Sparse factorization of the square all-ones matrix of arbitrary order[*]

Xin Jiang[†] Edward Duc Hien Nguyen[‡] César A. Uribe[‡] Bicheng Ying[§]

November 18, 2024

Abstract

In this paper, we study sparse factorization of the (scaled) square all-ones matrix J of arbitrary order. We introduce the concept of hierarchically banded matrices and propose two types of hierarchically banded factorization of J: the reduced hierarchically banded (RHB) factorization and the doubly stochastic hierarchically banded (DSHB) factorization. Based on the DSHB factorization, we propose the sequential doubly stochastic (SDS) factorization, in which J is decomposed as a product of sparse, doubly stochastic matrices. Finally, we discuss the application of the proposed sparse factorizations to the decentralized average consensus problem and decentralized optimization.

1 Introduction

We study sparse factorization of the real $n \times n$ matrix $J := \frac{1}{n}\mathbb{1}\mathbb{1}^\mathsf{T} \in \mathbb{R}^{n\times n}$; that is, we seek to find a (finite) sequence of matrices $\{W^{(k)}\}_{k=1}^{q} \subset \mathbb{R}^{n\times n}$ such that

$$W^{(q)}W^{(q-1)}\cdots W^{(1)} = \frac{1}{n}\begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}. \tag{1}$$

This problem finds applications in graph theory, systems and control, decentralized optimization, and other fields [6, 11, 13]. In this paper, we consider the general case where n is an arbitrary integer and propose several types of sparse factorization.

Previous work on the sparse factorization of J, or the all-ones matrix $\widetilde{J} = nJ = \mathbb{1}\mathbb{1}^\mathsf{T}$, can be roughly divided into two categories. The first class considers the case in which all the factors are identical, *i.e.*, $W^{(k)} = W$ for all $k \in [q]$. For example, the binary square root of $\widetilde{J}$ (when $n = p^2$ for some $p \in \mathbb{N}_{\geq 2}$) is studied in [3]. The De Bruijn matrix, first proposed in [5], serves as the q-th root of $\widetilde{J}$ when $n = p^q$, and has been extensively studied in the literature [6, 17]. For general n, the g-circulant binary solutions to $W^q = \widetilde{J}$ have also been investigated [7, 10, 18, 19].

[*] Author's final version. Accepted for publication in *SIAM Journal on Matrix Analysis and Applications*.

[†] School of Operations Research and Information Engineering, Cornell University. Email: xjiang@cornell.edu. Most of the work was completed when XJ was at Lehigh University.

[‡] Department of Electrical and Computer Engineering, Rice University. Email: {en18, cauribe}@rice.edu.

[§] Google Inc. Email: ybc@google.com.

The second class of solutions allows for differing factors of J. Among these solutions include one-peer exponential graphs (when $n = 2^q$) [20], one-peer hyper-cubes (when $n = 2^q$) [13], p-peer hyper-cuboids [11, 16], and deformable butterfly (DeBut) matrices [9]. Remarkably, the butterfly matrices [4], which were originally proposed for more general linear transforms and used in deep neural networks, reduce to one-peer hyper-cubes when we study the factorization (1) with $n = 2^q$. As extensions of the one-peer hyper-cubes (and butterfly matrices), the p-peer hyper-cuboids and the DeBut matrices serve as factorization of J for arbitrary n, and the sparsity of both factors depends on the prime factorization of n. In particular, when n is a large primal number, both p-peer hyper-cuboids and the DeBut matrices reduce to the fully dense matrix J. Allowing for different factors of J, in general, gives greater control over the sparsity of the factors compared to the case in which all the factors are identical [11].

In this paper, we consider the general case where $n \in \mathbb{N}_{\geq 2}$ is an arbitrary integer and study sparse factorization of J in the form

$$J = J_0 A J_0, \tag{2}$$

where $J_0 = J_1 \oplus \cdots \oplus J_\tau$ with $J_k := \frac{1}{n_k}\mathbb{1}\mathbb{1}^\intercal \in \mathbb{R}^{n_k \times n_k}$, $k \in [\tau]$. (Here, $\oplus$ denotes the direct sum of two matrices.) Throughout the paper, it is assumed that the partition $n = \sum_{k=1}^{\tau} n_k$ is given, with conditions that will be specified later (see (3)). Factorization (2) holds for arbitrary matrix order n and is inspired by the applications of J in decentralized averaging (and optimization). In decentralized averaging, for example, a group of agents each holds a piece of information and cooperates with other agents to compute a global quantity. The communication between agents is modeled by a graph (or a sequence of graphs) $\mathcal{G}^{(k)} = (\mathcal{V}, W^{(k)}, \mathcal{E}^{(k)})$. If the weight matrices $\{W^{(k)}\}$ satisfy (1), then the *exact* global average is computed in q communication rounds. In modern application scenarios, agents can be abstracted as high-performance computing (HPC) resources and naturally formed into clusters [1, 21]; see Section 6 for a more detailed discussion. Such clustering structure is captured by the proposed form of factorization (2). The block diagonal matrix J_0 models the intra-cluster communication, and each sub-block J_k, $k \in [\tau]$, can be further decomposed as (1) into, *e.g.*, p-peer hyper-cuboids. In contrast, the A-factor models the more expensive inter-cluster communication, and the main focus of this paper is to design *sparse* A-factors to reduce the communication overhead across clusters. Sparsity in A is desirable in decentralized averaging (and optimization) as the communication overhead is related to the total number of nonzeros $\mathrm{nnz}(A)$ as well as the largest node degree $d_{\max}(A) = \max_i\{\mathrm{nnz}(A_{i,:})\}$, where $A_{i,:}$ is the ith row of A.

Contributions In this paper, we study the form of factorization in (2) for arbitrary matrix order n and propose three types of A-factors. In the first two types, the sparse factor A has the so-called *hierarchically banded (HB)* structure, and additional properties of A distinguish these two types of HB factorization: (density) reduced HB and doubly stochastic HB. The third one is called the *sequential doubly stochastic (SDS) factorization* and admits an asymmetric, doubly stochastic factor A, which can be further decomposed as a product of several symmetric, doubly stochastic matrices. When applied to decentralized optimization, the proposed sparse factorizations provide more flexibility to balance communication costs and the total number of communication rounds in a decentralized optimization algorithm.

Notation Let $\mathbb{R}$ denote the set of real numbers (*i.e.*, scalars). Let $\mathbb{R}^n$ denote the set of n-dimensional (column) vectors. Let $\mathbb{R}^{m\times n}$ denote the set of m-by-n real matrices, and let $\mathbb{D}^n$ denote the set of $n \times n$ diagonal matrices. The set of natural numbers is denoted as $\mathbb{N} := \{0, 1, 2, \ldots\}$, and let $\mathbb{N}_{\geq r}$ denote the set of natural numbers greater than or equal to $r \in \mathbb{N}$. For any $n \in \mathbb{N}_{\geq 1}$, let $[n] := \{1, 2, \ldots, n\}$. Let $\mathbb{1}$ denote the all-ones (column) vector of compatible size. Let $\|\cdot\|$ denote the Euclidean norm of a vector. The direct sum of two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$ forms the block diagonal matrix $A \oplus B := \operatorname{blkdiag}(A, B) \in \mathbb{R}^{(m+p)\times(n+q)}$.

Outline In Section 2, we propose the notion of hierarchically banded (HB) matrices. Sections 3 and 4 study two types of HB factorization. The sequential doubly stochastic (SDS) factorization is discussed in Section 5. In Section 6, we present the potential usefulness of these sparse factorizations in decentralized averaging and optimization, and concluding remarks are offered in Section 7.

2 Hierarchically banded matrices

Factorization of the form (2) relies on a partition of $n \in \mathbb{N}_{\geq 2}$:

$$n = \sum_{k=1}^{\tau} n_k, \text{ where } \{n_k\}_{k=1}^{\tau} \subset \mathbb{N}_{\geq 1}, \text{ and } n_k \geq \sum_{j=k+1}^{\tau} n_j =: m_k \text{ for all } k \in [\tau-1]. \tag{3}$$

Such a partition can be constructed systematically, *e.g.*, via the base-p representation of n (with $p \in \mathbb{N}_{\geq 2}$). Overloading the binary representation, we denote the base-p representation of n as $(i_{\tau-1}i_{\tau-2}\cdots i_1 i_0)_p$, where $\tau = \lfloor \log_p(n) \rfloor + 1$. Then, any integer $n \in \mathbb{N}_{\geq 2}$ can be written as $n = \sum_{k=1}^{\tau} n_k$, where $n_k = i_{\tau-k}p^{\tau-k}$. (For those i_k's being zero, they are removed before the construction of the partition.) By construction, the condition $n_k \geq m_k$, for all $k \in [\tau-1]$, directly follows from the property of the base-p representation. A simple example is $(n, p) = (15, 2)$, and $(n_1, n_2, n_3, n_4) = (8, 4, 2, 1)$, which follows from the binary representation $15 = (1111)_2$.

Given such a decomposition (3), we study the factorization in the form of (2), where the matrix $A \in \mathbb{R}^{n\times n}$ has the so-called *hierarchically banded* structure.

Definition 1 (Hierarchically banded matrices)**.** *Given $n \in \mathbb{N}_{\geq 2}$ and a partition (3), a real symmetric $n \times n$ matrix A is called* hierarchically banded (HB) *if there exists a sequence of symmetric matrices $A^{(k)} \in \mathbb{R}^{m_k\times m_k}$, $k \in [\tau]$, such that the following three conditions hold.*

- *$A^{(1)} = A$.*
- *$A^{(\tau)} \in \mathbb{D}^{n_\tau}$ is diagonal.*
- *For all $k \in [\tau-1]$, the matrix $A^{(k)}$ can be partitioned as*

$$A^{(k)} = \begin{bmatrix} A_{11}^{(k)} & A_{12}^{(k)} \\ \big(A_{12}^{(k)}\big)^{\mathsf{T}} & A_{22}^{(k)} \end{bmatrix}, \tag{4}$$

where $A_{11}^{(k)} \in \mathbb{D}^{n_k}$, $A_{12}^{(k)} \in \mathbb{R}^{n_k\times m_k}$ have nonzero entries only on the diagonals, and the last submatrix satisfies $A_{22}^{(k)} = A^{(k+1)}$.

Such a sequence $\{A^{(k)}\}_{k=1}^{\tau}$ is called the hierarchically banded (HB) sequence *of A, and the set of $n \times n$ hierarchically banded matrices is denoted by $\mathbb{HB}^n$.*

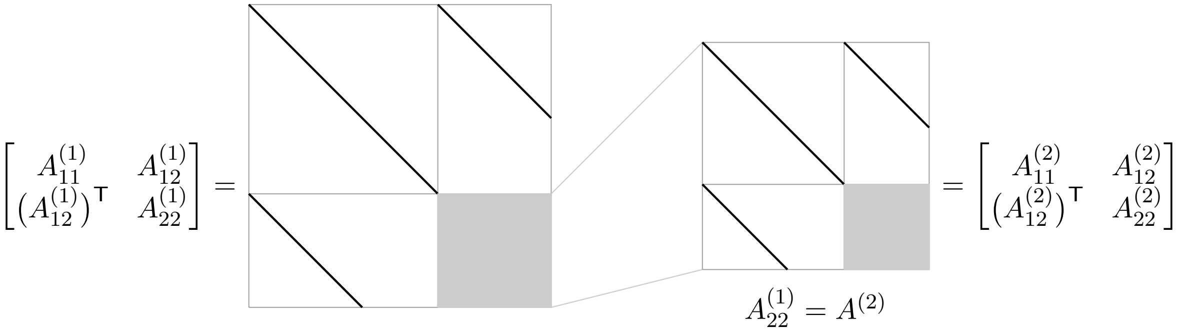
$\begin{bmatrix} A_{11}^{(1)} & A_{12}^{(1)} \\ (A_{12}^{(1)})^\mathsf{T} & A_{22}^{(1)} \end{bmatrix} =$
$= \begin{bmatrix} A_{11}^{(2)} & A_{12}^{(2)} \\ (A_{12}^{(2)})^\mathsf{T} & A_{22}^{(2)} \end{bmatrix}$
$A_{22}^{(1)} = A^{(2)}$

Figure 1: Illustration of a hierarchically banded matrix.

The hierarchically banded structure is illustrated in Figure 1. The word "hierarchically" means that the matrix can be hierarchically partitioned, and this term is inspired by the notion of hierarchical matrices (or $\mathcal{H}$-matrices) (see, *e.g.*, [2]). In addition, recall that a symmetric $n \times n$ banded matrix A satisfies

$$A_{ij} = 0 \quad \text{if } j < i - r \text{ or } j > i + r,$$

where $r \in [n]$ is called the *bandwidth* of A.

Factorization (2) with a hierarchically banded factor A is called the *hierarchically banded (HB) factorization* of J, and the matrix $A \in \mathbb{HB}^n$ is called the *hierarchically banded (HB) factor* of J. It turns out that the hierarchically banded factor A is not unique. In this paper, we study the following two types of hierarchically banded factorization, characterized by additional properties of A or the HB sequence $A^{(k)}$ defined in (4).

- *Reduced hierarchically banded (RHB) factorization.* To further promote the sparsity of the A-factor, we impose an additional condition that only a few elements in the two bands of each $A^{(k)}$ are nonzero:

  $$A_{12}^{(k)}[j,j] \neq 0, \quad \text{if } j = 1, 1 + n_{k+1}, 1 + n_{k+1} + n_{k+2}, \ldots, 1 + \sum_{\ell=1}^{\tau-k-1} n_{k+\ell},$$

  for all $k \in [\tau - 1]$. In $A^{(1)}$, for example, this condition means that the largest cluster (the one of size n_1) communicates with exactly *one* agent from each of the other clusters. Such a condition would further reduce the communication overhead, and the HB factor A designed for this purpose is called the (density) reduced HB factor, which is studied in Section 3.

- *Doubly stochastic hierarchically banded (DSHB) factorization.* In this case, the factor A is both hierarchically banded and *doubly stochastic*, *i.e.*, all the entries in A are nonnegative, $A\mathbb{1} = \mathbb{1}$, and $A^\mathsf{T}\mathbb{1} = \mathbb{1}$. This additional property of A would be useful in decentralized optimization. The details are discussed in Section 4.

Moreover, the DSHB factorization inspires another sparse factorization (2) of J, which is called the *sequential doubly stochastic (SDS) factorization.* In this factorization, the SDS factor A is doubly stochastic and can be written as the product of a sequence of symmetric, doubly stochastic matrices. Although the SDS factor is not hierarchically banded (nor symmetric), it is closely related to the DSHB factorization and finds its application in decentralized optimization. The details of the SDS factorization are presented in Section 5.

3 Reduced hierarchically banded factorization

As discussed in Section 2, the (density) reduced hierarchically banded (RHB) factorization further promotes sparsity in the HB factor A by requiring

$$A_{12}^{(k)}[j,j] \neq 0, \quad \text{if } j = 1,\ 1 + n_{k+1},\ 1 + n_{k+1} + n_{k+2},\ \ldots,\ 1 + \sum_{\ell=1}^{\tau-k-1} n_{k+\ell}, \tag{5}$$

i.e., only a few nonzeros exist in the diagonal entries of $A_{12}^{(k)}$. In addition, it is also assumed that only one diagonal entry in each $A_{11}^{(k)}$ is not one:

$$A_{11}^{(k)} = \operatorname{diag}(\alpha_k, 1, \ldots, 1), \tag{6}$$

for some $\alpha_k \in \mathbb{R}$. (Other requirements on the diagonal submatrices $\{A_{11}^{(k)}\}$ can be applied, and they do not affect the idea of density reduction in the RHB factorization. So (6) is chosen for simplicity.) The RHB factorization is illustrated in Section 3.1 via the simple example where $\tau = 2$, and Section 3.2 presents an algorithm for the RHB factorization in the general case.

3.1 A two-block example

To illustrate the idea of the RHB factorization, we consider the simple case: $\tau = 2$. In this case, suppose that $n = n_1 + n_2$ with $(n_1, n_2) \in \mathbb{N}_{\geq n_2} \times \mathbb{N}_{\geq 1}$. Then, the HB factorization (2) reduces to

$$J = \begin{bmatrix} J_1 & \\ & J_2 \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} \\ A_{12}^\mathsf{T} & A_{22} \end{bmatrix} \begin{bmatrix} J_1 & \\ & J_2 \end{bmatrix} = \begin{bmatrix} J_1 A_{11} J_1 & J_1 A_{12} J_2 \\ (J_1 A_{12} J_2)^\mathsf{T} & J_2 A_{22} J_2 \end{bmatrix}, \tag{7}$$

where $J_1 = \frac{1}{n_1}\mathbb{1}\mathbb{1}^\mathsf{T} \in \mathbb{R}^{n_1 \times n_1}$, $J_2 = \frac{1}{n_2}\mathbb{1}\mathbb{1}^\mathsf{T} \in \mathbb{R}^{n_2 \times n_2}$, $A_{11} \in \mathbb{D}^{n_1}$, $A_{12} \in \mathbb{R}^{n_1 \times n_2}$, and $A_{22} \in \mathbb{D}^{n_2}$. Expanding (7) yields

$$J_1 A_{11} J_1 = \frac{1}{n}\mathbb{1}_{n_1}\mathbb{1}_{n_1}^\mathsf{T}, \qquad J_1 A_{12} J_2 = \frac{1}{n}\mathbb{1}_{n_1}\mathbb{1}_{n_2}^\mathsf{T}, \qquad J_2 A_{22} J_2 = \frac{1}{n}\mathbb{1}_{n_2}\mathbb{1}_{n_2}^\mathsf{T}. \tag{8}$$

Recall that the condition (6) requires A_{11} to take the form $A_{11} = \operatorname{diag}(\alpha_1, 1, \ldots, 1)$ for some $\alpha_1 \in \mathbb{R}$. Substituting it into $J_1 A_{11} J_1$ gives

$$J_1 A_{11} J_1 = \frac{1}{n_1^2}\mathbb{1}_{n_1}\left(\mathbb{1}_{n_1}^\mathsf{T} \operatorname{diag}(\alpha_1, 1, \ldots, 1)\mathbb{1}_{n_1}\right)\mathbb{1}_{n_1}^\mathsf{T} = \frac{\alpha_1 + n_1 - 1}{n_1^2}\mathbb{1}_{n_1}\mathbb{1}_{n_1}^\mathsf{T}.$$

Then, combining it with the first condition in (8) yields

$$\alpha_1 = \frac{n_1^2}{n} - n_1 + 1.$$

Similarly, one obtains that $A_{22} = \operatorname{diag}(\alpha_2, 1, \ldots, 1)$ with $\alpha_2 = \frac{n_2^2}{n} - n_2 + 1$. Finally, the condition (5) implies that A_{12} is nonzero only at the first element: $A_{12}[1,1] := \beta$, and then expanding the second block $J_1 A_{12} J_2$ with the condition (6) gives

$$J_1 A_{12} J_2 = \frac{1}{n_1 n_2}\mathbb{1}_{n_1}\left(\mathbb{1}_{n_1}^\mathsf{T} A_{12}\mathbb{1}_{n_2}\right)\mathbb{1}_{n_2}^\mathsf{T} = \frac{\beta}{n_1 n_2}\mathbb{1}_{n_1}\mathbb{1}_{n_2}^\mathsf{T},$$

Combining it with the second condition in (8) gives $A_{12}[1,1] = \beta = \frac{n_1 n_2}{n}$.

In conclusion, when $n = n_1 + n_2$, the RHB factor A of J is given by

$$A = \left[\begin{array}{ccccc|cccc}
\alpha_1 & & & & & \beta & & & \\
& 1 & & & & & 0 & & \\
& & 1 & & & & & \ddots & \\
& & & \ddots & & & & & 0 \\
& & & & \ddots & & & & \\
& & & & 1 & & & & \\ \hline
\beta & & & & & \alpha_2 & & & \\
& 0 & & & & & 1 & & \\
& & \ddots & & & & & \ddots & \\
& & & 0 & & & & & 1
\end{array}\right], \tag{9}$$

where $\alpha_1 = \frac{n_1^2}{n} - n_1 + 1$, $\alpha_2 = \frac{n_2^2}{n} - n_2 + 1$, and $\beta = \frac{n_1 n_2}{n}$.

3.2 The RHB factorization algorithm

In this section, we extend the key idea in Section 3.1 to handle the general case $n = \sum_{k=1}^{\tau} n_k$, where we denote $m_k := \sum_{i=k+1}^{\tau} n_i$ for $k \in [\tau - 1]$ and assume that $n_k \geq m_k$ for all $k \in [\tau - 1]$. Then, the construction of the RHB factorization of J is summarized in Algorithm 1, which outputs the RHB factor A that satisfies (2), (5), and (6).

To verify the correctness of Algorithm 1, we start with the case $k = 1$ and write out the equality $J = J_0 A J_0$ for the partitioned matrices:

$$\frac{1}{n}\mathbb{1}_n\mathbb{1}_n^\top = \begin{bmatrix} J_1 & \\ & \overline{J}_1 \end{bmatrix} \begin{bmatrix} A_{11}^{(1)} & A_{12}^{(1)} \\ \left(A_{12}^{(1)}\right)^\top & A_{22}^{(1)} \end{bmatrix} \begin{bmatrix} J_1 & \\ & \overline{J}_1 \end{bmatrix}, \tag{12}$$

where $\overline{J}_1 := J_2 \oplus J_3 \oplus \cdots \oplus J_\tau \in \mathbb{R}^{m_1 \times m_1}$. Expanding the above equation gives three conditions similar to (8):

$$J_1 A_{11}^{(1)} J_1 = \frac{1}{n}\mathbb{1}_{n_1}\mathbb{1}_{n_1}^\top, \qquad J_1 A_{12}^{(1)} \overline{J}_1 = \frac{1}{n}\mathbb{1}_{n_1}\mathbb{1}_{m_1}^\top, \qquad \overline{J}_1 A_{22}^{(1)} \overline{J}_1 = \frac{1}{n}\mathbb{1}_{m_1}\mathbb{1}_{m_1}^\top. \tag{13}$$

It then follows from the condition (6) that

$$J_1 A^{(1)} J_1 = \frac{1}{n_1^2}\mathbb{1}_{n_1}\left(\mathbb{1}_{n_1}^\top \operatorname{diag}\left(\alpha_1, 1, \ldots, 1\right)\mathbb{1}_{n_1}\right)\mathbb{1}_{n_1}^\top = \frac{\alpha_1 + n_1 - 1}{n_1^2}\mathbb{1}_{n_1}\mathbb{1}_{n_1}^\top.$$

Combining it with the first condition in (13) yields $\alpha_1 = \frac{n_1^2}{n} - n_1 + 1$.

We now consider the $(1,2)$-block $J_1 A_{12}^{(1)} \overline{J}_1$. Recall from the condition (5) that the matrix

Algorithm 1 Reduced hierarchically banded (RHB) factorization algorithm

```
Input: $n \in \mathbb{N}_{\geq 2}$, and the factors $\{n_k\}_{k=1}^{\tau}$ satisfying $n = \sum_{k=1}^{\tau} n_k$ and $n_k \geq m_k = \sum_{i=k+1}^{\tau} n_i$
  for all $k \in [\tau - 1]$. Denote $m_0 = n$.
Output: The RHB factor $A$ of $J$, and the associated HB sequence $\{A^{(k)}\}_{k=1}^{\tau}$.
for $k = 1, 2, \ldots, \tau - 2$ do
    Compute the $(1,1)$-block $A_{11}^{(k)} \in \mathbb{D}^{n_k}$ of $A^{(k)}$:
        $A_{11}^{(k)} \leftarrow \operatorname{diag}\left(\frac{n_k^2}{n} - n_k + 1, 1, \ldots, 1\right).$
    Compute the $(1,2)$-block $A_{12}^{(k)} \in \mathbb{R}^{n_k \times m_k}$:
        $A_{12}^{(k)}[i,j] \leftarrow \begin{cases} \frac{n_k n_{k+1}}{n} & \text{if } i = j = 1 \\ \frac{n_k n_{k+\ell}}{n} & \text{if } i = j = 1 + \sum\limits_{r=1}^{\ell} n_{k+r} \text{ for } \ell = 1, 2, \ldots, \tau - k - 1 \\ 0 & \text{otherwise.} \end{cases}$
    Compute the $(2,2)$-block $A_{22}^{(k)} = A^{(k+1)}$ as the RHB factorization:
        $\frac{1}{m_k} \mathbb{1}_{m_k} \mathbb{1}_{m_k}^{\mathsf{T}} = \overline{J}_k A^{(k+1)} \overline{J}_k, \qquad (10)$
    where $\overline{J}_k := J_{k+1} \oplus \cdots \oplus J_{\tau}$, and the RHB factor $A^{(k+1)} = A_{22}^{(k)}$ is partitioned as
        $A^{(k+1)} = \begin{bmatrix} A_{11}^{(k+1)} & A_{12}^{(k+1)} \\ \left(A_{12}^{(k+1)}\right)^{\mathsf{T}} & A_{22}^{(k+1)} \end{bmatrix}. \qquad (11)$
end for
Set the RHB factor $A$: $A \leftarrow A^{(1)}$.
```

$A_{12}^{(1)} \in \mathbb{R}^{n_1 \times m_1}$ can be partitioned as

$$A_{12}^{(1)} = \begin{bmatrix} B_2^{(1)} & & & & \\ & B_3^{(1)} & & & \\ & & \ddots & & \\ & & & B_{\tau-1}^{(1)} & \\ & & & & B_\tau^{(1)} \\ \mathbf{0}_2 & \mathbf{0}_3 & \cdots & \mathbf{0}_{\tau-1} & \mathbf{0}_\tau \end{bmatrix},$$

where $B_j^{(1)} = \operatorname{diag}\left(\beta_j^{(1)}, 0, \ldots, 0\right) \in \mathbb{D}^{n_j}$, and $\mathbf{0}_j$ is the all-zeros matrix of size $(n_1 - m_1) \times n_j$, for $j = 2, \ldots, \tau$. In addition, we denote the diagonal entries of $B_j^{(1)}$ by the n_j-vector $b_j^{(1)} = (\beta_j^{(1)}, 0, \ldots, 0)$. Then, it holds that

$$\mathbb{1}_{n_1}^\mathsf{T} A_{12}^{(1)} = \begin{bmatrix} (b_2^{(1)})^\mathsf{T} & (b_3^{(1)})^\mathsf{T} & \cdots & (b_\tau^{(1)})^\mathsf{T} \end{bmatrix} \in \mathbb{R}^{1 \times m_1}.$$

Then, it holds that

$$\begin{aligned} J_1 A_{12}^{(1)} \overline{J} &= \frac{1}{n_1} \mathbb{1}_{n_1} (\mathbb{1}_{n_1}^\mathsf{T} A_{12}^{(1)}) \overline{J}_1 \\ &= \frac{1}{n_1} \mathbb{1}_{n_1} \begin{bmatrix} \left(b_2^{(1)}\right)^\mathsf{T} & \left(b_3^{(1)}\right)^\mathsf{T} & \cdots & \left(b_\tau^{(1)}\right)^\mathsf{T} \end{bmatrix} \left(J_2 \oplus J_3 \oplus \cdots \oplus J_\tau\right) \\ &= \frac{1}{n_1} \mathbb{1}_{n_1} \begin{bmatrix} \frac{1}{n_2}\left(b_2^{(1)}\right)^\mathsf{T} \mathbb{1}_{n_2} \mathbb{1}_{n_2}^\mathsf{T} & \frac{1}{n_3}\left(b_3^{(1)}\right)^\mathsf{T} \mathbb{1}_{n_3} \mathbb{1}_{n_3}^\mathsf{T} & \cdots & \frac{1}{n_\tau}\left(b_\tau^{(1)}\right)^\mathsf{T} \mathbb{1}_{n_\tau} \mathbb{1}_{n_\tau}^\mathsf{T} \end{bmatrix} \\ &= \frac{1}{n_1} \mathbb{1}_{n_1} \begin{bmatrix} \frac{\beta_2^{(1)}}{n_2} \mathbb{1}_{n_2}^\mathsf{T} & \frac{\beta_3^{(1)}}{n_2} \mathbb{1}_{n_3}^\mathsf{T} & \cdots & \frac{\beta_\tau^{(1)}}{n_\tau} \mathbb{1}_{n_\tau}^\mathsf{T} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\beta_2^{(1)}}{n_1 n_2} \mathbb{1}_{n_1} \mathbb{1}_{n_2}^\mathsf{T} & \frac{\beta_3^{(1)}}{n_1 n_3} \mathbb{1}_{n_1} \mathbb{1}_{n_3}^\mathsf{T} & \cdots & \frac{\beta_\tau^{(1)}}{n_1 n_\tau} \mathbb{1}_{n_1} \mathbb{1}_{n_\tau}^\mathsf{T} \end{bmatrix} \in \mathbb{R}^{n_1 \times m_1}. \end{aligned}$$

Hence, to satisfy the second condition in (13), we must have for all $j = 2, \ldots, \tau$ that

$$\frac{\beta_j^{(1)}}{n_1 n_j} = \frac{1}{n} \qquad \Longleftrightarrow \qquad \beta_j^{(1)} = \frac{n_1 n_j}{n}.$$

Finally, we consider the last condition in (13). Denote $A^{(2)} := A_{22}^{(1)}$ and consider the partition (11). Also notice that $\overline{J}_1 = J_2 \oplus (J_3 \oplus \cdots \oplus J_\tau) := J_2 \oplus \overline{J}_2$. Then, we write out the last condition (13) in the partitioned form:

$$\frac{1}{n} \mathbb{1}_{m_1} \mathbb{1}_{m_1}^\mathsf{T} = \begin{bmatrix} J_2 & \\ & \overline{J}_2 \end{bmatrix} \begin{bmatrix} A_{11}^{(2)} & A_{12}^{(2)} \\ \left(A_{12}^{(2)}\right)^\mathsf{T} & A_{22}^{(2)} \end{bmatrix} \begin{bmatrix} J_2 & \\ & \overline{J}_2 \end{bmatrix},$$

which takes the same form as (12). We can then repeat the above process for $k = 1, 2, \ldots, \tau - 2$. When Algorithm 1 reaches iteration $k = \tau - 2$, Line 6 computes the RHB factorization of the matrix

$$A^{(\tau-1)} = \begin{bmatrix} A_{11}^{(\tau-1)} & A_{12}^{(\tau-1)} \\ \left(A_{12}^{(\tau-1)}\right)^\mathsf{T} & A_{22}^{(\tau-1)} \end{bmatrix},$$

which is the two-block case studied in Section 3.1. Thus, the RHB factor of $A^{(\tau-1)}$ is in the form of (9) with

$$\alpha_1 = \frac{n_{\tau-1}^2}{n} - n_{\tau-1} + 1, \qquad \alpha_2 = \frac{n_\tau^2}{n} - n_\tau + 1, \qquad \beta = \frac{n_{\tau-1}n_\tau}{n}.$$

From the above discussion, we obtain the following result.

Theorem 1. *The $n \times n$ matrix A_{RHB} generated by Algorithm 1 is hierarchically banded and satisfies $J = J_0 A_{\mathrm{RHB}} J_0$ as well as conditions (5)–(6). In addition, the total number of nonzeros is $\mathrm{nnz}(A_{\mathrm{RHB}}) = n + \tau(\tau-1)$, and the largest node degree is $d_{\max}(A_{\mathrm{RHB}}) = \tau$.*

Proof. (The subscript in A_{RHB} is omitted in the proof for readability.) The hierarchically banded structure of A follows from the recursive nature of Algorithm 1, and in particular, the recursive partition of $A^{(k)}$ in Line 6 of Algorithm 1. Similarly, A satisfies the conditions (5)–(6) due to the assignment of values in $A_{11}^{(k)}$ and $A_{12}^{(k)}$ in Lines 4 and 5. Next, the factorization $J = J_0 A J_0 = J_0 A^{(1)} J_0$ holds by recursively applying (10) for $k = \tau-1, \tau-2, \dots, 1$. Finally, one has for all $k \in [\tau-1]$ that $\mathrm{nnz}(A_{11}^{(k)}) = n_k$, $\mathrm{nnz}(A_{12}^{(k)}) = \tau - k$, and $\mathrm{nnz}(A^{(\tau)}) = n_\tau$. So, the total number of nonzeros is

$$\mathrm{nnz}(A) = \sum_{k=1}^{\tau-1} \big(n_k + 2(\tau-k)\big) + n_\tau = n + \tau(\tau-1).$$

The row with the most nonzeros is row $n - n_\tau - n_{\tau-1} + 1$, where $A_{n-n_\tau-n_{\tau-1}+1,j} \neq 0$ if $j = 1, 1+n_1, 1+n_1+n_2, \dots, 1+\sum_{k=1}^{\tau-1} n_k$, and thus $d_{\max}(A) = \tau$. □

4 Doubly stochastic hierarchically banded factorization

Another useful type of hierarchically banded factorization, especially in decentralized optimization (see Section 6.2 for details), requires the HB factor A to be *doubly stochastic*, *i.e.*, all the entries are nonnegative and $A\mathbb{1} = \mathbb{1}$ (and $A^\intercal \mathbb{1} = \mathbb{1}$, which is guaranteed by the symmetry of A). Yet in this case, the HB sequence $\{A^{(k)}\}$ is *not* doubly stochastic. Instead, we show that each matrix in the *scaled* HB sequence $\{\widetilde{A}^{(k)}\}_{k=1}^{\tau}$ remains doubly stochastic, where

$$\widetilde{A}^{(k)} := \frac{n}{m_{k-1}} A^{(k)} \in \mathbb{HB}^{m_k}, \quad k \in [\tau]. \tag{14}$$

Again, the doubly stochastic hierarchically banded (DSHB) factorization is illustrated in Section 4.1 via the simple example where $\tau = 2$, and Section 4.2 presents an algorithm for DSHB factorization in the general case.

4.1 A two-block example

Similar to Section 3.1, we start with the simple case where $\tau = 2$, and assume $n = n_1 + n_2$ with $(n_1, n_2) \in \mathbb{N}_{\geq n_2} \times \mathbb{N}_{\geq 1}$. Then, the HB factorization takes the form of (7), which can be partitioned as in (8). Recall that in Section 3.1 we require both submatrices A_{11} and A_{12} to have only one nonzero entry. In the context of decentralized optimization, a larger cluster will communicate with exactly one agent from each of the smaller clusters. This section considers a different setting where all agents in subgroup 2 (recall $n_2 \leq n_1$) can communicate across subgroups. In particular, the submatrix $A_{12} \in \mathbb{R}^{n_1 \times n_2}$ takes the following form:

$$A_{12} = \begin{bmatrix} \operatorname{diag}(\beta \mathbb{1}_{n_2}) \\ 0 \end{bmatrix}.$$

Substituting into $J_1 A_{12} J_2$ gives

$$J_1 A_{12} J_2 = \frac{1}{n_1 n_2} \mathbb{1}_{n_1} \left(\mathbb{1}_{n_1}^\top A_{12} \mathbb{1}_{n_2} \right) \mathbb{1}_{n_2}^\top = \frac{\beta}{n_1} \mathbb{1}_{n_1} \mathbb{1}_{n_2}^\top.$$

Then, the second condition in (8) implies that

$$\beta = \frac{n_1}{n} = \frac{n_1}{n_1 + n_2}.$$

With the sub-block A_{12} settled, the doubly stochastic property of A implies that the sub-blocks $A_{11} \in \mathbb{D}^{n_1}$ and $A_{22} \in \mathbb{D}^{n_2}$ are diagonal matrices satisfying

$$A_{11} = \operatorname{diag}(\underbrace{1-\beta, \ldots, 1-\beta}_{n_2}, \underbrace{1, \ldots, 1}_{n_1 - n_2}), \qquad A_{22} = \operatorname{diag}\left((1-\beta) \mathbb{1}_{n_2} \right).$$

Finally, we confirm that this choice of A_{11} and A_{22} also satisfies the first and third conditions in (8):

$$\begin{aligned} \tfrac{1}{n_1^2} \mathbb{1}_{n_1} \mathbb{1}_{n_1}^\top A_{11} \mathbb{1}_{n_1} \mathbb{1}_{n_1}^\top &= \tfrac{\mathbb{1}_{n_1}^\top A_{11} \mathbb{1}_{n_1}}{n_1^2} \mathbb{1}_{n_1} \mathbb{1}_{n_1}^\top = \tfrac{(1-\beta) n_2 + (n_1 - n_2)}{n_1^2} \mathbb{1}_{n_1} \mathbb{1}_{n_1}^\top = \tfrac{1}{n} \mathbb{1}_{n_1} \mathbb{1}_{n_1}^\top, \\ \tfrac{1}{n_2^2} \mathbb{1}_{n_2} \mathbb{1}_{n_2}^\top A_{22} \mathbb{1}_{n_2} \mathbb{1}_{n_2}^\top &= \tfrac{\mathbb{1}_{n_2}^\top A_{22} \mathbb{1}_{n_2}}{n_2^2} \mathbb{1}_{n_2} \mathbb{1}_{n_2}^\top = \tfrac{(1-\beta) n_2}{n_2^2} \mathbb{1}_{n_2} \mathbb{1}_{n_2}^\top = \tfrac{1}{n} \mathbb{1}_{n_2} \mathbb{1}_{n_2}^\top. \end{aligned}$$

In conclusion, when $n = n_1 + n_2$, the doubly stochastic HB factor A of J is

$$A = \left[\begin{array}{cc|c} \frac{n_2}{n} I_{n_2} & 0 & \frac{n_1}{n} I_{n_2} \\ 0 & I_{n_1 - n_2} & 0 \\ \hline \frac{n_1}{n} I_{n_2} & 0 & \frac{n_2}{n} I_{n_2} \end{array} \right]. \tag{15}$$

4.2 The DSHB factorization algorithm

We extend the key idea in Section 3.1 to handle the general case where $n = \sum_{k=1}^{\tau} n_k$. The construction of the DSHB factor A, as well as the associated (scaled) HB sequence $\{A^{(k)}\}$ ($\{\widetilde{A}^{(k)}\}$ in (14)), is summarized in Algorithm 2.

To verify the correctness of Algorithm 2, we start with the iteration $k = 1$ and write out the equality $J = J_0 A J_0$ for the partitioned matrices:

$$\frac{1}{n} \mathbb{1}_n \mathbb{1}_n^\top = \begin{bmatrix} J_1 & \\ & \overline{J}_1 \end{bmatrix} \begin{bmatrix} A_{11}^{(1)} & A_{12}^{(1)} \\ (A_{12}^{(1)})^\top & A_{22}^{(1)} \end{bmatrix} \begin{bmatrix} J_1 & \\ & \overline{J}_1 \end{bmatrix},$$

where recall $\overline{J}_1 := J_2 \oplus J_3 \oplus \cdots \oplus J_\tau \in \mathbb{R}^{m_1 \times m_1}$ and $A^{(1)} = \widetilde{A}^{(1)}$. Expanding the above equation gives three conditions similar to (8):

$$J_1 A_{11}^{(1)} J_1 = \frac{1}{n} \mathbb{1}_{n_1} \mathbb{1}_{n_1}^\top, \qquad J_1 A_{12}^{(1)} \overline{J}_1 = \frac{1}{n} \mathbb{1}_{n_1} \mathbb{1}_{m_1}^\top, \qquad \overline{J}_1 A_{22}^{(1)} \overline{J}_1 = \frac{1}{n} \mathbb{1}_{m_1} \mathbb{1}_{m_1}^\top. \tag{19}$$

Algorithm 2 Doubly stochastic hierarchically banded (DSHB) factorization algorithm

```
Input: $n \in \mathbb{N}_{\geq 2}$, and the factors $\{n_k\}_{k=1}^{\tau}$ satisfying $n = \sum_{k=1}^{\tau} n_k$ and $n_k \geq m_k = \sum_{i=k+1}^{\tau} n_i$ for all $k \in [\tau - 1]$.
Output: The doubly stochastic HB factor $A$ of $J$, and the associated HB sequence $\{A^{(k)}\}_{k=1}^{\tau}$.
Set $m_{-1} \leftarrow n$ and $m_0 \leftarrow n$.
for $k = 1, 2, \ldots, \tau - 2$ do
    Compute the $(1,1)$-block $\widetilde{A}_{11}^{(k)} \in \mathbb{D}^{n_k}$ of $\widetilde{A}^{(k)}$:
        $$\widetilde{A}_{11}^{(k)} \leftarrow \operatorname{diag}\big(\underbrace{\tfrac{m_k}{m_{k-1}}, \ldots, \tfrac{m_k}{m_{k-1}}}_{m_k}, \underbrace{1, \ldots, 1}_{n_k - m_k}\big). \quad (16)$$
    Compute the $(1,2)$-block $\widetilde{A}_{12}^{(k)} \in \mathbb{R}^{n_k \times m_k}$:
        $$\widetilde{A}_{12}^{(k)}[i,j] \leftarrow \begin{cases} \frac{n_k}{m_{k-1}} & \text{if } i = j = 1, 2, \ldots, m_k \\ 0 & \text{otherwise.} \end{cases} \quad (17)$$
    Compute the $(2,2)$-block $\widetilde{A}_{22}^{(k)}$ from the DSHB factorization:
        $$\frac{1}{m_k} \mathbb{1}_{m_k} \mathbb{1}_{m_k}^{\mathsf{T}} = \overline{J}_k \widetilde{A}^{(k+1)} \overline{J}_k, \quad (18)$$
    where $\overline{J}_k := J_{k+1} \oplus \cdots \oplus J_\tau$, and the DSHB factor $\widetilde{A}^{(k+1)} \leftarrow \frac{m_{k-1}}{m_k} \widetilde{A}_{22}^{(k)}$ is partitioned as
        $$\widetilde{A}^{(k+1)} = \begin{bmatrix} \widetilde{A}_{11}^{(k+1)} & \widetilde{A}_{12}^{(k+1)} \\ \big(\widetilde{A}_{12}^{(k+1)}\big)^{\mathsf{T}} & \widetilde{A}_{22}^{(k+1)} \end{bmatrix}.$$
end for
Set the DSHB factor $A \leftarrow A^{(1)} \equiv \widetilde{A}^{(1)}$ and the associated HB sequence $A^{(k)} \leftarrow \frac{m_{k-1}}{n} \widetilde{A}^{(k)}$, for all $k \in [\tau]$.
```

For the $(1,2)$-block $A_{12}^{(1)}$, we follow the convention in Section 4.1 and assume that it has the structure

$$A_{12}^{(1)} = \begin{bmatrix} \operatorname{diag}\big(\beta^{(1)} \mathbb{1}_{m_1}\big) \\ 0 \end{bmatrix}.$$

Substituting into $J_1 A_{12}^{(1)} \overline{J}_1$ gives

$$J_1 A_{12}^{(1)} \overline{J}_1 = \frac{1}{n_1} \mathbb{1}_{n_1} \big(\mathbb{1}_{n_1}^{\mathsf{T}} A_{12}^{(1)}\big) \overline{J}_1 = \frac{\beta^{(1)}}{n_1} \mathbb{1}_{n_1} \mathbb{1}_{m_1}^{\mathsf{T}} \overline{J}_1 = \frac{\beta^{(1)}}{n_1} \mathbb{1}_{n_1} \mathbb{1}_{m_1}^{\mathsf{T}}.$$

Combining it with the second condition in (19) yields $\beta^{(1)} = \frac{n_1}{n}$. Then, the doubly stochastic property of A implies that

$$A_{11}^{(1)} = \operatorname{diag}\big(\underbrace{\tfrac{m_1}{n}, \ldots, \tfrac{m_1}{n}}_{m_1}, \underbrace{1, \ldots, 1}_{n_1 - m_1}\big), \qquad A_{22}^{(1)} \mathbb{1}_{m_1} = (1 - \beta^{(1)}) \mathbb{1}_{m_1} = \tfrac{m_1}{n} \mathbb{1}_{m_1}.$$

The second equation above is equivalent to the doubly stochastic property of the *scaled* matrix

$$\widetilde{A}^{(2)}\mathbb{1}_{m_1} = \mathbb{1}_{m_1}, \quad \text{where } \widetilde{A}^{(2)} := \frac{n}{m_1}A_{22}^{(1)} \in \mathbb{R}^{m_1\times m_1}.$$

Similarly, the third condition in (19) can be written in terms of $\widetilde{A}^{(2)}$ as

$$\overline{J}_1\widetilde{A}^{(2)}\overline{J}_1 = \frac{1}{m_1}\mathbb{1}_{m_1}\mathbb{1}_{m_1}^\mathsf{T}. \tag{20}$$

Therefore, to find a doubly stochastic, hierarchically banded matrix $\widetilde{A}^{(2)}$ that satisfies (20), we need to construct the DSHB factorization of $\frac{1}{m_1}\mathbb{1}_{m_1}\mathbb{1}_{m_1}^\mathsf{T}$, which requires recursive execution of the above process for $k = 1, 2, \ldots, \tau - 2$.

When Algorithm 2 reaches iteration $k = \tau - 2$, Line 7 computes the DSHB factor

$$\widetilde{A}^{(\tau-1)} = \begin{bmatrix} \widetilde{A}_{11}^{(\tau-1)} & \widetilde{A}_{12}^{(\tau-1)} \\ \big(\widetilde{A}_{12}^{(\tau-1)}\big)^\mathsf{T} & \widetilde{A}_{22}^{(\tau-1)} \end{bmatrix},$$

which is the two-block case studied in Section 4.1. Thus, the DSHB factor of $\frac{1}{m_{\tau-2}}\mathbb{1}_{m_{\tau-2}}\mathbb{1}_{m_{\tau-2}}^\mathsf{T}$ is in the form of (15):

$$\widetilde{A}^{(\tau-1)} = \left[\begin{array}{cc|c} \alpha I_{n_\tau} & 0 & \beta I_{n_\tau} \\ 0 & I_{n_{\tau-1}-n_\tau} & 0 \\ \hline \beta I_{n_\tau} & 0 & \alpha I_{n_\tau} \end{array}\right], \quad \text{where } \alpha = \frac{n_\tau}{n_{\tau-1}+n_\tau} \text{ and } \beta = \frac{n_{\tau-1}}{n_{\tau-1}+n_\tau}.$$

From the above discussion, we obtain the following result.

Theorem 2. *The $n \times n$ matrix A_{DSHB} generated by Algorithm 2 is doubly stochastic, hierarchically banded, and satisfies $J = J_0A_{\text{DSHB}}J_0$. Each matrix in the scaled HB sequence $\{\widetilde{A}_{\text{DSHB}}^{(k)}\}_{k=1}^{\tau}$ generated by Algorithm 2 is doubly stochastic. In addition, it holds that*

$$\text{nnz}(A_{\text{DSHB}}) = \sum_{k=1}^{\tau} kn_k, \qquad d_{\max}(A_{\text{DSHB}}) = \tau.$$

Proof. (The subscript in A_{DSHB} (and $\widetilde{A}_{\text{DSHB}}^{(k)}$) is omitted in the proof for readability.) The doubly stochastic property of A and $\{\widetilde{A}^{(k)}\}$ follows from the assignments (16)–(17) and condition (18). The hierarchically banded structure of A follows from the recursive nature of Algorithm 2, and in particular, the recursive partition of $\widetilde{A}^{(k)}$ in Line 7. Next, the factorization $J = J_0AJ_0$ holds by recursively applying (18) for $k = \tau-1, \tau-2, \ldots, 1$. Finally, the number of nonzeros and the largest node degree can be calculated using the same approach as for the RHB factor in Theorem 1. □

5 Sequential doubly stochastic factorization

The DSHB factorization inspires another type of factorization for J, in which the factor $A \in \mathbb{R}^{n\times n}$ in $J = J_0AJ_0$ is no longer symmetric (nor hierarchically banded) but remains doubly stochastic. Since the asymmetric, doubly stochastic matrix A can be written as the product of a sequence of doubly stochastic matrices, such a factorization is called the *sequential doubly stochastic (SDS) factorization* of J.

Theorem 3 (Sequential doubly stochastic (SDS) factorizations of J)**.** *Let $A \in \mathbb{HB}^n$ be the DSHB factor of J and $\{\widetilde{A}^{(k)}\}_{k=1}^{\tau}$ the associated scaled HB sequence, constructed via Algorithm 2. For all $k \in [\tau - 1]$, define*

$$T^{(k)} := \begin{bmatrix} \widetilde{A}_{11}^{(k)} & \widetilde{A}_{12}^{(k)} \\ \big(\widetilde{A}_{12}^{(k)}\big)^{\mathsf{T}} & \frac{m_k}{m_{k-1}} I_{m_k} \end{bmatrix} \in \mathbb{R}^{m_{k-1} \times m_{k-1}}, \tag{21}$$

with the convention $m_0 := n$, and $T^{(\tau)} := \widetilde{A}^{(\tau)} \equiv I_{n_\tau}$. The augmented matrices $\{\widehat{T}^{(k)}\}_{k=1}^{\tau} \subset \mathbb{R}^{n \times n}$ are defined as

$$\widehat{T}^{(k)} = I_{n_1} \oplus I_{n_2} \oplus \cdots \oplus I_{n_{k-1}} \oplus T^{(k)}, \quad k \in [\tau].$$

Then, the matrices $\{T^{(k)}\}_{k=1}^{\tau}$ (and $\{\widehat{T}^{(k)}\}_{k=1}^{\tau}$) are all symmetric, doubly stochastic, and the matrix $J = \frac{1}{n}\mathbb{1}_n\mathbb{1}_n^{\mathsf{T}}$ can be factored as

$$J = J_0 A_{\mathrm{L}} J_0 = J_0 A_{\mathrm{R}} J_0, \tag{22}$$

where

$$\begin{aligned} A_{\mathrm{L}} &:= \widehat{T}^{(1)} \widehat{T}^{(2)} \cdots \widehat{T}^{(\tau)} \\ &= T^{(1)} \cdot (I_{n_1} \oplus (T^{(2)} \cdot (I_{n_2} \oplus \cdots (T^{(\tau-1)} \cdot (I_{n_\tau} \oplus T^{(\tau)}))))), \\ A_{\mathrm{R}} &:= \widehat{T}^{(\tau)} \widehat{T}^{(\tau-1)} \cdots \widehat{T}^{(1)} \\ &= (I_{n_1} \oplus \cdots \oplus (I_{n_{\tau-1}} \oplus (I_{n_\tau} \oplus T^{(\tau)}) \cdot T^{(\tau-1)}) \cdot T^{(\tau-2)}) \cdots T^{(1)}. \end{aligned}$$

In addition, both factors A_{L} and A_{R} are doubly stochastic.

By definition, the matrices $\{T^{(k)}\}$ have nonzero entries only in three subdiagonals. The first factorization in (22) is called the *left SDS factorization* of J, and the second is called the *right SDS factorization*.

Proof. Define $\overline{J}_0 := J_0$, $m_0 := n$, and

$$\begin{aligned} V^{(k)} &:= T^{(k)} \cdot (I_{n_k} \oplus (T^{(k+1)} \cdot (I_{n_{k+1}} \oplus \cdots (T^{(\tau-1)} \cdot (I_{n_\tau} \oplus T^{(\tau)}))))) \\ &= T^{(k)} \cdot (I_{n_k} \oplus V^{(k+1)}) \in \mathbb{R}^{m_{k-1} \times m_{k-1}}, \end{aligned} \tag{23}$$

for all $k \in [\tau - 1]$, and $V^{(\tau)} := T^{(\tau)} \equiv I_{n_\tau}$. By definition, each matrix $V^{(k)}$ is doubly stochastic, because $T^{(k)}$ is doubly stochastic.

First, we apply mathematical induction to prove that for $k = \tau - 1, \ldots, 1$,

$$\overline{J}_{k-1} V^{(k)} \overline{J}_{k-1} = \tfrac{1}{m_{k-1}} \mathbb{1}_{m_{k-1}} \mathbb{1}_{m_{k-1}}^{\mathsf{T}}, \tag{24}$$

The base case $k \leftarrow \tau - 1$ holds because

$$\overline{J}_{\tau-2} V^{(\tau-1)} \overline{J}_{\tau-2} = \begin{bmatrix} J_{\tau-1} & \\ & J_\tau \end{bmatrix} T^{(\tau-1)} (I_{n_\tau} \oplus T^{(\tau)}) \begin{bmatrix} J_{\tau-1} & \\ & J_\tau \end{bmatrix} \tag{25a}$$

$$= \begin{bmatrix} J_{\tau-1} & \\ & J_\tau \end{bmatrix} T^{(\tau-1)} \begin{bmatrix} J_{\tau-1} & \\ & J_\tau \end{bmatrix} \tag{25b}$$

$$= \begin{bmatrix} J_{\tau-1} & \\ & J_\tau \end{bmatrix} \begin{bmatrix} \widetilde{A}_{11}^{(\tau-1)} & \widetilde{A}_{12}^{(\tau-1)} \\ \big(\widetilde{A}_{12}^{(\tau-1)}\big)^{\mathsf{T}} & \frac{m_{\tau-1}}{m_{\tau-2}} I \end{bmatrix} \begin{bmatrix} J_{\tau-1} & \\ & J_\tau \end{bmatrix} \tag{25c}$$

$$
\begin{aligned}
&= \begin{bmatrix} J_{\tau-1}\widetilde{A}_{11}^{(\tau-1)}J_{\tau-1} & J_{\tau-1}\widetilde{A}_{12}^{(\tau-1)}J_{\tau} \\ \big(J_{\tau-1}\widetilde{A}_{12}^{(\tau-1)}J_{\tau}\big)^{\mathsf{T}} & \frac{m_{\tau-1}}{m_{\tau-2}}J_{\tau}^2 \end{bmatrix} \\
&= \tfrac{1}{m_{\tau-2}}\mathbb{1}_{m_{\tau-2}}\mathbb{1}_{m_{\tau-2}}^{\mathsf{T}}. \qquad (25\text{d})
\end{aligned}
$$

The first equation (25a) uses $\overline{J}_{\tau-2} = J_{\tau-1} \oplus \overline{J}_{\tau-1} = J_{\tau-1} \oplus J_{\tau}$. Then, (25b) and (25c) use the definition $T^{(\tau)} = I_{n_\tau}$ and (21). Finally, (25d) follows from the updates (16) and (17) in Algorithm 2, as well as the fact $J_\tau^2 = J_\tau$.

Next, suppose the identity (24) holds for $k \in [\tau - 1]$, and we establish the same identity with $k \leftarrow k-1$:

$$
\begin{aligned}
&\overline{J}_{k-2}T^{(k-1)}(I_{n_k} \oplus V^{(k)})\overline{J}_{k-2} \\
&= \begin{bmatrix} J_{k-1} & \\ & \overline{J}_{k-1} \end{bmatrix} \begin{bmatrix} \widetilde{A}_{11}^{(k-1)} & \widetilde{A}_{12}^{(k-1)} \\ \big(\widetilde{A}^{(k-1)}\big)^{\mathsf{T}} & \frac{m_{k-1}}{m_{k-2}}I \end{bmatrix} \begin{bmatrix} I_{n_{k-1}} & \\ & V^{(k)} \end{bmatrix} \begin{bmatrix} J_{k-1} & \\ & \overline{J}_{k-1} \end{bmatrix} \qquad (26\text{a}) \\
&= \begin{bmatrix} J_{k-1}A_{11}^{(k-1)}J_{k-1} & J_{k-1}A_{12}^{(k-1)}V^{(k)}\overline{J}_{k-1} \\ \big(J_{k-1}A_{12}^{(k-1)}\overline{J}_{k-1}\big)^{\mathsf{T}} & \frac{m_{k-1}}{m_{k-2}}\overline{J}_{k-1}V^{(k)}\overline{J}_{k-1} \end{bmatrix} \qquad (26\text{b}) \\
&= \begin{bmatrix} \frac{1}{m_{k-2}}\mathbb{1}_{n_{k-1}}\mathbb{1}_{n_{k-1}}^{\mathsf{T}} & \frac{1}{m_{k-2}}\mathbb{1}_{n_{k-1}}\mathbb{1}_{m_{k-1}}^{\mathsf{T}} \\ \frac{1}{m_{k-2}}\mathbb{1}_{m_{k-1}}\mathbb{1}_{n_{k-1}}^{\mathsf{T}} & \frac{1}{m_{k-2}}\mathbb{1}_{m_{k-1}}\mathbb{1}_{m_{k-1}}^{\mathsf{T}} \end{bmatrix} \qquad (26\text{c}) \\
&= \tfrac{1}{m_{k-2}}\mathbb{1}_{m_{k-2}}\mathbb{1}_{m_{k-2}}^{\mathsf{T}}.
\end{aligned}
$$

In (26a), we use $\overline{J}_{k-2} = J_{k-1} \oplus \overline{J}_{k-1}$, the definition of $T^{(k)}$ in (21), and the definition of direct sum. After multiplying out all the matrices in (26b), the third equation (26c) follows from the definition of $A_{11}^{(k-1)}$ and $A_{12}^{(k-1)}$ (see (16)–(17)), the definition of $V^{(k)}$ in (23), and the assumption that identity (24) holds for $k \in [\tau - 1]$. In particular, the $(1,2)$-block of (26b) is further simplified as follows:

$$
\begin{aligned}
&J_{k-1}A_{12}^{(k-1)}V^{(k)}\overline{J}_{k-1} \\
&= \tfrac{1}{n_{k-1}}\mathbb{1}_{n_{k-1}}\left(\mathbb{1}_{n_{k-1}}^{\mathsf{T}}\begin{bmatrix} \operatorname{diag}\left(\frac{n_{k-1}}{m_{k-2}}\mathbb{1}_{m_{k-1}}\right) \\ 0 \end{bmatrix}\right)V^{(k)}\overline{J}_{k-1} \qquad (27\text{a}) \\
&= \tfrac{1}{m_{k-2}}\mathbb{1}_{n_{k-1}}\big(\mathbb{1}_{m_{k-1}}^{\mathsf{T}}V^{(k)}\big)\overline{J}_{k-1} \qquad (27\text{b}) \\
&= \tfrac{1}{m_{k-2}}\mathbb{1}_{n_{k-1}}\mathbb{1}_{m_{k-1}}^{\mathsf{T}}\overline{J}_{k-1} \qquad (27\text{c}) \\
&= \tfrac{1}{m_{k-2}}\mathbb{1}_{n_{k-1}}\mathbb{1}_{m_{k-1}}^{\mathsf{T}}. \qquad (27\text{d})
\end{aligned}
$$

In (27a), we use the definition of $\widetilde{A}_{12}^{(k-1)}$ in (17), and (27b) writes out $\mathbb{1}_{n_{k-1}}^{\mathsf{T}}\widetilde{A}_{12}^{(k-1)} = \frac{n_{k-1}}{m_{k-2}}\mathbb{1}_{m_{k-1}}^{\mathsf{T}}$. Then, (27c) and (27d) use the doubly stochastic property of $V^{(k)}$ and $\overline{J}_{k-1}$, respectively.

Therefore, the induction hypothesis is proved, and (24) holds for all $k \in [\tau - 1]$. In particular, (24) with $k \leftarrow 1$ gives the left SDS factorization:

$$J_0 A_{\mathrm{L}} J_0 = \overline{J}_0 V^{(1)} \overline{J}_0 = \tfrac{1}{m_0}\mathbb{1}_{m_0}\mathbb{1}_{m_0}^{\mathsf{T}} = J.$$

The first equation uses the convention $\overline{J}_0 = J_0$ and the relation $A_{\mathrm{L}} = T^{(1)}(I_{n_1} \oplus V^{(2)}) = V^{(1)}$. The second equation applies (24) with $k = 1$, and the last one follows from the convention $m_0 = n$. Then, the right SDS factorization follows directly from the fact that $A_{\mathrm{R}} = A_{\mathrm{L}}^{\mathsf{T}}$ and thus $J = (J_0 A_{\mathrm{L}} J_0)^{\mathsf{T}} = J_0 A_{\mathrm{L}}^{\mathsf{T}} J_0 = J_0 A_{\mathrm{R}} J_0$. Finally, the doubly stochastic property of A_{L} (and A_{R}) follows from that of $\{T^{(k)}\}$ and the fact that the product of doubly stochastic matrices is still doubly stochastic. □

In the context of decentralized optimization, if communication is modeled by the T-factors, then at each round of communication, each agent only needs to communicate with at most one neighbor (as $d_{\max}(T^{(k)}) = 2$ for all $k \in [\tau - 1]$). Such a property is called "one-peer" in decentralized optimization and holds for one-peer hyper-cubes [13] and one-peer exponential graphs [20].

We also note that the matrices $\{T^{(k)}\}$ represent the base-$(p+1)$ graphs introduced in [16]. Yet, the original work [16] fails to provide an explicit matrix representation for the base-$(p+1)$ graphs and does not prove that the weight matrices of their proposed base-$(p+1)$ graphs can be used to factorize the J matrix. Moreover, as explained in Section 2, the construction of all the matrices (A_{L}, A_{R}, $\{T^{(k)}\}$, and $\{\widetilde{A}^{(k)}\}$) does not necessarily rely on the base-p representation of the integer $n \in \mathbb{N}_{\geq 2}$, and only needs a decomposition $n = \sum_{k=1}^{\tau} n_k$ with $n_k \geq m_k = \sum_{i=k+1}^{\tau} n_i$ for all $k \in [\tau - 1]$. So, the original name "base-$(p+1)$" does not fully reveal the flexibility of the sequential doubly stochastic factorization proposed in this paper.

The following corollary presents the basic properties of the two SDS factors and the T-factors.

Corollary 4. *The total number of nonzeros in the matrix $T^{(k)}$ is $\operatorname{nnz}(T^{(k)}) = n_k + 2\sum_{i=k+1}^{\tau} n_i$, for $k \in [\tau]$, and the largest node degree is $d_{\max}(T^{(k)}) = 2$. In addition,*

$$\operatorname{nnz}(A_{\mathrm{L}}) = \operatorname{nnz}(A_{\mathrm{R}}) = \sum_{k=1}^{\tau} (2^k - 1) n_k, \qquad d_{\max}(A_{\mathrm{L}}) = \tau, \qquad d_{\max}(A_{\mathrm{R}}) = 2^{\tau-1}.$$

Proof. The total number of nonzeros in the matrix $T^{(k)}$ and the largest node degree $d_{\max}(T^{(k)})$ hold from the definition (21).

It follows from the definition of $V^{(k)}$ (23) that

$$\operatorname{nnz}(V^{(k)}) = n_k + m_k + 2\operatorname{nnz}(V^{(k+1)}), \quad \text{for all } k \in [\tau - 1],$$

and $\operatorname{nnz}(V^{(\tau)}) = n_\tau$. Then, recursion over k yields

$$\begin{aligned}
\operatorname{nnz}(A_{\mathrm{L}}) = \operatorname{nnz}(A_{\mathrm{R}}) = \operatorname{nnz}(V^{(1)}) &= n_1 + m_1 + 2\operatorname{nnz}(V^{(2)}) \\
&= n_1 + m_1 + 2(n_2 + m_2) + 4\operatorname{nnz}(V^{(3)}) \\
&\vdots \\
&= \sum_{k=1}^{\tau-1} 2^{k-1}(n_k + m_k) + 2^{\tau-1}\operatorname{nnz}(V^{(\tau)}) \\
&= \sum_{k=1}^{\tau-1} 2^{k-1}(n_k + m_k) + 2^{\tau-1} n_\tau \\
&= \sum_{k=1}^{\tau} 2^{k-1} n_k + \sum_{k=1}^{\tau-1} 2^{k-1} m_\tau \\
&= \sum_{k=1}^{\tau} 2^{k-1} n_k + \sum_{k=1}^{\tau-1} 2^{k-1} \sum_{i=k+1}^{\tau} n_i \\
&= \sum_{k=1}^{\tau} 2^{k-1} n_k + \sum_{k=2}^{\tau} (2^{k-1} - 1) n_k
\end{aligned}$$

$$= \sum_{k=1}^{\tau} (2^k - 1) n_k.$$

Similarly, the largest node degree of A_{L} (and A_{R}) can be calculated as

$$d_{\max}(A_{\mathrm{L}}) = d_{\max}(V^{(1)}) = d_{\max}(V^{(2)}) + 1 = \cdots = d_{\max}(V^{(\tau)}) + \tau - 1 = \tau,$$
$$d_{\max}(A_{\mathrm{R}}) = d_{\max}((V^{(1)})^{\mathsf{T}}) = 2d_{\max}(V^{(2)}) = \cdots = 2^{\tau-1} d_{\max}(V^{(\tau)}) = 2^{\tau-1}. \qquad \square$$

6 Application in decentralized averaging and optimization

In this section, we show how the presented factorizations of the form (2) can be used in decentralized averaging (in Section 6.1) and then describe extensions to decentralized optimization (in Section 6.2).

The application scenario considered here involves abstracting agents as high-performance computing (HPC) resources. In modern data centers, machines are organized (or clustered) into racks, and switches and routers are used to connect machines physically. These connections form the so-called *physical topology*, which is often fixed [15, §2]. In comparison, *virtual topology* refers to the logical network layout between virtual machines (VMs) and other HPC resources. It models how data are transferred between VMs and has the following properties (see, *e.g.*, [1]).

- *Flexibility.* Virtual topology can be dynamically redesigned without reconfiguring the physical hardware.
- *Scalability.* Virtual topology can span multiple physical data centers.
- *Automation.* Management of virtual topology can be automated via software-defined network (SDN) controllers.

These properties are crucial for adapting to evolving traffic demands with the purpose of, *e.g.*, reducing power consumption, network congestion, end-to-end delay, or blocking probability. Hence, with virtual topology, the connection between machines (agents) can be easily manipulated in a dynamic manner. However, the communication cost between machines is based on their physical locations. For example, communication within a network segment (*e.g.*, a rack or a section of the data center) is often more efficient and economical than that between network segments [15, §2]. The proposed sparse factorization (2) of J offers a robust solution for designing dynamic virtual topologies as it exploits the clustering and hierarchical structure in data centers, leverages the flexibility of virtual topology, and takes into account the non-uniform communication costs within a virtual topology.

6.1 Decentralized average consensus

The decentralized average consensus (or decentralized averaging) problem can be formally formulated as follows. In a group of n agents, each one holds a piece of information, denoted by $x_i^{(0)} \in \mathbb{R}^d$, and the entire group aims to compute the average $\overline{x} := \frac{1}{n}\sum_{i=1}^{n} x_i^{(0)}$ via communication. The communication (or connection) between agents is modeled by a sequence of (undirected) graphs (or topologies) $\mathcal{G}^{(k)} = (\mathcal{V}, W^{(k)}, \mathcal{E}^{(k)})$, where $\mathcal{V} = \{1, \ldots, n\}$ is the node set representing agents, each $\mathcal{E}^{(k)} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges (or connections), and the weighted adjacency matrix $W^{(k)} \in \mathbb{R}^{n\times n}$ stores the weights of the edges. It is assumed that the set of agents remains static while the set of edges can be time-varying. In this context, $W^{(k)}$ is often called the *mixing matrix*, and its entry $w_{ij}^{(k)} \in \mathbb{R}_{\geq 0}$ applies a weighting factor to the information exchanged between agent j and agent i. If $w_{ij}^{(k)} = 0$, it means agent i is not a neighbor of agent j in $\mathcal{G}^{(k)}$; *i.e.*, $(i,j) \notin \mathcal{E}^{(k)}$. We do not distinguish between a graph $\mathcal{G}$ and its weight matrix W. The state of agent i (or the information held by i) at iteration k is designated as $x_i^{(k)}$ and evolves according to the following recursion: for $k \in \mathbb{N}$,

$$x_i^{(k+1)} = \sum_{j:\,(i,j)\in\mathcal{E}^{(k)}} w_{ij}^{(k)} x_j^{(k)}, \quad \text{for all } i \in [n].$$

The above recursion can be written more compactly as

$$X^{(k+1)} = W^{(k)} X^{(k)}, \quad \text{where } X^{(k)} = \begin{bmatrix} x_1^{(k)} & x_2^{(k)} & \cdots & x_n^{(k)} \end{bmatrix}^\mathsf{T} \in \mathbb{R}^{n\times d}. \tag{28}$$

We say average consensus is achieved if either of the following conditions is satisfied.

1. The limit of each $x_i^{(k)}$ is $\bar{x}$: $\lim_{k\to\infty} x_i^{(k)} = \bar{x}$ for all $i \in [n]$.

2. There exists $\bar{k} \in \mathbb{N}$ such that $X^{(\bar{k})} = \bar{x}\mathbb{1}^\mathsf{T}$ and $X^{(k)} = \bar{x}\mathbb{1}^\mathsf{T}$ for all $k \in \mathbb{N}_{\geq \bar{k}}$.

In this section, we are interested in modern applications where virtual machines in HPC scenarios are abstracted as agents. In this case, the communication topology can be customized and easily altered during the averaging process. Hence, the proposed sparse factorization of J helps design topologies that achieve consensus within a finite number of communication rounds. To see this, consider a set of sparse matrices $\{W^{(i)}\}_{i=1}^q$ that satisfies (1). When the associated graph sequence $\{\mathcal{G}^{(i)}\}_{i=1}^q$ is used as the (time-varying) topologies for decentralized averaging, the iteration (28) yields

$$X^{(q)} = W^{(q)} W^{(q-1)} \cdots W^{(2)} W^{(1)} X^{(0)} = \tfrac{1}{n}\mathbb{1}\mathbb{1}^\mathsf{T} X^{(0)} = \mathbb{1}\bar{x}^\mathsf{T}.$$

Therefore, unlike classical results where consensus is achieved only asymptotically, *finite-time consensus* (*i.e.*, consensus in *exactly* q communication rounds) in decentralized averaging is achieved by exploiting sparse factorization of J. Moreover, the sparsity of all the factors in the proposed factorizations of J helps reduce the per-round communication costs. In decentralized average consensus (as well as decentralized optimization), the communication cost at each round is modeled by either the total number of nonzeros in the mixing matrix W or the largest node degree $d_{\max}$; see, *e.g.*, the recent book [12, §11.3].

To this end, the proposed HB and SDS factorizations of J can be used to construct sparse graph sequences with cheap per-round communication costs and the desirable finite-time consensus property for *arbitrary* number of agents $n \in \mathbb{N}_{\geq 2}$. This is in contrast to most previous work, which has requirements on the matrix order n (*e.g.*, $n = p^\tau$ for some $(p,\tau) \in \mathbb{N}_{\geq 2} \times \mathbb{N}_{\geq 1}$). Below, we describe in detail how to exploit the factorization $J = J_0 A J_0$ to construct graph sequences $\{W^{(i)}\}$ with finite-time consensus, and then discuss two additional advantages of the proposed HB and SDS factorizations.

- **Phase 1.** The communication network is constructed via a sparse factorization of $J_0 = J_1 \oplus \cdots \oplus J_\tau$. For example, each smaller matrix $J_j \in \mathbb{R}^{n_j \times n_j}$ can be decomposed as product of p-peer hyper-cuboids [11]. Then, each mixing matrix in Phase 1 is a direct sum of several p-peer hyper-cuboids (and identity matrices).

Matrices in phase 2	A_{RHB}	A_{DSHB}	A_{L}	A_{R}	T-factors
Num. of nonzeros	$n+\tau(\tau-1)$	$\sum_{k=1}^{\tau} kn_k$	$\sum_{k=1}^{\tau}(2^k-1)n_k$	$\sum_{k=1}^{\tau}(2^k-1)n_k$	$n_k+2\sum_{i=k+1}^{\tau} n_i$
Largest node degree d_{max}	τ	τ	τ	$2^{\tau-1}$	2
Num. of iter. in Phase 2	1	1	1	1	$\tau-1$

Table 1: Trade-offs between the communication cost (modeled by either the number of nonzeros or the largest node degree d_{max}) and the number of communication rounds in Phase 2.

- **Phase 2.** This phase corresponds to the A matrix in (2), which can be the RHB factor, the DSHB factor, the (left or right) SDS factor, or even a sequence of T-factors. A detailed comparison between these choices is discussed in the next paragraph and presented in Table 1.

- **Phase 3.** It corresponds to a sparse factorization of J_0, and can be the same as Phase 1.

In addition to the ability to handle an arbitrary number of agents, the proposed factorizations (RHB, DSHB, SDS) provide more flexibility to balance the communication costs and the number of communication rounds toward consensus. Recall that the communication cost involved in each iteration (28) is related to the total number of nonzeros and the largest node degree in the communication topology. Thus, using sparser graphs would reduce communication costs but likely increase the total number of iterations toward consensus. For example, using A_{L} (or A_{R}) completes Phase 2 in one iteration, while using the "one-peer" T-factors in (21) results in $\tau-1$ iterations in Phase 2. Such a trade-off is summarized in Table 1.

Moreover, the proposed form of factorization (2) handles the issue of non-uniform communication costs mentioned at the beginning of Section 6. In classical decentralized settings, it is typically assumed that the distance between agents is equidistant and that each agent is indistinguishable from another. However, this is not the case in the virtual topology of modern data centers. Recall that *inter-cluster* communication between machines in a rack (or a section of the data center) is often cheaper and swifter than *intra-cluster* communication. Such a characteristic is easily exploited by factorization $J = J_0AJ_0$. Communication in Phases 1 and 3 is all intra-cluster and can be modeled by different sparse factorizations of J_k, $k \in [\tau]$. The more expensive inter-cluster communication only happens in Phase 2 and is modeled by the sparse matrix A. So, the proposed factorization form (2) promotes cheap, intra-cluster communications and limits the more expensive, inter-cluster ones.

Numerical experiments Here, we verify that the proposed sparse factors of J satisfy the finite-time consensus property (1) in numerical experiments. To do so, we simulate an average consensus problem. Each agent is initialized with a random vector $x_i^{(0)} \sim \mathcal{N}(0, \Sigma)$ drawn from a Gaussian distribution (with Σ symmetric positive definite). The iterates $x_i^{(k)}$ evolve according to the recursion (28), and the consensus error at each iteration is defined as $\Xi^{(k)} := \frac{1}{n}\sum_{i=1}^{n} \|x_i^{(k)} - \bar{x}\|^2$.

Figure 2 presents the consensus error using the proposed graph sequences, the one-peer exponential graphs [20] and p-peer hyper-cuboids [11] for various numbers of agents. (The experimental settings and the presentation style strictly follow from [11, 20].) It is known that when n is not a power of 2, using the one-peer exponential graphs cannot achieve finite-time consensus [20]. This result, together with Theorems 1 to 3, is verified numerically in Figure 2. More specifically, the proposed factorizations have a steep drop in the consensus error, indicating the vanishing of the

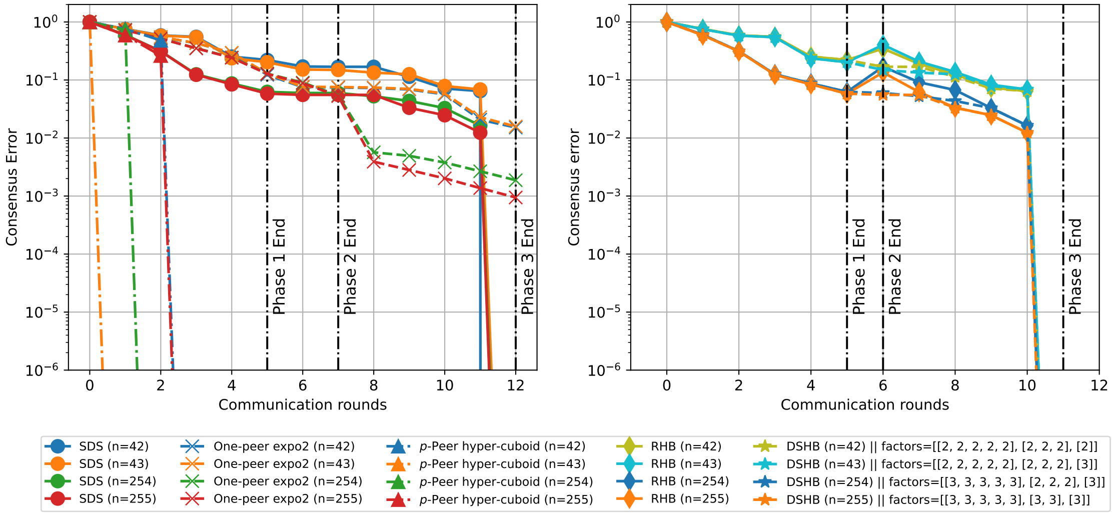

Figure 2: Consensus error versus the number of communication rounds. We examine five graph sequences over four different numbers of agents. The results for SDS factorization, the one-peer exponential graphs, and the p-peer hyper-cuboids are presented on the left; the results for RHB and DSHB factorizations are presented on the right. For SDS, RHB, and DSHB, the partition of n (3) and the factors used in each cluster are listed at the right-most of legend. For example, $[[2,2,2,2,2],[2,2,2],[2]]$ means the 42 agents are partitioned into three clusters with sizes 32, 8, and 2 and each cluster is further binary partitioned for J_0 generation. The factors used for p-peer hyper-cuboid are $42=[2,3,7]$, $43=[43]$, $254=[2,127]$, and $255=[3,5,17]$ respectively.

consensus error, while for one-peer exponential graphs, the consensus error decreases asymptotically. Moreover, note that the number of communication rounds is not the only criterion considered in practice. The per-round communication cost is also important. For example, when $n=43$ (a prime number), the p-peer hyper-cuboid reduces to the fully-connected graph and reaches consensus in just one round, while SDS takes 12 rounds. However, the total communication cost (measured by total number of nonzeros; see Table 1) for p-peer hyper-cuboid is 1849, compared to only 488 for the SDS factorization.

6.2 Decentralized optimization

Besides decentralized averaging, sparse factorization of J is also useful in decentralized optimization. In decentralized optimization, agents collaborate to solve the following optimization problem

$$\text{minimize} \quad f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x), \tag{29}$$

where the optimization variable is $x \in \mathbb{R}^d$, and each component function $f_i\colon \mathbb{R}^d \to \mathbb{R}$ is continuously differentiable and potentially nonconvex. Each agent $i \in \mathcal{V}$ only has access to one component function f_i, and agents communicate with each other via (time-varying) topologies $\{\mathcal{G}^{(k)}\}$. It can be shown that the decentralized average consensus problem is a special case of (29) with $f_i(x) = \frac{1}{2}\|x - x_i^{(0)}\|_2^2$.

In the context of decentralized optimization, a sparse factorization of J offers sequences of graphs that satisfy the finite-time consensus property, and incorporating such graph sequences in decentralized optimization algorithms could significantly reduce the communication cost in the algorithm while achieving a comparable convergence rate (compared with decentralized algorithms using traditional communication protocols) [11, 20]. Consider, for example, the decentralized gradient descent (DGD) algorithm, an extension of the gradient descent method to the decentralized setting:

$$X^{(k+1)} = W^{(k)}X^{(k)} - \alpha\nabla \boldsymbol{f}(X^{(k)}),$$

where $\nabla \boldsymbol{f}(X) := \begin{bmatrix}\nabla f_1(x_1) & \cdots & \nabla f_n(x_n)\end{bmatrix}^\mathsf{T} \in \mathbb{R}^{n\times d}$ is a matrix of all the component gradient functions and $\alpha \in \mathbb{R}_{>0}$ is the step size. At each DGD iteration, the matrix $W^{(k)}$ is the weighted adjacency matrix of a graph $\mathcal{G}^{(k)}$, and it can be sparse factors of J. More specifically, in DGD, one iterates from $W^{(1)}$ to $W^{(q)}$ in (2) and restart with $W^{(1)}$ again. In most existing analyses for DGD (and other decentralized optimization algorithms), the weight matrices $\{W^{(k)}\}$ are assumed to be connected and doubly stochastic (see, *e.g.*, the book [12, §11.3]). Recall that the T-factors (21) are doubly stochastic yet not connected. So when the T-factors are used in DGD, the convergence guarantee is different from classical results. To see this, recall that the convergence rate of DGD with a *connected* static graph is inversely proportional to the so-called *spectral gap* $(1 - \rho(W))$ [22], where $\rho(W)$ is the second largest eigenvalue (in modulus) of W. For time-varying connected graphs, the convergence rate is inversely proportional to the *worst-case* spectral gap $(1 - \rho_{\max})$ [8], where $\rho_{\max} := \max\{\rho(W^{(k)}) \colon k \in \mathbb{N}\}$. In contrast, when T-factors are used, the convergence rate of DGD follows from [20, Theorem 1] (because the topology sequence constructed using (2) and T-factors satisfies all the assumptions stated in [20]) and reads as

$$\frac{1}{K}\sum_{k=1}^{K}\|\nabla \boldsymbol{f}(X^{(k)})\|^2 = O\left(\frac{nq^2}{K}\right), \tag{30}$$

where q is the finite-time consensus parameter in (1). Unlike the classical convergence results that depend on the spectral gap, the rate (30) is independent of the connectivity of any of the individual graphs, but instead considers the joint effect of all the topologies used in the algorithm. Therefore, the proposed factorization enables DGD to handle potentially sparser topologies that can be disconnected during certain iterations. The same conclusion can also be drawn for DGD with momentum [20] and the (decentralized) gradient tracking algorithm [11].

Numerical experiments Numerical evidence is presented to demonstrate the potential benefits of using the T-factors in decentralized optimization algorithms. We apply DGD to solve the least squares problem (*i.e.*, (29) with each $f_i(x) = \|A_i x - b_i\|^2$). The entries in each $A_i \in \mathbb{R}^{m\times d}$ are independently and identically distributed (IID) random variables drawn from the standard distribution, and so are the vectors $\{\tilde{x}_i\}_{i=1}^n \subset \mathbb{R}^d$. The vector $b_i \in \mathbb{R}^d$ is then computed by $b_i = A_i\tilde{x}_i + \delta z_i$, where $\delta \in \mathbb{R}_{>0}$ is a prescribed constant and $z_i \in \mathbb{R}^d$ is Gaussian random noise. In the experiments, we set $n = 241$ (a prime number), $m = 100$, $d = 50$, and $\delta = 0.1$. We construct the partition (3) via binary representation of n, use the one-peer hyper-cubes in Phases 1 and 3, and use the proposed T-factors (21) in Phase 2.

Figure 3 presents the simulation results for various sparse topologies that have a maximum degree of 1. We see that DGD with T-factors has similar, if not better, performance compared to DGD with other sparse topologies. Yet, only the proposed SDS factorization considers the clustering

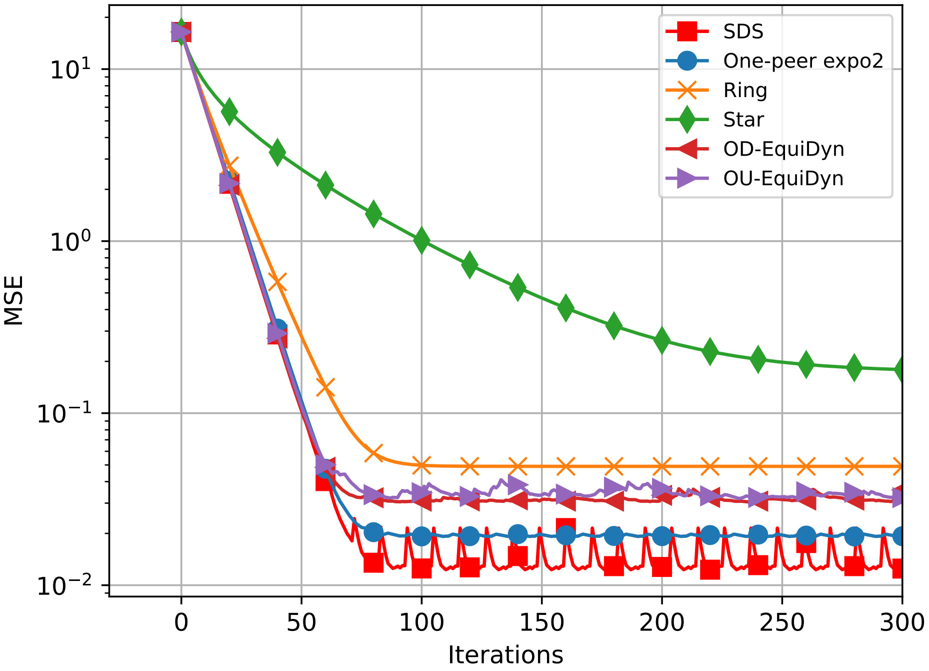

Figure 3: Mean-square error (MSE) versus the number of DGD iterations for the decentralized least squares problem, with various sparse graphs. The convergence of DGD with one-peer exponential graphs is plotted in blue, and that with one-peer undirected EquiTopo graphs [14] is in green.

and hierarchical structures in modern application scenarios. So, despite the similar convergence rate, using the proposed T-factors takes fewer inter-cluster communication and has lower total communication cost. An intriguing observation is that the proposed SDS factorization with T-factors exhibits oscillation in the mean-square error (MSE) over time, whereas DGD with one-peer exponential graphs approaches a more stable MSE asymptotically. A closer examination finds that the oscillation period equals to q, the length of the factorization in (2). This suggests that the oscillation may stem from the three-phase nature of SDS factorization and the presence of some disconnected factors in SDS factorization. Consequently, during each period, agents across different clusters may have different iterates $x_i^{(k)}$ and only achieve consensus at the end of each period.

7 Conclusion

In this paper, we study the sparse factorization $J = J_0 A J_0$, where $J = \frac{1}{n}\mathbb{1}_n\mathbb{1}_n^\mathsf{T}$ is the (scaled) all-ones matrix and $J_0 = J_1 \oplus \cdots \oplus J_\tau$ is the direct sum of several smaller (scaled) all-ones matrices. We introduce the hierarchically banded structure of a symmetric matrix, based on which we present two types of hierarchically banded factorization of J: the reduced hierarchically banded (RHB) factorization and the doubly stochastic hierarchically banded (DSHB) factorization. Moreover, inspired by the DSHB factorization, we propose the sequential doubly stochastic (SDS) factorization, which further factorizes the matrix A as the product of a sequence of symmetric, doubly stochastic matrices. We then discuss the usefulness of the proposed factorizations in decentralized average consensus and decentralized optimization. The presented three types of sparse factorization offer much flexibility in handling the trade-off between the per-iteration communication cost and the total number of communication rounds in decentralized averaging (and optimization).

Finally, recall that the partition (3) is assumed to be given and fixed throughout the paper. Further investigation is needed in the design of this partition to fully leverage the power of the proposed sparse factorizations in decentralized optimization.

References

[1] Samaresh Bera, Sudip Misra, and Athanasios V. Vasilakos. Software-defined networking for internet of things: A survey. *IEEE Internet of Things Journal*, 4(6):1994–2008, 2017.

[2] Steffen Börm, Lars Grasedyck, and Wolfgang Hackbusch. Introduction to hierarchical matrices with applications. *Engineering Analysis with Boundary Elements*, 27(5):405–422, 2003.

[3] Frank E. Curtis, John Drew, Chi-Kwong Li, and Daniel Pragel. Central groupoids, central digraphs, and zero-one matrices A satisfying $A^2 = J$. *Journal of Combinatorial Theory, Series A*, 105(1):35–50, 2004.

[4] Tri Dao, Albert Gu, Matthew Eichhorn, Atri Rudra, and Christopher Ré. Learning fast algorithms for linear transforms using butterfly factorizations. In *International Conference on Machine Learning*, pages 1517–1527, 2019.

[5] Nicolaas G. de Bruijn. A combinatorial problem. *Proceedings of the Section of Sciences of the Koninklijke Nederlandse Akademie van Wetenschappen te Amsterdam*, 49(7):758–764, 1946.

[6] Jean-Charles Delvenne, Ruggero Carli, and Sandro Zampieri. Optimal strategies in the average consensus problem. *Systems & Control Letters*, 58(10-11):759–765, 2009.

[7] Fenn King and Kai Wang. On the g-circulant solutions to the matrix equation $A^m = \lambda J$. *Journal of Combinatorial Theory, Series A*, 38(2):182–186, 1985.

[8] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized SGD with changing topology and local updates. In *International Conference on Machine Learning*, pages 5381–5393, 2020.

[9] Rui Lin, Jie Ran, King Hung Chiu, Graziano Chesi, and Ngai Wong. Deformable butterfly: A highly structured and sparse linear transform. In *Advances in Neural Information Processing Systems*, volume 34, pages 16145–16157, 2021.

[10] S. L. Ma and William C. Waterhouse. The g-circulant solutions of $A^m = \lambda J$. *Linear Algebra and its Applications*, 85:211–220, 1987.

[11] Edward Duc Hien Nguyen, Xin Jiang, Bicheng Ying, and César A. Uribe. On graphs with finite-time consensus and their use in gradient tracking. *arXiv preprint*, arXiv:2311.01317, 2023.

[12] Ernest K. Ryu and Wotao Yin. *Large-Scale Convex Optimization: Algorithms & Analyses via Monotone Operators.* Cambridge University Press, 2022.

[13] Guodong Shi, Bo Li, Mikael Johansson, and Karl Henrik Johansson. Finite-time convergent gossiping. *IEEE/ACM Transactions on Networking*, 24(5):2782–2794, 2016.

[14] Zhuoqing Song, Weijian Li, Kexin Jin, Lei Shi, Ming Yan, Wotao Yin, and Kun Yuan. Communication-efficient topologies for decentralized learning with $O(1)$ consensus rate. In *Advances in Neural Information Processing Systems*, volume 35, pages 1073–1085, 2022.

[15] Thomas Sterling, Maciej Brodowicz, and Matthew Anderson. *High Performance Computing: Modern Systems and Practices.* Morgan Kaufmann, 2017.

[16] Yuki Takezawa, Ryoma Sato, Han Bao, Kenta Niwa, and Makoto Yamada. Beyond exponential graph: Communication-efficient topologies for decentralized learning via finite-time convergence. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[17] Maguy Trefois, Paul Van Dooren, and Jean-Charles Delvenne. Binary factorizations of the matrix of all ones. *Linear Algebra and its Applications*, 468:63–79, 2015.

[18] Kai Wang. On the g-circulant solutions to the matrix equation $A^m = \lambda J$. *Journal of Combinatorial Theory, Series A*, 33(3):287–296, 1982.

[19] Yao-Kun Wu, Rui-Zhong Jia, and Qiao Li. g-Circulant solutions to the $(0, 1)$ matrix equation $A^m = J_n$. *Linear Algebra and its Applications*, 345(1):195–224, 2002.

[20] Bicheng Ying, Kun Yuan, Yiming Chen, Hanbin Hu, Pan Pan, and Wotao Yin. Exponential graph is provably efficient for decentralized deep training. In *Advances in Neural Information Processing Systems*, volume 34, pages 13975–13987, 2021.

[21] Bicheng Ying, Kun Yuan, Hanbin Hu, Yiming Chen, and Wotao Yin. BlueFog: Make decentralized algorithms practical for optimization and deep learning. *arXiv e-prints*, arXiv:2111.04287, 2021.

[22] Kun Yuan, Qing Ling, and Wotao Yin. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3):1835–1854, 2016.