Random Space–Time Line Code With Proportional Fairness Scheduling

JINGON JOUNG, (Senior Member, IEEE)
School of Electrical and Electronics Engineering, Chung-Ang University, Seoul 06974, South Korea
e-mail: jgjoung@cau.ac.kr

This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government, Ministry of Science and ICT (MSIT), under Grant 2019R1A2C1084168.

ABSTRACT
As the demand for and services of the Internet of Things (IoT) increase, the dense deployment of a massive number of devices is expected in future IoT networks. Accordingly, an appropriate scheduling method will be required to fairly support the overwhelmingly large number of devices or users operating in practical environments. In this paper, we consider a single-cell downlink network in which one transmitter, such as an access point and base station, supports the massive number of users by using a space–time line code (STLC) scheme. Here, to resolve a prohibitively large amount of channel state information (CSI) feedback from the users to the transmitter, we propose a new STLC scheme called random STLC that uses a random vector for the STLC encoding. Furthermore, a proportional fairness (PF) scheduler is modified with a priority factor that is proportional to the devised alternative signal-to-interference-plus-noise ratio. By a comparison with an optimal STLC system that uses full CSI with a PF scheduler, we justify the benefit of the proposed random STLC with a PF scheduler using partial CSI for massive-user networks. Numerical results obtained from a rigorous simulation confirm that the proposed scheme can provide a comparable to achievable rate and fairness with massive number of users together with affordable feedback overhead in massive networks.

INDEX TERMS
Space–time line code (STLC), proportional fairness (PF) scheduling, Internet of Things (IoT) networks, partial channel state information (CSI).

I. INTRODUCTION
The Internet of Things (IoT) or Everything (IoE) involves an intelligent network that is able to connect all things through the Internet [1]–[3]. Various studies have thus mainly focused on networking topics, such as network congestion, delay, routing, and security. The true benefit of the IoT can be achieved when Things can be connected to the networks wirelessly as well. For the Olympic Winter Games PyeongChang 2018 opening ceremony, 1,218 unmanned aerial vehicles (i.e., Intel® Shooting Star™ drones) were airborne simultaneously, which is a demonstration that was enabled by massive connectivity through the air interface [4]. As the new era of 5G begins, massive connectivity of IoT devices (from a few tens of thousands to one million connections per square kilometer [5]–[7]) has been addressed for integration with various powerful analytics, services, and management of the massive number of IoT devices [1], such as sensors, actuators, smart devices, vehicles [8], and many other personal/public smart appliances [3], [9].

A. RESOURCE ALLOCATION, MULTIPLE ACCESS, AND SCHEDULING FOR MASSIVE DEVICES
For massive connectivity of IoT devices, resource management is critical. The energy resources of IoT networks have been studied to improve the energy efficiency through a precaching mechanism and radio frequency chain on-and-off scheduling [10]–[12]. Frequency and/or time resources have been managed by exploiting the heterogeneity and dynam-
massive number of devices, with each method having its advantages and disadvantages. It was reported that NOMA can achieve better spectral efficiency and connectivity performance than OFDMA [17], and to reap the benefits, IoT devices require complex operation, i.e., successive interference cancellation, to reduce the interference due to non-orthogonality. On the other hand, OFDMA enables low-cost functional receivers (i.e., IoT devices) as there is no interference among the devices. To obtain lower system coverage, lower power consumption, and low-cost devices, the subcarrier bandwidth has to be narrow. Therefore, narrow-band OFDMA (NB-OFDMA) has recently attracted the attention of industry and academia. The subcarrier spacing of NB-IoT in the 180 kHz spectrum is 3.75 kHz for uplink NB-IoT and 15 kHz for downlink long-term evolution (LTE) of 3GPP [18]–[20]. As a novel type of NOMA, a sparse code multiple access (SCMA) scheme that combines the concepts of code division multiple access (CDMA) and OFDMA has been also rigorously studied [21], [22].

To support the massive number of devices and increase the energy efficiency, by exploiting the IoT data characteristics, that is, sporadically and randomly generated short data, a discontinuous reception mechanism is defined in LTE and employed by NB-IoT. To exploit the increased degree of freedom of the radio resource in the time domain as well as the frequency domain from the narrow band and to support the massive number of devices, many rigorous studies have been performed by designing scheduling and association methods in the higher layers above the physical layer (PHY) in communication systems. For example, IoT devices are associated (connected to) with the access point (AP) or base stations (BSs) for the uplink transmission through a radio resource control with a scheduling request or random access [20], [23], [24]. Recently, user (device) scheduling in PHY, specifically in the time domain, was studied to further improve the energy efficiency of NOMA-based IoT networks, which is indirectly equivalent to connectivity [25]. As reported in [25], earlier studies implicitly assumed that full connectivity of massive number of devices is possible simultaneously; however, this assumption has not been borne out in practice. Therefore, time-domain scheduling in PHY is required to implement massive connectivity in a practical setting.

For the time-domain scheduling of multiple devices, the simplest scheduler transmits data to each user in a round-robin (RR) fashion, regardless of the channel condition [26]. On the other hand, to exploit multiuser diversity so that the throughput of the network is maximized, a maximum rate (MR) scheduler can be considered [27]. In addition to the rate performance, fairness among the users can be provided by a proportional fairness (PF) scheduler, which has been used for IS-856 systems and is one of the most widely used schedulers [26]–[30]. The PF scheduler provides a designed trade-off between the current rate and throughput (the average rate or achievable rate). The PF scheduler allows a longer delay for a specific device whose throughput is high to improve fairness and/or whose channel condition is poor so that it can wait until the channel condition improves. If the scheduling time scale is much larger than the coherence time of the channels, the throughput of each user converges to the same quantity, resulting in perfect fairness [26], [28]. The convergence of a PF scheduler in uplink communications is analytically proved in [29], and the throughput after the convergence is asymptotically analyzed in [30]. Due to the low complexity and heterogeneous scalability supporting different devices, for example, from WiFi to cellular networks, and different generations of cellular networks [1], the PF scheduler would be one promising function of massive IoT networks.

B. CONTRIBUTION OF THIS STUDY

In this paper, we consider a PF scheduler in downlink communications to support the massive number of users. It is assumed that $K$ users (i.e., a receiver) are associated with a transmitter (Tx), such as an AP or BS, during an uplink communication period before the downlink communications. The $K$ associated users are scheduled in orthogonal time by using a PF scheduler to transmit data. The Tx employs space–time line code (STLC), which was recently proposed as a symbol-level precoding scheme in [31]–[35]. Note that any of the existing source and channel coding schemes can be applied to the considered system in this study. Because the STLC scheme provides full spatial diversity gain and simplifies the receiver (Rx) structure (decoding/combining structure), it has been applied to various systems, such as a two-way relay system [36], [37], secure wireless communication system [38], spatial multiplexing system [39], and a random access system with uplink NOMA [40]. Therefore, the STLC scheme is an appropriate transmission strategy to support multi-antenna (at least two) devices, e.g., the user equipment in cellular networks [31], [32]. Although the STLC Rx can operate without channel state information (CSI) [41], full CSI is required for the optimal STLC design at the Tx. The condition of full CSI at the Tx would be infeasible if the CSI needed to be fed back from all users, and the overall amount of feedback is huge. This is a typical case in the networks with the massive number of IoT devices operating in frequency division duplex (FDD). To resolve this problem, we propose a random STLC that encodes symbols with a randomly generated vector at the Tx. Evidently, the random STLC, which encodes symbols with a random vector, is different from the conventional STLC schemes in [31]–[35] that encode symbols by using the channel gains. However, to compensate for the loss of spatial diversity gain of the random STLC, we propose a PF scheduling method to extract the spatial diversity gain opportunistically from multiple users, such as the opportunistic beamforming strategy used in [42]. For the PF scheduling of massive number of random STLC users, a novel metric is designed based on the signal-to-interference-plus-noise ratio (SINR), called an alternative SINR ($\alpha$SINR). Throughout this study, a homogeneous network is considered, but the results can be readily
extended to heterogeneous networks due to the heterogeneous scalability of the PF scheduler. Numerical results show that the proposed random STLC with a PF scheduler and partial CSI can provide an achievable rate and fairness that are comparable to that of the optimal STLC system with full CSI. Thus, it is confirmed that the proposed scheme is an appropriate candidate for massive networks.

Some of the contributions of this study are summarized as follows:

- We propose a new random STLC scheme that uses a randomly generated vector at a transmitter for the STLC.
- We devise an alternative SINR (aSINR) metric for PF scheduling of the massive number of devices that are supported by the random STLC.
- We modify a PF scheduling algorithm to support the massive number of devices that are supported by the random STLC in homogeneous IoT networks.

The rest of this paper is organized as follows. In Section II, a system model is introduced, and the motivation of this study is further clarified. Section III presents the design of the proposed random STLC and PF scheduling methods. Various numerical results and a discussion are provided in Section IV. Section V concludes this paper with a summary.

Notations: Superscripts $T$, $*$, and $-1$ denote transposition, the complex conjugate, and inversion, respectively, for any scalar, vector, or matrix. $|x|$ and $\|x\|$ denote the absolute value of $x$ and the 2-norm of vector $x$, respectively; and $x \sim \mathcal{CN}(0, \sigma^2)$ indicates that a complex random variable $x$ conforms to a normal distribution with zero mean and variance $\sigma^2$. $E\{x\}$ represents the expectation of a random variable $x$.

II. SYSTEM MODEL AND MOTIVATION

We consider multi-device networks in which one Tx supports $K$ users, as shown in Fig. 1. To simplify the derivation, it is assumed that the TX, namely, AP or BS, has a single antenna for STLC symbol transmission, whereas each user, i.e., an Rx, has two receive antennas. Here, note that the extension to a system with a general number of Tx antennas becomes straightforward if the multi-antenna STLC scheme in [32] is used. The information symbols are denoted by $x_1$ and $x_2$, and the STLC symbols after the encoding are denoted by $s_1$ and $s_2$. With channel vector $h_k = [h_{k,1} h_{k,2}]^T$ of the $k$th device, the typical STLC symbols are constructed as follows [31], [32]:

$$s_1 = \gamma_k \left( h_{k,1}^* x_1 + h_{k,2}^* x_2^2 \right),$$  
$$s_2 = \gamma_k \left( h_{k,2} x_1^* - h_{k,1}^* x_2 \right),$$

where $s_1$ is the STLC symbol that is transmitted at time $t$, and $\gamma_k$ is a transmit power normalization factor that is derived as $\gamma_k = \sqrt{P/\|h_k\|^2}$ such that $E\{s_t^2\} = P$. Here, $P$ is the maximum transmit power of the Tx. Throughout the paper, we assume that each channel coefficient of $h_k$ conforms to an independent complex Gaussian distribution and that the channels are static during $T_c$ and vary after $T_c$ i.e. $T_c$ is the coherence time and the channels are block fading channels. To be specific, the channels from the Tx antenna to the nth antenna of user $k$ are denoted by $h_{k,n} = \sqrt{\rho_k} h_{k,n}$, where $n \in \{1, 2\}$ and $k \in \{1, \ldots, K\}$; $\rho_k$ is a large-scale fading factor; and $h_{k,n}$ is the small-scale fading channel, which is an independent and identically distributed (i.i.d.) random variable with $\mathcal{CN}(0, 1)$ distribution, i.e., Rayleigh fading channels.

Multiple users can be supported by using various types of scheduling methods, such as an RR and PF scheduling methods. The RR scheduler can be simply implemented at the Tx, but does not provide sufficiently fair service to all the users, and the channel capacity is not well utilized, resulting in a relatively low average achievable rate compared to the PF scheduler. A PF scheduler is thus more widely used than an RR scheduler. To employ the PF scheduler, however, the Tx should estimate the expected achievable rate of all users, which is achieved by estimating the effective signal-to-noise ratios (SNRs) or SINRs. For the STLC with PF scheduling, the full CSI, $\{h_1, \ldots, h_K\}$, should be known at the Tx. Note that the SNRs or SINRs can be obtained from the full CSI. To this end, all users need to estimate the CSI and feed it back to the Tx. This is true for an FDD system, in which the downlink channel from the Tx to the users and uplink channels from the users to the Tx are asymmetric. In other words, during the scheduling period $T$ for $K$ users, $2K [T/T_c]$ complex values should be fed back from the users. This would be infeasible owing to the significant uplink network overhead if $K$ is very large or the coherent time of the channels, denoted by $T_c$, is very small. To resolve the problem, we propose a new STLC method called random STLC that uses a randomly generated vector to construct STLC symbols. Furthermore, for PF scheduling of STLC users, we devise a novel metric, i.e., the aSINR.

III. PROPOSED RANDOM STLC WITH PF SCHEDULING

In this section, to reduce the signaling overhead for scheduling the massive number of users while maintaining the STLC Rx structure, we propose a new STLC using a randomly generated vector at the Tx, and devise a new metric for PF scheduling of multiple users.
A. RANDOM STLC METHOD

Instead of the channel vector shown in (1), the new STLC employs a randomly generated vector \( \mathbf{w} = [w_1 \ w_2]^T \in \mathbb{C}^{2 \times 1} \) for the STLC as follows:

\[
\begin{align*}
    s_1 &= \gamma (w_1^* x_1 + w_2^* x_2^*), \\
    s_2 &= \gamma (w_2^* x_1^* - w_1^* x_2^*),
\end{align*}
\]  

where \( w_1 \sim \mathcal{CN}(0, 1) \), \( w_2 \sim \mathcal{CN}(0, 1) \), and \( \gamma = \sqrt{\mathbb{E}[|w|^2]}^{-1} \). The received signals of user \( k \) after two consecutive transmissions of \( s_1 \) and \( s_2 \) in (2) are then written as follows:

\[
\begin{align*}
    r_{k,1,1} &= h_{k,1}s_1 + v_{k,1,1}, \\
    r_{k,1,2} &= h_{k,1}s_2 + v_{k,1,2}, \\
    r_{k,2,1} &= h_{k,2}s_1 + v_{k,2,1}, \\
    r_{k,2,2} &= h_{k,2}s_2 + v_{k,2,2},
\end{align*}
\]

where \( r_{k,n,t} \) is the received signal at the \( n \)th antenna of user \( k \) at time \( t \), and \( v_{k,n,t} \) is the additive white Gaussian noise (AWGN) with zero mean and variance \( \sigma^2 \), i.e., \( v_{k,n,t} \sim \mathcal{CN}(0, \sigma^2) \).

User \( k \) then combines the received signals, i.e., STLC decoding as follows [31, 32]:

\[
\begin{align*}
    r_{k,1,1} + r_{k,2,2} &= \gamma h_{k,1}(w_1^* x_1 + w_2^* x_2^*) \\
    &\quad + \gamma h_{k,2}(w_2 x_1 - w_1 x_2) \\
    &\quad + v_{k,1,1} + v_{k,2,2}, \\
    r_{k,2,1} - r_{k,1,2} &= \gamma h_{k,2}(w_1 x_1^* + w_2 x_2) \\
    &\quad - \gamma h_{k,1}(w_2 x_1^* - w_1 x_2) \\
    &\quad + v_{k,2,1} - v_{k,1,2},
\end{align*}
\]

where \( v_{k,1} = v_{k,1,1} + v_{k,2,2} \) and \( v_{k,2} = v_{k,2,1} - v_{k,1,2} \).

Here, (4a) and (4c) are the intended signals for estimates \( x_1 \) and \( x_2 \), respectively, and (4b) and (4d) are the inter-symbol-interferences (ISIs) plus AWGNs. Therefore, from (4), the effective SINR (eSINR) of user \( k \) is readily derived as follows:

\[
\text{eSINR}_k = \frac{\mathbb{E}|\gamma (h_{k,1} x_1^* + h_{k,2} x_2)|^2}{\mathbb{E}|\gamma (h_{k,1} x_2^* - h_{k,2} x_1)|^2 + \mathbb{E}|h_{k,1} x_1^* + h_{k,2} x_2|^2} = \frac{|h_{k,1} x_1^* + h_{k,2} x_2|^2}{|h_{k,1} x_2^* - h_{k,2} x_1|^2 + 2\sigma^2/\gamma^2}. 
\]  

Note that (5) is derived from (4a) and (4b) for \( x_1 \), and it is identical to the eSINR derived from (4c) and (4d) for \( x_2 \). Because the STLC is performed by using the randomly generated vector \( \mathbf{w} \), ISI appears after the STLC decoding, and the corresponding eSINR depends on how well the random vector \( \mathbf{w} \) matches the actual channel vectors. Thus, to operate a PF scheduler for the random STLC, the eSINRs of all users are required at the Tx.

Evidently, the eSINRs in (5) should be estimated at each user and fed back to the Tx for the operation of PF scheduling. Note that the eSINR is represented by a single real value, which is a smaller amount of information compared to the full CSI consisting of two complex values (i.e., four real values). Thus, for each feedback of the eSINR instead of the full CSI, the feedback amount is reduced by one quarter. This is a crucial benefit that mainly stems from the random STLC. However, there are two disadvantages of employing the eSINR for the PF scheduler. The first one is that user \( k \) needs to estimate the channels and all the elements of the eSINR, namely, \( h_{k,1}, h_{k,2}, h_{k,1} x_1^*, h_{k,2} x_2, h_{k,1} w_1, h_{k,2} w_2, h_{k,1} w_2^*, \) and \( h_{k,2} w_1^* \). This estimation increases the computational complexity of user \( k \) compared to the exact estimation of \( h_{k,1} \) and \( h_{k,2} \). The second drawback is that the eSINR estimation requires an additional function/module at the user’s end, because the structure of the training symbols broadcasted from the Tx, i.e., \( 1, w_1^*, \) and \( w_2^* \), is not an STLC structure. To circumvent the drawbacks of the eSINR, in the next subsection, we devise an alternative SINR (aSINR) and a method to estimate it while retaining the STLC decoding structure at the user’s end.

B. ALTERNATIVE SINR

To acquire an aSINR at the Rx, it is considered that the Tx sends two additional STLC signals \( s_3 \) and \( s_4 \) after transmitting \( s_1 \) and \( s_2 \) in (2). The additional STLC symbols are constructed in the new STLC form as follows:

\[
\begin{align*}
    s_3 &= \gamma (w_1^* x_1 - w_2^* x_2^*), \\
    s_4 &= \gamma (w_2^* x_1^* + w_1^* x_2^*).
\end{align*}
\]

In (6), we see that the encoding structure is the same as that for \( s_1 \) and \( s_2 \), whereas the second information symbol \( x_2 \) has a different sign. Similarly, the received signals are written as

\[
\begin{align*}
    r_{k,1,3} &= h_{k,1}s_3 + v_{k,1,3}, \\
    r_{k,1,4} &= h_{k,1}s_4 + v_{k,1,4}, \\
    r_{k,2,3} &= h_{k,2}s_3 + v_{k,2,3}, \\
    r_{k,2,4} &= h_{k,2}s_4 + v_{k,2,4},
\end{align*}
\]

and the combined signals are then obtained at user \( k \) as follows:

\[
\begin{align*}
    r_{k,1,3} + r_{k,2,4} &= \gamma h_{k,1}(w_1^* x_1 - w_2^* x_2^*) \\
    &\quad + \gamma h_{k,2}(w_2 x_1 + w_1 x_2) \\
    &\quad + v_{k,1,3} + v_{k,2,4}, \\
    r_{k,2,3} - r_{k,1,4} &= \gamma h_{k,2}(w_1 x_1^* + w_2 x_2) \\
    &\quad - \gamma h_{k,1}(w_2 x_1^* - w_1 x_2) \\
    &\quad + v_{k,2,3} - v_{k,1,4},
\end{align*}
\]

where \( v_{k,3} = v_{k,1,3} + v_{k,2,4} \) and \( v_{k,4} = v_{k,2,3} - v_{k,1,4} \).
Again, the user adds the previously combined STLC signals in (4) to the newly combined STLC signals in (8), and obtains the following signals:

\[
(r_k,1,1 + r_k,2,2) + (r_k,1,3 + r_k,2,4) \\
= 2\gamma(h_k,1w_1^* + h_k,2w_2)x_1 + v_k,1 + v_k,3,
\]

\[(9a)\]

\[
(r_k,1,1 - r_k,3,2) - (r_k,2,3 - r_k,1,4) \\
= 2\gamma(h_k,1w_1^* - h_k,2w_2)x_1 + v_k,2 - v_k,4,
\]

\[(9b)\]

\[
(r_k,1,1 + r_k,2,2) - (r_k,1,3 + r_k,2,4) \\
= 2\gamma(h_k,1w_1^* + h_k,2w_2)x_1 + v_k,1 - v_k,3,
\]

\[(9c)\]

\[
(r_k,2,1 - r_k,1,2) + (r_k,2,3 - r_k,1,4) \\
= 2\gamma(h_k,1w_1^* - h_k,2w_2)x_1 + v_k,2 + v_k,4,
\]

\[(9d)\]

where \((9a)\) and \((9b)\) are the intended signals for \(x_1\) and \(x_2\), respectively, and \((9c)\) and \((9d)\) are the ISI signals for \(x_1\) and \(x_2\), respectively. From the right-hand side (RHS) of \((9a)\) and \((9c)\), user \(k\) then calculates the following ratio:

\[
E[(\text{RHS of } (9a))^2] \\
E[(\text{RHS of } (9c))^2] + 4\gamma^2 \\
= E|\sqrt{\gamma}(h_k,1w_1^* + h_k,2w_2)x_1 + v_k,1 + v_k,3|^2 \\
= E|\sqrt{\gamma}(h_k,1w_1^* + h_k,2w_2)x_1 + v_k,2 - v_k,4|^2 + 4\gamma^2 \\
= |h_k,1w_1^* + h_k,2w_2|^2 + \sigma_v^2/\gamma^2 \\
= |h_k,1w_1^* - h_k,2w_2|^2 + 2\gamma^2/\gamma^2 \\
= a\text{SINR}_k + \epsilon(\sigma_v^2),
\]

\[(10)\]

where \(\epsilon\) is the SINR estimation error defined as

\[
\epsilon(\sigma_v^2) = \frac{\sigma_v^2/\gamma^2}{|h_k,1w_1^* - h_k,2w_2|^2 + 2\gamma^2/\gamma^2}.
\]

\[(11)\]

Note that \(\lim_{\sigma_v^2 \to 0} \epsilon(\sigma_v^2) \to 0\), whereas \(\lim_{\sigma_v^2 \to \infty} \epsilon(\sigma_v^2) \to 1/2\). Because \(\epsilon(\sigma_v^2)\) is a monotonically increasing function bounded by 0.5, \((12)\) is a good alternative to the effective SINR in \((5)\).

Further comparing \((5)\) and \((10)\), it is evident that both \((5)\) and \((10)\) approach the signal-to-interference ratio as \(\sigma_v\) decreases. In other words, \((10)\) is a good alternative to the eSINR when the noise power is small. On the other hand, if the random vector \(w\) is perfectly matched with the channels of user \(k\), i.e., \(w = h_k\), the aSINR in \((10)\) becomes an upper bound of the eSINR in \((5)\) owing to the nonnegative \(\epsilon(\sigma_v^2)\). From these properties, the aSINR is again justified as a good alternative to the eSINR. Here, dropping the expectation by transmitting the minimum number of required training sequences, i.e., only two symbols \(x_1 = x_2 = 1\), and using \((9)\), we formally define the aSINR as follows:

\[
a\text{SINR}_k \triangleq \frac{|r_k,1,1 + r_k,2,2 + r_k,1,3 + r_k,2,4|^2}{|r_k,1,1 + r_k,2,2 - r_k,1,3 - r_k,2,4|^2 + 4\gamma^2}. 
\]

\[(12)\]

Additional validation of the designed aSINR is obtained from Fig. 2, where we observe that the aSINR matches well with the eSINR as \(1/\sigma_v^2\) increases. As further corroborated by the numerical results in Section IV, the new metric aSINR \(_k\) operates well (i.e., it provides an improved achievable rate with fairness) with a PF scheduler, and it is a good alternative to the eSINR when \(\sigma_v^2\) is low. Furthermore, it is worth reemphasizing that users can obtain aSINRs by just combining their received signals without significant changes in the Rx structure and the explicit channel estimation. The comparison of the eSINR and aSINR is summarized in Table 1.

### C. PF SCHEDULING METHOD

We modify a PF scheduler at a Tx to support random STLC users. The main purpose of a PF scheduler is to improve the overall throughput of the networks and to support users fairly as much as possible. Thus, the fundamental policy of the PF scheduler is to select a user based on a scheduling priority. The scheduling priority of a user is proportional to the requested data rate, which is the achievable rate at the scheduling time (or slot), denoted by \(i\), to improve the overall network throughput. On the other hand, the priority is inversely proportional to the average throughput over the duration before the scheduling time \(i\) for fairness among the users. To be specific, the scheduling priority of user \(k\) at

| Metric | Training symbols (STLC structure) | Estimation at user \(k\) | Feedback information | Total feedback amount |
|--------|----------------------------------|-------------------------|---------------------|----------------------|
| eSINR  | \(1, w_1^*, w_2^*\) (No)        | \(h_{k,1}, h_{k,2}, h_{k,1}w_1^*, h_{k,2}w_2^*\), \(\text{eSINR}_k\) in \((5)\) | \{eSINR\}_k | \(K\) real values |
| aSINR  | \(s_1, s_2, s_3, s_4\) (Yes)    | \(a\text{SINR}_k\) in \((12)\) | \{aSINR\}_k | \(K\) real values |

### TABLE 1. Comparison of metrics.

**FIGURE 2. Comparison of exact SINR SINR\(_k\) in \((5)\) and an alternative SINR a\text{SINR}_k\) in \((12)\).**
scheduling time/slot $i$ is defined as follows:

$$p_k[i] = \frac{R_k[i]}{(i-1)^{-1} \sum_{i'=1}^{i-1} \overline{R}_k[i']}, \quad \forall i \in \{1, \ldots, T\}, \quad (13)$$

where $R_k[i]$ is the requested data rate from user $k$ at $i$, and it is obtained by the Shannon limit as follows [12], [26]:

$$R_k[i] = \log_2(1 + \text{eSINR}_k), \quad (14)$$

by treating the interference as noise. In (13), $\overline{R}_k[i']$ is the actually supported data rate of user $k$ at $i'$, i.e.,

$$\overline{R}_k[i'] = \begin{cases} R_k[i'], & \text{if user } k \text{ is scheduled at } i', \\ 0, & \text{o.w.} \end{cases} \quad (15)$$

At scheduling time $i$, the Tx selects the one user who has the highest priority as

$$k' = \max_k p_k[i], \quad (16)$$

and transmits the STLC symbols generated in (2) to user $k'$. The average achievable rate of scheduled user $k'$ increases at the next scheduling time $i + 1$, which decreases the priority of this user at the $(i+1)$th scheduling time. Thus, the fairness can be improved.

**D. PROPOSED PF SCHEDULING ALGORITHM FOR THE MASSIVE NUMBER OF USERS**

In this subsection, we summarize the procedure of PF scheduling with the designed aSINR$_k$ at scheduling time $i$:

**Step 1:**

1) The Tx generates a random vector $w = [w_1 \ w_2]^T$.
2) The Tx broadcasts STLC training symbols defined as

$$s_1 = \|w\|^{-1}(w_1^* + w_2^*),$$
$$s_2 = \|w\|^{-1}(w_2^* - w_1^*),$$
$$s_3 = \|w\|^{-1}(w_2^* - w_1^*),$$
$$s_4 = \|w\|^{-1}(w_1^* + w_2^*),$$

in (2) and (6), where $x_1 = x_2 = 1$.
3) User $k$ calculates the aSINR$_k$ in (12) by combining the eight received signals $r_{k,n,i}$, where $n = 1, 2$ and $t = 1, 2, 3, 4$.
4) User $k$ sends the aSINR$_k$ back to the Tx.
5) The Tx calculates the expected achievable rates of all users based on aSINR$_k$ as follows:

$$R_k[i] \triangleq \log_2(1 + \text{aSINR}_k), \quad \forall k.$$  

6) The Tx calculates the priority of user $k$ as follows:

$$p_k[i] = \frac{R_k[i]}{(i-1)^{-1} \sum_{i'=1}^{i-1} \overline{R}_k[i']}, \quad \forall k,$$

where $\overline{R}_k[i']$ is the actually supported data rate of user $k$ at $i'$ similar to (15).
7) The Tx transmits STLC data symbols to the selected/scheduled user $k'$ who has the highest priority $p_k[i]$ during the $(T_c - T_{\text{sig}})$ period, where $T_{\text{sig}}$ is the signaling time from Step 1 to Step 5.
8) Update $i = i + 1$, and repeat Steps from 1 to 6 if $i \leq [T/T_c]$.

Throughout the simulation in the next section, we assume that $T_{\text{sig}}$ and $T_c$ are the four- and six-symbol times, respectively. In other words, fast block fading channels are assumed in our simulation.

**IV. NUMERICAL RESULTS**

The average/sum achievable rates and fairness are evaluated numerically to verify the proposed random STLC with the PF scheduling method. In the comparison, we consider the following four systems as follows:

- **Optimal STLC and maximum rate scheduler with full CSI (Opt-MR-FCSI):** Under the assumption that the Tx knows the full CSI, the optimal STLC is performed with the exact channels, i.e., $w = h_k$. The maximum rate scheduler uses the exact SINR in (5) and schedules a user who achieves the maximum rate in each scheduling time. This scheme provides the upper bound of rate.
- **Optimal STLC and a PF scheduler with full CSI (Opt-PF-FCSI):** Under the assumption that the Tx knows the full CSI, the optimal STLC is performed with the exact channels, i.e., $w = h_k$. The PF scheduler uses the exact SINR in (5). This scheme is a benchmarking scheme for the proposed scheme.
- **Random STLC and a PF scheduler with partial CSI (Ran-PF-PCS):** Random STLC is used with the randomly generated channel vector $w$. The PF scheduler uses the partial CSIs, i.e., the alternative SINRs in (12). This is our proposed system.
- **Random STLC and an RR scheduler with no CSI (Ran-RR-NCSI):** Random STLC is used with the randomly generated channel vector $w$. No CSI is used for the RR scheduling. This benchmarking system is used as the performance lower bound.

Jain’s fairness index is used as the fairness metric, which is defined for one scheduling period $T$ as [43]

$$J = \frac{\left( \sum_{k=1}^{K} \sum_{i=1}^{T} \overline{R}_k[i] \right)^2}{K \sum_{k=1}^{K} \sum_{i=1}^{T} \overline{R}_k[i]^2}. \quad (17)$$

The fairness index is obtained by averaging out the results over multiple scheduling realizations for the independent channels.

**A. RESULTS FOR VARIOUS $P/\sigma^2$ AND $K$**

The achievable rates and the fairness indices of three systems are compared in Fig. 3. The results in Figs. 3(a) and 3(b) are obtained when $K = 100$ and $T = 5,000$, whereas the results in Figs. 3(c) and 3(d) are obtained when $K = 1,000$ and $T = 5,000$. As shown in Figs. 3(a) and 3(c), the achievable rate of each user increases as the SNR $P/\sigma^2$ increases; however, the fairness does not vary according to the SNR as shown in Figs. 3(b) and 3(d). From Figs. 3(a) and 3(c), as expected, Opt-MR-FCSI achieves the highest achievable
rate per user irrespective of the number of users, scheduling period, and SNR, i.e., $K$, $T$, and $P/\sigma^2_0$, respectively. However, as shown in Figs. 3(b) and 3(d), the fairness performance of the Opt-MR-FCSI scheme is relatively worse than that of the other schemes. It is evident that there exists an tradeoff between the achievable rate and fairness. In contrast, the Ran-RR-NCSI scheme provides the worst achievable rate performance with insufficient fairness. Note that Ran-RR-NCSI does not require any CSI and feedback information. From the results, it is evident that the Opt-PF-FCSI scheme provides an almost optimal performance with respect to both the achievable rate and fairness. To implement the Opt-PF-FCSI scheme, however, full CSI is required at the transmitter, and simultaneously, the eSINR is also required for the scheduling, which increases the feedback amount to $5K$ (refer to Table 1). On the other hand, the proposed Ran-PF-PCSI scheme achieves a performance that is comparable that of the Opt-PF-FCSI scheme, but with a lower feedback amount $K$.

In Figs. 4(a) and 4(b), the achievable rates and fairness indices among the users are compared over the number of users, $K$. As shown in Fig. 4(a), the achievable rate of all the schemes decreases as $K$ increases owing to the increase in interferences among the users. From the results, it is also evident that the proposed Ran-PF-PCSI scheme achieves a similar achievable rate to that of the Opt-PF-FCSI scheme. The Ran-RR-NCSI scheme exhibits poor performance. On the other hand, as shown in Fig. 4(b), the fairness of the Opt-PF-FCSI scheme is almost stable against the number of users. The fairness of the other schemes decreases as $K$ increases. Here, we note that the fairness decrease is marginal for the proposed Ran-PF-PCSI scheme, whereas the decrease is severe for the STLC system that uses MR or RR scheduling, especially, when $K$ is large. Moreover, it is observed that
FIGURE 5. Achievable rate and fairness comparison over the mean-squared error (MSE) when $K = 1000$ and $T = 5000$. (a) Achievable rate. (b) Fairness index.

In Fig. 5, the achievable rate and fairness are evaluated across the channel estimation errors. We denote an estimated CSI at the transmitter by $\tilde{h}_{k,n} \triangleq h_{k,n} + \epsilon$, where $h_{k,n}$ is the actual channel element, and $\epsilon$ is the estimation error that conforms to a normal distribution with zero mean and variance $\sigma^2_\epsilon$. We represent the mean-squared error (MSE) of the estimation by $\epsilon^2$, i.e., $\mathbb{E} |h_{k,n} - \tilde{h}_{k,n}|^2 = \sigma^2_\epsilon$. As shown in Fig. 5, because the random STLC schemes, i.e., Ran-PF-PCSI and Ran-RR-NCSI, do not use CSI, their achievable rates do not vary with respect to the MSE. On the other hand, the achievable rates of the conventional STLC schemes using CSI, i.e., Opt-MR-FCSI and Opt-PF-FCSI, decrease as the MSE increases. This is because the mismatch between STLC precoding and actual channels increases as MSE increase, which causes the received SINR degradation and also the uncertainty of the priority of users in the scheduling. When the CSI error is severe, the proposed Ran-PF-PCSI scheme outperforms the conventional STLC-based schemes. From the results in Fig. 5, it is observed that CSI uncertainty does not affect the fairness of all the schemes.

B. RESULTS WITH DISTRIBUTED USERS

In this subsection, practical parameters, such as the transmit power of the Tx, large-scale fading, and AWGN, are considered. To this end, a simplified path-loss model in [44] is used: $\rho_k = G + 10\log_{10}(d_{k}^{-\mu})$ in the dB scale, where $G = -23.4$ dB is a constant path-loss factor that is the sum of the antenna gain, $15$ dBi, and a path-loss factor $(c/(4\pi f))^2$, carrier frequency $f = 2$ GHz, and $c = 3 \times 10^8$ m/s; $d_k$ is the distance between the Tx and user $k$; and $\mu$ is a path-loss exponent. We set the path-loss exponent as $\mu = 3.76$, and assume that the small-scale fading conforms to the Rayleigh distribution with zero mean and unit variance. The maximum transmit power of the Tx, $P$, is either $23$ dBm or $43$ dBm. The low transmit power $P = 23$ dBm is for low-power transmitters, such as WiFi APs, fusion centers, and micro BSs, while the high...
transmit power $P = 43$ dBm is for high-power transmitters, such as macro BSs. The noise figure is set to $-174$ dBm/Hz. The users are located outside a circle of a radius of 5 m, and within a circle of radius 500 m, where the centers of the two circles are coincide. Here, the Tx is located at the center. Under these conditions, the cumulative distribution function (CDF) of the achievable rates is obtained from $10^3$ independent location realizations of the users.

Figs. 6(a) and 6(b) show the CDFs of the achievable rates when $P = 23$ dBm and $P = 43$ dBm, respectively. The slope of the CDF lines implies fairness among the users. The steeper the line, the better the fairness e.g., an infinite slope means perfect fairness, and zero slope means no fairness at all. A CDF line is located further to the RHS when $K$ means a higher achievable rate. As expected, a better performance is achieved when $K$ is large (i.e., $K = 20$) and $P$ is large (i.e., $P = 43$ dBm in Fig. 6(b)). From the results, it is evident that the proposed Ran-PF-PCSI outperforms Ran-RR-NCSI with respect to both the achievable rate and fairness. Furthermore, Ran-PF-PCSI is comparable to an optimal Opt-PF-FCSI.

For a quantitative comparison, the average values of the achievable rates and fairness indices are shown in Table 2, and the values normalized by the maximum value in each environment are shown in Fig. 7. From a comparison of the results in the table and bar chart (i.e., Figs. 7(a)–7(d)), we observe that the proposed Ran-PF-PCSI scheme provides a greater achievable rate compared to the Opt-PF-FCSI scheme, with marginal fairness loss, when $P = 23$ dBm regardless of the number of devices. On the other hand, a non-negligible rate decrease and marginal fairness loss of Ran-PF-PCSI are observed when $K = 20$ and $P = 43$ dBm in Figs. 7(e) and 7(f). The reason for this is that the diversity gain from the opportunistic scheduling is marginal, and the achievable rate is sensitive to the scheduling accuracy because the effect of noise is negligible. However, note that the gain of Ran-PF-PCSI compared to Ran-RR-NCSI is very large, i.e., a rate improvement of more than 200%. Furthermore, as the number of users, $K$, increases, as shown in Figs. 7(g) and 7(h), the proposed Ran-PF-PCSI achieves almost the same performance as the Opt-PF-FCSI when $K = 500$ and $P = 43$ dBm.

In conclusion, if the channel varies slowly or is static, the number of users is small, and sufficient power is supplied to the Tx, full CSI feedback is relevant, and Opt-PF-FCSI would be a good option. Otherwise, in general, the proposed Ran-PF-PCSI is recommended as a promising option for an STLC-based IoT system.

### REFERENCES

1. Q. Chi, H. Yan, C. Zhang, Z. Pang, and L. Da Xu, “A reconfigurable smart sensor interface for industrial WSN in IoT environment,” *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1417–1425, May 2014.
2. S. Chen, H. Xu, D. Liu, B. Hu, and H. Wang, “A vision of IoT: Applications, challenges, and opportunities with China perspective,” *IEEE Internet Things J.*, vol. 1, no. 4, pp. 349–359, Aug. 2014.
3. M. Chiang and T. Zhang, “Fog and IoT: An overview of research opportunities,” *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854–864, Dec. 2016.
4. Intel Breaks Guinness World Records Title and Lights Up the Sky at Winter Olympics With Intel Shooting Star Drones. Intel News Fact Sheet, Feb. 2018. [Online]. Available: https://newsroom.intel.com
5. Study on RAN Improvements for MTC, document TR 37.868, 3GPP, Sep. 2011.
