illustrated the state of the mean volume of buffered data under different data rates for different SIC receivers, where Z is for empty buffer, S is for stable, and I is for infinite. For an objective performance evaluation, we also test the classic TDMA. The experiment results reveal that our scheduling algorithm on SIC is prominently superior to the classic TDMA. Besides, our scheduling algorithm will have better scheduling capability for k-SIC receiver with larger k.

D. Performance of Delay and Throughput

We reveal how the delay is influenced by network load with various SIC receivers. With the default simulation parameters, we evaluate the average delay and list results in Table VI. Obviously, with the increasing k of k-SIC, much scheduling opportunities are provided, and thus, it brings less delays.

To evaluate the throughput performance, we only vary the area size. We set the TDMA throughput with edge length being 1 km as the performance benchmark. The throughput performances in different scenarios are revealed by their ratios to the benchmark. Just as illustrated by Table VI, with the decreasing of area sizes,

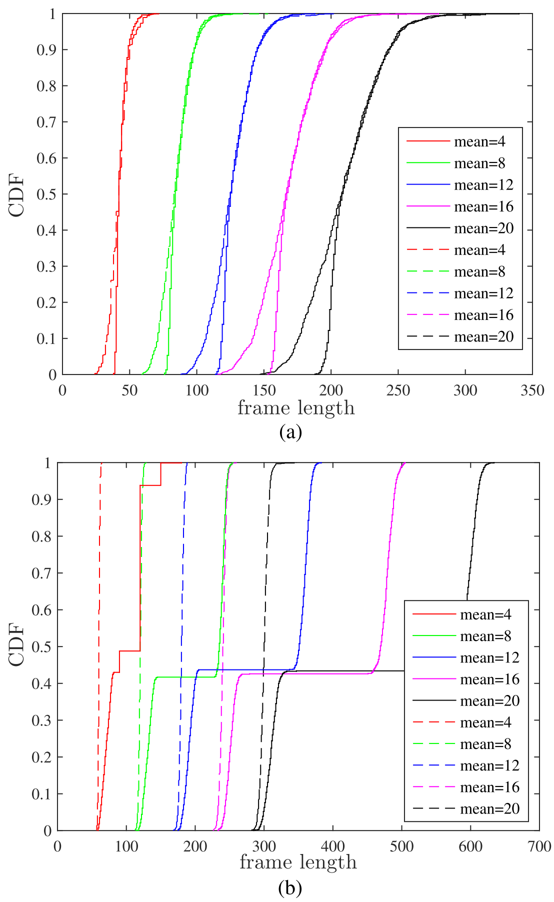


Fig. 6. Frame length CDFs by Algorithms 1 and 3.

the MDL of a WS will increase. Larger MDL will provide more scheduling flexibility, and thus, it brings larger throughput.

We also reveal the relationship between throughput and SIC type. We regard the throughput of TDMA as the performance benchmark, and the value in Table VI is the throughput ratio of different scenarios to the benchmark. We can know the ratio is roughly positively correlated with value of k for k-SIC receiver. The shorter the edge length is, the more obvious the phenomenon is.

E. Performance of Heuristic Algorithm

We evaluate the performances of the two heuristic algorithms for 4-SIC receiver. For finding their performance gaps with the respective optima, we use a simulated annealing algorithm to find their minimum frame lengths. The distributions of frame lengths are illustrated in Fig. 6, where the solid lines represent the results of our scheduling algorithms and the dotted lines are for the optima.

From the above-mentioned results, for both the continuous transmit powers and the discrete transmit powers, the performance deviation from the optimum always increases with the traffic loads. Besides, for the case of continuous transmit powers, the performance deviation mainly lies in the short frames, while it lies in the long frames for the case of discrete transmit powers. The results reveal that the performance deviation caused by the nonoptimum of Algorithm 1 is further amplified by the discrete transmit powers.

VII. Conclusion

Power-domain NOMA has broad prospects in IWNs for its strong capability of supporting parallel transmissions. How to effectively utilize the capability is the key to performances of applications. We model the SUS problem by joint WS scheduling and power allocation, and solve it based on rigorous mathematical derivations.

Our conclusions are as follows.

1) Two key terms, MDL, and power exclusiveness, can accurately describe the properties of SIC, and they have great potential for SIC-related researches.
2) The method of the joint WS scheduling and power allocation is valid for the SUS problem under SIC, and it surely improves the real-time performances.
3) The optimality of the proposed algorithms has been proven for regular cases, which guarantees their usefulness.

Since the perfect SIC is difficult to be realized in practice, the influence of the residual error on the frame length is meaningful to be investigated, which will be our future works.

Appendix

A. Proof of Theorem 1

Proof: It is easy to be proven using mathematical induction. When $r = 1$, it is very obvious that $\widetilde{X}_1 \geq \widehat{X}_1$. We assume that when $r = n$, Theorem 1 is established, i.e., $\widetilde{X}_i \geq \widehat{X}_i$ for $\forall i \in [1, n]$. Next, we will prove when $r = n + 1$, Theorem 1 is also established. First, we assume $\widetilde{X}_{n+1} < \widehat{X}_{n+1}$. So, $\frac{\widetilde{X}_{n+1}}{\sum_{i=1}^{n} \widetilde{X}_i + n_0} < \frac{\widehat{X}_{n+1}}{\sum_{i=1}^{n} \widetilde{X}_i + n_0} < \frac{\widehat{X}_{n+1}}{\sum_{i=1}^{n} \widehat{X}_i + n_0} = \gamma$. However, $\frac{\widetilde{X}_{n+1}}{\sum_{i=1}^{n} \widetilde{X}_i + n_0} \geq \gamma$. Thus, our assumption is wrong, i.e., $\widetilde{X}_{n+1} \geq \widehat{X}_{n+1}$. Theorem 1 is thus proven. ■

B. Proof of Lemma 1

Proof: We prove it by contradiction. For convenience, we assume that $WS_1, WS_2, \ldots, WS_w$, where $w > l$, transmit a packet simultaneously in a slot and all of them are decoded correctly. Since $\gamma > 1$ and these packets are decoded successfully, their received powers must be distinct. W.l.o.g., assume $rp_1 < rp_2 < \cdots < rp_w$. Therefore, there is only one feasible decoding order $\langle w, w-1, \ldots, 1 \rangle$, and $rp_1 \geq \widehat{X}_1, rp_2 \geq \widehat{X}_2, \ldots, rp_w \geq \widehat{X}_w$ holds based on Theorem 1. Since $w \geq l$, $\widehat{X}_w \geq \widehat{X}_{l+1}$, therefore, $rp_w \geq \widehat{X}_{l+1}$ holds, which contradicts the presumption of the lemma. ■

C. Proof of Lemma 2

Proof: Necessity: We prove it by contradiction. Assume there is a WS that is decoded in phase i and whose MDL is less than i. Since the WS is decoded in phase i, there are at least $i - 1$ parallel WSs that must be decoded in $[1, i - 1]$. On the other hand, based on the assumption, there are at most $i - 2$

parallel WSs that are decoded in phase $[1, i-1]$. Therefore, the two inferences contradict.

Sufficiency: For the WS, which is successfully decoded in decoding phase i, say WS_j, since its MDL is no less than i, i.e., $\frac{\widehat{X}_i}{G_j} \leq P_j^{\max}$, we can set its transmit power as $\frac{\widehat{X}_i}{G_j}$. We do the similar power allocation for every $i \in [1, w]$. It can be verified that SINR of every WS is γ, that is, all of these w packets can be decoded correctly. ∎

D. Proof of Lemma 3

Proof: The lemma can be proven by contradictions. Assume it is not the fact, there are two slots, w.l.o.g., assume they are the jth and the $(j+1)$th slot, where the jth slot is noncompound while the $(j+1)$th is compound.

Case 1: The MDL of the WS in the jth slot is 1. The emergence of the $(j+1)$th slot that is compound is impossible by Algorithm 1, or else the type-2 WS in the $(j+1)$th slot must be scheduled in the jth slot according to Algorithm 1.

Case 2: The MDL of the WS in the jth slot is 2. The emergence of the $(j+1)$th slot is impossible, because either the type-1 WS or the type-2 WS must be scheduled in the jth slot according to Algorithm 1. ∎

E. Proof of Lemma 4

Proof: The lemma can be proven by contradictions. For two noncompound slots, if they are monopolized by a type-1 and a type-2 WS, respectively, they will be combined as one compound slot according to Algorithm 1. In other words, the multiple noncompound slots could not be monopolized simultaneously by type-1 and type-2 WSs. Furthermore, if two noncompund slots are monopolized by two distinct type-2 WSs, the two WSs must be same, or else they will be combined as one compound slot by Algorithm 1. ∎

F. Proof of Proposition 1

Proof: Based on the same notations and the conclusion of Lemma 4, the first T_2 slots are compound and the remaining T_1 slots are noncompound. Therefore, at the beginning of the T_2th slot, the unscheduled traffic load of type-1 WS ensemble is larger than that of any a type-2 WS, because there are more than one noncompound slots, which are monopolized by type-1 WSs. Based on the lines 4–6 of Algorithm 1, a type-1 WS will be chosen as $usr1$, i.e., the T_2th slot must contain a type-1 WS. Thus, at the beginning of the (T_2-1)th slot, the traffic load of all type-1 WS ensemble is larger than that of any type-2 WS. The above-mentioned procedure goes iteratively until the beginning of the first slot. The proposition is thus proven. ∎

G. Proof of Proposition 2

Proof: Based on the same notations and the conclusion of Lemma 4, the first T_2 slots are compound and the remaining T_1 slots are noncompound. And, all noncompound slots are monopolized by u_i. Therefore, at the beginning of the T_2th slot, the unscheduled traffic load of u_i is not only larger than that of any other type-2 WS but also larger than the unscheduled traffic loads of type-1 WS ensemble. Based on the lines 4–6 of Algorithm 1, u_i will be chosen as $usr1$, i.e., the T_2th slot must contain u_i. Thus, at the beginning of the (T_2-1)th slot, the unscheduled traffic load of u_i is not only larger than that of any other type-2 WS but also larger than the unscheduled traffic loads of type-1 WS ensemble. The above-mentioned procedure goes iteratively until the beginning of the first slot. The proposition is thus proven. ∎

H. Proof of Lemma 5

Proof: It is easy to be proven by contradictions. For brief, notate the WS by u_i. First, there is no empty phase from 1 to j in any slot except for the last slot, or else, u_i will be allocated to the empty phase according to Algorithm 1.

Second, the MDL of these WSs allocated to phases 1 to j in every slot except for the last one are no larger than j. Or else, w.l.o.g., assume u_l, whose MDL is greater than j, is among them. In this case, u_l would not be chosen when Algorithm 1 chooses a WS for the position of u_l, since u_i have higher priority than u_l based on lines 6, 10, or 14 in Algorithm 1. Therefore, u_i could not be contained in the last slot. ∎

I. Proof of Theorem 3

Proof: We prove it by contradictions. Assume the optimal slot number is T_{opt}. Therefore, the frame length by Algorithm 1 is at least $T_{\text{opt}}+1$. W.l.o.g., if the MDL of the variable $usr1$ in the $T_{\text{opt}}+1$th slot is j, based on Lemma 5, there are at least $jT_{\text{opt}}+1$ WSs whose MDL is no larger than j. So, according to lemma 2, the frame length of any a feasible sensor scheduling strategy, including the optimal one, is at least $\lceil\frac{(jT_{\text{opt}}+1)}{j}\rceil$, i.e., $T_{\text{opt}}+1$, which contradicts the assumption. ∎

J. Proof of Theorem 4

Proof: According to Lemma 5, if the MDL of the $usr1$ in the last slot is j, the scheduling length is $\lceil\frac{\sum_{i=1}^{j} n_i}{j}\rceil$. We now try to prove that $\lceil\frac{\sum_{i=1}^{j} n_i}{j}\rceil = \max\{\lceil\frac{n_1}{1}\rceil, \lceil\frac{n_1+n_2}{2}\rceil, \ldots, \lceil\frac{\sum_{i=1}^{k} n_i}{k}\rceil\}$ Notate $\lceil\frac{\sum_{i=1}^{j} n_i}{j}\rceil$ by $T_{\min}$.

Case 1: For all $1 \leq l \leq j$

Based on Lemma 5, $\sum_{i=1}^{l} n_i \leq lT_{\min}$, i.e., $T_{\min} \geq \frac{\sum_{i=1}^{l} n_i}{l}$. Therefore, $T_{\min} \geq \lceil\frac{\sum_{i=1}^{l} n_i}{l}\rceil$ since $T_{\min}$ is an integer.

Case 2: for all $j+1 \leq l \leq k$

Case 2.1. MDLs of all WSs allocated for phase $j \sim k$ in the last slot are distinct.

Based on Lemma 5, $\sum_{i=1}^{l} n_i = l(T_{\min}-1)+2$. Therefore, $T_{\min} \geq \lceil\frac{\sum_{i=1}^{l} n_i}{l}\rceil$ since $l \geq 2$.

Case 2.2. MDLs of all WSs allocated for phase $j \sim k$ in the last slot are not distinct.

Based on Lemma 5, $\sum_{i=1}^{l} n_i \le lT_{\min}$, and $\sum_{i=1}^{l} n_i - 2 \ge (T_{\min} - 1)l$. Therefore, $\frac{\sum_{i=1}^{l} n_i}{l} \le T_{\min} \le \frac{(\sum_{i=1}^{l} n_i)-2+l}{l}$. So, $T_{\min} = \lceil \frac{\sum_{i=1}^{l} n_i}{l} \rceil$, since $l \ge 2$.

In conclusions, the frame length of the optimal sensor scheduling strategy is $\max\{\lceil \frac{n_1}{1} \rceil, \lceil \frac{n_1+n_2}{2} \rceil, \ldots, \lceil \frac{\sum_{i=1}^{k} n_i}{k} \rceil\}$ ■

K. Proof of Lemma 6

Proof: $\frac{a_i}{\sum_{j=1}^{j=i-1} a_j + c} \ge \frac{a_i}{\frac{a_i}{q+1} + \frac{a_i}{(q+1)^2} + \ldots + \frac{a_i}{(q+1)^{i-1}} + \frac{a_i}{(q+1)^{i-1} q}} \ge \frac{a_i}{\sum_{j=1}^{j=i} \frac{a_i}{(q+1)^j}} \ge \frac{a_i}{\sum_{j=1}^{j=\infty} \frac{a_i}{(q+1)^j}} \ge q.$ ■

L. Proof of Theorem 5

Proof:

Case 1: There is only one WS u_i in the slot. Its transmit power is $\min\{p_i^{\max}, tp_1\}$, and the transmission from u_i can be decoded correctly.

Case 2: There is more than one WS in the slot. W.l.o.g., assume two WSs, u_i and u_j, are allocated to phase $l+1$ and l by line 9 of Algorithm 3, respectively. In this case, $\frac{\mathrm{tp}_{l+1} * G_i}{\mathrm{tp}_l * G_j} = \frac{[\![\mathrm{tp}_l * \frac{G_{\max}}{G_{\min}} * (1+\gamma)]\!] * G_i}{\mathrm{tp}_l * G_j} \ge \frac{\mathrm{tp}_l * \frac{G_{\max}}{G_{\min}} * (1+\gamma) * G_i}{\mathrm{tp}_l * G_j} = \frac{G_{\max} * G_i}{G_{\min} * G_j} * (1+\gamma) \ge (1+\gamma)$.

Besides, $\frac{\min\{p_i^{\max}, tp_1\}}{n_0} \ge \gamma$ always holds. Therefore, based on Lemma 6, all parallel transmissions are successful. ■

REFERENCES

[1] C. Xu, K. Ma, Y. Xu, Y. Xu, and Y. Fang, "Optimal power scheduling for uplink transmissions in SIC-based industrial wireless networks with guaranteed real-time performance," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3216–3228, Apr. 2018.

[2] C. Xu, M. Wu, Y. Xu, and Y. Fang, "Uplink low-power scheduling for delay-bounded industrial wireless networks based on imperfect power-domain NOMA," *IEEE Syst. J.*, vol. 14, no. 2, pp. 2443–2454, Jun. 2020.

[3] C. Xu, H. Ding, and Y. Xu, "Low-complexity uplink scheduling algorithms with power control in successive interference cancellation based wireless mud-logging systems," *Wireless Netw.*, vol. 25, no. 1, pp. 321–334, 2019.

[4] Y. Liu, Z. Qin, M. Elkashlan, Z. Ding, A. Nallanathan, and L. Hanzo, "Nonorthogonal multiple access for 5G and beyond," *Proc. IEEE*, vol. 105, no. 12, pp. 2347–2381, Dec. 2017.

[5] L. Liu, C. Yuen, Y. L. Guan, Y. Li, and C. Huang, "Gaussian message passing for overloaded massive mimo-noma," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 210–226, Jan. 2019.

[6] Y. Chi, L. Liu, G. Song, C. Yuen, Y. L. Guan, and Y. Li, "Practical MIMO-NOMA: Low complexity and capacity-approaching solution," *IEEE Trans. Wireless Commun.*, vol. 17, no. 9, pp. 6251–6264, Sep. 2018.

[7] P. Swami, V. Bhatia, S. Vuppala, and T. Ratnarajah, "On user offloading in NOMA-hetnet using repulsive point process," *IEEE Syst. J.*, vol. 13, no. 2, pp. 1409–1420, Jun. 2019.

[8] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Autom. Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.

[9] A. Behzad and I. Rubin, "Optimum integrated link scheduling and power control for multihop wireless networks," *IEEE Trans. Veh. Technol.*, vol. 56, no. 1, pp. 194–205, Jan. 2007.

[10] A. Pantelidou and A. Ephremides, "Minimum schedule lengths with rate control in wireless networks," in *Proc. IEEE Mil. Commun. Conf.*, San Diego, CA, USA, 2008, pp. 1–7.

[11] S. A. Borbash and A. Ephremides, "Wireless link scheduling with power control and SINR constraints," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5106–5111, Nov. 2006.

[12] Y. Sadi and S. C. Ergen, "Minimum length scheduling with packet traffic demands in wireless ad hoc networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3738–3751, Jul. 2014.

[13] 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.

[14] D. Yuan, V. Angelakis, L. Chen, E. Karipidis, and E. G. Larsson, "On optimal link activation with interference cancelation in wireless networking," *IEEE Trans. Veh. Technol.*, vol. 62, no. 2, pp. 939–945, Feb. 2013.

[15] C. Xu, K. Ma, and Y. Xu, "Complexity of minimum uplink scheduling in backbone-assisted successive interference cancellation-based wireless networks," *Comput. Netw.*, vol. 133, pp. 120–129, 2018.

[16] M. Kontik and S. C. Ergen, "Scheduling in single-hop multiple access wireless networks with successive interference cancellation," *IEEE Wireless Commun. Lett.*, vol. 3, no. 2, pp. 197–200, Apr. 2014.

[17] M. Kontik and S. C. Ergen, "Scheduling in successive interference cancellation based wireless ad hoc networks," *IEEE Commun. Lett.*, vol. 19, no. 9, pp. 1524–1527, Sep. 2015.

[18] O. Goussevskaia and R. Wattenhofer, "Scheduling with interference decoding: Complexity and algorithms," *Ad Hoc Netw.*, vol. 11, no. 6, pp. 1732–1745, 2013.

[19] L. Liu, Y. Chi, C. Yuen, Y. L. Guan, and Y. Li, "Capacity-achieving MIMO-NOMA: Iterative LMMSE detection," *IEEE Trans. Signal Process.*, vol. 67, no. 7, pp. 1758–1773, Apr. 2019.

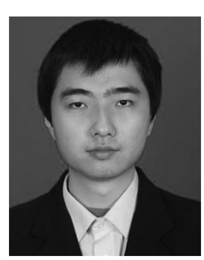
[20] D. B. Green and A. S. Obaidat, "An accurate line of sight propagation performance model for ad-hoc 802.11 wireless lan (WLAN) devices," in *Proc. IEEE Int. Conf. Commun. Conf. Proc.*, 2002, vol. 5, pp. 3424–3428.

Chaonong Xu received the B.S. and M.S. degrees in computer sciences from the Hefei University of Technology, Hefei, China, in 1998 and 2001, respectively, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2007.

He joined Tsinghua University, Beijing, China, as a Postdoctoral in 2007. He is currently an Associate Professor of Computer Science with the China University of Petroleum, Beijing, China. His research interests cover embedded system and wireless networks.

Mianze Wu received the B.S. degree in computer science and technology in 2017, from the China University of Petroleum, Beijing, China, where he is currently working toward the M.S. degree with the Department of Computer Science and Technology.

His research interests cover wireless communication and networks.



Yida Xu received the B.S. degree in computer science and technology from the University of Science and Technology of China, Hefei, China, in 2011, and the M.S. degree in computer science and technology from the China University of Petroleum, Beijing, China, in 2017. He is currently working toward the Ph.D. degree with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. His research interests include wireless communication and networks.

Yongjun Xu received the B.S. degree in communication engineering from the Xi'an University of Posts and Telecommunications, Xi'an, China, in 2001, and the Ph.D. degree in computer sciences with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2006.

He is currently a Professor with the Institute of Computing Technology, Chinese Academy of Sciences. His research interests include low power system and wireless sensor networks.