Normalized Delivery Time of Wireless MapReduce

Yue Bi[*†], Michèle Wigger[†], Yue Wu[*]
[*]*School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University*, China;
wuyue@sjtu.edu.cn
[†]*LTCI, Telecom Paris, IP Paris*, 91120 Palaiseau, France
{bi, michele.wigger}@telecom-paris.fr

Abstract

We consider a full-duplex wireless Distributed Computing (DC) system under the MapReduce framework. New upper and lower bounds on the optimal tradeoff between Normalized Delivery Time (NDT) and computation load are presented. The upper bound strictly improves over the previous reported upper bounds and is based on two novel interference alignment (IA) schemes tailored to the interference cancellation capabilities of the nodes. Our second IA scheme additionally applies a zero-forcing strategy that allows to accumulate all interference at any of the nodes on the same (small) subspace, leaving the remaining space for useful signals. The lower bound is proved through information-theoretic converse arguments based on carefully chosen multi-access channel (MAC) type arguments and by finding solutions to the optimization problems resulting from these arguments. The lower bound matches an existing upper bound based on zero-forcing and interference cancelation (but no IA) in the regime where each node can store at least half of the files. While optimal in this regime, zero-forcing and interference cancelation are not sufficient to obtain the optimal NDT in scenarios where each node cannot store half of the files. This follows from the previously established optimal NDT under zero-forcing and interference cancelation and our new IA-schemes.

Index Terms

Wireless distributed computing, MapReduce, coded computing, interference alignment.

I. Introduction

Distributed Computing (DC) systems are computer networks that through task-parallelization reduce execution times of complex computing tasks such as data mining or computer vision. MapReduce is a popular such framework and runs in three phases [1], [2]. In the first *map phase*, nodes calculate intermediate values (IVA) from their associated input files. In the following *shuffle phase*, nodes exchange these IVAs in a way that each node obtains all IVAs required to compute its assigned output function in the final *reduce phase*. MapReduce is primarily applied to wired systems where it has been noticed that a significant part of the MapReduce execution time stems from the IVA *delivery time* during the shuffle phase [2], [3]. Various coding schemes [3]–[8] were proposed to reduce this IVA delivery time, and consequently speed up execution time compared to naive approaches.

In recent years, MapReduce systems became increasingly popular also for wireless scenarios, such as vehicular networks [9] or distributed e-health applications [10], thus creating a need for coding schemes that perform well over wireless networks in the context of MapReduce systems. Similarly to the wired case [4]–[6], delivery time in wireless MapReduce systems can be decreased by sending appropriate linear combinations of the IVAs, from which the receiving nodes can extract their desired IVAs by bootstrapping the IVAs that they can compute from their locally stored input files. Further improvements are however possible by exploiting the superposition nature of wireless networks, e.g., by cooperatively encoding messages, zero-forcing transmissions at specific sets of nodes, or aligning interferences at nodes.

The focus of this paper is on the high Signal-to-Noise Ratio (SNR) regime, and on the following two key metrics of wireless MapReduce systems:

- *Computation load* r: This describes the average number of nodes to which each file is assigned. In other words, it is the ratio of the total number of assigned input files (including replications) normalized by the total number of files.
- *Normalized Delivery Time (NDT)* Δ: This is the wireless shuffle duration normalized by the number of reduce functions and input files and by the transmission time of one IVA over a point-to-point channel in the high SNR regime.

We are interested in the minimal NDT under a fixed computation load, which we define as the *NDT-computation tradeoff*.

Several works have analyzed NDT-computation tradeoffs for different wireless networks. For example, [11], [12] studied the NDT-computation tradeoff of wireless *cellular* networks, and proposed schemes to reduce the NDT by sending appropriate linear combinations of the IVAs and applying simple interference cancellation (bootstrapping of known IVAs) at the receiving nodes. (The energy-efficiency latency tradeoff in such wireless cellular systems has been studied in [13].) Interference networks were studied in [14]–[17]. More specifically, [14] considered a half-duplex interference network and proposed a scheme that converts the network into a fully-connected X-channel and applies the IA-scheme in [18] for this X-channel. The NDT-computation tradeoff of full-duplex interference networks was considered in [15], [16]. The work [15] proposed a coding scheme based on one-shot beamforming and zero-forcing, and showed that this scheme is optimal for this class of strategies. The works in [14]–[16] all assumed perfect channel state information at the transmitters. For scenarios with imperfect (delayed) channel-state information a better NDT is achievable by combing zero-forcing and interference cancellation with superposition coding, see [17].

The benefits of interference alignment (IA) [19] for DC systems were first proved in [16] for full-duplex wireless networks. Specifically, [16] proposed to divide the nodes into groups and to use a combination of IA and zero-forcing so that signals intended for a given node do not interfere the signals received at the other nodes in the same group.

In this paper, we further improve the NDT-computation tradeoff of MapReduce over full-duplex wireless interference channels with two novel IA schemes. Our first scheme is inspired by the IA scheme in [18], where multi-cast messages are sent over a fully-connected interference network. We however adapt this scheme to our DC setup, where nodes simultaneously act as a transmitter and a receiver, allowing to achieve improved performances. In fact, the equivalence of transmitting and receiving terminals implies only a partial connectivity in the corresponding interference network, which we can exploit through an improved reutilization of IA precoding matrices compared to [18]. More in detail, the IA scheme in [18] utilizes a different IA precoding matrix for all transmissions to a given set of r receivers. In our scheme we can omit the precoding matrices that correspond to a receive set containing index 1. As a consequence, nodes $2, \ldots, \mathsf{K}$ suffer from fewer interference spaces, thus leaving a larger part of their receive dimensions as signal space and resulting in a improved performance. In the special case of $\mathsf{r} = 1$, i.e., when each file is stored only at a single node, this idea was already proposed by the authors in [16], and in fact in this special case $\mathsf{r} = 1$, our first scheme coincides with the scheme in [16].

We present a second IA-DC scheme for systems with an odd number of users K and computation load $\mathsf{r} = \frac{\mathsf{K}-1}{2\mathsf{K}}$, i.e., when each node can store almost half of the input files. In this second scheme each node only sees interference pertaining to one of the K utilized IA-precoding matrices, while all other transmissions at this node are zero-forced. Each set of r nodes that cooperatively transmits an IVA to a given receiver zero-forces this transmission to all other receivers except for the intended receiver and the receiver that corresponds to the utilized IA-precoding matrix. We point out that zero-forcing influences the construction of the precoding matrices. Under these new constructions it is more complicated to prove that at any receiver all signals corresponding to a given precoding matrix either align or vanish due to zero-forcing. We facilitate this proof by a careful assignment of the precoding matrices to the transmitted IVAs.

The upper bound on the NDT implied by our two new IA-schemes improves over the previously proposed bounds in [15], [16] whenever the computation load $1 < \mathsf{r} < \frac{\mathsf{K}}{2}$, i.e., when each file can be stored at more than one node, but each node cannot store half of the total number of files. As already mentioned, for $\mathsf{r} = 1$ our upper bound recovers the NDT-bound in [16], which for this value improves over the NDT upper bound in [15]. Since [15] proved that its proposed scheme achieves the optimal NDT when one restricts to zero-forcing and interference cancellation, our results show that these techniques are strictly suboptimal for all computation loads $1 \leq \mathsf{r} \leq \frac{\mathsf{K}}{2}$, i.e., whenever nodes cannot store half of the number of input files. On the contrary, in this manuscript we show that for $\mathsf{r} \geq \frac{\mathsf{K}}{2}$ the zero-forcing and interference cancellation scheme in [15] is optimal also without any restriction on the utilized coding scheme.

In fact, we also present an information-theoretic lower bound on the NDT-computation tradeoff based on a multi-access channel (MAC) type argument that is applied in parallel to a set of well-selected sub-systems and by solving a resulting linear programme. For computation load $\mathsf{r} < \frac{\mathsf{K}}{2}$ the lower bound on the NDT is close to the proposed upper bound, but they do not match. As mentioned, for $\mathsf{r} \geq \frac{\mathsf{K}}{2}$ the lower bound matches the upper bound in [15] thus establishing the exact NDT of wireless MapReduce over full-duplex networks.

To summarize, the main contributions of this paper are:

- Improved coding schemes based on IA for wireless MapReduce over full-duplex interference networks. (Sections IV–VI and Theorem 1 and Corollary 2)
- A lower bound (converse) on the NDT of wireless MapReduce systems. (Theorem 1)
- The exact NDT of wireless MapReduce for computation loads $\mathsf{r} \geq \lceil \frac{\mathsf{K}}{2} \rceil$. (Corollary 1)
- Proof that zero-forcing and interference cancellation cannot achieve the NDT tradeoff for all computation loads $1 < \mathsf{r} < \lceil \frac{\mathsf{K}}{2} \rceil$. (Remark 2)

Organization: We terminate this section with notation. The following Section II describes the detailed system model, while Section III presents and discusses our bounds on the NDT tradeoff. Sections V and VI explain our two novel IA schemes, where in Section IV, we first describe our first novel IA scheme using some simple examples. Section VII proofs our NDT-lower bound.

Notations: We use sans serif font for constants or matrices, bold for vectors, and calligraphic font for most sets. The sets of complex numbers and positive integers are denoted $\mathbb{C}$ and $\mathbb{Z}^+$. For a finite set $\mathcal{A}$, let $|\mathcal{A}|$ denote its cardinality. For any $n \in \mathbb{Z}^+$, define $[n] \triangleq \{1, 2, \ldots, n\}$ and define $[\mathcal{A}]^n$ as the collection of all the subsets of $\mathcal{A}$ with cardinality n, i.e. $[\mathcal{A}]^t \triangleq \{\mathcal{T} \colon \mathcal{T} \subset \mathcal{A}, |\mathcal{T}| = t\}$. In particular, $[[n]]^t$ denotes the set of all size-t subsets of $[n]$. For vectors $\mathbf{v}$ we use $|\mathbf{v}|$, and we denote its i-th element by v_i. The transpose of a vector $\mathbf{v}$ is denoted $\mathbf{v}^T$. By writing $[\boldsymbol{v}_i \colon i \in \mathcal{A}]$ or $[\boldsymbol{v}_i]_{i \in \mathcal{A}}$ we mean the matrix consisting of the columns $\{\mathbf{v}_i\}_{i \in \mathcal{A}}$. We denote the transpose of a matrix $\mathbf{A}$ by $\mathbf{A}^T$ and We abbreviate *independent and identically distributed* by *i.i.d.*, and for any function f we denote by $\mathrm{lowc}\,(f(\ell))$ the convex lower enveloppe of the curve $\{(\ell, f(\ell))\}_{\ell}$.

II. Wireless MapReduce Framework

Consider a distributed computing (DC) system with a fixed number of K nodes labelled $1, \ldots, \mathsf{K}$; a large number N of input files $W_1, \ldots, W_\mathsf{N}$; and K output functions $\phi_1, \ldots, \phi_\mathsf{K}$ mapping the input files to the desired computations. We assume that the output function ϕ_k is assigned to node k, for $k \in [\mathsf{K}]$.

A *MapReduce* System decomposes the output functions as:

$$\phi_q(W_1, \ldots, W_\mathsf{N}) = v_q(a_{q,1}, \ldots, a_{q,\mathsf{N}}), \qquad q \in [\mathsf{K}], \tag{1}$$

where v_q is an appropriate *reduce function* and $a_{q,p}$ is an *intermediate value (IVA)* calculated from input file W_p through an appropriate *map function*:

$$a_{q,p} = u_{q,p}(W_p), \qquad p \in [\mathsf{N}]. \tag{2}$$

For simplicity, all IVAs are assumed independent and consisting of A i.i.d. bits.

The MapReduce framework has 3 phases:

Map phase: A subset of all input files $\mathcal{M}_k \subseteq [\mathsf{N}]$ is assigned to each node $k \in [\mathsf{K}]$. Node k computes all IVAs $\{a_{q,p} \colon p \in \mathcal{M}_k,\ q \in [\mathsf{K}]\}$ associated with these input files. Notice that the set $\{\mathcal{M}_k\}_{k \in [\mathsf{K}]}$ is a design factor.

Shuffle phase: Computation of the k-th output function is assigned to the k-th node.

The K nodes in the system communicate over T uses of a wireless network in a full-duplex mode, where T is a design parameter. During this communication, nodes communicate IVAs that they calculated in the Map phase to nodes that are missing these IVAs for the computations of their assigned output functions. So, node $k \in [\mathsf{K}]$ produces complex channel inputs of the form

$$\mathbf{X}_k \triangleq (X_k(1), \ldots, X_k(\mathsf{T}))^T = f_k^{(\mathsf{T})}\left(\{a_{1,p}, \ldots, a_{\mathsf{K},p}\}_{p \in \mathcal{M}_k}\right), \tag{3}$$

by means of an encoding function $f_k^{(\mathsf{T})}$ on appropriate domains and so that the inputs satisfy the block-power constraint

$$\frac{1}{\mathsf{T}} \sum_{t=1}^{\mathsf{T}} \mathbb{E}\left[|X_k(t)|^2\right] \leq \mathsf{P}, \qquad k \in [\mathsf{K}]. \tag{4}$$

Given the full-duplex nature of the network, Node k also observes the complex channel outputs

$$Y_k(t) = \sum_{k' \in [\mathsf{K}] \backslash \{k\}} H_{k,k'}(t) X_{k'}(t) + Z_k(t), \quad t \in [\mathsf{T}], \tag{5}$$

where the sequences of complex-valued channel coefficients $\{H_{k,k'}(t)\}$ and standard circularly symmetric Gaussian noises $\{Z_k(t)\}$ are both i.i.d. and independent of each other and of all other channel coefficients and noises.

Based on its outputs $\mathbf{Y}_k \triangleq (Y_k(1), \ldots, Y_k(\mathsf{T}))^T$ and the IVAs $\{a_{q,p} \colon p \in \mathcal{M}_k,\ q \in [\mathsf{K}]\}$ it computed during the Map phase, Node k decodes the missing IVAs $\{a_{k,p} \colon p \notin \mathcal{M}_k\}$ required to compute its assigned output functions ϕ_k as:

$$\hat{a}_{k,p} = g_{k,p}^{(\mathsf{T})}\left(\{a_{1,i}, \ldots, a_{\mathsf{K},i}\}_{i \in \mathcal{M}_k}, \mathbf{Y}_k\right), \quad p \notin \mathcal{M}_k. \tag{6}$$

Reduce phase: Each node applies the reduce functions to the appropriate IVAs calculated during the Map phase or decoded in the Shuffle phase.

The performance of the distributed computing system is measured in terms of its *computation load*

$$\mathsf{r} \triangleq \sum_{k \in [\mathsf{K}]} \frac{|\mathcal{M}_k|}{\mathsf{N}}, \tag{7}$$

and the *normalized delivery time (NDT)*

$$\Delta \triangleq \varlimsup_{\mathsf{P} \to \infty} \varlimsup_{\mathsf{A} \to \infty} \frac{\mathsf{T}}{\mathsf{A} \cdot \mathsf{K} \cdot \mathsf{N}} \cdot \log \mathsf{P}. \tag{8}$$

We focus on the *fundamental NDT-computation tradeoff* $\Delta^*(\mathsf{r})$, which is defined as the infimum over all values of Δ satisfying (8) for some choice of file assignments $\{\mathcal{M}_k\}$ and sequence (in T) of encoding and decoding functions $\{f_k^{(\mathsf{T})}\}$ and $\{g_{k,p}^{(\mathsf{T})}\}$ in (3) and (6), all depending on A so that the probability of IVA decoding error

$$\Pr\left[\bigcup_{k \in [\mathsf{K}]} \bigcup_{p \notin \mathcal{M}_k} \hat{a}_{k,p} \neq a_{k,p}\right] \to 0 \quad \text{as} \quad \mathsf{A} \to \infty. \tag{9}$$

We terminate this section with some remarks. Notice first the trivial extreme point $\Delta^*(\mathsf{K}) = 0$, because for $\mathsf{r} = \mathsf{K}$ all files can be stored at all nodes and the nodes can thus calculate their desired output functions locally without communicating with the other nodes.

A. Sufficiency of Symmetric File Assignments

Our model exhibits a perfect symmetry between the various nodes in the network in the sense that the channels from any Tx-node to any Rx-node has same statistical behaviour and the various channels are independent of each other. The optimal NDT-computation tradeoff is therefore achieved by a symmetric file assignment where any subset of nodes $\mathcal{T} \subseteq [\mathsf{K}]$ of size i is assigned the same number of files to be stored at all nodes in $\mathcal{T}$. In fact, any non-symmetric file assignment can be symmetrized without decreasing the NDT-computation tradeoff. It suffices to time-share $\mathsf{K}!$ instances of the original scheme for a IVA size that is also multiplied by $\mathsf{K}!$, where in each instance the K nodes are relabeled according to a different permutation. The resulting scheme has a symmetric file assignment and achieves the same NDT-computation tradeoff as the original scheme because both T and A are multiplied by $\mathsf{K}!$ while the other parameters remain unchanged and because the new scheme still satisfies (9).

By the optimality of symmetric file assignments, the optimization problem over the optimal file assignment reduces to finding the optimal fraction of files that should be assigned to exactly i nodes, for any $i \in [\mathsf{K}]$. It is well-known that when communication is over noiseless broadcast links, then it suffices to assign some of the files to $\lfloor \mathsf{r} \rfloor$ nodes and the remaining files to $\lceil \mathsf{r} \rceil$ nodes. We apply the same strategy in this paper. For $\mathsf{r} < \lceil \frac{\mathsf{K}}{2} \rceil$ however we cannot prove optimality of these file assignments.

B. Relation to the Network's Sum-DoF with r-fold Cooperation

A well-studied property of wireless networks is the *Sum Degrees of Freedom (sum-DoF)*, which characterizes the maximum throughput of a network. In this work we are specifically interested in the sum-DoF that one can achieve over the wireless network described by (5), when the inputs are subject to the average power constraints (4) and any set of r nodes $\mathcal{T} \in [\mathsf{K}]^{\mathsf{r}}$ has a message $M_{\mathcal{T}}^{j}$ that it wishes to convey to Node j, for any $j \in [\mathsf{K}] \backslash \mathcal{T}$. Each message $M_{\mathcal{T}}^{j}$ is uniformly distributed over a set $\{1, \ldots, 2^{nR_{\mathcal{T}}^{j}}\}$ and a rate-tuple $(R_{\mathcal{T}}^{j} \colon \mathcal{T} \in [\mathsf{K}]^{\mathsf{r}},\ j \in [\mathsf{K}] \backslash \mathcal{T})$ is called achievable if there exists a sequence of encoding and decoding functions such that the probabilities of error tend to 0 in the asymptotic regime of infinite blocklengths. The sum-DoF is then defined as

$$\text{Sum-DoF}(\mathsf{r}) \triangleq \sup \varlimsup_{P \to \infty} \frac{\sum_{\mathcal{T} \in [\mathsf{K}]^{\mathsf{r}},\, j \in [\mathsf{K}] \backslash \mathcal{T}} R_{\mathcal{T}}^{j}(\mathsf{P})}{\frac{1}{2} \log \mathsf{P}}, \tag{10}$$

where the supremum is over all sequences of rate tuples $\{(R_{\mathcal{T}}^{j}(\mathsf{P}) \colon \mathcal{T} \in [\mathsf{K}]^{\mathsf{r}},\ j \in [\mathsf{K}] \backslash \mathcal{T})\}_{\mathsf{P}>0}$ so that for each $\mathsf{P} > 0$ each tuple $(R_{\mathcal{T}}^{j}(\mathsf{P}) \colon \mathcal{T} \in [\mathsf{K}]^{\mathsf{r}},\ j \in [\mathsf{K}] \backslash \mathcal{T})$ is achievable under power P.

We have the following lemma, which we use in this paper:

Lemma 1. *For any* $\mathsf{r} \in [\mathsf{K}]$*:*

$$\Delta(\mathsf{r}) \geq \left(1 - \frac{\mathsf{r}}{\mathsf{K}}\right) \frac{1}{\text{Sum-DoF}(\mathsf{r})}. \tag{11}$$

Proof: We show how to construct a distributed computing scheme achieving the NDT upper bound in (11). We shall assume a sequence (in $\mathsf{P} > 0$) of rates $(R_{\mathcal{T}}^{j} \colon \mathcal{T} \in [\mathsf{K}]^{\mathsf{r}},\ j \in [\mathsf{K}] \backslash \mathcal{T})$ that achieves the sum-DoF Sum-DoF(r) and is completely symmetric with respect to indices j and sets $\mathcal{T}$. By the same time-sharing and relabeling arguments as described in Subsection II such a sequence must exist.

In the Map Phase we choose a regular file assignment. Partition the input files $\{W_1, \ldots, W_{\mathsf{N}}\}$ into $\binom{\mathsf{K}}{\mathsf{r}}$ disjoint bundles and assign each bundle to a size-r subset $\mathcal{T} \in [[\mathsf{K}]]^{\mathsf{r}}$. The bundle associated to subset $\mathcal{T}$ is denoted $\mathcal{W}_{\mathcal{T}} \subseteq \{W_1, \ldots, W_{\mathsf{N}}\}$. Notice that the proposed assignment satisfies the constraint on the computation load, because the number of files stored at Node k is:

$$\binom{\mathsf{K}-1}{\mathsf{r}-1} \frac{\mathsf{N}}{\binom{\mathsf{K}}{\mathsf{r}}} = \frac{\mathsf{r}}{\mathsf{K}} \mathsf{N}. \tag{12}$$

Each node computes all IVAs associated to its stored files.

During the Wireless Shuffle Phase, each transmit set $\mathcal{T}$ communicates to any receive node $j \notin \mathcal{T}$ all IVAs that can be calculated from bundle $\mathcal{W}_{\mathcal{T}}$. To this end, all nodes use the encoding and decoding functions achieving Sum-DoF(r).

Each transmit group $\mathcal{T}$ has to send $\frac{\mathsf{N}}{\binom{\mathsf{K}}{\mathsf{r}}}$ IVAs to each Node $j \notin \mathcal{T}$ and in total there are $\binom{\mathsf{K}}{\mathsf{r}}(\mathsf{K}-\mathsf{r})$ rates in the symmetric rate vector $(R_{\mathcal{T}}^{j} \colon \mathcal{T} \in [\mathsf{K}]^{\mathsf{r}},\ j \in [\mathsf{K}] \backslash \mathcal{T})$. By definition, the probability of error in reconstructing all missing IVAs tends to 0 as $\mathsf{T} \to \infty$ if

$$\varlimsup_{P \to \infty} \varlimsup_{A \to \infty} \frac{\mathsf{A} \cdot \mathsf{N}}{\mathsf{T} \cdot \binom{\mathsf{K}}{\mathsf{r}} \cdot \log \mathsf{P}} < \frac{\text{Sum-DoF}(\mathsf{r})}{\binom{\mathsf{K}}{\mathsf{r}}(\mathsf{K}-\mathsf{r})}. \tag{13}$$

We thus conclude that

$$\Delta^{\star}(\mathsf{r}) \leq \varliminf_{P \to \infty} \varliminf_{A \to \infty} \frac{\mathsf{T}}{\mathsf{A} \cdot \mathsf{K} \cdot \mathsf{N}} \cdot \log \mathsf{P} = \frac{\mathsf{K}-\mathsf{r}}{\mathsf{K}} \frac{1}{\text{Sum-DoF}(\mathsf{r})}, \tag{14}$$

which proves the desired achievability result. ∎

III. Main Results

The main results of this paper are new upper and lower bounds on the NDT-computation tradeoff of the wireless DC system described in Section II.

For fixed K, define for each $\mathsf{r} \in \{1, \lceil \mathsf{K}/2 \rceil - 1\}$:

$$\Delta_{\mathrm{Ub},1}(\mathsf{r}) \triangleq \left(1 - \frac{\mathsf{r}}{\mathsf{K}}\right) \cdot \frac{\mathsf{r}(\mathsf{K}-1) + \mathsf{K} - \mathsf{r} - 1}{\mathsf{r}(\mathsf{K}-1)^2 + \mathsf{r}(\mathsf{K}-2)}. \tag{15}$$

Further, for $\mathsf{K} = 5$ and $\mathsf{r} = 2$, define

$$\Delta_{\mathrm{Ub},2}(\mathsf{r}) \triangleq \left(1 - \frac{\mathsf{r}}{\mathsf{K}}\right)\left(1 + \frac{1}{6}\right) \tag{16}$$

and for all odd values $\mathsf{K} \geq 7$ and $\mathsf{r} = (\mathsf{K}-1)/2$ set:

$$\Delta_{\mathrm{Ub},2}(\mathsf{r}) \triangleq \frac{1}{\mathsf{K}}\left(1 - \frac{\mathsf{r}}{\mathsf{K}}\right)\left(1 + \frac{1}{(\mathsf{K}-\mathsf{r}-1)(\mathsf{K}-1)}\right). \tag{17}$$

For all other values of r and K set $\Delta_{\mathrm{Ub},2}(\mathsf{r}) = \infty$.[1]

Define for any integer value $\mathsf{r} \in [\mathsf{K}]$:

$$\Delta_{\mathrm{Ub}}(\mathsf{r}) \triangleq \begin{cases} \min_{i \in \{1,2\}} \Delta_{\mathrm{Ub},i}(\mathsf{r}) & \text{if } \mathsf{r} < \mathsf{K}/2 \\ \frac{1}{\mathsf{K}}\left(1 - \frac{\mathsf{r}}{\mathsf{K}}\right) & \text{if } \mathsf{r} \geq \mathsf{K}/2 \end{cases}. \tag{18}$$

Also, let

$$\Delta_{\mathrm{Lb}}(\mathsf{r}) \triangleq \begin{cases} \frac{1}{\mathsf{K}}\left(2 - \frac{3}{\mathsf{K}}\right) & \text{if } \mathsf{r} = 1, \\ \frac{1}{\mathsf{K}}\left(1 - \frac{\mathsf{r}}{\mathsf{K}} + \max\limits_{t \in [\lfloor \mathsf{K}/2 \rfloor]} \operatorname{lowc}\left(C_t(\mathsf{r})\right)\right) & \text{if } \mathsf{r} \in (1,2), \\ \frac{1}{\mathsf{K}}\left(1 - \frac{\mathsf{r}}{\mathsf{K}} + \operatorname{lowc}\left(C_{\lfloor \mathsf{K}/2 \rfloor}(\mathsf{r})\right)\right) & \text{if } \mathsf{r} \in [2, \mathsf{K}], \end{cases} \tag{19}$$

where for any $t \in [\lfloor \mathsf{K}/2 \rfloor]$:

$$C_t(i) = \begin{cases} \frac{\binom{\mathsf{K}-i}{t-i}}{\binom{\mathsf{K}}{t} \cdot t} \cdot (\mathsf{K} - t - i), & \text{if } i \in [t], \\ 0, & \text{if } i \in [\mathsf{K}] \backslash [t] \end{cases} \tag{20}$$

and recall that for any function f we denote by $\operatorname{lowc}(f(\ell))$ the lower convex enveloppe of the curve $\{(\ell, f(\ell))\}$.

Theorem 1. *The NDT-computation tradeoff $\Delta^*(\mathsf{r})$ is upper- and lower-bounded as:*

$$\Delta_{\mathrm{Lb}}(\mathsf{r}) \leq \Delta^*(\mathsf{r}) \leq \operatorname{lowc}\left(\Delta_{\mathrm{Ub}}(\mathsf{r})\right). \tag{21}$$

Proof: For integers $\mathsf{r} \geq \mathsf{K}/2$ achievability of the upper bound $\Delta_{\mathrm{Ub}}(\mathsf{r})$ is proved in [15]. For integers $\mathsf{r} < \mathsf{K}/2$ achievability of the two upper bounds $\Delta_{\mathrm{Ub},1}(\mathsf{r})$ and $\Delta_{\mathrm{Ub},2}(\mathsf{r})$ follows by Lemma 1 and the coding schemes described in Sections V and VI. (Section IV illustrates simple special cases of the scheme in Section V.) Achievability of the lower convex enveloppe follows by selecting the best strategy for each value of r and by simple time- and memory-sharing strategies. The lower bound is proved in Section VII. ∎

Remark 1. *The upper bound is convex and piece-wise constant. The lower bound is piecewise constant with segments spanning the intervals $[i, i+1]$, for $i = 2, \ldots, \mathsf{K}-1$. On the interval $[1, 2)$, the lower bound is constant over smaller sub-intervals only but not over the entire segment.*

For all $\mathsf{r} \geq \lceil \mathsf{K}/2 \rceil$ the lower bound (18) and the upper bound (19) match.

Corollary 1. *For all $\mathsf{r} \geq \lceil \mathsf{K}/2 \rceil$:*

$$\Delta^*(\mathsf{r}) = \left(1 - \frac{\mathsf{r}}{\mathsf{K}}\right) \cdot \frac{1}{\mathsf{K}}. \tag{22}$$

Proof: For $\mathsf{r} \geq \lceil \mathsf{K}/2 \rceil$ the upper bound $\operatorname{lowc}(\Delta_{\mathrm{Ub}}(\mathsf{r}))$ is equal to the lower bound $\Delta_{\mathrm{Lb}}(\mathsf{r})$ because $C_{\lfloor \mathsf{K}/2 \rfloor}(i) = 0$ for all $i \geq \lceil \mathsf{K}/2 \rceil$. ∎

[1]This second upper bound is interesting and nontrivial only when $\mathsf{K} = 2\mathsf{r} + 1$ and $\mathsf{r} = 2, 3, \ldots$.

Remark 2. *The NDT-computation tradeoff in* (22)*, is achieved with linear zero-forcing and side-information cancellation, see [15]. These simple strategies are thus sufficient to achieve the optimal NDT-computation tradeoff in the regime* $\mathsf{r} \geq \lceil \mathsf{K}/2 \rceil$*. This statement however does not apply for smaller values of* r *where more sophisticated strategies such as interference alignment (IA) strategies are necessary to achieve the optimal NDT-computation tradeoff. This follows from the converse result in [15] and our achievability part in Theorem 1, see Corollary 2 ahead.*

A. Comparison to Previous Upper Bounds

We compare the bounds in Theorem 1 to the upper bound in [15] and [16]. The upper bound in [15] is given as follows:

$$\Delta^*(\mathsf{r}) \leq \Delta_{\text{UB-BF}}(\mathsf{r}) \triangleq \text{lowc}\left\{ \left(\mathsf{r}, \frac{1-\mathsf{r}/\mathsf{K}}{\min(\mathsf{K}, 2\mathsf{r})}\right) : \mathsf{r} \in [\mathsf{K}] \right\}, \tag{23}$$

and is tight when restricting to zero-forcing, one-shot beamforming, and side-information cancellation. The upper bound in [16] has the form:

$$\Delta^*(\mathsf{r}) \leq \Delta_{\text{Ub-Groups}}(\mathsf{r}) \triangleq \text{lowc}\left((\mathsf{K}, 0) \cup \left\{ \left(\mathsf{r}, \frac{1-\mathsf{r}/\mathsf{K}}{\text{Sum-DoF}_{\text{Lb}}(\mathsf{r})}\right) : 1 \leq \mathsf{r} < \mathsf{K}, \mathsf{r}|\mathsf{K} \right\} \right), \tag{24}$$

where

$$\text{Sum-DoF}_{\text{Lb}}(\mathsf{r}) \triangleq \begin{cases} 2\mathsf{r} & \text{if } \mathsf{K}/\mathsf{r} \in \{2, 3\}, \\ \frac{\mathsf{K}(\mathsf{K}-\mathsf{r})-\mathsf{r}^2}{2\mathsf{K}-3\mathsf{r}} & \text{if } \mathsf{K}/\mathsf{r} \geq 4, \end{cases} \tag{25}$$

and coincides with the upper bound in Theorem 1 for $\mathsf{r} = 1$, i.e., $\Delta_{\text{Ub}}(1) = \Delta_{\text{Ub-Groups}}(1)$.

Notice that the sequences $\{\Delta_{\text{UB-BF}}(\mathsf{r})\colon \mathsf{r} = 1, \ldots, \lceil \frac{\mathsf{K}}{2} \rceil\}$ and $\{\Delta_{\text{UB-Groups}}(\mathsf{r})\colon \mathsf{r} = 1, \ldots, \lceil \frac{\mathsf{K}}{2} \rceil\}$ are strictly convex and the lower convex-envelope of these points, i.e., $\Delta_{\text{UB-BF}}(\mathsf{r})$ and $\Delta_{\text{UB-Groups}}(\mathsf{r})$ are thus piece-wise linear.

Corollary 2. *For all* $1 < \mathsf{r} < \lceil \frac{\mathsf{K}}{2} \rceil$*:*

$$\Delta^*(\mathsf{r}) \leq \text{lowc}\left(\Delta_{\text{Ub}}(\mathsf{r})\right) < \Delta_{\text{Ub-Groups}}(\mathsf{r}), \tag{26}$$

and for all $1 \leq \mathsf{r} < \lceil \frac{\mathsf{K}}{2} \rceil$*:*

$$\Delta^*(\mathsf{r}) \leq \text{lowc}\left(\Delta_{\text{Ub}}(\mathsf{r})\right) < \Delta_{\text{Ub-ZF}}(\mathsf{r}). \tag{27}$$

Figures 1 and 2 show the bounds in Theorem 1 and compare them to the previous upper bounds $\Delta_{\text{Ub-Groups}}(\mathsf{r})$ and $\Delta_{\text{Ub-ZF}}(\mathsf{r})$.

Proof of the Corollary: For $\mathsf{r} < \lceil \frac{\mathsf{K}}{2} \rceil$, our upper bound in Theorem 1 is strictly better than the bounds in [15] and [16], as we argue in the following. We start by noticing that for $\mathsf{r} = \lceil \frac{\mathsf{K}}{2} \rceil$ all three upper bounds coincide:

$$\Delta_{\text{Ub}}\left(\left\lceil \frac{\mathsf{K}}{2} \right\rceil\right) = \Delta_{\text{Ub-Groups}}\left(\left\lceil \frac{\mathsf{K}}{2} \right\rceil\right) = \Delta_{\text{Ub-ZF}}\left(\left\lceil \frac{\mathsf{K}}{2} \right\rceil\right). \tag{28}$$

Consider now $\mathsf{r} < \lceil \frac{\mathsf{K}}{2} \rceil$. For any integer computation load

$$\mathsf{r} \leq (\mathsf{K}-2)/2 \tag{29}$$

the upper bound in (15) is lower than the upper bound in (23) because under (29)

$$(\mathsf{K}-1)(\mathsf{K}-2\mathsf{r}-2) > -1 \tag{30}$$

which is equivalent to

$$\frac{\mathsf{r}(\mathsf{K}-1) + \mathsf{K} - \mathsf{r} - 1}{\mathsf{r}(\mathsf{K}-1)^2 + \mathsf{r}(\mathsf{K}-2)} < \frac{1}{2\mathsf{r}}. \tag{31}$$

We thus conclude that for K even and integer-valued $\mathsf{r} \geq \frac{\mathsf{K}}{2} - 1$ as well as for K odd and integer-valued $\mathsf{r} \geq \frac{\mathsf{K}-3}{2}$ the upper bound in Theorem 1 is strictly lower than the previous upper bound $\Delta_{\text{Ub-ZF}}(\mathsf{r})$. We continue to prove that for K odd and $\mathsf{r} = \frac{\mathsf{K}-1}{2}$ the upper bound in (16) (for $\mathsf{K} = 5$) or in (17) (for $\mathsf{K} \geq 7$) is strictly lower than the upper bound in (23). Combined with (28) and the piece-wise linearity of the bounds, this proves (26).

Notice that for $\mathsf{K} \geq 5$:

$$\left(\frac{(\mathsf{K}-1)^2 + 2}{(\mathsf{K}-1)\mathsf{K}}\right) = \frac{\mathsf{K}^2 - 2\mathsf{K} + 3}{\mathsf{K}^2 - \mathsf{K}} < 1, \tag{32}$$

and thus for $\mathsf{r} = \frac{\mathsf{K}-1}{2}$:

$$\frac{1}{\mathsf{K}}\left(1 + \frac{1}{(\mathsf{K}-\mathsf{r}-1)(\mathsf{K}-1)}\right) < \frac{1}{2\mathsf{r}} = \frac{1}{\mathsf{K}-1}, \tag{33}$$

establishing that the bound in (17) is lower than (23). For $\mathsf{K}=5$ and $\mathsf{r}=2$ it is easily verified that bound (23) evaluates to $\left(1-\frac{\mathsf{r}}{\mathsf{K}}\right)\cdot\frac{1}{4}$ and is thus strictly higher than bound (16).

We continue to prove (26). For integers r satisfying $1<\mathsf{r}<\left\lceil\frac{\mathsf{K}}{2}\right\rceil$ we have

$$(\mathsf{r}-1)(\mathsf{K}-2\mathsf{r}-1)\geq 0, \tag{34}$$

which is equivalent to

$$\mathsf{r}(2\mathsf{K}-3\mathsf{r})\leq \mathsf{r}(\mathsf{K}-1)+\mathsf{K}-\mathsf{r}-1. \tag{35}$$

Moreover, for all integers $\mathsf{r}>1$:

$$(\mathsf{K}-1)^2+(\mathsf{K}-2)>\mathsf{K}(\mathsf{K}-\mathsf{r})-\mathsf{r}^2, \tag{36}$$

which combined with (35) establishes that

$$\frac{\mathsf{r}(\mathsf{K}-1)+\mathsf{K}-\mathsf{r}-1}{\mathsf{r}(\mathsf{K}-1)^2+\mathsf{r}(\mathsf{K}-2)}<\frac{2\mathsf{K}-3\mathsf{r}}{\mathsf{K}(\mathsf{K}-\mathsf{r})-\mathsf{r}^2}. \tag{37}$$

This implies (27) except for values of K where $\mathrm{lowc}\left(\Delta_{\text{Ub}}(\mathsf{r})\right)$ and $\Delta_{\text{Ub-Groups}}(\mathsf{r})$ are both given by the straight line between $\Delta_{\text{Ub-Groups}}(1)$ and $\Delta_{\text{Ub-Groups}}\left(\left\lceil\frac{\mathsf{K}}{2}\right\rceil\right)$. While this is the case for $\Delta_{\text{Ub-Groups}}(\mathsf{r})$ when $\mathsf{K}=2\mathsf{K}'$ and K' is a prime number, it is never true for $\Delta_{\text{Ub}}(\mathsf{r})$ because the sequence $\{\Delta_{\text{Ub}}(\mathsf{r})\colon \mathsf{r}=1,\ldots,\left\lceil\frac{\mathsf{K}}{2}\right\rceil\}$ is strictly convex. ■

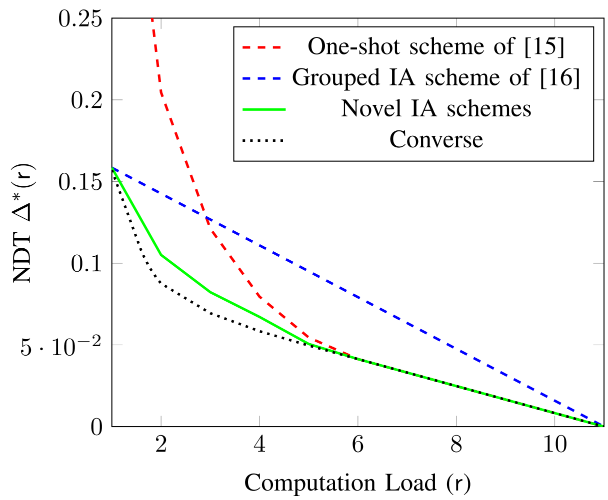

Fig. 1. Bounds on $\Delta^*(\mathsf{r})$ from Theorem 1 compared to the optimal zero-forcing and interference cancelation scheme in [15] and to the upper bound obtained by the grouped IA scheme in [16] for $\mathsf{K}=11$.

IV. Examples of our IA Scheme without zero-forcing

In the next-following two sections we describe our first coding scheme based on IA but without zero-forcing, and we evaluate the lower bound on Sum-DoF(r) that it achieves. In this section we only present some simple examples and attempt to build up intuition. The scheme and its corresponding upper bound on the sum-DoF are described and analyzed in detail in the next-following Section V.

A. *Example 1:* $\mathsf{K}\geq 3, \mathsf{r}=1$

Consider first the simple case with computation load $\mathsf{r}=1$, i.e., when each IVA can be stored only at a single node.

In our scheme we transmit the $\mathsf{K}(\mathsf{K}-1)-1$ Messages (or IVAs)

$$\{M_k^j\colon j,k\in[\mathsf{K}],\ j\neq k,\ (j,k)\neq(1,\mathsf{K})\}. \tag{38}$$

That means, each Node k transmits a message M_k^j to each other node $j\neq k$, except for node K that only transmits messages to nodes $2,\ldots,\mathsf{K}-1$ but not to Node 1.

All messages $\{M_k^j\}_k$ intended for Node j, for $j=2,\ldots,\mathsf{K}$, are precoded by the precoding matrix $\mathbf{U}_j$, whose construction we shall present shortly. In contrast to the standard IA-scheme in [19], here Node 1 does not have a dedicated IA precoding matrix. Instead, each Node k precodes the message b_k^1 that it sends to Node 1 with its own dedicated precoding matrix $\mathbf{U}_k$.

Table I depicts the precoding matrices used to transmit information from a given Node k to another Node j. The entry "x" indicates that nodes do not transmit messages to themselves. The entry "o" indicates that a given Node k chooses not to send

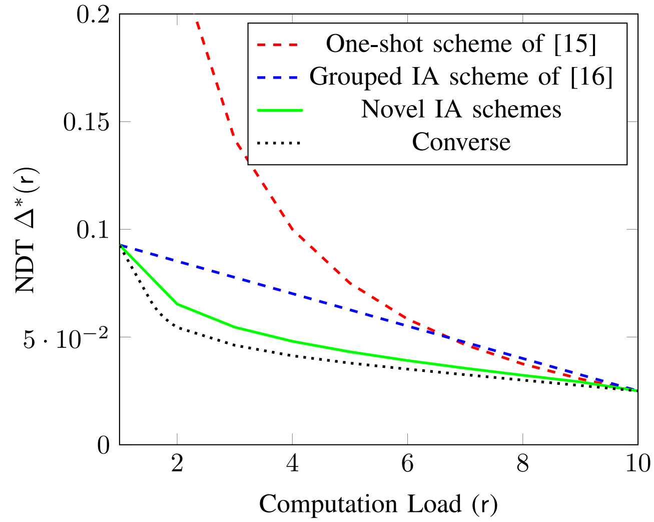

Fig. 2. Bounds on $\Delta^*(\mathsf{r})$ from Theorem 1 compared to the optimal NDT achieved by the optimal zero-forcing and interference cancelation scheme in [15] and to the upper bound obtained by the grouped IA scheme in [16] for $\mathsf{K} = 20$.

TABLE I
TABLE SHOWING THE PRECODING MATRIX USED TO SEND EACH MESSAGE M_k^j.

$k\backslash j$	1	2	3	4	5	6	7
1	x	$\mathbf{U}_2$	$\mathbf{U}_3$	$\mathbf{U}_4$	$\mathbf{U}_5$	$\mathbf{U}_6$	$\mathbf{U}_7$
2	$\mathbf{U}_2$	x	$\mathbf{U}_3$	$\mathbf{U}_4$	$\mathbf{U}_5$	$\mathbf{U}_6$	$\mathbf{U}_7$
3	$\mathbf{U}_3$	$\mathbf{U}_2$	x	$\mathbf{U}_4$	$\mathbf{U}_5$	$\mathbf{U}_6$	$\mathbf{U}_7$
4	$\mathbf{U}_4$	$\mathbf{U}_2$	$\mathbf{U}_3$	x	$\mathbf{U}_5$	$\mathbf{U}_6$	$\mathbf{U}_7$
5	$\mathbf{U}_5$	$\mathbf{U}_2$	$\mathbf{U}_3$	$\mathbf{U}_4$	x	$\mathbf{U}_6$	$\mathbf{U}_7$
6	$\mathbf{U}_6$	$\mathbf{U}_2$	$\mathbf{U}_3$	$\mathbf{U}_4$	$\mathbf{U}_5$	x	$\mathbf{U}_7$
7	o	$\mathbf{U}_2$	$\mathbf{U}_3$	$\mathbf{U}_4$	x	$\mathbf{U}_6$	x

a message to a given Node j. The motivation for our precoding assignment is to have no duplications in a given row (because otherwise the corresponding receive nodes won't be able to distinguish their intended signals from interference) and to use as few precoding matrices as possible so as to keep the nodes' interference spaces small (this will become more clear shortly).

Node 1 thus transmits the signal

$$\mathbf{X}_1 = \sum_{j \in [\mathsf{K}] \backslash \{1\}} \mathbf{U}_j \mathbf{b}_1^j, \tag{39}$$

Nodes $2, \ldots, \mathsf{K}-1$ transmit the signals

$$\mathbf{X}_k = \mathbf{U}_k \mathbf{b}_k^1 + \sum_{j \in [\mathsf{K}] \backslash \{1,k\}} \mathbf{U}_j \mathbf{b}_k^j, \quad k \in [\mathsf{K}-1] \backslash \{1\}, \tag{40}$$

and Node K transmits the signal

$$\mathbf{X}_\mathsf{K} = \sum_{j \in [\mathsf{K}-1] \backslash \{1\}} \mathbf{U}_j \mathbf{b}_k^j, \tag{41}$$

where $\mathbf{b}_k^j$ is a Gaussian codeword encoding Message M_k^j.

Node 1 observes the receive signal

$$\mathbf{Y}_1 = \underbrace{\sum_{k \in [\mathsf{K}-1] \backslash \{1\}} \mathbf{H}_{1,k} \mathbf{U}_k \mathbf{b}_k^1}_{\text{desired signal}} + \underbrace{\sum_{\substack{k,\ell \in [\mathsf{K}] \backslash \{1\} \\ k \neq \ell}} \mathbf{H}_{1,k} \mathbf{U}_\ell \mathbf{b}_k^\ell}_{\text{interference}} + \mathbf{Z}_1, \tag{42}$$

and Nodes $2, \ldots, \mathsf{K}$ the receive signals

$$\mathbf{Y}_j = \underbrace{\sum_{k \in [\mathsf{K}] \backslash \{j\}} \mathbf{H}_{j,k} \mathbf{U}_j \mathbf{b}_k^j}_{\text{desired signal}} + \underbrace{\sum_{\substack{\ell,k \in [\mathsf{K}] \backslash \{j\} \\ \ell \neq k, \ell \neq 1}} \mathbf{H}_{j,k} \mathbf{U}_\ell \mathbf{b}_k^\ell + \sum_{k \in [\mathsf{K}-1] \backslash \{1,j\}} \mathbf{H}_{j,k} \mathbf{U}_k \mathbf{b}_k^1}_{\text{interference}} + \mathbf{Z}_j, \tag{43}$$

where $\mathbf{H}_{j,k}$ denotes the diagonal T-by-T channel matrix consisting of the entries $\{H_{j,k}(t)\}_{t=1}^{\mathsf{T}}$.

Remark 3. *The $\mathsf{K}-1$ desired signals at any Node $j \in [\mathsf{K}]\backslash\{1\}$ are precoded by the same precoding matrix $\mathbf{U}_j$, while its interference signals are precoded by the remaining $\mathsf{K}-2$ precoding matrices $\mathbf{U}_2, \ldots, \mathbf{U}_{j-1}, \mathbf{U}_{j+1}, \ldots, \mathbf{U}_{\mathsf{K}}$. In contrast, for Node 1, matrices $\mathbf{U}_2, \ldots, \mathbf{U}_{\mathsf{K}-1}$ precode both desired and interference signals while matrix $\mathbf{U}_{\mathsf{K}}$ only precodes interference.*

We choose the precoding matrices $\mathbf{U}_1, \ldots, \mathbf{U}_{\mathsf{K}}$ according to the IA principle [19]. That means, we construct each column of matrix $\mathbf{U}_j$ using all channel matrices that pre-multiply $\mathbf{U}_j$ in the interference terms of the receive signals $\mathbf{Y}_1, \ldots, \mathbf{Y}_{\mathsf{K}}$ and exponentiate these the channel matrices with a different set of exponents for each column. More formally, we choose[2]

$$\mathbf{U}_\ell \triangleq \left[\prod_{\mathbf{H} \in \mathcal{H}_\ell} \mathbf{H}^{\alpha_{\ell,\mathbf{H}}} \cdot \mathbf{\Xi}_\ell \colon \;\; \forall \boldsymbol{\alpha}_\ell \in [\eta]^\Gamma \right], \tag{44}$$

where each column of the matrix is constructed using a different exponent-vector $\boldsymbol{\alpha}_\ell = (\alpha_{\ell,\mathbf{H}} \colon \mathbf{H} \in \mathcal{H}_\ell) \in [\eta]^\Gamma$; η is a large number depending on the blocklength T that tends to ∞ with T; $\{\mathbf{\Xi}_\ell\}_{\ell \in [\mathsf{K}]\backslash\{1\}}$ are i.i.d. random vectors independent of all channel matrices, noises, and messages; and

$$\begin{aligned} \mathcal{H}_\ell = \{&\mathbf{H}_{j,k} \colon j \in [\mathsf{K}]\backslash\{1,\ell\}, k \in [\mathsf{K}]\backslash\{\ell\}, \; j \neq k\} \\ &\cup \{\mathbf{H}_{1,k} \colon k \in [\mathsf{K}]\backslash\{1,\ell\}\}, \end{aligned} \tag{45}$$

and $\Gamma \triangleq |\mathcal{H}_\ell|$ does not depend on ℓ.

With the proposed construction, for any $j \in [\mathsf{K}]$ and $\ell \in [\mathsf{K}]\backslash\{1,j\}$, the signals that are precoded by matrix $\mathbf{U}_\ell$ and interfere at Node j lie in the column space of the matrix

$$\mathbf{W}_\ell \triangleq \left[\prod_{\mathbf{H} \in \mathcal{H}_\ell} \mathbf{H}^{\alpha_{\ell,\mathbf{H}}} \cdot \mathbf{\Xi}_\ell \colon \;\; \forall \boldsymbol{\alpha}_\ell \in [\eta+1]^\Gamma \right]. \tag{46}$$

The signals that are desired at Node $j \in \{2, \ldots, \mathsf{K}\}$ correspond to the columns of the matrix

$$\mathbf{D}_j \triangleq \Big[\mathbf{H}_{j,k}\mathbf{U}_j\Big]_{k \in [\mathsf{K}]\backslash\{j\}}, \quad j \in \{2, \ldots, \mathsf{K}\}. \tag{47}$$

The signals desired at Node 1 correspond to the column space of the matrix

$$\mathbf{D}_1 \triangleq \Big[\mathbf{H}_{1,k}\mathbf{U}_k\Big]_{k \in [\mathsf{K}-1]\backslash\{1\}}. \tag{48}$$

As is proved in [16] and follows from our analysis in Section V, with probability 1 the matrices

$$\mathbf{\Lambda}_j = [\mathbf{D}_j \; \mathbf{W}_2 \cdots \mathbf{W}_{j-1} \mathbf{W}_{j+1} \cdots \mathbf{W}_{\mathsf{K}}], \quad j \in [\mathsf{K}]\backslash\{1\}, \tag{49}$$

and

$$\mathbf{\Lambda}_1 = [\mathbf{D}_1 \; \mathbf{W}_2 \cdots \mathbf{W}_{\mathsf{K}}] \tag{50}$$

have full column-rank.

Since the matrices have full column-rank, a simple zero-forcing strategy at the receiving nodes allows to achieve DoF

$$\frac{\#\text{columns}(\mathbf{D}_j)}{\#\text{columns}(\mathbf{\Lambda}_j)} \tag{51}$$

to each Node j. I.e., in the limit as $\eta \to \infty$ (and thus $\frac{\eta}{\eta+1} \to 1$) a DoF $\frac{\mathsf{K}-1}{2\mathsf{K}-3}$ at Nodes $2, \ldots, \mathsf{K}$ and DoF $\frac{\mathsf{K}-2}{2\mathsf{K}-3}$ at Node 1 is achievable, yielding a sum-DoF of

$$\text{Sum-DoF}_{\text{LB}} = \frac{(\mathsf{K}-1)^2 + \mathsf{K} - 2}{2\mathsf{K}-3}. \tag{52}$$

B. Example 2: $\mathsf{K} = 4, \mathsf{r} = 2$

Consider now a computation load of $\mathsf{r} = 2$ and only $\mathsf{K} = 4$ nodes. In this case our scheme transmits 18 different messages depicted in (53). Here, Message $M_{k,\mathcal{T}}^j$ is a message that is known by the set of nodes $\mathcal{T}$ and intended to Node $j \notin \mathcal{T}$. Though known to the entire set $\mathcal{T}$, Message $M_{k,\mathcal{T}}^j$ is only transmitted by a single Node $k \in \mathcal{T}$. The remaining nodes in $\mathcal{T}\backslash\{k\}$ do not participate in the transmission. They however exploit their knowledge of $M_{k,\mathcal{T}}^j$ to remove cancel the corresponding transmission from their receive signal.

Notice that for certain sets $\mathcal{T}$ and receive nodes $j \notin \mathcal{T}$ our scheme sends two messages to the same node j: $M_{k_1,\mathcal{T}}^j$ and $M_{k_2,\mathcal{T}}^j$ for different indices $k_1, k_2 \in \mathcal{T}$. (In (53) the two messages $M_{1,\{1,4\}}^2$ and $M_{4,\{1,4\}}^2$ for example have this form.) These

[2]By the memorylessness of the channel the matrices $\mathbf{H}_{j,k}$ are diagonal and their multiplications and exponentiations are effectively multiplications and exponentiations of the corresponding diagonal elements.

messages $M^j_{k_1,\mathcal{T}}$ and $M^j_{k_2,\mathcal{T}}$ actually represent two independent submessages of Message $M^j_{\mathcal{T}}$ as we defined it in Section II-B. For the sets $\mathcal{T}$ and Nodes $j \notin \mathcal{T}$ for which there exists only a single Message $M^j_{k,\mathcal{T}}$, this message is really the message $M^j_{\mathcal{T}}$ as we defined it in Section II-B. As we will only analyze the sum-DoF of our scheme, this distinction between submessages and messages is not important and we shall simply omit it in the following.

We send the following messages in our scheme. To Node 1, we send messages

$$M^1_{2,\{2,3\}},\ M^1_{3,\{2,3\}},\ M^1_{2,\{2,4\}},\ M^1_{3,\{3,4\}}; \tag{53a}$$

to Node 2 we send messages

$$\begin{aligned}M^2_{1,\{1,3\}},\ M^2_{3,\{1,3\}},\ M^2_{1,\{1,4\}},\ M^2_{4,\{1,4\}}\\ M^2_{3,\{3,4\}}\ M^2_{4,\{3,4\}};\end{aligned} \tag{53b}$$

to Node 3 we send messages

$$\begin{aligned}M^3_{1,\{1,2\}},\ M^3_{2,\{1,2\}},\ M^3_{1,\{1,4\}},\ M^3_{4,\{1,4\}},\\ M^3_{2,\{2,4\}},\ M^3_{4,\{2,4\}};\end{aligned} \tag{53c}$$

and to Node 4 we send messages

$$\begin{aligned}M^4_{1,\{1,2\}},\ M^4_{2,\{1,2\}},\ M^4_{1,\{1,3\}},\ M^4_{3,\{1,3\}},\\ M^4_{2,\{2,3\}},\ M^4_{3,\{2,3\}}.\end{aligned} \tag{53d}$$

We observe that each Node j obtains two messages from each subset of nodes $\mathcal{T}$ of size 2 not containing j, where each node in $\mathcal{T}$ sends one of the two messages. An exception are messages $M^1_{4,\{2,4\}}$ and $M^1_{4,\{3,4\}}$ which are not transmitted because in our scheme the last Node $\mathsf{K} = 4$ does not send any message to the first Node 1. (As we will see, this omission allows to reuse some of the precoding matrices, similarly to the scheme for $\mathsf{r} = 1$, and thus achieve an improved Sum-DoF.) Prior to transmission, each message $M^j_{k,\mathcal{T}}$ is encoded into a Gaussian codeword $\mathbf{b}^j_{k,\mathcal{T}}$. We use the interference-alignment (IA) technique with three precoding matrices $\mathbf{U}_{\{2,3\}}, \mathbf{U}_{\{2,4\}}$, and $\mathbf{U}_{\{3,4\}}$. Precoding matrix $\mathbf{U}_{\{2,3\}}$ is used to send codewords

$$\mathbf{b}^2_{1,\{1,3\}},\ \mathbf{b}^3_{1,\{1,2\}},\ \mathbf{b}^2_{4,\{3,4\}},\ \mathbf{b}^3_{4,\{2,4\}}, \tag{54}$$

$$\mathbf{b}^1_{2,\{2,3\}},\ \mathbf{b}^1_{3,\{2,3\}},\ \mathbf{b}^2_{3,\{1,3\}},\ \mathbf{b}^3_{2,\{1,2\}}, \tag{55}$$

precoding matrix $\mathbf{U}_{\{2,4\}}$ is used to send codewords

$$\mathbf{b}^2_{1,\{1,4\}},\ \mathbf{b}^4_{1,\{1,2\}},\ \mathbf{b}^2_{3,\{3,4\}},\ \mathbf{b}^4_{3,\{2,3\}}, \tag{56}$$

$$\mathbf{b}^1_{2,\{2,4\}},\ \mathbf{b}^4_{2,\{1,2\}},\ \mathbf{b}^2_{4,\{1,4\}}, \tag{57}$$

and precoding matrix $\mathbf{U}_{\{3,4\}}$ is used to send codewords

$$\mathbf{b}^3_{1,\{1,4\}},\ \mathbf{b}^4_{1,\{1,3\}},\ \mathbf{b}^3_{2,\{2,4\}},\ \mathbf{b}^4_{2,\{2,3\}}, \tag{58}$$

$$\mathbf{b}^1_{3,\{3,4\}},\ \mathbf{b}^3_{4,\{1,4\}},\ \mathbf{b}^4_{3,\{1,3\}}. \tag{59}$$

Remark 4. *The choice of precoding matrices is inspired by [18] where Message $M^j_{k,\mathcal{T}}$ is precoded by the matrix $\mathbf{U}_{\mathcal{R}}$ for $\mathcal{R} = \mathcal{T}\backslash\{j\} \cup \{k\}$. The idea behind the choice of precoding matrices in [18] is that any node in $\mathcal{R}$ is either interested in learning Message $M^j_{k,\mathcal{T}}$ or it can compute it itself and remove the interference from its receive signal. A given node j thus only experiences interference from precoding matrices $\mathbf{U}_{\mathcal{R}}$ for which $j \notin \mathcal{R}$.*

In contrast to [18], in our IA scheme we de not use precoding matrices $\mathbf{U}_{\mathcal{R}'}$ for sets $\mathcal{R}'$ containing index 1, but reuse precoding matrices $\mathbf{U}_{\mathcal{R}}$ for sets $\mathcal{R}$ not containing 1. Specifically, we use the precoding matrix $\mathbf{U}_{\mathcal{R}}$ also to send the codewords (if they exist)

$$\mathbf{b}^1_{k,\mathcal{R}}, \qquad \mathbf{b}^j_{k,\mathcal{R}\cup\{1\}\backslash\{j\}}, \qquad \forall j,k \in \mathcal{R},\ j \neq k. \tag{60}$$

One can verify that the codewords in lines (55)*,* (57)*,* (59) *are of the form in* (60)*.*

We illustrate our assignment of the precoding matrices also using the following table. The entries in column 1 or in rows $\{1,2\},\{1,3\},\{1,4\}$ correspond to two submessages $M^j_{k_1,\mathcal{T}}$ and $M^j_{k_2,\mathcal{T}}$, where k_1 and k_2 denote the two entries in $\mathcal{T}$. For all other entries in Table II not equal to "x", we have only one message per precoding matrix, see (54), (56), and (58).

During the shuffling phase, Nodes 1–4 send the following signals. Node 1 sends:

$$\begin{aligned}\mathbf{X}_1 = \mathbf{U}_{\{2,3\}}&\left(\mathbf{b}^2_{1,\{1,3\}} + \mathbf{b}^3_{1,\{1,2\}}\right)\\ +\mathbf{U}_{\{2,4\}}&\left(\mathbf{b}^2_{1,\{1,4\}} + \mathbf{b}^4_{1,\{1,2\}}\right)\\ +\mathbf{U}_{\{3,4\}}&\left(\mathbf{b}^3_{1,\{1,4\}} + \mathbf{b}^4_{1,\{1,3\}}\right).\end{aligned} \tag{61}$$

TABLE II
MESSAGES $M_{k,\mathcal{T}}^{j}$ PRECODED BY THE THREE PRECODING MATRICES $\mathbf{U}_{\{2,3\}}$, $\mathbf{U}_{\{2,4\}}$, AND $\mathbf{U}_{\{3,4\}}$.

$\mathcal{T} \setminus j$	1	2	3	4
$\{1,2\}$	x	x	$\mathbf{U}_{\{2,3\}}$	$\mathbf{U}_{\{2,4\}}$
$\{1,3\}$	x	$\mathbf{U}_{\{2,3\}}$	x	$\mathbf{U}_{\{3,4\}}$
$\{1,4\}$	x	$\mathbf{U}_{\{2,4\}}$	$\mathbf{U}_{\{3,4\}}$	x
$\{2,3\}$	$\mathbf{U}_{\{2,3\}}$	x	x	$\mathbf{U}_{\{2,4\}}$,$\mathbf{U}_{\{3,4\}}$
$\{2,4\}$	$\mathbf{U}_{\{2,4\}}$	x	$\mathbf{U}_{\{2,3\}}$, $\mathbf{U}_{\{3,4\}}$	x
$\{3,4\}$	$\mathbf{U}_{\{3,4\}}$	$\mathbf{U}_{\{2,3\}}$, $\mathbf{U}_{\{2,4\}}$	x	x

Node 2 sends:

$$\begin{aligned}\mathbf{X}_2 = \mathbf{U}_{\{2,3\}} &\left(\mathbf{b}^1_{2,\{2,3\}} + \mathbf{b}^3_{2,\{1,2\}}\right) \\ +\mathbf{U}_{\{2,4\}} &\left(\mathbf{b}^1_{2,\{2,4\}} + \mathbf{b}^4_{2,\{1,2\}}\right) \\ +\mathbf{U}_{\{3,4\}} &\left(\mathbf{b}^3_{2,\{2,4\}} + \mathbf{b}^4_{2,\{2,3\}}\right). \end{aligned} \tag{62}$$

Node 3 sends:

$$\begin{aligned}\mathbf{X}_3 = \mathbf{U}_{\{2,3\}} &\left(\mathbf{b}^1_{3,\{2,3\}} + \mathbf{b}^2_{3,\{1,3\}}\right) \\ +\mathbf{U}_{\{2,4\}} &\left(\mathbf{b}^2_{3,\{3,4\}} + \mathbf{b}^4_{3,\{2,3\}}\right) \\ +\mathbf{U}_{\{3,4\}} &\left(\mathbf{b}^1_{3,\{3,4\}} + \mathbf{b}^4_{3,\{1,3\}}\right). \end{aligned} \tag{63}$$

Node 4 sends:

$$\begin{aligned}\mathbf{X}_4 = \mathbf{U}_{\{2,3\}} &\left(\mathbf{b}^2_{4,\{3,4\}} + \mathbf{b}^3_{4,\{2,4\}}\right) \\ &+\mathbf{U}_{\{2,4\}}\mathbf{b}^2_{4,\{1,4\}} + \mathbf{U}_{\{3,4\}}\mathbf{b}^3_{4,\{1,4\}}. \end{aligned} \tag{64}$$

As mentioned, each receiving node can subtract all the interference of the signals that it can compute itself. We can rewrite the four receive signals after this interference elimination step as follows. Node 1 can construct:

$$\begin{aligned}\mathbf{Y}'_1 = &\underbrace{\mathbf{H}_{1,2}\mathbf{U}_{\{2,3\}}\mathbf{b}^1_{2,\{2,3\}} + \mathbf{H}_{1,3}\mathbf{U}_{\{2,3\}}\mathbf{b}^1_{3,\{2,3\}}}_{\text{desired signal}} \\ &\underbrace{+\mathbf{H}_{1,2}\mathbf{U}_{\{2,4\}}\mathbf{b}^1_{2,\{2,4\}} + \mathbf{H}_{1,3}\mathbf{U}_{\{3,4\}}\mathbf{b}^1_{3,\{3,4\}}}_{\text{desired signal}} \\ &+\mathbf{H}_{1,2}\mathbf{U}_{\{3,4\}}\left(\mathbf{b}^3_{2,\{2,4\}} + \mathbf{b}^4_{2,\{2,3\}}\right) \\ &+\mathbf{H}_{1,3}\mathbf{U}_{\{2,4\}}\left(\mathbf{b}^2_{3,\{3,4\}} + \mathbf{b}^4_{3,\{2,3\}}\right) \\ &+\mathbf{H}_{1,4}\mathbf{U}_{\{2,3\}}\left(\mathbf{b}^2_{4,\{3,4\}} + \mathbf{b}^3_{4,\{2,4\}}\right) + \mathbf{Z}_1, \end{aligned} \tag{65}$$

Node 2 can construct:

$$\begin{aligned}\mathbf{Y}'_2 = &\underbrace{\mathbf{H}_{2,1}\mathbf{U}_{\{2,3\}}\mathbf{b}^2_{1,\{1,3\}} + \mathbf{H}_{2,1}\mathbf{U}_{\{2,4\}}\mathbf{b}^2_{1,\{1,4\}}}_{\text{desired signal}} \\ &\underbrace{+\mathbf{H}_{2,3}\mathbf{U}_{\{2,3\}}\mathbf{b}^2_{3,\{1,3\}} + \mathbf{H}_{2,3}\mathbf{U}_{\{2,4\}}\mathbf{b}^2_{3,\{3,4\}}}_{\text{desired signal}} \\ &\underbrace{+\mathbf{H}_{2,4}\mathbf{U}_{\{2,3\}}\mathbf{b}^2_{4,\{3,4\}} + \mathbf{H}_{2,4}\mathbf{U}_{\{2,4\}}\mathbf{b}^2_{4,\{1,4\}}}_{\text{desired signal}} \\ &+\mathbf{H}_{2,1}\mathbf{U}_{\{3,4\}}\left(\mathbf{b}^3_{1,\{1,4\}} + \mathbf{b}^4_{1,\{1,3\}}\right) \\ &+\mathbf{H}_{2,3}\mathbf{U}_{\{3,4\}}\left(\mathbf{b}^1_{3,\{3,4\}} + \mathbf{b}^4_{3,\{1,3\}}\right) \\ &+\mathbf{H}_{2,4}\mathbf{U}_{\{3,4\}}\mathbf{b}^3_{4,\{2,4\}} + \mathbf{Z}_2, \end{aligned} \tag{66}$$

Node 3 can construct:

$$\begin{aligned}\mathbf{Y}_3' = &\underbrace{\mathbf{H}_{3,1}\mathbf{U}_{\{2,3\}}\mathbf{b}^3_{1,\{1,2\}} + \mathbf{H}_{3,1}\mathbf{U}_{\{3,4\}}\mathbf{b}^3_{1,\{1,4\}}}_{\text{desired signal}} \\ &\underbrace{+\mathbf{H}_{3,2}\mathbf{U}_{\{2,3\}}\mathbf{b}^3_{2,\{1,2\}} + \mathbf{H}_{3,2}\mathbf{U}_{\{3,4\}}\mathbf{b}^3_{2,\{2,4\}}}_{\text{desired signal}} \\ &\underbrace{+\mathbf{H}_{3,4}\mathbf{U}_{\{2,3\}}\mathbf{b}^3_{4,\{3,4\}} + \mathbf{H}_{3,2}\mathbf{U}_{\{3,4\}}\mathbf{b}^3_{4,\{1,4\}}}_{\text{desired signal}} \\ &+\mathbf{H}_{3,1}\mathbf{U}_{\{2,4\}}\left(\mathbf{b}^2_{1,\{1,4\}} + \mathbf{b}^4_{1,\{1,2\}}\right) \\ &+\mathbf{H}_{3,2}\mathbf{U}_{\{2,4\}}\left(\mathbf{b}^1_{2,\{2,4\}} + \mathbf{b}^4_{2,\{1,2\}}\right) \\ &+\mathbf{H}_{3,4}\mathbf{U}_{\{2,4\}}\mathbf{b}^2_{4,\{1,4\}} + \mathbf{Z}_3, \end{aligned} \tag{67}$$

Node 4 can construct:

$$\begin{aligned}\mathbf{Y}_4' = &\underbrace{\mathbf{H}_{4,1}\mathbf{U}_{\{2,4\}}\mathbf{b}^4_{1,\{1,2\}} + \mathbf{H}_{4,1}\mathbf{U}_{\{2,4\}}\mathbf{b}^4_{1,\{1,3\}}}_{\text{desired signal}} \\ &\underbrace{+\mathbf{H}_{4,2}\mathbf{U}_{\{2,4\}}\mathbf{b}^4_{2,\{1,2\}} + \mathbf{H}_{4,2}\mathbf{U}_{\{3,4\}}\mathbf{b}^4_{2,\{2,3\}}}_{\text{desired signal}} \\ &\underbrace{+\mathbf{H}_{4,3}\mathbf{U}_{\{2,4\}}\mathbf{b}^4_{3,\{2,3\}} + \mathbf{H}_{4,3}\mathbf{U}_{\{3,4\}}\mathbf{b}^4_{3,\{1,3\}}}_{\text{desired signal}} \\ &+\mathbf{H}_{4,1}\mathbf{U}_{\{2,3\}}\left(\mathbf{b}^2_{1,\{1,3\}} + \mathbf{b}^3_{1,\{1,2\}}\right) \\ &+\mathbf{H}_{4,2}\mathbf{U}_{\{2,3\}}\left(\mathbf{b}^1_{2,\{2,3\}} + \mathbf{b}^3_{2,\{1,2\}}\right) \\ &+\mathbf{H}_{4,3}\mathbf{U}_{\{2,3\}}\left(\mathbf{b}^1_{3,\{2,3\}} + \mathbf{b}^2_{3,\{1,3\}}\right) + \mathbf{Z}_4. \end{aligned} \tag{68}$$

Remark 5. *We remark that Node 2's desired signals are all precoded by precoding matrices $\mathbf{U}_{\{2,3\}}$ and $\mathbf{U}_{\{2,4\}}$ while all interference signals are precoded by matrix $\mathbf{U}_{\{3,4\}}$. Similar observations hold for Nodes 3 and 4. Node 1 instead observes desired and interference signals precoded by all three precoding matrices.*

In more general terms, each node $j \in [\mathsf{K}]\backslash\{1\}$, observes desired signals multiplied by the precoding matrices $\{\mathbf{U}_{\mathcal{R}}\}_{j\in\mathcal{R}}$ and interference signals multiplied by the precoding matrices $\{\mathbf{U}_{\mathcal{R}}\}_{j\notin\mathcal{R}}$. For Node 1, both desired and interference signals are multiplied by all possible precoding matrices.

The IA matrices $\mathbf{U}_{\{2,3\}}$, $\mathbf{U}_{\{2,4\}}$, and $\mathbf{U}_{\{3,4\}}$ are constructed based on the interference alignment idea in [19] taking into account the channel matrices that premultiply the IA matrices in the interference signals of (65)–(68). Specifically, we choose

$$\mathbf{U}_{\mathcal{R}} \triangleq \left[\prod_{\mathbf{H}\in\mathcal{H}_{\mathcal{R}}} \mathbf{H}^{\alpha_{\mathcal{R},\mathbf{H}}} \cdot \mathbf{\Xi}_{\mathcal{R}} \colon \ \ \forall \boldsymbol{\alpha}_{\mathcal{R}} \in [\eta]^4\right], \tag{69}$$

where each column of the matrix is constructed using a different exponent-vector $\boldsymbol{\alpha}_{\mathcal{R}} = (\alpha_{\mathcal{R},\mathbf{H}} \colon \mathbf{H} \in \mathcal{H}_{\mathcal{R}}) \in [\eta]^4$; η is a large number depending on the blocklength T that tends to ∞ with T; $\mathbf{\Xi}_{\mathcal{R}}$ are i.i.d. random vectors drawn according to a continuous distribution,

$$\mathcal{H}_{\{2,3\}} \triangleq \left\{\mathbf{H}_{1,4},\ \mathbf{H}_{4,1},\ \mathbf{H}_{4,2},\ \mathbf{H}_{4,3}\right\}, \tag{70}$$
$$\mathcal{H}_{\{2,4\}} \triangleq \left\{\mathbf{H}_{1,3},\ \mathbf{H}_{3,1},\ \mathbf{H}_{3,2},\ \mathbf{H}_{3,4}\right\}, \tag{71}$$
$$\mathcal{H}_{\{3,4\}} \triangleq \left\{\mathbf{H}_{1,2},\ \mathbf{H}_{2,1},\ \mathbf{H}_{2,3},\ \mathbf{H}_{2,4}\right\}. \tag{72}$$

By this choice of the precoding matrices, all interference signals at a Node 2 will lie in the columnspace of the matrix

$$\mathbf{W}_{\{3,4\}} \triangleq \left[\prod_{\mathbf{H}\in\mathcal{H}_{\{3,4\}}} \mathbf{H}^{\alpha_{\mathcal{R},\mathbf{H}}} \cdot \mathbf{\Xi}_{\{3,4\}} \colon \ \ \forall \boldsymbol{\alpha}_{\mathcal{R}} \in [\eta+1]^4\right], \tag{73}$$

while the desired signals will be separable from each other and from this interference space. The DoF achieved to this Node 2 is thus

$$\frac{6}{7}. \tag{74}$$

Similar observations hold for Nodes 3 and 4. Node 1 has a larger interference space and smaller desired signal space and only achieves a DoF of $4/7$. The SumDoF achieved by the scheme is thus

$$\text{Sum-DoF} = 22/7. \tag{75}$$

V. The General IA-Scheme without ZF

We fix a large parameter $\eta \in \mathbb{Z}^+$ (which we shall let tend to ∞) and let

$$\Gamma \triangleq \mathsf{K} \cdot (\mathsf{K} - \mathsf{r} - 1) \tag{76}$$

$$\mathsf{T} \triangleq (\mathsf{K} - 2) \cdot \binom{\mathsf{K}-2}{\mathsf{r}-1} \cdot \eta^\Gamma + \binom{\mathsf{K}-1}{\mathsf{r}} \cdot (\eta+1)^\Gamma. \tag{77}$$

In our scheme, we send the following messages to any Node $j \in [\mathsf{K}]\backslash\{1\}$:

$$\left\{ M^j_{k,\mathcal{T}} \colon \mathcal{T} \in [[\mathsf{K}]\backslash\{j\}]^{\mathsf{r}},\ k \in \mathcal{T} \right\} \tag{78}$$

and to Node 1 we send messages

$$\left\{ M^1_{k,\mathcal{T}} \colon \mathcal{T} \in [[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}},\ k \in \mathcal{T}\backslash\{\mathsf{K}\} \right\}. \tag{79}$$

Thus, as in the examples of the previous section, the last node K does not send any message to the first node 1.

For each message, construct a Gaussian codebook of power $\mathsf{P}/\binom{\mathsf{K}-1}{\mathsf{r}}$ and length η^Γ to encode each Message $M^j_{k,\mathcal{T}}$ into a codeword $\mathbf{b}^j_{k,\mathcal{T}}$. As in the previous sections, we shall use a linear precoding scheme, and thus Node $i \in [\mathsf{K}]$ can mitigate the interference caused by the codewords

$$\left\{ \mathbf{b}^j_{k,\mathcal{T}} \right\}_{\forall \mathcal{T} \colon i \in \mathcal{T}}. \tag{80}$$

As a consequence, for each set $\mathcal{R} \in [[\mathsf{K}]]^{\mathsf{r}}$, without causing non-desired interference to nodes in $\mathcal{R}$, we can use the same precoding matrix $\mathbf{U}_\mathcal{R}$ (whose choice we describe later) for all the codewords:

$$\left\{ \mathbf{b}^j_{k,\mathcal{R}\cup\{k\}\backslash\{j\}} \right\}_{\substack{k\in[\mathsf{K}]\backslash\mathcal{R}\\ j\in\mathcal{R}}}. \tag{81}$$

This idea was already used in the related works [14], [18]. In contrast to these previous works, here we do not introduce the precoding matrices $\mathbf{U}_\mathcal{R}$ for sets $\mathcal{R}$ containing 1. Instead, for any $\mathcal{R}$ not containing 1 and any $k \in \mathcal{R}$, we use the matrix $\mathbf{U}_\mathcal{R}$ also to precode the set of codewords

$$\left\{ \mathbf{b}^j_{k,\mathcal{R}\cup\{1\}\backslash\{j\}} \right\}_{j,k\in\mathcal{R}, j\neq k} \quad \cup \quad \left\{ \mathbf{b}^j_{1,\mathcal{R}\cup\{1\}\backslash\{j\}} \right\}_{j\in\mathcal{R}}, \tag{82}$$

and

$$\{\mathbf{b}^1_{k,\mathcal{R}}\}_{k\in\mathcal{R}\backslash\{\mathsf{K}\}}. \tag{83}$$

All non-intended nodes in $\mathcal{R}$ can subtract these interferences from their receive signals because they know the codewords. This trick allows us to reduce the dimension of the interference space and thus improve performance.

In Table III we illustrate which codewords $\mathbf{b}^j_{k,\mathcal{T}}$ are premultiplied by the precoding matrix $\mathbf{U}_{\{2,3\}}$ when $\mathsf{r} = 2$ and $\mathsf{K} > 3$. The entry "o" indicates that a given Node k chooses not to send a message to a given Node j or the message is premultiplied by other matrices. According to (81) the entry in column-3 and row-$\{k, 2\}$, for each $k \in \{1, 4, \ldots, \mathsf{K}\}$, corresponds to the codeword $\mathbf{b}^3_{k,\{k,2\}}$, and the entry in column-2 and row-$\{k, 3\}$, for each $k \in \{1, 4, \ldots, \mathsf{K}\}$, corresponds to codeword $\mathbf{b}^2_{k,\{k,3\}}$. According to (82), the entries in rows $\mathcal{T}$ containing index 1 correspond to the r codewords $\{\mathbf{b}^j_{k,\mathcal{T}}\}_{k\in\mathcal{T}}$. And finally, according to (83), the entry in column-1 and row-$\{2, 3\}$ corresponds to the two codewords $\mathbf{b}^1_{2,\{2,3\}}$ and $\mathbf{b}^1_{3,\{2,3\}}$. We thus conclude that all entries of the table in rows $\mathcal{T}$ containing index 1 and in row $\{2, 3\}$ correspond to $\mathsf{r} = 2$ different codewords, while all other entries correspond to only a single codeword. Similar tables can be drawn for all pairs $(k_1, k_2) \in [\mathsf{K}]$, where recall however that node K does not send any information to node 1.

Similarly, Table IV illustrates which codesymbols $\mathbf{b}^j_{k,\mathcal{T}}$ are premultiplied by the precoding matrix $\mathbf{U}_{\{2,3,4\}}$, when $\mathsf{r} = 3$ and $\mathsf{K} \geq 5$. The entries in rows $\mathcal{T}$ containing index 1 correspond to $\mathsf{r} = 3$ different codewords $\mathbf{b}^j_{k,\mathcal{T}}$, one for each $k \in \mathcal{T}$, see (82). Similarly, the entry in column-1 and row $\{2, 3, 4\}$ corresponds to the r codewords $\mathbf{b}^1_{k,\{2,3,4\}}$, for each $k \in \{2, 3, 4\}$. Any other entry of the table showing $\mathbf{U}_{\{2,3,4\}}$ corresponds to a single codeword $\mathbf{b}^j_{k,\mathcal{T}}$, where k is the single element in $\mathcal{T}\backslash\{2, 3, 4\}$.

We now describe encodings and decodings.

Encoding: Define the T-length vector of channel inputs $\mathbf{X}_k \triangleq (X_k(1), \ldots, X_k(\mathsf{T}))^T$ for each Node k. Nodes $1, \ldots, \mathsf{K}$ form the channel inputs as:

$$\mathbf{X}_1 = \sum_{\mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}} \sum_{j\in\mathcal{R}} \mathbf{U}_\mathcal{R} \mathbf{b}^j_{1,\mathcal{R}\cup\{1\}\backslash\{j\}}, \tag{84}$$

TABLE III

LET $r = 2$. THE TABLE ILLUSTRATES THE CODESYMBOLS $\mathbf{b}^j_{k,\mathcal{T}}$ THAT ARE PREMULTIPLIED BY THE PRECODING MATRIX $\mathbf{U}_{\{2,3\}}$. ENTRIES FOR SETS $\mathcal{T}$ EITHER EQUAL TO $\{2,3\}$ OR CONTAINING 1, CORRESPOND TO r TRANSMITTED CODEWORDS, ONE FROM EACH NODE IN $\mathcal{T}$. ALL OTHER ENTRIES CORRESPOND ONLY TO A SINGLE CODEWORD FROM THE NODE NOT IN $\{2,3\}$.

$\mathcal{T} \setminus j$	1	2	3
$\{1,2\}$	x	x	$\mathbf{U}_{\{2,3\}}$
$\{1,3\}$	x	$\mathbf{U}_{\{2,3\}}$	x
$\{2,3\}$	$\mathbf{U}_{\{2,3\}}$	x	x
$\{2,4\}$	o	x	$\mathbf{U}_{\{2,3\}}$
$\{2,5\}$	o	x	$\mathbf{U}_{\{2,3\}}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\{2,\mathsf{K}\}$	o	x	$\mathbf{U}_{\{2,3\}}$
$\{3,4\}$	o	$\mathbf{U}_{\{2,3\}}$	x
$\{3,5\}$	o	$\mathbf{U}_{\{2,3\}}$	x
$\vdots$	o	$\vdots$	x
$\{3,\mathsf{K}\}$	o	$\mathbf{U}_{\{2,3\}}$	x
$\{4,5\}$	o	o	o
$\vdots$	o	$\vdots$	o
$\{\mathsf{K}-1,\mathsf{K}\}$	o	o	o

TABLE IV

LET $\mathsf{r} = 3$. THE TABLE ILLUSTRATES THE CODESYMBOLS $\mathbf{b}^j_{k,\mathcal{T}}$ THAT ARE PREMULTIPLIED BY THE PRECODING MATRIX $\mathbf{U}_{\{2,3,4\}}$. ENTRIES FOR SETS $\mathcal{T}$ EITHER EQUAL TO $\{2,3,4\}$ OR CONTAINING 1, CORRESPOND TO r TRANSMITTED CODEWORDS, ONE FROM EACH NODE IN $\mathcal{T}$. ALL OTHER ENTRIES CORRESPOND ONLY TO A SINGLE CODEWORD FROM THE NODE NOT IN $\{2,3,4\}$.

$\mathcal{T} \setminus j$	1	2	3	4
$\{1,2,3\}$	x	x	x	$\mathbf{U}_{\{2,3,4\}}$
$\{1,2,4\}$	x	x	$\mathbf{U}_{\{2,3,4\}}$	x
$\{1,3,4\}$	x	$\mathbf{U}_{\{2,3,4\}}$	x	x
$\{2,3,4\}$	$\mathbf{U}_{\{2,3,4\}}$	x	x	x
$\{2,3,5\}$	o	x	x	$\mathbf{U}_{\{2,3,4\}}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\{2,3,\mathsf{K}\}$	o	x	x	$\mathbf{U}_{\{2,3,4\}}$
$\{2,3,5\}$	o	x	$\mathbf{U}_{\{2,3,4\}}$	x
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\{2,3,\mathsf{K}\}$	o	x	$\mathbf{U}_{\{2,3,4\}}$	x
$\{3,4,5\}$	o	$\mathbf{U}_{\{2,3,4\}}$	x	x
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\{3,4,\mathsf{K}\}$	o	$\mathbf{U}_{\{2,3,4\}}$	x	x

$$\mathbf{X}_k = \sum_{\mathcal{R}\in[[\mathsf{K}]\setminus\{1,k\}]^{\mathsf{r}}} \sum_{j\in\mathcal{R}} \mathbf{U}_{\mathcal{R}} \mathbf{b}^j_{k,\mathcal{R}\cup\{k\}\setminus\{j\}} + \sum_{\substack{\mathcal{R}\in[[\mathsf{K}]\setminus\{k\}]^{\mathsf{r}}:\\ 1\in\mathcal{R}}} \sum_{j\in\mathcal{R}} \mathbf{U}_{\mathcal{R}\cup\{k\}\setminus\{1\}} \mathbf{b}^j_{k,\mathcal{R}\cup\{k\}\setminus\{j\}}, \quad k\in[\mathsf{K}-1]\setminus\{1\}, \tag{85}$$

$$\mathbf{X}_{\mathsf{K}} = \sum_{\mathcal{R}\in[[\mathsf{K}-1]\setminus\{1\}]^{\mathsf{r}}} \sum_{j\in\mathcal{R}} \mathbf{U}_{\mathcal{R}} \mathbf{b}^j_{\mathsf{K},\mathcal{R}\cup\{\mathsf{K}\}\setminus\{j\}} + \sum_{\substack{\mathcal{R}\in[[\mathsf{K}-1]]^{\mathsf{r}}:\\ 1\in\mathcal{R}}} \sum_{j\in\mathcal{R}\setminus\{1\}} \mathbf{U}_{\mathcal{R}\cup\{\mathsf{K}\}\setminus\{1\}} \mathbf{b}^j_{\mathsf{K},\mathcal{R}\cup\{\mathsf{K}\}\setminus\{j\}}, \tag{86}$$

where the precoding matrices $\{\mathbf{U}_{\mathcal{R}}\}_{\mathcal{R}\in[[\mathsf{K}]\setminus\{1\}]^{\mathsf{r}}}$ are described shortly.

Decoding: After receiving the respective sequence of T channel outputs $\mathbf{Y}_j \triangleq (Y_{j,1},\ldots,Y_{j,\mathsf{T}})$, for $j\in[\mathsf{K}]$, each node removes the influence of the codewords corresponding to the messages that it can compute itself. The nodes' "cleaned" signals can then be written as:

$$\mathbf{Y}'_1 = \underbrace{\sum_{\substack{\mathcal{R}\in[[\mathsf{K}]]^{\mathsf{r}}:\\ 1\in\mathcal{R}}} \sum_{k\in[\mathsf{K}-1]\setminus\mathcal{R}} \mathbf{H}_{1,k} \mathbf{U}_{\mathcal{R}\cup\{k\}\setminus\{1\}} \mathbf{b}^1_{k,\mathcal{R}\cup\{k\}\setminus\{1\}}}_{\text{desired signal}}$$

$$+\sum_{\mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}}\ \sum_{k\in[\mathsf{K}]\backslash\mathcal{R}}\mathbf{H}_{1,k}\mathbf{U}_{\mathcal{R}}\mathbf{v}_{\mathcal{R},k}+\mathbf{Z}_1, \tag{87a}$$

$$\mathbf{Y}'_j=\underbrace{\sum_{\substack{\mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}:\\ j\in\mathcal{R}}}\ \sum_{k\in[\mathsf{K}]\backslash\mathcal{R}}\mathbf{H}_{j,k}\mathbf{U}_{\mathcal{R}}\mathbf{b}^{j}_{k,\mathcal{R}\cup\{k\}\backslash\{j\}}}_{\text{desired signal}}$$
$$+\underbrace{\sum_{\substack{\mathcal{R}\in[[\mathsf{K}]]^{\mathsf{r}}:\\ 1,j\in\mathcal{R}}}\ \sum_{k\in[\mathsf{K}]\backslash\mathcal{R}}\mathbf{H}_{j,k}\mathbf{U}_{\mathcal{R}\cup\{k\}\backslash\{1\}}\mathbf{b}^{j}_{k,\mathcal{R}}}_{\text{desired signal}}$$
$$+\sum_{\substack{\mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}:\\ j\notin\mathcal{R}}}\ \sum_{\substack{k\in[\mathsf{K}]\backslash\mathcal{R}:\\ k\neq j}}\mathbf{H}_{j,k}\mathbf{U}_{\mathcal{R}}\mathbf{v}_{\mathcal{R},k}$$
$$+\sum_{\substack{\mathcal{R}\in[[\mathsf{K}]]^{\mathsf{r}}:\\ 1\in\mathcal{R},j\notin\mathcal{R}}}\ \sum_{k\in[\mathsf{K}]\backslash\mathcal{R}}\mathbf{H}_{j,k}\mathbf{U}_{\mathcal{R}\cup\{k\}\backslash\{1\}}\mathbf{v}_{\mathcal{R},k}+\mathbf{Z}_j,$$
$$j\in[\mathsf{K}]\backslash\{1\}, \tag{87b}$$

where for ease of notation we defined for Nodes $k\in[\mathsf{K}-1]$:

$$\mathbf{v}_{\mathcal{R},k}\triangleq\sum_{j\in\mathcal{R}}\mathbf{b}^{j}_{k,\mathcal{R}\cup\{k\}\backslash\{j\}},\qquad\forall\mathcal{R}\in[[\mathsf{K}]\backslash\{k\}]^{\mathsf{r}}, \tag{88}$$

and for the last Node K, since its signal to Node 1 is absent:

$$\mathbf{v}_{\mathcal{R},\mathsf{K}}\triangleq\sum_{j\in\mathcal{R}\backslash\{1\}}\mathbf{b}^{j}_{k,\mathcal{R}\cup\{k\}\backslash\{j\}},\qquad\forall\mathcal{R}\in[[\mathsf{K}-1]]^{\mathsf{r}}. \tag{89}$$

Each Node j zero-forces the non-desired interference terms of its "cleaned" signal and decodes its intended messages $\{M^{j}_{k,\mathcal{T}}\}$.

Choice of IA Matrices $\{\mathbf{U}_{\mathcal{R}}\}$ and Analysis of Signal and Interference Spaces: Inspired by the IA scheme in [19], we choose each $\mathsf{T}\times\eta^{\Gamma}$ precoding matrix $\mathbf{U}_{\mathcal{R}}$ so that its column-span includes all power products (with powers from 1 to η) of the channel matrices $\mathbf{H}_{j,k}$ that premultiply $\mathbf{U}_{\mathcal{R}}$ in (87) in the non-desired interference terms. Thus, for $\mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}$:

$$\mathbf{U}_{\mathcal{R}}\triangleq\left[\prod_{\mathbf{H}\in\mathcal{H}_{\mathcal{R}}}\mathbf{H}^{\alpha_{\mathcal{R},\mathbf{H}}}\cdot\mathbf{\Xi}_{\mathcal{R}}\colon\ \forall\boldsymbol{\alpha}_{\mathcal{R}}\in[\eta]^{\Gamma}\right], \tag{90}$$

where $\{\mathbf{\Xi}_{\mathcal{R}}\}_{\mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}}$ are i.i.d. random vectors independent of all channel matrices, noises, and messages,

$$\mathcal{H}_{\mathcal{R}}\triangleq\big\{\mathbf{H}_{j,k}\colon\ j\in[\mathsf{K}]\backslash\mathcal{R},\ k\in[\mathsf{K}]\backslash\{j\}\big\}\backslash\big\{\mathbf{H}_{1,k}\colon k\in\mathcal{R}\big\}, \tag{91}$$

and $\boldsymbol{\alpha}_{\mathcal{R}}\triangleq(\alpha_{\mathcal{R},\mathbf{H}}\colon\mathbf{H}\in\mathcal{H}_{\mathcal{R}})$. Notice that $|\mathcal{H}_{\mathcal{R}}|=\Gamma$ for any $\mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}$.

Since the column-span of $\mathbf{U}_{\mathcal{R}}$ contains all power products of powers 1 to η of the channel matrices $\mathbf{H}\in\mathcal{H}_{\mathcal{R}}$, we have

$$\mathrm{span}(\mathbf{H}\cdot\mathbf{U}_{\mathcal{R}})\subseteq\mathrm{span}(\mathbf{W}_{\mathcal{R}}),\qquad\mathbf{H}\in\mathcal{H}_{\mathcal{R}}, \tag{92}$$

where we defined the $\mathsf{T}\times\eta^{\Gamma}$-matrix

$$\mathbf{W}_{\mathcal{R}}=\left[\prod_{\mathbf{H}\in\mathcal{H}_{\mathcal{R}}}\mathbf{H}^{\alpha_{\mathcal{R},\mathbf{H}}}\cdot\mathbf{\Xi}_{\mathcal{R}}\colon\ \forall\boldsymbol{\alpha}_{\mathcal{R}}\in[\eta+1]^{\Gamma}\right],$$
$$\text{for }\mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}. \tag{93}$$

As a consequence, and by the choice of T in (77), the signal and interference space at Rx 1 is represented by the $\mathsf{T}\times\mathsf{T}$-matrix:

$$\mathbf{\Lambda}_1=\Big[\underbrace{\mathbf{D}_1}_{\text{signal space}},\ \underbrace{[\mathbf{W}_{\mathcal{R}}]_{\mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}}}_{\text{interference space}}\Big], \tag{94}$$

where the signal subspace is given by a collection of $\mathsf{T}\times\Big((\mathsf{K}-2)\cdot\binom{\mathsf{K}-2}{\mathsf{r}-1}\cdot\eta^{\Gamma}\Big)$-matrices

$$\mathbf{D}_1\triangleq\Big[\mathbf{H}_{1,k}\mathbf{U}_{\mathcal{R}}\Big]_{\substack{k\in[\mathsf{K}-1]\backslash\{1\},\\ \mathcal{R}\in[[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}:\\ k\in\mathcal{R}}}. \tag{95}$$

The signal space at Rx $j \in [\mathsf{K}]\backslash\{1\}$ is represented by the $\mathsf{T} \times \tilde{\mathsf{T}}$ matrix:

$$\mathbf{\Lambda}_j \triangleq \Big[\underbrace{\mathbf{D}_j}_{\text{signal space}} , \underbrace{[\mathbf{W}_{\mathcal{R}}]_{\mathcal{R} \in [[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}:\, j \notin \mathcal{R}}}_{\text{interference space}} \Big]. \tag{96}$$

where the signal subspace $\mathbf{D}_j$ is given by the collection of $\mathsf{T} \times \left(\mathsf{r} \cdot \binom{\mathsf{K}-1}{\mathsf{r}} \cdot \eta^{\Gamma}\right)$-matrices

$$\mathbf{D}_j \triangleq [\mathbf{H}_{j,k}\mathbf{U}_{\mathcal{R}}]_{\substack{\mathcal{R} \in [[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}: \\ j \in \mathcal{R}, \\ k \in [\mathsf{K}]\backslash\{j\}}} \tag{97}$$

and

$$\tilde{\mathsf{T}} \triangleq \mathsf{r} \cdot \binom{\mathsf{K}-1}{\mathsf{r}} \cdot \eta^{\Gamma} + \binom{\mathsf{K}-2}{\mathsf{r}} \cdot (\eta+1)^{\Gamma}. \tag{98}$$

According to Lemmas 2 and 3 below, $\{\mathbf{\Lambda}_j\}$ is full column-rank if each column has different exponent vector $\boldsymbol{\alpha}$, which follows by the way we constructed the matrices $\mathbf{U}_{\mathcal{R}}$ and $\mathbf{W}_{\mathcal{R}}$. Indeed:

- For each $\mathcal{R} \in [[\mathsf{K}]\backslash\{1\}]^{\mathsf{r}}$, matrices $\mathbf{U}_{\mathcal{R}}$ and $\mathbf{W}_{\mathcal{R}}$ are constructed using a dedicated i.i.d. vector $\mathbf{\Xi}_{\mathcal{R}}$ that is independent of all other random variables in the system and thus the vectors $\mathbf{\Xi}_{\mathcal{R}}$ can play the roles of the vectors $\mathbf{\Xi}_i$ in Lemma 3.
- For each term $\mathbf{H}\mathbf{U}_{\mathcal{R}}$ in (95) and (97), we have $\mathbf{H} \notin \mathcal{H}_{\mathcal{R}}$. Thus $\mathbf{H}$ is not used in the construction of neither $\mathbf{U}_{\mathcal{R}}$ nor $\mathbf{W}_{\mathcal{R}}$ and induces a unique exponent on the corresponding columns in the signal space which is 0 in all columns of the interference space $\mathbf{W}_{\mathcal{R}}$.

This proves that based on the "cleaned" signal (87), each receiving node j can separate the various desired signals from each other as well as from the non-desired interfering signals. Since each codeword $\mathbf{b}^j_{k,\mathcal{T}}$ occupies η^{Γ} dimensions out of the T dimensions, we obtain that whenever

$$\frac{|\mathbf{b}^j_{k,\mathcal{T}}|}{\mathsf{T}} \leq \frac{\eta^{\Gamma}}{\mathsf{T}} \log \mathsf{P} + o(\log \mathsf{P}), \tag{99}$$

for an appropriate function $o(\log \mathsf{P})$ that grows slowlier than $\log \mathsf{P}$, each codeword $\mathbf{b}^j_{k,\mathcal{T}}$ can be decoded with arbitrary small probability of error as $\eta \to \infty$.

Since $(\mathsf{K}-2) \cdot \binom{\mathsf{K}-2}{\mathsf{r}-1}$ codewords are sent to Node 1, and $\mathsf{r}\binom{\mathsf{K}-1}{\mathsf{r}}$ codewords to any other Node $j = 2, \ldots, \mathsf{K}$, and since

$$\lim_{\eta \to \infty} \frac{\eta^{\Gamma}}{\mathsf{T}} = \frac{1}{(\mathsf{K}-2)\binom{\mathsf{K}-2}{\mathsf{r}-1} + \binom{\mathsf{K}-1}{\mathsf{r}}}, \tag{100}$$

we conclude that a sum-DoF of

$$\begin{aligned} \text{Sum-DoF} &= \frac{(\mathsf{K}-2) \cdot \binom{\mathsf{K}-2}{\mathsf{r}-1} + (\mathsf{K}-1)\mathsf{r}\binom{\mathsf{K}-1}{\mathsf{r}}}{(\mathsf{K}-2)\binom{\mathsf{K}-2}{\mathsf{r}-1} + \binom{\mathsf{K}-1}{\mathsf{r}}} \\ &= \frac{\mathsf{r}(\mathsf{K}-1)^2 + \mathsf{r}(\mathsf{K}-2)}{\mathsf{r}(\mathsf{K}-2) + \mathsf{K} - 1} \end{aligned} \tag{101}$$

is achievable over the system. This establishes achievability of (15).

Lemma 2. *Let $\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_m$ be independent random vectors with i.i.d. entries drawn according to continuous distributions. for any $L \leq m$ and L* different *exponent vectors*

$$\boldsymbol{\alpha}_j = (\alpha_{j,1}, \ldots, \alpha_{j,m}) \in \mathbb{Z}_+^m, \quad j \in [L],$$

the $m \times L$ matrix $\mathbf{M}$ with row-i and column-j entry

$$M_{i,j} = \prod_{k=1}^{m} (s_{i,})^{\alpha_{j,k}}, \quad i \in [m],\ j \in [L], \tag{102}$$

is full rank almost surely.

Lemma 3. *Consider numbers $\{n_1, n_2, \cdots, n_{\tilde{\mathsf{K}}}\} \in \mathbb{Z}_+^{\tilde{\mathsf{K}}}$ so that their sum $C \triangleq \sum_{i=1}^{\tilde{\mathsf{K}}} n_i \leq \mathsf{T}$. Assume that for each $i \in [\tilde{\mathsf{K}}]$ and $k \in [n_i]$, $\mathbf{B}_{i,k} \in \mathbb{C}^{\mathsf{T}\times\mathsf{T}}$ is a diagonal matrix so that all square sub-matrices of the following matrices $\{\mathbf{B}_i\}_{i \in \tilde{\mathsf{K}}}$ are full rank:*

$$\mathbf{B}_i \triangleq \left[\mathbf{B}_{i,1} \cdot \mathbf{1}_{\mathsf{T}}, \mathbf{B}_{i,2} \cdot \mathbf{1}_{\mathsf{T}}, \cdots, \mathbf{B}_{i,n_i} \cdot \mathbf{1}_{\mathsf{T}}\right], \quad i \in [\tilde{\mathsf{K}}], \tag{103}$$

where $\mathbf{1}_{\mathsf{T}}$ denotes a T-dimensional all-one column vector.

Let further $\{\Xi_i\}_{i\in\tilde{\mathsf{K}}}$ be independent T-vectors with entries drawn i.i.d. from continuous distributions and define the $\mathsf{T}\times n_i$-matrices

$$\mathbf{A}_i \triangleq [\mathbf{B}_{i,1}\cdot\Xi_i, \mathbf{B}_{i,2}\cdot\Xi_i, \cdots, \mathbf{B}_{i,n_i}\cdot\Xi_i], \quad i\in[\tilde{\mathsf{K}}]. \tag{104}$$

Then, the $\mathsf{T}\times C$-matrix

$$\mathbf{\Lambda} \triangleq \left[\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_{\tilde{\mathsf{K}}}\right] \tag{105}$$

has full column rank almost surely.

Proof: We assume that the matrix $\mathbf{\Lambda}$ is a square matrix i.e. $C=\mathsf{T}$. If $\mathsf{T}>C$, we take a square submatrix of $\mathbf{\Lambda}$ and perform the same proof steps on the submatrix.

Define

$$F\left(\Xi_1,\ldots,\Xi_{\tilde{\mathsf{K}}}\right) \triangleq \det(\mathbf{\Lambda}) \tag{106}$$

which is a polynomial of $\Xi_1,\Xi_2,\cdots,\Xi_{\tilde{\mathsf{K}}}$ as the determinant is a polynomial of the entries of $\mathbf{\Lambda}$.

For the vectors

$$\boldsymbol{d}_i = [\ \underbrace{0,\cdots 0,}_{(n_1+\cdots+n_{i-1})\text{ 0s}}\ \underbrace{1,\cdots 1,}_{n_i\text{ 1s}}\ \underbrace{0,\cdots 0}_{(n_{i+1}+\cdots+n_{\tilde{\mathsf{K}}})\text{ 0s}}\ ]^T, \quad i\in\tilde{\mathsf{K}}, \tag{107}$$

the polynomial evaluates to

$$F\left(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_{\tilde{\mathsf{K}}}\right) = \det\begin{pmatrix}\mathbf{B}'_1 & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & \mathbf{B}'_2 & \cdots & \mathbf{0}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{B}'_{\tilde{\mathsf{K}}}\end{pmatrix} \tag{108}$$

$$= \prod_{i=1}^{\tilde{\mathsf{K}}}\det(\mathbf{B}'_i) \neq 0 \tag{109}$$

where $\mathbf{B}'_i$ is the $n_i\times n_i$ square sub-matrix of $\mathbf{B}_i$ consisting of its rows $(n_1+\cdots+n_{i-1}+1)$ to $(n_1+\cdots+n_{i-1}+n_i)$. The inequality holds by our assumption that all square sub-matrices of $\mathbf{B}_i$ are full rank.

We conclude that F is a non-zero polynomial and thus $F\left(\Xi_1,\ldots,\Xi_{\tilde{\mathsf{K}}}\right)$ equals 0 with probability 0 because the entries of $\Xi_1,\Xi_2,\cdots,\Xi_{\tilde{\mathsf{K}}}$ are drawn independently from continuous distributions. ■

VI. A Scheme with IA and ZF for K odd and $\mathsf{r}=(\mathsf{K}-1)/2$

Our second scheme has a similar precoding structure as the first scheme. However, now each signal sent from a set $\mathcal{T}$ to a node $j\in[\mathsf{K}]\backslash\mathcal{T}$ is zero-forced at a group of $\mathsf{r}-1$ nodes in $[\mathsf{K}]\backslash\{\mathcal{T}\cup\{j\}\}$ and thus effectively causes interference only at a single remaining node. This necessitates that the r nodes storing a single message cooperate in their transmission.

We choose to precode all Messages that cause interference at a given Node ℓ by the same precoding matrix $\mathbf{U}_\ell$, for $\ell\in[\mathsf{K}]$, and construct this precoding matrix so that all interferences at this Node ℓ align, thus leaving the remaining space for signaling dimensions. In other words, if we use precoding matrix $\mathbf{U}_\ell$ for the transmission of a message from group $\mathcal{T}$ to Node j, then we zero-force this signal at all nodes in $[\mathsf{K}]\backslash\{\mathcal{T}\cup\{j,\ell\}\}$.

Table V indicates the precoding matrix used to transmit each Message $M_{\mathcal{T}}^j$, when $\mathsf{K}=5$ and $\mathsf{r}=2$. We observe that in this case our scheme transmits a message for each set $\mathcal{T}\in[[\mathsf{K}]]^{\mathsf{r}}$ and $j\in[\mathsf{K}]\backslash\mathcal{T}$. The table implicitly also indicates how to zero-force the transmit codewords. For example, Message $M_{\{1,2\}}^3$ is precoded by matrix $\mathbf{U}_5$ and its transmission is thus zero-forced at node 4.

For $\mathsf{K}=7$ and $\mathsf{r}=3$, Table VI indicates the messages precoded by the matrix $\mathbf{U}_7$ and thus causing interference to Node 7. The general rule for $\mathsf{K}\geq 7$ with K odd, is that we use precoding matrix $\mathbf{U}_\ell$ exactly $\mathsf{K}-\mathsf{r}-1=\mathsf{r}$ times in each column j, namely for the sets

$$\mathcal{T}\in\{\{j-\mathsf{r}+1,\ldots,j-1,t\}\colon t\in[\mathsf{K}]\backslash\{j-\mathsf{r}+1,\ldots,j,\ell\}, \tag{110}$$

where the indices in (110) need to be understood with omission of the index ℓ and then taken modulo $\mathsf{K}-1$. For example, as depicted in Table VI, for $\mathsf{K}=7$, $\mathsf{r}=3$, and $j=2$ the follwing sets $\mathcal{T}$ send a message to Node 2 using precoding matrix $\mathbf{U}_7$:

$$\mathcal{T}\in\{\{1,3,6\},\quad\{1,4,6\},\quad\{1,5,6\}\}, \tag{111}$$

where here we associated the index 1 with $j-1$, the index 6 as $j-2$, and t runs over the set remaining set $[7]\backslash\{1,2,6\}$. Notice that in each column there

Using above rule, for large values of K there are transmit sets $\mathcal{T}\in[[\mathsf{K}]]^{\mathsf{r}}$ and receiving Nodes j so that two different precoding matrices $\mathbf{U}_\ell$ and $\mathbf{U}_{\ell'}$, for $\ell,\ell'\in[\mathsf{K}]\backslash(\mathcal{T}\cup\{j\})$ and $\ell\neq\ell'$, are assigned to the row-$\mathcal{T}$ and column-j entry of the table. We capture this phenomena in the set $\mathcal{L}_{\mathcal{T}}^j$, which for each $\mathcal{T}\in[[\mathsf{K}]]^{\mathsf{r}}$ and $j\in[\mathsf{K}]\backslash\mathcal{T}$ contains all indices ℓ so that the row-$\mathcal{T}$ and column-j entry contains matrix $\mathbf{U}_\ell$. We can then rephrase above observation as the remark that $|\mathcal{L}_{\mathcal{T}}^j|$ can be larger than 1. If this is the case, in our scheme Message $M_{\mathcal{T}}^j$ needs to be split into two submessages, which are then precoded by the matrices $\mathbf{U}_\ell$ and $\mathbf{U}_{\ell'}$, respectively. We shall therefore introduce the general notation $M_{\mathcal{T}}^{j,\ell}$ to denote the message that a transmit set $\mathcal{T}$ sends to the receiving Node j using precoding matrix $\mathbf{U}_\ell$. Obviously, the messages $M_{\mathcal{T}}^{j,\ell}$ either denotes a submessage of $M_{\mathcal{T}}^j$ or a submessage thereof. We further introduce for each

We make the following observations:

Remark 6. 1) *For a given j, consider all rows $\mathcal{T}_1, \ldots, \mathcal{T}_{\mathsf{r}}$ showing precoding matrix $\mathbf{U}_\ell$ in column j. Each set $\mathcal{T}_i$ contains a unique index (namely t in* (110)*) that is not present in the other sets $\mathcal{T}_k$, for $k \in [\mathsf{r}]\backslash\{i\}$.*

2) *For each set $\mathcal{T}$ there is a unique column-index j with entry $\mathbf{U}_\ell$ in the table.*

3) *For each $\ell \in [\mathsf{K}]$, each node in $[\mathsf{K}]\backslash\{\ell\}$ receives $\mathsf{K}-\mathsf{r}-1=\mathsf{r}$ different codewords that are precoded with matrix $\mathbf{U}_\ell$.*

Items 1) and 2) apply also to the precoding matrix assignment in Table V for $\mathsf{K}=5$ and $\mathsf{r}=2$.

To see item 2) in above remark, notice that for $\mathsf{r} \geq 3$ the tuple of $\mathsf{r}-1$ consecutive (omitting index ℓ and modulo $\mathsf{K}-1$) numbers $(j-\mathsf{r}+1, \ldots, j-1)$ can only be present in a set $\mathcal{T}$ that we use in column j but not in other columns.

TABLE V

CHOICE OF PRECODING MATRICES IN OUR SCHEME FOR $\mathsf{K}=5$ AND $r=2$. EACH SIGNAL THAT IS PRECODED WITH MATRIX $\mathbf{U}_\ell$ IS ZERO-FORCED AT THE UNIQUE NODE $i \neq \ell$ THAT IS NEITHER THE INTENDED NODE j NOR PART OF THE TRANSMITTING SET $\mathcal{T}$.

$\mathcal{T} \setminus j$	1	2	3	4	5
$\{1,2\}$	x	x	$\mathbf{U}_5$	$\mathbf{U}_3$	$\mathbf{U}_4$
$\{1,3\}$	x	$\mathbf{U}_5$	x	$\mathbf{U}_2$	$\mathbf{U}_4$
$\{1,4\}$	x	$\mathbf{U}_5$	$\mathbf{U}_2$	x	$\mathbf{U}_3$
$\{1,5\}$	x	$\mathbf{U}_4$	$\mathbf{U}_2$	$\mathbf{U}_3$	x
$\{2,3\}$	$\mathbf{U}_4$	x	x	$\mathbf{U}_5$	$\mathbf{U}_1$
$\{2,4\}$	$\mathbf{U}_5$	x	$\mathbf{U}_1$	x	$\mathbf{U}_3$
$\{2,5\}$	$\mathbf{U}_3$	x	$\mathbf{U}_4$	$\mathbf{U}_1$	x
$\{3,4\}$	$\mathbf{U}_5$	$\mathbf{U}_1$	x	x	$\mathbf{U}_2$
$\{3,5\}$	$\mathbf{U}_4$	$\mathbf{U}_1$	x	$\mathbf{U}_2$	x
$\{4,5\}$	$\mathbf{U}_2$	$\mathbf{U}_3$	$\mathbf{U}_1$	x	x

We now describe the encoding and decodings and analyze the signal and interference spaces.

We fix a large parameter $\eta \in \mathbb{Z}^+$ (which we shall let tend to ∞) and define

$$\Gamma \triangleq \mathsf{r}(\mathsf{K}-1) \tag{112}$$

$$\mathsf{T} \triangleq \mathsf{r}(\mathsf{K}-1)\cdot\eta^\Gamma + (\eta+1)^\Gamma. \tag{113}$$

For each message $M_{\mathcal{T}}^{j,\ell}$, construct a Gaussian codebook of power P and length η^Γ to encode each Message $M_{\mathcal{T}}^{j,\ell}$ into a codeword $\mathbf{b}_{\mathcal{T}}^{j,\ell}$.

1) Encoding: Tx $q \in [\mathsf{K}]$ forms its inputs as:

$$\mathbf{X}_q = \sum_{\substack{\mathcal{T}\subseteq[[\mathsf{K}]]^{\mathsf{r}}:\\ q\in\mathcal{T}}} \sum_{j\in[\mathsf{K}]\backslash\mathcal{T}} \sum_{\ell\in\mathcal{L}_{\mathcal{T}}^j} \mathbf{V}^q_{[\mathsf{K}]\backslash\{j,\ell\},\mathcal{T}}\, \mathbf{U}_\ell\, \mathbf{b}_{\mathcal{T}}^{j,\ell}, \tag{114}$$

where $\mathbf{U}_1, \ldots, \mathbf{U}_K$ denote the precoding-matrices that we will construct shortly and $\mathbf{V}_{\mathcal{R},\mathcal{T}}$ denotes the node-q component of a matrix that zero-forces the signals emitted by the set of nodes $\mathcal{T}$ at the receiving nodes $\mathcal{R}\backslash\mathcal{T}$ but not at the other nodes. This precoding matrix is also scaled in a way to satisfy the block-power constraint for all channel input signals. (Implicitly here we assume that $\mathcal{T}$ and $\mathcal{R}\backslash\mathcal{T}$ are of sizes r and $\mathsf{r}-1$, respectively, so that the desired precoding matrix exists with probability 1.) For any set $\mathcal{T}=\{q_1,\ldots,q_{\mathsf{r}}\}$ and $\mathcal{R}$, define

$$\mathbf{V}_{\mathcal{R},\mathcal{T}} \triangleq \begin{pmatrix} \mathbf{V}^{q_1}_{\mathcal{R},\mathcal{T}} \\ \vdots \\ \mathbf{V}^{q_{\mathsf{r}}}_{\mathcal{R},\mathcal{T}} \end{pmatrix}. \tag{115}$$

With the proposed precoding matrices, and after each receive Node p removes the signals it can produce itself (i.e., the signals stemming from sets $\mathcal{T}$ containing p), we can rewrite Node p's equivalent receive signal as:

$$\begin{aligned}\mathbf{Y}'_p = &\sum_{\substack{\mathcal{T}\subseteq[[\mathsf{K}]]^{\mathsf{r}}:\\ p\notin\mathcal{T}}} \mathbf{H}_{p,\mathcal{T}} \sum_{\ell\in\mathcal{L}_{\mathcal{T}}^p} \mathbf{V}_{[\mathsf{K}]\backslash\{p,\ell\},\mathcal{T}}\, \mathbf{S}_{\mathcal{T}}^p\, \mathbf{U}_\ell\, \mathbf{b}_{\mathcal{T}}^{p,\ell} \\ &+ \sum_{j\in[\mathsf{K}]\backslash p} \sum_{\substack{\mathcal{T}\subseteq[[\mathsf{K}]\backslash\{j,p\}]^{\mathsf{r}}:\\ p\in\mathcal{L}_{\mathcal{T}}^j}} \underbrace{\mathbf{H}_{p,\mathcal{T}}\mathbf{V}_{[\mathsf{K}]\backslash\{j,p\},\mathcal{T}}\, \mathbf{S}_{\mathcal{T}}^j}_{\triangleq\mathbf{G}_{\mathcal{T}}^{j,p}} \mathbf{U}_p \mathbf{b}_{\mathcal{T}}^{j,p} + \mathbf{Z}_p.\end{aligned} \tag{116}$$

TABLE VI

CODEWORDS PRECODED BY MATRIX $\mathbf{U}_7$ WHEN $\mathsf{K}=7$ AND $\mathsf{r}=3$. EACH CODEWORD IS ZERO-FORCED AT THE TWO NODES NOT BELONGING TO THE TRANSMIT $\mathcal{T}$ AND NOT EQUAL TO THE RECEIVE NODE j OR TO 7.

$\mathcal{T}\setminus j$	1	2	3	4	5	6	7
$\{1,2,3\}$	x	x	x	$\mathbf{U}_7$	o	o	o
$\{1,2,4\}$	x	x	$\mathbf{U}_7$	x	o	o	o
$\{1,2,5\}$	x	x	$\mathbf{U}_7$	o	x	o	o
$\{1,2,6\}$	x	x	$\mathbf{U}_7$	o	o	x	o
$\{1,2,7\}$	x	x	o	o	o	o	x
$\{1,3,4\}$	x	o	x	x	$\mathbf{U}_7$	o	o
$\{1,3,5\}$	x	o	x	o	x	o	o
$\{1,3,6\}$	x	$\mathbf{U}_7$	x	o	o	x	o
$\{1,3,7\}$	x	o	x	o	o	o	x
$\{1,4,5\}$	x	o	o	x	x	$\mathbf{U}_7$	o
$\{1,4,6\}$	x	$\mathbf{U}_7$	o	x	o	x	o
$\{1,4,7\}$	x	o	o	x	o	o	x
$\{1,5,6\}$	x	$\mathbf{U}_7$	o	o	x	x	o
$\{1,5,7\}$	x	o	o	o	x	o	x
$\{1,6,7\}$	x	o	o	o	o	x	x
$\{2,3,4\}$	o	x	x	x	$\mathbf{U}_7$	o	o
$\{2,3,5\}$	o	x	x	$\mathbf{U}_7$	x	o	o
$\{2,3,6\}$	o	x	x	$\mathbf{U}_7$	o	x	o
$\{2,3,7\}$	o	x	x	o	o	o	x
$\{2,4,5\}$	o	x	o	x	x	$\mathbf{U}_7$	o
$\{2,4,6\}$	o	x	o	x	o	x	o
$\{2,4,7\}$	o	x	o	x	o	o	x
$\{2,5,6\}$	$\mathbf{U}_7$	x	o	o	x	x	o
$\{2,5,7\}$	o	x	o	o	x	o	x
$\{2,6,7\}$	o	x	o	o	o	x	x
$\{3,4,5\}$	o	o	x	x	x	$\mathbf{U}_7$	o
$\{3,4,6\}$	o	o	x	x	$\mathbf{U}_7$	x	o
$\{3,4,7\}$	o	o	x	x	o	o	x
$\{3,5,6\}$	$\mathbf{U}_7$	x	o	x	x	o	o
$\{3,5,7\}$	o	o	x	o	x	o	x
$\{3,6,7\}$	o	o	x	o	o	x	x
$\{4,5,6\}$	$\mathbf{U}_7$	o	o	x	x	x	o
$\{4,5,7\}$	o	o	o	x	x	o	x
$\{4,6,7\}$	o	o	o	x	o	x	x
$\{5,6,7\}$	o	o	o	o	x	x	x

Remark 7. *Notice that all interfering signals at receiving Node p are precoded by the same matrix $\mathbf{U}_p$.*

2) IA Matrices $\{\mathbf{U}_p\}$: Inspired by the IA schemes in [19], [20], we choose each $\mathsf{T}\times\eta^{\Gamma}$ precoding matrix $\mathbf{U}_\ell$ so that its column-span includes all power products (with powers from 1 to η) of the "generalized" channel matrices $\mathbf{G}_{\mathcal{T}}^{j,\ell}$ that premultiply $\mathbf{U}_\ell$ in (116):

$$\mathcal{G}_\ell \triangleq \left\{\mathbf{G}_{\mathcal{T}}^{j,\ell}\colon \forall j\in[\mathsf{K}]\backslash\{\ell\},\;\; \forall\mathcal{T}\;\text{ s.t. }\ell\in\mathcal{L}_{\mathcal{T}}^{j}\right\}. \tag{117}$$

Since the network is memoryless, the "generalized" channel matrices $\mathbf{G}_{\mathcal{T}}^{j,\ell}$ are diagonal T-by-T matrices. Then, for each $k\in[\mathsf{K}]$ construct the T-by-η^{Γ} matrix $\mathbf{U}_\ell$ by selecting each of its columns as a product of the elements in $\mathcal{G}_\ell$ multiplied with an independent i.i.d. random vector $\mathbf{\Xi}_\ell$:

$$\mathbf{U}_\ell=\left[\prod_{\mathbf{G}\in\mathcal{G}_\ell}\mathbf{G}^{\alpha_{\ell,\mathbf{G}}}\cdot\mathbf{\Xi}_\ell\colon\;\;\forall\boldsymbol{\alpha}_\ell\in[\eta]^{\Gamma}\right], \tag{118}$$

where $\boldsymbol{\alpha}_\ell\triangleq(\alpha_{\ell,\mathbf{G}}\colon\quad\mathbf{G}\in\mathcal{G}_\ell)$ are exponent vectors of length Γ.

3) Decoding at Rx p: The way we constructed our precoding matrices, we have:

$$\text{span}(\mathbf{G}\cdot\mathbf{U}_p)\subseteq\text{span}(\mathbf{W}_p),\qquad\mathbf{G}\in\mathcal{G}_p, \tag{119}$$

where we defined the $\mathsf{T}\times(\eta+1)^{\Gamma}$-matrix

$$\mathbf{W}_p=\left[\prod_{\mathbf{G}\in\mathcal{G}_p}\mathbf{G}^{\alpha_{p,\mathbf{G}}}\cdot\mathbf{\Xi}_p\colon\;\;\forall\boldsymbol{\alpha}_p\in[\eta+1]^{\Gamma}\right]. \tag{120}$$

The signal subspace at Rx p is given by:

$$\mathbf{D}_p\triangleq\left[\mathbf{H}_{p,\mathcal{T}}\,\mathbf{V}_{[\mathsf{K}]\backslash\{p,\ell\},\mathcal{T}}\,\mathbf{S}_{\mathcal{T}}^{p}\,\mathbf{U}_\ell\right]_{\substack{\mathcal{T}\in[\mathsf{K}]^{\mathsf{r}}\colon\, p\notin\mathcal{T}\\ \ell\in\mathcal{L}_{\mathcal{T}}^{p}}} \tag{121}$$

and its interference subspace is included in $\mathbf{W}_p$.

For each column of the signal space $\mathbf{D}_p$, Rx p projects its receive signal $\mathbf{Y}'_p$ onto a vector that is orthogonal to all columns in the interference space $\mathbf{W}_p$ and also to all other columns of $\mathbf{D}_p$. It can then decode the desired Messages in an interference-free manner based on the various projections.

4) Analysis of Signal and Interference Subspaces: If the columns of the matrix $\mathbf{D}_p$ are linearly independent of each other and of the columns of $\mathbf{W}_p$, the following DoF is achievable to each Node p when we let $\eta \to \infty$:

$$\begin{cases} 6/7, & \mathsf{K}=5 \\ \frac{\mathsf{r}(\mathsf{K}-1)}{\mathsf{r}(\mathsf{K}-1)+1}, & \mathsf{K} \geq 7 \end{cases}. \tag{122}$$

Remark 8. *Notice the difference in our expressions* (122) *for* $\mathsf{K}=5$ *and* $\mathsf{K}\geq 7$*. In fact, for* $\mathsf{K}=5$*, we fill all* $\binom{\mathsf{K}-1}{\mathsf{r}}$ *rows of each column with one of the precoding matrices. For* $\mathsf{K}\geq 7$ *however we use each precoding matrix only* $\mathsf{K}-\mathsf{r}-1$ *times in each column, and since in each row we can use* $(\mathsf{K}-1)$ *precoding matrices and* $(\mathsf{K}-\mathsf{r}-1)(\mathsf{K}-1) \leq \binom{\mathsf{K}-1}{\mathsf{r}}$ *for* $\mathsf{K}\geq 7$*, some of the entries in the table remain empty.*

Accordingly, the Sum-DoF of the entire system is given by

$$\text{Sum-DoF} = \begin{cases} 30/7, & \mathsf{K}=5 \\ \frac{\mathsf{K}\mathsf{r}(\mathsf{K}-1)}{\mathsf{r}(\mathsf{K}-1)+1}, & \mathsf{K} \geq 7 \end{cases}. \tag{123}$$

The way we constructed the precoding matrices and by Lemma 3 at the end of the previous Section V, it suffices to show that for each $\ell \neq p$ the matrix

$$\mathbf{\Lambda}_{p,\ell} = \left[\bar{\mathbf{G}}^{p,\ell}_{\mathcal{T}} \prod_{\substack{i\in[\mathsf{K}]\backslash\{\ell\} \\ \tilde{\mathcal{T}}\in[[\mathsf{K}]\backslash\{i,\ell\}]^{\mathsf{r}}: \\ \ell\in\mathcal{L}_{\tilde{\mathcal{T}}^i}}} \left(\mathbf{G}^{i,\ell}_{\tilde{\mathcal{T}}}\right)^{\alpha_{\ell,i,\tilde{\mathcal{T}}}} \mathbf{1}_{\mathsf{T}} : \ \ \forall \boldsymbol{\alpha}_\ell \in [\eta]^{\Gamma} \right]_{\substack{\mathcal{T}\in[[\mathsf{K}]\backslash\{\ell,p\}]^{\mathsf{r}} \\ \ell\in\mathcal{L}^p_{\mathcal{T}}}} \tag{124}$$

has only full-rank square submatrices, where

$$\bar{\mathbf{G}}^{p,\ell}_{\mathcal{T}} \triangleq \mathbf{H}_{p,\mathcal{T}} \mathbf{V}_{[\mathsf{K}]\backslash\{p,\ell\},\mathcal{T}}\ \mathbf{S}^p_{\mathcal{T}} \tag{125}$$

is a diagonal T-by-T matrix.

By construction and the diagonal structure of the "generalized" channel coefficients, any square sub-matrix of the matrix $\mathbf{\Lambda}_{p,\ell}$, for $\ell \neq p$, has the same form as matrix $\mathbf{M}$ in Lemma 4 ahead, Equation (126), when one considers the diagonal entries of the "generalized" channel matrices $\{\bar{\mathbf{G}}^{p,\ell}_{\mathcal{T}}\}_{\mathcal{T}:\ \ell\in\mathcal{L}^p_{\mathcal{T}}}$ and $\{\mathbf{G}^{i,\ell}_{\mathcal{T}}\}_{\substack{i\in[\mathsf{K}]\backslash\{\ell\} \\ \mathcal{T}:\ \ell\in\mathcal{L}^i_{\mathcal{T}}}}$ as the outcomes of the functions $\mathbf{f}_1,\ldots,\mathbf{f}_L$. The inputs of these functions are the random channel coefficients $\{H_{p',q}(t)\}$ and the entries of the diagonal matrices $\{\mathbf{S}^j_{\mathcal{T}}\}$ which in Lemma 4 can thus play the role of the i.i.d. random variables in the vector $\mathbf{x}_t$. By Lemma 4 it thus suffices to show that the "generalized" channel matrices $\{\bar{\mathbf{G}}^{p,\ell}_{\mathcal{T}}\}_{\ell\in\mathcal{L}^p_{\mathcal{T}}}$ and $\{\mathbf{G}^{i,\ell}_{\mathcal{T}}\}_{\ell\in\mathcal{L}^i_{\mathcal{T}}}$ are algebraically independent functions of the channel coefficients $\{H_{p',q}(t)\}$ and the entries of $\{\mathbf{S}^j_{\mathcal{T}}\}$.

Notice that $\{\bar{\mathbf{G}}^{p,\ell}_{\mathcal{T}}\}_{\ell\in\mathcal{L}^p_{\mathcal{T}}}$ and $\{\mathbf{G}^{i,\ell}_{\mathcal{T}}\}_{\ell\in\mathcal{L}^i_{\mathcal{T}}}$ are all diagonal matrices with the t-th elements only depending on the time-t channel coefficients $\{H_{p',q}(t)\}$ and the t-th components of the diagonal matrices $\{\mathbf{S}^j_{\mathcal{T}}\}$. We restrict to a single time-instance $t \in \{1,\ldots,\mathsf{T}\}$ and drop this time-index for convenience. Henceforth, the random variables $\{\bar{G}^{p,\ell}_{\mathcal{T}}\}_{\ell\in\mathcal{L}^p_{\mathcal{T}}}$ and $\{G^{i,\ell}_{\mathcal{T}}\}_{\ell\in\mathcal{L}^i_{\mathcal{T}}}$, and $\{S^j_{\mathcal{T}}\}$ refer to the t-th diagonal elements of the corresponding matrices and $\{H_{p',q}\}$ to the corresponding time-t channel coefficients.

Recall that p and ℓ are fixed and notice the following:

- Since each transmitted codeword interferes only at a single node, each element $G^{i,\ell}_{\mathcal{T}}$, with $\mathcal{T}$ and i so that $\ell \in \mathcal{L}^i_{\mathcal{T}}$, depends on a different auxiliary random variable $S^i_{\mathcal{T}}$. The functions $\{G^{i,\ell}_{\mathcal{T}}\}_{\ell\in\mathcal{L}^i_{\mathcal{T}}}$ are thus algebraically independent because the factors in front of these auxiliary random variables $\mathbf{H}_{\ell,\mathcal{T}}\mathbf{V}_{[\mathsf{K}]\backslash\{i,\ell\},\mathcal{T}}$ are non-zero with probability 1.
- Each function $\bar{G}^{p,\ell}_{\mathcal{T}}$ only depends on $S^p_{\mathcal{T}}$ but not on the other S-random variables.
- Each set $\mathcal{T}$ for which $\ell \in \mathcal{L}^p_{\mathcal{T}}$ has a distinct element t (see (110) and Item 1) of Remark 6). For a given such set $\mathcal{T}$ for which $\ell \in \mathcal{L}^p_{\mathcal{T}}$, the two functions $\bar{G}^{p,\ell}_{\mathcal{T}}$ and $G^{p,\ell}_{\mathcal{T}}$ thus each depends on a different channel coefficient $H_{p,t}$ and $H_{\ell,t}$, respectively, that does not influence any of the other functions $\bar{G}^{p,\ell}_{\tilde{\mathcal{T}}}$ and $G^{p,\ell}_{\tilde{\mathcal{T}}}$ for sets $\tilde{\mathcal{T}} \neq \mathcal{T}$ satisfying $\ell \in \mathcal{L}_{p,\tilde{\mathcal{T}}}$. Again, with probability 1 the factors in front of the distinct channel coefficients $H_{p,t}$ and $H_{\ell,t}$ are non-zero, which establishes algebraic independence of our functions.

All these considerations can be combined to conclude that the functions $\{\bar{G}^{p,\ell}_{\mathcal{T}}\}_{\ell\in\mathcal{L}^p_{\mathcal{T}}}$ and $\{G^{i,\ell}_{\mathcal{T}}\}_{\substack{i\in[\mathsf{K}]\backslash\{\ell\} \\ \ell\in\mathcal{L}^i_{\mathcal{T}}}}$ are algebraically independent.

Lemma 4 (Lemmas 3 and 4 in [20]). *Let $\mathbf{f} = (f_1 f_2, ..., f_m) \in \mathbb{C}^m$ be a vector of rational functions and let $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_\tau$ be i.i.d. random vectors with i.i.d. entries drawn according to continuous distributions. Define*

$$\mathbf{s}_i \triangleq \mathbf{f}(\mathbf{x}_i), \quad i \in [\tau].$$

For any $L \leq \tau$ and L different *exponent vectors*

$$\boldsymbol{\alpha}_j = (\alpha_{j,1}, \ldots, \alpha_{j,m}) \in \mathbb{Z}_+^m, \quad j \in [L],$$

the $T \times L$ matrix $\mathbf{M}$ with row-i and column-j entry

$$M_{i,j} = \prod_{k=1}^{m} (s_{i,k})^{\alpha_{j,k}}, \quad i \in [\tau], \ j \in [L], \tag{126}$$

is full rank almost surely, if and only if the functions $\mathbf{f}$ are algebraically independent, i.e., if and only if the Jacobian $[\frac{\partial f_n}{\partial x_i}]_{(i,n)}$ is of rank m.

VII. Proof of the NDT Lower Bound in Theorem 1

Consider a fixed file assignment (map phase), and for any positive power P a sequence (in T) of wireless distributed computing systems satisfying (9) for the given file assignment. (Since for finite N there are only a finite number of different file assignments irrespective of P and T, we can fix the assignment.) The following limiting behaviour must hold.

Lemma 5. *Consider two disjoint sets $\mathcal{T}$ and $\mathcal{R}$ of same size*

$$|\mathcal{T}| = |\mathcal{R}|, \tag{127}$$

and define $\mathcal{F} \triangleq [\mathsf{K}] \backslash (\mathcal{R} \cup \mathcal{T})$. Let $\mathcal{M} \subseteq [\mathsf{N}]$ be the set of files known only to nodes $\mathcal{T}$ but not to any other node and partition the set of all IVAs $\mathcal{A}$ it into the following disjoint subsets:

$$\mathcal{W}_r \triangleq \{a_{j,m}\}_{\substack{j \in \mathcal{R} \\ m \in [\mathsf{N}] \backslash \mathcal{M}_j}}, \tag{128}$$

$$\mathcal{W}_t \triangleq \{a_{j,m}\}_{\substack{j \in (\mathcal{T} \cup \mathcal{F}) \\ m \in \mathcal{M} \backslash \mathcal{M}_j}}. \tag{129}$$

For any sequence of distributed computing systems:

$$d \triangleq \varlimsup_{P \to \infty} \varlimsup_{\mathsf{T} \to \infty} \frac{\mathsf{A}}{\mathsf{T} \log \mathsf{P}} \leq \frac{|\mathcal{T}|}{|\mathcal{W}_t| + |\mathcal{W}_r|} \tag{130}$$

(Notice that $\mathcal{W}_r$ denotes the set of all IVAs intended to nodes in $\mathcal{R}$ and $\mathcal{W}_t$ the set of IVAs deduced from files in $\mathcal{M}$ and intended for nodes not in $\mathcal{R}$.)

Proof. Denote by $\mathcal{H}$ the set of all channel coefficients to all nodes in the system and define $\mathcal{W}_c \triangleq \mathcal{A} \backslash (\mathcal{W}_r \cup \mathcal{W}_t$. Since channel coefficients and IVAs are independent, we have

$$\begin{aligned}
H(\mathcal{W}_t, \mathcal{W}_r) &= H(\mathcal{W}_t, \mathcal{W}_r | \mathcal{W}_c, \mathcal{H}) & (131)\\
&= I(\mathcal{W}_t, \mathcal{W}_r; \mathbf{Y}_\mathcal{R} | \mathcal{W}_c, \mathcal{H}) + H(\mathcal{W}_t, \mathcal{W}_r | \mathcal{W}_c, \mathbf{Y}_\mathcal{R}, \mathcal{H}) & (132)\\
&= h(\mathbf{Y}_\mathcal{R} | \mathcal{W}_c, \mathcal{H}) - h(\mathbf{Z}_\mathcal{R}) \\
&\quad + H(\mathcal{W}_r | \mathcal{W}_c, \mathbf{Y}_\mathcal{R}, \mathcal{H}) + H(\mathcal{W}_t | \mathcal{W}_r, \mathcal{W}_c, \mathbf{Y}_\mathcal{R}, \mathcal{H}) & (133)\\
&\leq h(\mathbf{Y}_\mathcal{R} | \mathcal{W}_c, \mathcal{H}) - h(\mathbf{Z}_\mathcal{R}) \\
&\quad + \mathsf{T}\epsilon_\mathsf{T} + H(\mathcal{W}_t | \mathcal{W}_r, \mathcal{W}_c, \mathbf{Y}_\mathcal{R}, \mathcal{H}), & (134)
\end{aligned}$$

where we defined $\mathbf{Y}_\mathcal{A} \triangleq [\mathbf{Y}_j]_{j \in \mathcal{A}}$ for a set $\mathcal{A} \subseteq [\mathsf{K}]$ and ϵ_T is a vanishing sequence as $\mathsf{T} \to \infty$. Here the inequality holds by Fano's inequality, because $\mathcal{W}_r$ is decoded from $\mathbf{Y}_\mathcal{R}$ and $\mathcal{W}_c$, and because we impose vanishing probability of error (9).

Again by Fano's inequality and by (9), there exists a vanishing sequence ϵ'_T such that

$$\begin{aligned}
H(\mathcal{W}_t | \mathcal{W}_r, \mathcal{W}_c, \mathbf{Y}_\mathcal{R}, \mathcal{H}) &\leq I(\mathcal{W}_t; \mathbf{Y}_{(\mathcal{F} \cup \mathcal{T})} | \mathcal{W}_r, \mathcal{W}_c, \mathbf{Y}_\mathcal{R}, \mathcal{H}) + \mathsf{T}\epsilon'_\mathsf{T} & (135)\\
&= h(\mathbf{Y}_{(\mathcal{F} \cup \mathcal{T})} | \mathcal{W}_r, \mathcal{W}_c, \mathbf{Y}_\mathcal{R}, \mathcal{H}) & (136)\\
&\quad - h(\mathbf{Y}_{(\mathcal{F} \cup \mathcal{T})} | \mathcal{W}_r, \mathcal{W}_t, \mathcal{W}_c, \mathbf{Y}_\mathcal{R}, \mathcal{H}) + \mathsf{T}\epsilon'_\mathsf{T} & (137)\\
&\leq h(\bar{\mathbf{Y}}_{(\mathcal{F} \cup \mathcal{T})} | \bar{\mathbf{Y}}_\mathcal{R}, \mathcal{H}) - h(\mathbf{Z}_{(\mathcal{F} \cup \mathcal{T})}) + \mathsf{T}\epsilon'_\mathsf{T}, & (138)
\end{aligned}$$

where $\bar{\mathbf{Y}}_\mathcal{A} \triangleq [\bar{\mathbf{Y}}_j]_{j \in \mathcal{A}}$ and $\bar{\mathbf{Y}}_j$ denotes Node j's "cleaned" signal without the inputs that do not depend on files in $\mathcal{M}$ but only on IVAs $\mathcal{W}_r \cup \mathcal{W}_c$:

$$\bar{\mathbf{Y}}_j \triangleq \mathbf{H}_{j,\mathcal{T}} \mathbf{X}_\mathcal{T} + \mathbf{Z}_j, \qquad j \in \mathcal{T} \cup \mathcal{F}.$$

Here, $\mathbf{H}_{\mathcal{A},\mathcal{B}}$ denotes the channel matrix from set $\mathcal{B}$ to set $\mathcal{A}$.

To bound the first term in (138), we introduce a random variable E indicating whether the matrix $\mathbf{H}_{\mathcal{R},\mathcal{T}}$ is invertible ($E=1$) or not ($E=0$). If this matrix is invertible and $E=1$, then the input vector $\mathbf{X}_{\mathcal{T}}$ can be computed from $\bar{\mathbf{Y}}_{\mathcal{R}}$ up to noise terms. Based on this observation and defining the residual noise terms

$$\bar{\mathbf{Z}}_j \triangleq \mathbf{Z}_j - \mathbf{H}_{j,\mathcal{T}}\mathbf{H}_{\mathcal{R},\mathcal{T}}^{-1}\mathbf{Z}_{\mathcal{R}}, \quad \text{if } E=1, \tag{139}$$

we obtain:

$$\begin{aligned}
h(\bar{\mathbf{Y}}_{(\mathcal{F}\cup\mathcal{T})}|\bar{\mathbf{Y}}_{\mathcal{R}},\mathcal{H}) &\leq \mathbb{P}(E=1)\cdot h\left(\bar{\mathbf{Z}}_{(\mathcal{F}\cup\mathcal{T})}|\bar{\mathbf{Y}}_{\mathcal{R}},\mathcal{H},E=1\right) \\
&\quad + \mathbb{P}(E=0)\cdot h\left(\bar{\mathbf{Y}}_{(\mathcal{F}\cup\mathcal{T})}|\bar{\mathbf{Y}}_{\mathcal{R}},\mathcal{H},E=0\right) && (140)\\
&\leq h\left(\bar{\mathbf{Z}}_{(\mathcal{F}\cup\mathcal{T})}\right) + \mathbb{P}(E=0) h\left(\bar{\mathbf{Y}}_{(\mathcal{F}\cup\mathcal{T})}|\bar{\mathbf{Y}}_{\mathcal{R}},\mathcal{H},E=0\right). && (141)
\end{aligned}$$

Since the channel coefficients follow continuous distribution, $\mathbf{H}_{\mathcal{R},\mathcal{T}}$ is invertible almost surely, implying $\mathbb{P}(E=0)=0$. By the boundedness of the entropy term $h\left(\bar{\mathbf{Y}}_{(\mathcal{F}\cup\mathcal{T})}|\bar{\mathbf{Y}}_{\mathcal{R}},\mathcal{H},E=0\right)$ (since power P and channel coefficients are bounded), this implies

$$h(\bar{\mathbf{Y}}_{(\mathcal{F}\cup\mathcal{T})}|\bar{\mathbf{Y}}_{\mathcal{R}},\mathcal{H}) \leq h(\bar{\mathbf{Z}}_{(\mathcal{F}\cup\mathcal{T})}),$$

which combined with (134) and (138) yields:

$$\begin{aligned}
H(\mathcal{W}_t,\mathcal{W}_r) &\leq h(\mathbf{Y}_{\mathcal{R}}|\mathcal{H}) - h(\mathbf{Z}_{\mathcal{R}}) + h(\bar{\mathbf{Z}}_{(\mathcal{F}\cup\mathcal{T})}) \\
&\quad - h(\mathbf{Z}_{(\mathcal{F}\cup\mathcal{T})}) + \mathsf{T}(\epsilon_{\mathsf{T}}+\epsilon'_{\mathsf{T}}) \\
&\leq \mathsf{T}|\mathcal{R}|\log(\mathsf{P}) + \mathsf{T}C_{\mathsf{T},\mathcal{H}}, && (142)
\end{aligned}$$

where $C_{\mathsf{T},\mathcal{H}}$ is a function that is uniformly bounded over all realizations of channel matrices and powers P. Noticing

$$H(\mathcal{W}_t,\mathcal{W}_r) = \mathsf{A}(|\mathcal{W}_t|+|\mathcal{W}_r|), \tag{143}$$

dividing (142) by $\mathsf{T}\log(\mathsf{P})$, and letting $\mathsf{P}\to\infty$, establishes the lemma because $|\mathcal{R}|=|\mathcal{T}|$ and $\mathsf{T}C_{\mathsf{T},\mathcal{H}}$ is bounded. □

For each subset $\mathcal{T}\subseteq[\mathsf{K}]$, let $\mathcal{B}_{\mathcal{T}}^j$ denote the set of IVAs that are computed exclusively at nodes in set $\mathcal{T}$ and intended for reduce function j. Define $b_{\mathcal{T}}=|\mathcal{B}_{\mathcal{T}}^j|$, which does not depend on the index of the reduce function $j\in[\mathsf{K}]\backslash\mathcal{T}$.

Choose two disjoint subsets $\mathcal{T}$ and $\mathcal{R}$ of same size $|\mathcal{T}|=|\mathcal{R}|$. By Lemma 5, and rewriting the sets $\mathcal{W}_t$ and $\mathcal{W}_r$ in the lemma in terms of the sets $\{\mathcal{B}_{\mathcal{T}}^j\}$, we obtain:

$$\begin{aligned}
\frac{|\mathcal{T}|}{d} &\geq \sum_{\mathcal{T}\subseteq[\mathsf{K}]}\sum_{j\in\mathcal{R}\backslash\mathcal{T}}|\mathcal{B}_{\mathcal{T}}^j| + \sum_{\mathcal{G}\subseteq\mathcal{T}}\sum_{j\in[\mathsf{K}]\backslash(\mathcal{R}\cup\mathcal{G})}|\mathcal{B}_{\mathcal{G}}^j| && (144)\\
&= \sum_{\mathcal{T}\subseteq[\mathsf{K}]}|\mathcal{R}\backslash\mathcal{T}|\cdot b_{\mathcal{T}} + \sum_{\mathcal{G}\subseteq\mathcal{T}}(\mathsf{K}-|\mathcal{R}|-|\mathcal{G}|)\cdot b_{\mathcal{G}}. && (145)
\end{aligned}$$

Summing up Equality (145) over all sets $\mathcal{T}$ and $\mathcal{R}$ of constant size $t\leq\mathsf{K}/2$, we obtain:

$$\begin{aligned}
\binom{\mathsf{K}}{t}\cdot\binom{\mathsf{K}-t}{t}\cdot\frac{t}{d} &\geq \sum_{\mathcal{T}\in[[\mathsf{K}]]^t}\sum_{\mathcal{R}\in[[\mathsf{K}]\backslash\mathcal{T}]^t}\sum_{\mathcal{T}\subseteq[\mathsf{K}]}|\mathcal{R}\backslash\mathcal{T}|\cdot b_{\mathcal{T}} \\
&\quad + \sum_{\mathcal{T}\in[[\mathsf{K}]]^t}\sum_{\mathcal{R}\in[[\mathsf{K}]\backslash\mathcal{T}]^t}\sum_{\mathcal{G}\subseteq\mathcal{T}}(\mathsf{K}-t-|\mathcal{G}|)\cdot b_{\mathcal{G}} && (146)\\
&= \sum_{\mathcal{T}\subseteq[\mathsf{K}]}\binom{\mathsf{K}}{t}\cdot\binom{\mathsf{K}-t}{t}\cdot(\mathsf{K}-|\mathcal{T}|)\frac{t}{\mathsf{K}}b_{\mathcal{T}} \\
&\quad + \sum_{\substack{\mathcal{G}\subseteq[\mathsf{K}]:\\|\mathcal{G}|\leq t}}\binom{\mathsf{K}-|\mathcal{G}|}{t-|\mathcal{G}|}\cdot\binom{\mathsf{K}-t}{t}\cdot(\mathsf{K}-t-|\mathcal{G}|)b_{\mathcal{G}} && (147)\\
&= \binom{\mathsf{K}}{t}\cdot\binom{\mathsf{K}-t}{t}\cdot t\left(\mathsf{N}-\frac{\mathsf{rN}}{\mathsf{K}}\right) \\
&\quad + \sum_{\substack{\mathcal{G}\subseteq[\mathsf{K}]:\\|\mathcal{G}|\leq t}}\binom{\mathsf{K}-|\mathcal{G}|}{t-|\mathcal{G}|}\cdot\binom{\mathsf{K}-t}{t}\cdot(\mathsf{K}-t-|\mathcal{G}|)b_{\mathcal{G}}, && (148)
\end{aligned}$$

where we define $\binom{a}{0}=1$ for any positive integer a. The first equality holds because for a given set $\mathcal{T}$, each element of $[\mathsf{K}]\backslash\mathcal{T}$ is present in a fraction of t/K pairs of the admissible sets $(\mathcal{R},\mathcal{T})$ and the last equality holds because

$$\sum_{\mathcal{T}\subseteq[\mathsf{K}]} b_{\mathcal{T}}=\mathsf{N}, \qquad \sum_{\mathcal{T}\subseteq[\mathsf{K}]} |\mathcal{T}|\cdot b_{\mathcal{T}}\leq \mathsf{r}\cdot\mathsf{N}. \tag{149}$$

Dividing both sides of (148) by $\binom{\mathsf{K}}{t}\binom{\mathsf{K}-t}{t}t$, and defining $b_i\triangleq\sum_{\mathcal{T}\in[[\mathsf{K}]]^i} b_{\mathcal{T}}$, for any $t\in[\lfloor \mathsf{K}/2\rfloor]$ we obtain:

$$\frac{1}{d}\geq \mathsf{N}-\frac{\mathsf{r}\cdot\mathsf{N}}{\mathsf{K}}+\min_{\substack{b_1,\ldots,b_{\mathsf{K}}\in\mathbb{Z}^+:\\ \sum_{i=1}^{\mathsf{K}} b_i=\mathsf{N}\\ \sum_{i=1}^{\mathsf{K}} ib_i\leq \mathsf{r}\mathsf{N}}} \sum_{i=1}^{t} C_t(i)b_i, \quad t\in[\lfloor \mathsf{K}/2\rfloor], \tag{150}$$

where $C_t(i)$ is defined in (20).

For any $t\in[\lfloor \mathsf{K}/2\rfloor]$, the sequence of coefficients $C_t(1),C_t(2),\ldots,C_t(t)$ is convex and non-increasing, see Appendix A. Based on this convexity, it can be shown (see Appendix B) that for any $\mathsf{r}<t+1$ there exists a solution to the minimization problem in (150) putting only positive masses on $b^*_{\lfloor \mathsf{r}\rfloor}$ and $b^*_{\lceil \mathsf{r}\rceil}$ in the unique way satisfying

$$b^*_{\lfloor \mathsf{r}\rfloor}+b^*_{\lceil \mathsf{r}\rceil}=\mathsf{N} \tag{151}$$

$$\lfloor \mathsf{r}\rfloor b^*_{\lfloor \mathsf{r}\rfloor}+\lceil \mathsf{r}\rceil b^*_{\lceil \mathsf{r}\rceil}=\mathsf{r}\mathsf{N}. \tag{152}$$

For $\mathsf{r}\geq t+1$ an optimal solution consists of setting $b^*_{\lfloor \mathsf{r}\rfloor}=\mathsf{N}$, in which case the minimization in (150) evaluates to 0.

For $\mathsf{r}\geq 2$, the lower bound on the NDT in the theorem is then obtained by plugging these optimum values into bound (150) for the choice $t=\lfloor \mathsf{K}/2\rfloor$. For $\mathsf{r}=1$ we choose $t=1$, and for $\mathsf{r}\in(1,2)$ we maximize over the value of t.

Acknowledgements

The authors would like to thank P. Ciblat for helpful discussions at the beginning of this work. This work has been supported by National Key R&D Program of China under Grant No 2020YFB1807504 and National Science Foundation of China Key Project under Grant No 61831007.

Appendix A
Proof of Monotonicity and Convexity of values $C_i^{(t)}$

We shall prove monotonicity and convexity of the values

$$D_i^{(t)}\triangleq\frac{\mathsf{K}!}{t!}t\cdot C_i^{(t)} \tag{153}$$

$$=\frac{(\mathsf{K}-i)!}{(t-i)!}(\mathsf{K}-t-i), \qquad i\in[t]. \tag{154}$$

Notice that

$$D_{i-1}^{(t)}=D_i^{(t)}\frac{\mathsf{K}-i+1}{t-i+1}\cdot\frac{\mathsf{K}-t-i+1}{\mathsf{K}-t-i} \tag{155}$$

$$D_{i+1}^{(t)}=D_i^{(t)}\frac{t-i}{\mathsf{K}-i}\cdot\frac{\mathsf{K}-t-i-1}{\mathsf{K}-t-i}, \tag{156}$$

and the monotonicity

$$D_{i-1}^{(t)}>D_i^{(t)} \tag{157}$$

simply follows because $K>t$ implies

$$\mathsf{K}-i+1>t-i+1 \quad \text{and} \quad \mathsf{K}-t-i+1>\mathsf{K}-t-i. \tag{158}$$

To prove convexity, we shall prove that

$$D_{i+1}^{(t)}+D_{i-1}^{(t)}\geq 2D_i^{(t)}, \tag{159}$$

or equivalently

$$\frac{t-i}{\mathsf{K}-i}\cdot\frac{\mathsf{K}-t-i-1}{\mathsf{K}-t-i}+\frac{\mathsf{K}-i+1}{t-i+1}\cdot\frac{\mathsf{K}-t-i+1}{\mathsf{K}-t-i}\geq 2. \tag{160}$$

Multiplying both sides with the denominators, we see that this condition is equivalent to

$$\begin{aligned}&(t-i)(\mathsf{K}-t-i-1)(t-i+1)\\&+(\mathsf{K}-i+1)(\mathsf{K}-t-i+1)(\mathsf{K}-i)\end{aligned}$$

$$\geq 2(\mathsf{K}-i)(t-i+1)(\mathsf{K}-t-i). \tag{161}$$

and after rearranging the terms:

$$\begin{aligned}&(\mathsf{K}-t)(\mathsf{K}-i)(\mathsf{K}-t-i)+(\mathsf{K}-i)(\mathsf{K}-i+1)\\ &\geq (\mathsf{K}-t)(t-i+1)(\mathsf{K}-t-i)+(t-i)(t-i+1).\end{aligned} \tag{162}$$

This last inequality is easily verified by noting that $\mathsf{K} \geq t+1$.

APPENDIX B
PROOF OF STRUCTURE OF MINIMIZER

Start with any feasible vector $b_1, \ldots, b_{\mathsf{K}}$ and consider two indices $i < j$ with non-zero masses, $b_i > 0$ and $b_j > 0$. Updating this vector as

$$b'_i = b_i - \Delta, \qquad \text{and} \qquad b'_{i+1} = b_{i+1} + \Delta, \tag{163}$$

$$b'_{j-1} = b_{j-1} + \Delta, \qquad \text{and} \qquad b'_j = b_j - \Delta, \tag{164}$$

for any $\Delta \in [0, \min\{b_i, b_j\}]$, results again in a feasible solution vector, which has smaller objective function due to the convexity of the coefficients $\{C_i^{(t)}\}$.

Applying this argument iteratively, one can conclude that there must exist an optimal solution vector where all entries are zero except for two masses $b_k > 0$ and $b_{k+1} \geq 0$. Since $\sum_{i=1}^{\mathsf{K}} i b_i \leq \mathsf{rN}$, the index k cannot exceed r. By the decreasing monotonicity of the coefficients $C_i^{(t)}$, the optimal solution must then be to choose $b_{\lfloor \mathsf{r} \rfloor} > 0$ and $b_{\lfloor \mathsf{r} \rfloor + 1} \geq 0$ and all other masses equal to 0. Since there is a unique such choice satisfying $\sum_{i=1}^{\mathsf{K}} i b_i \leq \mathsf{rN}$ and $\sum_{i=1}^{\mathsf{K}} b_i = \mathsf{N}$, this concludes the proof.

REFERENCES

[1] J. S. Ng, W. Y. B. Lim, N. C. Luong, Z. Xiong, A. Asheralieva, D. Niyato, C. Leung, and C. Miao, "A comprehensive survey on coded distributed computing: Fundamentals, challenges, and networking applications," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1800–1837, Jun. 2021.

[2] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008.

[3] M. Chowdhury, M. Zaharia, J. Ma, M. I. Jordan, and I. Stoica, "Managing data transfers in computer clusters with orchestra," *ACM SIGCOMM computer communication review*, vol. 41, no. 4, pp. 98–109, 2011.

[4] S. Li, M. A. Maddah-Ali, Q. Yu, and A. S. Avestimehr, "A fundamental tradeoff between computation and communication in distributed computing," *IEEE Transactions on Information Theory*, vol. 64, no. 1, pp. 109–128, Jan. 2018.

[5] Q. Yan, S. Yang, and M. Wigger, "Storage-computation-communication tradeoff in distributed computing: Fundamental limits and complexity," 2019. [Online]. Available: https://hal.telecom-paris.fr/hal-02288592/document

[6] F. Xu, S. Shao, and M. Tao, "New results on the computation-communication tradeoff for heterogeneous coded distributed computing," *IEEE Transactions on Communications*, vol. 69, no. 4, pp. 2254–2270, Apr. 2021.

[7] S. Li, M. A. Maddah-Ali, and A. S. Avestimehr, "A unified coding framework for distributed computing with straggling servers," in *2016 IEEE Globecom Workshops (GC Wkshps)*, Dec. 2016, pp. 1–6.

[8] J. Zhang and O. Simeone, "Improved latency-communication trade-off for map-shuffle-reduce systems with stragglers," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 8172–8176, iSSN: 2379-190X.

[9] L. Zheng, M. Sun, Y. Luo, X. Song, C. Yang, F. Hu, and M. Yu, "Utilizing mapreduce to improve probe-car track data mining," *ISPRS International Journal of Geo-Information*, vol. 7, no. 7, 2018.

[10] Y. Khourdifi and M. Bahaj, "Hadoop and mapreduce technology as a solution for wireless body area networks in e-health," in *Proceedings of the 2nd International Conference on Computing and Wireless Communication Systems*, ser. ICCWCS'17. New York, NY, USA: Association for Computing Machinery, 2017. [Online]. Available: https://doi.org/10.1145/3167486.3167503

[11] S. Li, Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "A scalable framework for wireless distributed computing," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2643–2654, Oct. 2017.

[12] K. Yang, Y. Shi, and Z. Ding, "Data shuffling in wireless distributed computing via low-rank optimization," *IEEE Transactions on Signal Processing*, vol. 67, no. 12, pp. 3087–3099, Jun. 2019.

[13] A. Paris, H. Mirghasemi, I. Stupia, and L. Vandendorpe, "Energy-efficient edge-facilitated wireless collaborative computing using map-reduce," 07 2019, pp. 1–5.

[14] K. Yuan and Y. Wu, "Coded wireless distributed computing via interference alignment," *IEEE International Symposium on Information Theory*, p. 6, 2022.

[15] F. Li, J. Chen, and Z. Wang, "Wireless mapreduce distributed computing," *IEEE Transactions on Information Theory*, vol. 65, no. 10, pp. 6101–6114, Oct. 2019.

[16] Y. Bi, P. Ciblat, M. Wigger, and Y. Wu, "DoF of a cooperative X-channel with an application to distributed computing," in *2022 IEEE International Symposium on Information Theory (ISIT)*, Jun. 2022, pp. 566–571, iSSN: 2157-8117.

[17] F. Li, J. Chen, and Z. Wang, "Wireless mapreduce distributed computing," 06 2018, pp. 1286–1290.

[18] J. Hachem, U. Niesen, and S. N. Diggavi, "Degrees of freedom of cache-aided wireless interference networks," *IEEE Transactions on Information Theory*, vol. 64, no. 7, pp. 5359–5380, Jul. 2018.

[19] S. Jafar and S. Shamai, "Degrees of freedom region of the MIMO X-channel," *IEEE Transactions on Information Theory*, vol. 54, no. 1, pp. 151–170, Jan. 2008.

[20] V. S. Annapureddy, A. El Gamal, and V. V. Veeravalli, "Degrees of freedom of interference channels with CoMP transmission and reception," *IEEE Transactions on Information Theory*, vol. 58, no. 9, pp. 5740–5760, Sep. 2012.