SIAM J. MATRIX ANAL. APPL.
Vol. 40, No. 1, pp. 320–346

MAX-BALANCED HUNGARIAN SCALINGS*

JAMES HOOK†, JENNIFER PESTANA‡, FRANÇOISE TISSEUR§, AND JONATHAN HOGG¶

Abstract. A Hungarian scaling is a diagonal scaling of a matrix that is typically applied along with a permutation to a sparse linear system before calling a direct or iterative solver. A matrix that has been Hungarian scaled and reordered has all entries of modulus less than or equal to 1 and entries of modulus 1 on the diagonal. An important fact that has been largely overlooked by the previous research into Hungarian scaling of linear systems is that a given matrix typically has a range of possible Hungarian scalings, and direct or iterative solvers may behave quite differently under each of these scalings. Since standard algorithms for computing Hungarian scalings return only one scaling, it is natural to ask whether a superior performing scaling can be obtained by searching within the set of all possible Hungarian scalings. To this end we propose a method for computing a Hungarian scaling that is optimal from the point of view of a measure of diagonal dominance. Our method uses max-balancing, which minimizes the largest off-diagonal entries in the scaled and permuted matrix. Numerical experiments illustrate the increased diagonal dominance produced by max-balanced Hungarian scaling as well as the reduced need for row interchanges in Gaussian elimination with partial pivoting and the improved stability of LU factorizations without pivoting. We additionally find that applying the max-balancing scaling before computing incomplete LU preconditioners improves the convergence rate of certain iterative methods. Our numerical experiments also show that the Hungarian scaling returned by the HSL code MC64 has performance very close to that of the optimal max-balanced Hungarian scaling, which further supports the use of this code in practice.

Key words. max-plus algebra, diagonal scaling, Hungarian scaling, max-balancing, diagonal dominance, linear systems of equations, sparse matrices, incomplete LU preconditioner

AMS subject classifications. 65F35, 15A12, 15A60, 15A80

DOI. 10.1137/15M1024871

1. Introduction.

A Hungarian scaling is a two-sided diagonal scaling of a matrix that is applied along with a permutation P to a linear system $Ax = b$, with $A \in \mathbb{C}^{n\times n}$ and $b \in \mathbb{C}^n$, yielding

$$H = PD_1AD_2, \quad Hy = PD_1b, \quad x = D_2y, \tag{1.1}$$

where $D_1, D_2 \in \mathbb{R}^{n\times n}$ are diagonal and nonsingular. The scaled and reordered matrix $H = (h_{ij})$ is such that $|h_{ij}| \le 1$ and $|h_{ii}| = 1$ for $i, j = 1, \ldots, n$. A permutation matrix P, such that (1.1) holds, is commonly referred to as an optimal assignment for A.

Olschowka and Neumaier [16] propose applying a Hungarian scaling together with a permutation to matrices prior to performing Gaussian elimination. They prove that

*Received by the editors June 8, 2015; accepted for publication (in revised form) December 19, 2018; published electronically February 26, 2019.
http://www.siam.org/journals/simax/40-1/M102487.html
Funding: The work of the first, second, and third authors was supported by Engineering and Physical Sciences Research Council grant EP/I005293. The work of the third author was also supported by a Royal Society-Wolfson Research Merit Award.
†Department of Mathematical Sciences, University of Bath, Bath, BA2 7AY, UK (j.l.hook@bath.ac.uk).
‡Department of Mathematics and Statistics, University of Strathclyde, Glasgow, G1 1XH, UK (jennifer.pestana@strath.ac.uk).
§School of Mathematics, The University of Manchester, Manchester, M13 9PL, UK (francoise.tisseur@manchester.ac.uk).
¶Scientific Computing Department, STFC Rutherford Appleton Laboratory, Harwell, Oxford, Didcot, OX11 0QX, UK (jhogg41@gmail.com).

this preprocessing step eliminates the need for row interchanges for some special class of matrices. Some intuitive explanation for this widely observed fact is provided in Hook and Tisseur [13, Thm. 3.9] for general matrices. Duff and Koster [7, 8] describe an efficient algorithm for computing a Hungarian scaling, on which the HSL code MC64 is based [14]. They show that applying the scaling and permutation significantly reduces the number of delayed pivots during factorization of sparse nonsymmetric matrices by a multifrontal direct solver [8]. The authors explain this phenomenon by pointing out that the Hungarian scaled matrix H tends to be more diagonally dominant than the original matrix A.

Benzi, Haws, and Tůma [1] show that Hungarian scaling is an effective preprocessing step before applying BiCGSTAB, GMRES, or TFQMR to sparse indefinite nonsymmetric matrices. The scaled matrices require significantly fewer iterations for convergence. Again, the authors explain this phenomenon by pointing out that the Hungarian scaled matrix H in (1.1) tends to be more diagonally dominant than the original matrix A. The authors also experiment with using Hungarian scaling as a preprocessing step before applying preconditioned BiCGSTAB with an incomplete LU (ILU) preconditioner. Without scaling they show that there are many problems for which attempts to compute a very sparse ILU preconditioner break down. In these cases, to reliably compute effective ILU preconditioners they are forced to compute denser ILU factors at a considerably increased cost. However, they show that after Hungarian scaling has been applied it is possible to reliably compute very sparse ILU preconditioners.

In the symmetric case, rather than permuting matched entries (unsymmetrically) to the main diagonal, these entries can instead be permuted (symmetrically) to the subdiagonal and used in 2×2 block pivots. However, in the sparse case, doing so conflicts with the minimization of fill-in. Various compromises have been proposed. In [11] and [12] Hogg and Scott show that for most matrices merely using the symmetrized Hungarian scaling is sufficient to eliminate the need for significant amounts of pivoting in LDL^T factorizations with threshold partial pivoting. For the class of problems where this is not the case, reordering roughly half the matched entries onto the subdiagonal and then applying a constrained fill-reducing ordering is sufficient to reduce pivoting to a manageable level.

Olschowka and Neumaier [16, Alg. 4.2] describe a second round of scaling with a nonsingular diagonal matrix S applied to the Hungarian scaled matrix H in (1.1). When the optimal assignment permutation is unique, this second round of scaling yields the doubly scaled matrix $S^{-1}HS$ with all off-diagonal entries of modulus strictly less than one [16, Thm. 4.3]. This is equivalent to choosing a different pair of diagonal matrices D_1, D_2 in the initial Hungarian scaling (1.1). However, the fact that the Hungarian scaling and reordering associated with a matrix $A \in \mathbb{C}^{n\times n}$ are not necessarily unique has been overlooked in subsequent research into Hungarian scaling of linear systems, and there have not been any numerical experiments that compare the effectiveness of different Hungarian scalings. In general there is a range of different diagonal matrix pairs $D_1, D_2 \in \mathbb{R}^{n\times n}$ and permutation matrices $P \in \mathbb{R}^{n\times n}$ which results in different Hungarian scaled and reordered matrices, for which direct or iterative solvers may behave quite differently. Since the increased diagonal dominance of the Hungarian scaled matrices has been repeatedly cited as responsible for their improved numerical characteristics, we focus in this paper on trying to obtain Hungarian scaled matrices that are as diagonally dominant as possible. For this, we

consider row-wise diagonal dominance measures of the form

$$\Delta(A) = g\left(\left(\sqrt[p]{\frac{\sum_{j\neq i}|a_{ij}|^p}{|a_{ii}|^p}}\right)_{i=1,\dots,n}\right) \tag{1.2}$$

for some $p \in [1,\infty)$ and where $g : \mathbb{R}_+^n \mapsto \mathbb{R}_+$ is some function that amalgamates the individual row p-norm scores into a single score for the whole matrix.

While the choice of optimal assignment permutation may impact the number of row interchanges required during Gaussian elimination, it is difficult to predict which permutations will work best in advance. Although the choice of optimal assignment permutation might affect the diagonal dominance of a general matrix, we will show that once a matrix has been Hungarian scaled, all of the possible choices of optimal assignment permutation result in scaled and reordered matrices with the same measure of diagonal dominance. Hence we focus on the choice of diagonal matrices D_1 and D_2 defining the Hungarian scaling and, in particular, on the following two questions: What does the set of all Hungarian scalings of a matrix look like? And how do we choose the best possible Hungarian scaling for a particular problem?

To answer these questions we will use results from max-plus algebra, to which we give a brief introduction in section 2. It turns out that the different Hungarian scalings of a matrix A are all related by diagonal similarities, so that if $H = PD_1AD_2$ and $H' = PD_1'AD_2'$ are both Hungarian scaled, then there exists a diagonal matrix S such that $H' = S^{-1}HS$. Therefore, starting from one Hungarian scaling, we can generate new Hungarian scalings by applying "special" diagonal similarities. The diagonal matrix S must be such that H' retains the properties of a Hungarian scaled matrix, i.e., $|h_{ij}'| \leq 1$ and $|h_{ii}'| = 1$ for all i, j. These conditions on S are very naturally expressed in terms of max-plus algebra, and that is why it proves so useful here; see Theorem 2.5.

In order to compute a Hungarian scaling that is as diagonally dominant as possible, we use a technique called max-balancing. Max-balanced graphs were introduced by Schneider and Schneider in connection with certain network flow problems [20]. A directed weighted graph is max-balanced if for any subset of vertices the maximum weight of an edge into that subset is equal to the maximum weight of an edge out of that subset. We can use the max-balancing algorithm of Schneider and Schneider to compute a nonsingular diagonal matrix $S \in \mathbb{R}^{n\times n}$ such that the scaled matrix $M = S^{-1}HS$ is max-balanced. Intuitively, max-balancing is the similarity scaling obtained by first minimizing the largest off-diagonal entry in the matrix, then minimizing the next largest entry subject to minimizing the first, and so on.

We show in section 3 that max-balancing (a) preserves the property of a matrix being Hungarian scaled and (b) minimizes the entrywise p-norm over all diagonal similarity scalings of A in the limit as p tends to infinity; see Theorem 3.2. As a result, the max-balancing of a Hungarian scaled matrix tends to be more diagonally dominant than the initial Hungarian scaling H. Theorem 3.8, which is the main theoretical result of this paper, states that the max-balanced Hungarian scaling of A is the unique optimal scaling and reordering of A with respect to a particular p-norm diagonal dominance measure in the limit as p tends to infinity. If we were to attempt to minimize some other measure of diagonal dominance via similarity scaling, then there would be no guarantee that we would be able to preserve the properties of being Hungarian scaled, i.e., that no off-diagonal entries have modulus greater than 1. The elegance of the max-balanced Hungarian scaling is that both diagonal dominance and Hungarian scaling are achieved simultaneously.

To demonstrate the effectiveness of max-balancing we include numerical experiments in section 4. We focus on solving $Ax = b$ via LU factorization, where $A \in \mathbb{C}^{n\times n}$ is sparse and nonsymmetric. Our experiments confirm that max-balancing improves diagonal dominance. Additionally, the condition number and number of row interchanges in Gaussian elimination with partial pivoting, reduced by Hungarian scaling, are further reduced by max-balancing Hungarian scaling. Finally, we apply the max-balancing scaling before computing incomplete LU preconditioners for GMRES and BiCGStab and find that doing so reduces the number of iterations for both methods.

2. Introduction to max-plus algebra and Hungarian scaling. We introduce in this section the basic max-plus algebra concepts that are needed to understand the theoretical results in our paper. Max-plus algebra concerns the max-plus semiring $\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}$, which is equipped with the binary operations max and plus,

$$a \oplus b = \max\{a, b\}, \quad a \otimes b = a + b \quad \text{for all } a, b \in \mathbb{R}_{\max},$$

with $-\infty$ and 0 playing the role of additive and multiplicative identities. Throughout this paper we use calligraphic letters for elements of $\mathbb{R}_{\max}$. A max-plus matrix $\mathcal{A} \in \mathbb{R}_{\max}^{n\times m}$ is simply an array of elements from $\mathbb{R}_{\max}$.

Max-plus matrix multiplication is defined analogously to classical matrix multiplication so that if $\mathcal{A} \in \mathbb{R}_{\max}^{n\times m}$ and $\mathcal{B} \in \mathbb{R}_{\max}^{m\times \ell}$, then $\mathcal{A} \otimes \mathcal{B} \in \mathbb{R}_{\max}^{n\times \ell}$ with

$$\big(\mathcal{A} \otimes \mathcal{B}\big)_{ij} = \bigoplus_{k=1}^{m} a_{ik} \otimes b_{kj} = \max_{1\le k\le m} a_{ik} + b_{kj}.$$

A max-plus diagonal matrix has all off-diagonal entries equal to minus infinity. Let $\operatorname{diag}_\infty(d)$ denote the max-plus diagonal matrix with diagonal entries given by some vector $d \in \mathbb{R}_{\max}^n$; we use the subscript ∞ to distinguish them from classical $n \times n$ complex diagonal matrices, which we denote by $\operatorname{diag}(b)$ for some $b \in \mathbb{C}^n$. For $\mathcal{A} \in \mathbb{R}_{\max}^{n\times n}$ and for $u, v \in \mathbb{R}^n$, we have

$$\big(\operatorname{diag}_\infty(-u) \otimes \mathcal{A} \otimes \operatorname{diag}_\infty(-v)\big)_{ij} = a_{ij} - u_i - v_j.$$

The *max-plus permanent* of $\mathcal{A} \in \mathbb{R}_{\max}^{n\times n}$ is given by

$$\operatorname{perm}(\mathcal{A}) = \max_{\pi\in\Pi(n)} \sum_{j=1}^{n} a_{\pi(j)j}, \tag{2.1}$$

where the maximum is taken over the set $\Pi(n)$ of all permutations of $\{1, \dots, n\}$. We denote by $\pi = \mathrm{id}$ the identity permutation, i.e., $\mathrm{id} = \{1, \dots, n\}$. A permutation π which attains the maximum in (2.1) is called an *optimal assignment* of $\mathcal{A}$. When $\operatorname{perm}(\mathcal{A}) \neq -\infty$, the max-plus permanent of $\mathcal{A}$ can be rewritten as a minimization problem,

$$\operatorname{perm}(\mathcal{A}) = \min\left\{ \sum_{i=1}^{n} (u_i + v_i) : \ u, v \in \mathbb{R}^n, \ a_{ij} - u_i - v_j \le 0 \right\}. \tag{2.2}$$

A *Hungarian pair* of $\mathcal{A}$ is an optimal solution (u, v) to (2.2). It is named after the Hungarian algorithm, which is a widely used primal-dual algorithm for solving the optimal assignment problem.

For $\pi \in \Pi(n)$ denote by P_π the $n \times n$ classical permutation and by $\mathcal{P}_\pi$ the $n \times n$ max-plus permutation matrix, both defined by

$$(P_\pi)_{ij} = \begin{cases} 1 & \text{for } j = \pi(i), \\ 0 & \text{otherwise,} \end{cases} \qquad (\mathcal{P}_\pi)_{ij} = \begin{cases} 0 & \text{for } j = \pi(i), \\ -\infty & \text{otherwise.} \end{cases} \tag{2.3}$$

The following theorem, or, more precisely, its corollary for complex matrices, appears in [16, Thm. 2.8].

THEOREM 2.1 (Hungarian scaling). *For $\mathcal{A} \in \mathbb{R}_{\max}^{n\times n}$, with $\operatorname{perm}(\mathcal{A}) \neq -\infty$, let π and (u, v) be an optimal assignment and Hungarian pair of $\mathcal{A}$, respectively. Then the max-plus Hungarian scaled and reordered matrix*

$$\mathcal{H} = \mathcal{P}_\pi \otimes \operatorname{diag}_\otimes(-u) \otimes \mathcal{A} \otimes \operatorname{diag}_\otimes(-v)$$

is such that $h_{ij} \leq 0$ and $h_{ii} = 0$ for all $i, j = 1, \ldots, n$.

To link the classical algebra of complex matrices with standard addition and multiplication to the max-plus algebra, we use the following transformation, which is known as a non-Archimedean valuation:

$$\mathcal{V} : \mathbb{C} \mapsto \mathbb{R}_{\max}, \quad \mathcal{V}(x) = \log|x|, \tag{2.4}$$

with the convention that $\log 0 = -\infty$. For matrices, we apply the valuation componentwise; that is, for $A \in \mathbb{C}^{n\times n}$, $\mathcal{V}(A) = \mathcal{A} = \big(\log|a_{ij}|\big) \in \mathbb{R}_{\max}^{n\times n}$. Note that $\operatorname{perm}(\mathcal{A}) \neq -\infty$ with $\mathcal{A} = \mathcal{V}(A)$ means that A is not structurally rank deficient. The next result, which holds for complex or real matrices, is a direct consequence of Theorem 2.1.

COROLLARY 2.2. *Let $A \in \mathbb{C}^{n\times n}$ be of full structural rank. Let π and (u, v) be an optimal assignment and a Hungarian pair of $\mathcal{V}(A)$, respectively. Then the Hungarian scaled and reordered matrix*

$$H = P_\pi \operatorname{diag}\big(\exp(-u)\big) A \operatorname{diag}\big(\exp(-v)\big) \tag{2.5}$$

is such that $|h_{ij}| \leq 1$ and $|h_{ii}| = 1$ for all $i, j = 1, \ldots, n$.

We note that the max-plus matrix $\mathcal{H}$ in Theorem 2.1 is the componentwise log-of-absolute-value of the matrix H in Corollary 2.2, that is, $\mathcal{H} = \mathcal{V}(H)$.

The max-plus matrix $\mathcal{A} = \mathcal{V}(A)$ may have more than one optimal assignment and the optimal solution (u, v) to (2.2) is in general not unique. Let us look at a simple example to illustrate the latter point.

EXAMPLE 2.3. *Let $A \in \mathbb{R}^{3\times 3}$ and $\mathcal{A} := \mathcal{V}(A) \in \mathbb{R}_{\max}^{3\times 3}$ be given by*

$$A = \begin{bmatrix} \exp(6) & \exp(6) & \exp(9) \\ \exp(-4) & \exp(-3) & \exp(-2) \\ 0 & \exp(-7) & 1 \end{bmatrix}, \qquad \mathcal{A} = \begin{bmatrix} 6 & 6 & 9 \\ -4 & -3 & -2 \\ -\infty & -7 & 0 \end{bmatrix}.$$

It is easy to check that the max-plus matrix $\mathcal{A}$ has a unique optimal assignment $\pi = (1, 2, 3)$ and that (u, v) with $u = [0, -9, -9]^T$ and $v = [6, 6, 9]^T$ is a Hungarian pair for $\mathcal{A}$ yielding the Hungarian scaled matrix

$$H = \operatorname{diag}\big(\exp(-u)\big) A \operatorname{diag}\big(\exp(-v)\big) = \begin{bmatrix} 1 & 1 & 1 \\ \exp(-1) & 1 & \exp(-2) \\ 0 & \exp(-4) & 1 \end{bmatrix}. \tag{2.6}$$

Hungarian scaling tends to significantly reduce the matrix 2-norm condition number $\kappa_2(A) = \|A\|_2\|A^{-1}\|_2$. *For this example we have* $\kappa_2(A) = 4.1 \times 10^5 \gg \kappa_2(H) = 6.2$. *Note also that* H *is more diagonally dominant than* A.

We will show in the next section that if (u_s, v_s) *is another Hungarian pair for* $\mathcal{A}$, *then there exists* $s \in \mathbb{R}^3$ *such that* $(u_s, v_s) = (u + s, v - s)$. *This means that the Hungarian scaled matrices*

$$H_s = \operatorname{diag}\big(\exp(-u_s)\big) A \operatorname{diag}\big(\exp(-v_s)\big) = \operatorname{diag}\big(\exp(-s)\big) H \operatorname{diag}\big(\exp(s)\big)$$

and H *are similar. But not all diagonal similarity scalings of* H *are Hungarian scalings of* A: *the vector* s *must be such that* H_s *is a Hungarian matrix. Indeed,* H_s *is Hungarian if and only if* $|(H_s)_{ij}| \le 1$ *for all* $i, j = 1, 2, 3$. *This yields the following constraints on the entries of* s:

$$-1 + s_1 - s_2 \le 0, \quad s_2 - s_1 \le 0, \quad -4 + s_2 - s_3 \le 0, \quad s_3 - s_1 \le 0. \tag{2.7}$$

Now for all $\alpha \in \mathbb{R}$, $s \in \mathbb{R}^3$ *satisfies* (2.7) *if and only if* $\tilde{s} := s + \alpha[1, 1, 1]^T$ *satisfies* (2.7). *Therefore, the set* $\mathcal{S}(H) := \{s \in \mathbb{R}^3 : H_s \text{ is Hungarian}\}$ *is a prism extruded in the* $[1, 1, 1]^T$ *direction. It is not difficult to see that for any* $\alpha \in \mathbb{R}$, s *and* $\tilde{s}$ *give rise to the same scaling of* H, *so there is no loss of generality in choosing* α *such that* $s_1 = 0$. *Then the intersection of* $\mathcal{S}(H)$ *with the plane* $s_1 = 0$ *is the set of solutions to*

$$s_2 \ge -1, \quad s_2 \le 0, \quad s_2 - s_3 \le 4, \quad s_3 \le 0,$$

which is given by the quadrilateral shown in Figure 2.1(a). *The vertices* a, b, c, d *of the quadrilateral are are given by*

$$a = [0, -1, 0]^T, \quad b = [0, -1, -5]^T, \quad c = [0, 0, -4]^T, \quad d = [0, 0, 0]^T.$$

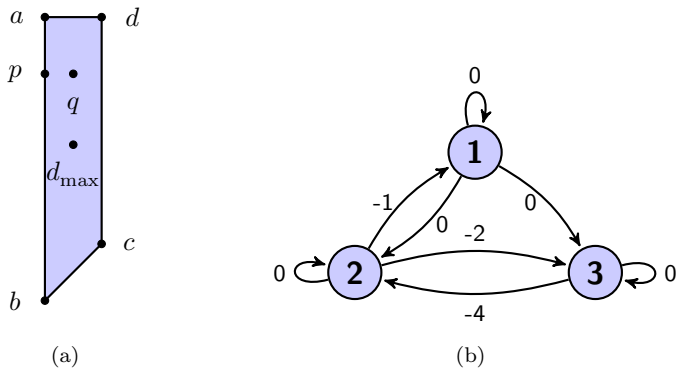
They correspond to extremal Hungarian scalings of A *given by*

$$H_a = \begin{bmatrix} 1 & \exp(-1) & 1 \\ 1 & 1 & \exp(-1) \\ 0 & \exp(-5) & 1 \end{bmatrix}, \quad H_b = \begin{bmatrix} 1 & \exp(-1) & \exp(-5) \\ 1 & 1 & \exp(-6) \\ 0 & 1 & 1 \end{bmatrix},$$
$$H_c = \begin{bmatrix} 1 & 1 & \exp(-4) \\ \exp(-1) & 1 & \exp(-6) \\ 0 & 1 & 1 \end{bmatrix}, \quad H_d = \begin{bmatrix} 1 & 1 & 1 \\ \exp(-1) & 1 & \exp(-2) \\ 0 & \exp(-4) & 1 \end{bmatrix}.$$

Each of these Hungarian scaled matrices contain precisely five entries equal to one. If we scale using any parameter from the relative interior of an edge of the quadrilateral, then we obtain a scaled matrix with exactly four entries equal to one. At the end of section 2.2 *we will see that if we take any scaling parameter from the interior of this quadrilateral, then we obtain a scaled matrix with exactly three entries equal to one. For example,* $p = [0, -1, -1]^T$ *and* $q = [0, -0.5, -1]^T$ *yield*

$$H_p = \begin{bmatrix} 1 & \exp(-1) & \exp(-1) \\ 1 & 1 & \exp(-2) \\ 0 & \exp(-4) & 1 \end{bmatrix}, \quad H_q = \begin{bmatrix} 1 & \exp(-\frac{1}{2}) & \exp(-1) \\ \exp(-\frac{1}{2}) & 1 & \exp(-\frac{5}{2}) \\ 0 & -\exp(\frac{7}{2}) & 1 \end{bmatrix}.$$

The 2-norm and 2-norm condition number of these matrices are provided in Table 2.1. *The scalings* a, b, c, d *which are taken from extreme points of the quadrilateral all result in scaled matrices with very similar condition numbers and norms. The scaling*



(a) (b)

FIG. 2.1. (a) *shows* $\mathcal{S}(H) \cap \{s_1 = 0\}$ *for the matrix* $H \in \mathbb{R}^{3\times 3}$ *of Example* 2.3 *and different scaling vectors;* (b) *shows the precedence graph* $\Gamma(\mathcal{H})$ *for* $\mathcal{H} = \mathcal{V}(H)$.

TABLE 2.1
Frobenius norm and 2*-norm condition number for the matrices of Examples* 2.3 *and* 3.11 *(for* $H_{d_{\max}}$*).*

Matrix X	A	$H\ (= H_d)$	H_a	H_b	H_c	H_p	H_q	$H_{d_{\max}}$
$\|X\|_F$	8.12e3	2.27	2.30	2.27	2.27	2.07	1.97	**1.94**
$\kappa_2(X)$	4.14e5	6.19	6.56	6.98	6.40	4.96	4.27	**4.08**

p taken from an edge of the quadrilateral results in a scaled matrix with a slightly smaller condition number and norm compared to the previous Hungarian scalings. The scaling q taken from the interior of the quadrilateral results in a scaled matrix that has a further reduced condition number and norm.

We show in Theorem 2.5 *that the set of all Hungarian pairs of a matrix, in this example the extruded quadrilateral* $\mathcal{S}(\mathcal{H})$*, is actually given by the column space of a related max-plus matrix. We also show how max-balancing provides a way to automatically select a vector from the middle of the interior of this set. Just as max-plus algebra provides a neat characterization of the set of Hungarian all pairs, it also provides the perfect framework to describe the max-balancing algorithm and prove results about the properties of max-balanced scaled matrices, as we shall see. In Table* 2.1 *the max-balancing scaling vector* $d_{\max}$*, which we show how to calculate in Example* 3.11*, performs the best at reducing the norm and condition number. We explain this performance from the improved diagonal dominance brought about by max-balancing. Theorem* 3.8 *states that the max-balanced Hungarian scaling of a matrix is optimal with respect to a particular measure of diagonal dominance.*

2.1. Set of all optimal assignments. In this section we argue that, although a matrix may have more than one optimal assignment, from the point of view of diagonal dominance, it does not matter which one we choose.

The set of all optimal assignments

$$\operatorname{oas}(\mathcal{A}) = \left\{ \pi \in \Pi(n) : \sum_{j=1}^{n} a_{\pi(j)j} = \operatorname{perm}(\mathcal{A}) \right\}$$

for $\mathcal{A} \in \mathbb{R}_{\max}^{n\times n}$ may contain several different permutations. Let $\mathrm{w}(\mathcal{A}, \pi) = \sum_{j=1}^{n} a_{\pi(j)j}$ denote the weight of the permutation $\pi \in \Pi(n)$. It is easy to show that for any

$u, v \in \mathbb{R}^n$ we have

$$\mathrm{w}\big(\mathrm{diag}_\infty(-u) \otimes \mathcal{A} \otimes \mathrm{diag}_\infty(-v), \pi\big) = \mathrm{w}(\mathcal{A}, \pi) - \sum_{i=1}^{n} u_i + v_i,$$

so that $\mathrm{oas}\big(\mathrm{diag}_\infty(-u) \otimes \mathcal{A} \otimes \mathrm{diag}_\infty(-v)\big) = \mathrm{oas}(\mathcal{A})$ (see [2, Lem. 1.6.32]). Also, for any permutation $\varpi \in \Pi(n)$ and corresponding max-plus permutation matrix $\mathcal{P}_\varpi$, we have

$$\mathrm{w}(\mathcal{P}_\varpi \otimes \mathcal{A}, \pi) = \mathrm{w}(\mathcal{A}, \varpi \circ \pi).$$

Thus if we choose a particular optimal assignment $\widetilde{\pi}$ and Hungarian pair (u, v) of $\mathcal{A}$, then the set of all optimal assignments of the scaled and reordered matrix $\mathcal{H} = \mathcal{P}_{\widetilde{\pi}} \otimes \mathrm{diag}_\infty(-u) \otimes \mathcal{A} \otimes \mathrm{diag}_\infty(-v)$ is given by

$$\mathrm{oas}(\mathcal{H}) = \{\widetilde{\pi}^{-1} \circ \pi : \pi \in \mathrm{oas}(\mathcal{A})\}, \tag{2.8}$$

and for all $\omega \in \mathrm{oas}(\mathcal{H})$ we have $\mathcal{h}_{\omega(j)j} = 0$ for $j = 1, \ldots, n$.

We are interested in quantifying the diagonal dominance of Hungarian scaled matrices, which could potentially be affected by the choice of optimal assignment. For this, we use the general row-wise diagonal dominance measure in (1.2),

$$\Delta(A) = g\left(\left(\sqrt[p]{\frac{\sum_{j \neq i} |a_{ij}|^p}{|a_{ii}|^p}}\right)_{i=1,\ldots,n}\right),$$

where $p \in [1, \infty)$ and $g : \mathbb{R}_+^n \mapsto \mathbb{R}_+$ forms a single score from the individual row p-norm scores. If we assume that g is invariant to permutations in its n arguments, then it is easy to prove that $\Delta(H)$ with H as in (2.5) does not depend on the choice of $\pi \in \mathrm{oas}(\mathcal{V}(A))$. The same is true for any equivalent columnwise diagonal dominance measure. However, as we will demonstrate in section 4, the choice of Hungarian pair (u, v) can cause large changes to different diagonal dominance measures.

2.2. Set of all Hungarian pairs. In this section we give a max-plus algebraic characterization of the set of all Hungarian pairs of a matrix. Before we can state our results we need to introduce a few more important definitions.

The *precedence graph* $\Gamma(\mathcal{A})$ of $\mathcal{A} \in \mathbb{R}_{\max}^{n \times n}$ is the weighted directed graph with vertices $\{1, \ldots, n\}$ and an edge from i to j with weight $\mathcal{a}_{ij}$ whenever $\mathcal{a}_{ij} \neq -\infty$. Equivalently, $\Gamma(\mathcal{A})$ is the graph such that $\mathcal{A}$ is the weighted adjacency matrix of $\Gamma(\mathcal{A})$, with minus infinity entries whenever there is an edge missing. See Figure 2.1(b) for an example. The *maximum cycle mean* of $\mathcal{A} \in \mathbb{R}_{\max}^{n \times n}$ is defined by

$$\lambda_{\max}(\mathcal{A}) := \max_C \mathrm{w}(C)/\mathrm{l}(C), \tag{2.9}$$

where the maximum is taken over all elementary cycles C through $\Gamma(\mathcal{A})$. Here $\mathrm{w}(C)$ is the *weight of the cycle* C, that is, the sum of the weights of its constituent edges, and $\mathrm{l}(C)$ is the *length of the cycle* C, that is, the number of edges C contains.

For clarity we denote powers of $\mathcal{A} \in \mathbb{R}_{\max}^{n \times n}$ by the $\otimes$ symbol so that, for example, $\mathcal{A}^{\otimes 3} = \mathcal{A} \otimes \mathcal{A} \otimes \mathcal{A}$. In terms of the precedence graph we have that $(\mathcal{A}^{\otimes k})_{ij}$ is equal to the weight of the maximally weighted path of length k through $\Gamma(\mathcal{A})$ from i to j.

The *Kleene star* of $\mathcal{A} \in \mathbb{R}_{\max}^{n \times n}$, denoted by $\mathcal{A}^\star$, is given by

$$\mathcal{A}^\star = \mathcal{I} \oplus \mathcal{A} \oplus \mathcal{A}^{\otimes 2} \oplus \cdots.$$

It is known that the Kleene star $\mathcal{A}^\star$ exists if and only if $\lambda_{\max}(\mathcal{A}) \leq 0$ (see [2, Prop. 1.6.10], for example). In terms of the precedence graph we have that $(\mathcal{A}^\star)_{ij}$ is equal to the weight of the maximally weighted path through $\Gamma(\mathcal{A})$ from i to j. Thus if $\lambda_{\max}(\mathcal{A}) > 0$, then $\Gamma(\mathcal{A})$ contains a positively weighted cycle, and the maximally weighted path through $\Gamma(\mathcal{A})$ from i to j will not exist as it will be possible to find paths with arbitrarily high weight. Otherwise, if $\lambda_{\max}(\mathcal{A}) \leq 0$, then

$$\mathcal{A}^\star = \mathcal{I} \oplus \mathcal{A} \oplus \mathcal{A}^{\otimes 2} \oplus \cdots \oplus \mathcal{A}^{\otimes(n-1)}.$$

Now consider a Hungarian matrix $\mathcal{H} \in \mathbb{R}_{\max}^{n\times n}$. Since the diagonal entries of $\mathcal{H}$ correspond to length one cycles of weight zero in $\Gamma(\mathcal{H})$ and no cycle can have strictly positive weight, it follows that $\lambda_{\max}(\mathcal{H}) = 0$, and so the Kleene star of a Hungarian matrix $\mathcal{H}$ always exists.

For $\mathcal{A}, \mathcal{B} \in \mathbb{R}_{\max}^{n\times n}$ with $\mathcal{B}$ having finite entries, define $\mathcal{A}/\mathcal{B} \in \mathbb{R}_{\max}^{n\times n}$ by

$$(\mathcal{A}/\mathcal{B})_{ij} = a_{ij} - b_{ij}.$$

To characterize the set of Hungarian pairs, we need a result by Butkovič and Schneider in [3], which they state for nonnegative matrices in the max-times algebra rather than max-plus matrices in the max-plus algebra, but the transformation from one to the other is very straightforward. The solution to [3, Problem 3.1] we state below is for the max-plus algebra.

THEOREM 2.4 (one-sided inequality). *For $\mathcal{A}, \mathcal{B} \in \mathbb{R}_{\max}^{n\times n}$, $\mathcal{B}$ with finite entries,*

$$\left\{ s \in \mathbb{R}^n : \operatorname{diag}_\infty(-s) \otimes \mathcal{A} \otimes \operatorname{diag}_\infty(s) \leq \mathcal{B} \right\} = \begin{cases} \operatorname{col}\big((\mathcal{A}/\mathcal{B})^\star\big) \cap \mathbb{R}^n & \text{if } \lambda_{\max}(\mathcal{A}/\mathcal{B}) \leq 0, \\ \emptyset & \text{otherwise,} \end{cases}$$

where $\operatorname{col}(\mathcal{A}) := \{\mathcal{A} \otimes \mathcal{X} : \ \mathcal{X} \in \mathbb{R}_{\max}^n\}$ *denotes the column space of* $\mathcal{A}$.

Theorem 2.4 allows a neat characterization for the set of all Hungarian pairs.

THEOREM 2.5 (set of all Hungarian pairs). *Let $\mathcal{A} \in \mathbb{R}_{\max}^{n\times n}$ with* $\operatorname{perm}(\mathcal{A}) \neq -\infty$, *and let π and (u, v) be an optimal assignment and a Hungarian pair of $\mathcal{A}$, respectively. Then the set of all Hungarian pairs* $\operatorname{Hung}(\mathcal{A})$ *of $\mathcal{A}$ is given by*

$$\operatorname{Hung}(\mathcal{A}) = \{(u + s_{\pi^{-1}}, v - s) : s \in \operatorname{col}(\mathcal{H}^\star) \cap \mathbb{R}^n\},$$

where $\mathcal{H} = \mathcal{P}_\pi \otimes \operatorname{diag}_\infty(-u) \otimes \mathcal{A} \otimes \operatorname{diag}_\infty(-v)$ *and* $(s_{\pi^{-1}})_i = s_{\pi^{-1}(i)}$.

Proof. Since $\mathcal{H}/\mathcal{O}_n = \mathcal{H}$ and $\lambda_{\max}(\mathcal{H}) = 0$, we have $\lambda_{\max}(\mathcal{H}/\mathcal{O}_n) \leq 0$. Therefore, from Theorem 2.4 we have

$$\begin{aligned} s \in \operatorname{col}(\mathcal{H}^\star) \cap \mathbb{R}^n &\iff \operatorname{diag}_\infty(-s) \otimes \mathcal{H} \otimes \operatorname{diag}_\infty(s) \leq \mathcal{O}_n \\ &\iff -s_i - u_{\pi(i)} + a_{\pi(i)j} - v_j + s_j \leq 0, \quad i, j = 1, \ldots, n, \\ &\iff a_{ij} - (u_i + s_{\pi^{-1}(i)}) - (v_j - s_j) \leq 0, \quad i, j = 1, \ldots, n, \end{aligned} \tag{2.10}$$

which is equivalent to saying that $(u + s_{\pi^{-1}}, v - s)$ is a feasible solution to the dual problem (2.2). Finally, since

$$\sum_{i=1}^n (u_i + s_{\pi^{-1}(i)} + v_i - s_i) = \sum_{i=1}^n u_i + v_i = \operatorname{perm}(\mathcal{A}),$$

the pair $(u + s_{\pi^{-1}}, v - s)$ must also be an optimal solution to (2.2) and therefore a Hungarian pair of $\mathcal{A}$.

Conversely, suppose that (u', v') is a Hungarian pair of $\mathcal{A}$, and let $\mathcal{H}' = \mathcal{P}_\pi \otimes \operatorname{diag}_\infty(-u') \otimes \mathcal{A} \otimes \operatorname{diag}_\infty(-v')$. From Theorem 2.1 we have $h_{ij}, h'_{ij} \le 0$ and $h_{ii} = h'_{ii} = 0$ for all $i, j = 1, \dots, n$. Therefore,

$$h_{ii} = h'_{ii} \iff a_{\pi(i),i} - u_{\pi(i)} - v_i = a_{\pi(i),i} - u'_{\pi(i)} - v'_i \iff u'_{\pi(i)} - u_{\pi(i)} = v_i - v'_i$$

so that $(u', v') = (u + s_{\pi^{-1}}, v - s)$ for some $s \in \mathbb{R}^n$. Also,

$$h'_{ij} \le 0 \iff a_{\pi(i),i} - u'_{\pi(i)} - v'_i \le 0 \iff a_{ij} - (u_i + s_{\pi^{-1}(i)}) - (v_j - s_j) \le 0$$

for $i, j = 1, \dots, n$, which by (2.10) is equivalent to $s \in \operatorname{col}(\mathcal{H}^\star) \cap \mathbb{R}^n$. ∎

The following theorem is equivalent to results presented in [21], except it is stated for the special case of Hungarian matrices. This result relates the secondary scaling method of Olschowka and Neumaier [16, Alg. 4.2] to the geometric characterization of the set of all Hungarian scalings given in Theorem 2.5 and illustrated in Example 2.3. When $\operatorname{col}(\mathcal{H}^\star)$ is of dimension n, their algorithm returns a scaling vector from the relative interior of $\operatorname{col}(\mathcal{H}^\star)$. Our max-balancing approach also returns a scaling vector from the relative interior of $\operatorname{col}(\mathcal{H}^\star)$ but goes further by choosing this vector to optimize the diagonal dominance of the scaled matrix.

THEOREM 2.6. *Let $\mathcal{A}$ and $\mathcal{H}$ be as in Theorem 2.5. For any s in the relative interior of $\operatorname{col}(\mathcal{H}^\star)$, the Hungarian matrix $\operatorname{diag}_\infty(-s) \otimes \mathcal{H} \otimes \operatorname{diag}_\infty(s)$ has exactly k entries equal to zero with all other entries strictly less than zero, where*

$$k = |\{(i, j) : \exists\ \pi \in \operatorname{oas}(\mathcal{A}) \text{ with } \pi(i) = j\}|.$$

Moreover, this is the least possible number of zero entries for a Hungarian scaling and reordering of $\mathcal{A}$.

REMARK 2.7 (reducible case). *If the matrix $\mathcal{A}$ is irreducible, i.e., if $\Gamma(\mathcal{A})$ is strongly connected, then the Kleene star $\mathcal{H}^\star$ will have finite entries. As a result, $\operatorname{col}(\mathcal{H}^\star)$ will only contain vectors with finite entries apart from the vector with all entries equal to $-\infty$. This is not the case when $\mathcal{A}$ is reducible. Indeed, for $\mathcal{A} = \mathcal{H} = \mathcal{H}^\star = \left[\begin{smallmatrix} 0 & 0 \\ -\infty & 0 \end{smallmatrix}\right]$, we have that $s_p = \left[\begin{smallmatrix} 0 \\ p \end{smallmatrix}\right] \in \operatorname{col}(\mathcal{H}^\star)$ for $p \in [-\infty, 0]$. By scaling with the vector s_p, $\operatorname{diag}_\infty(-s_p) \otimes \mathcal{A} \otimes \operatorname{diag}_\infty(s_p) = \left[\begin{smallmatrix} 0 & -p \\ -\infty & 0 \end{smallmatrix}\right]$, we can make the $(1, 2)$ entry arbitrarily small. However, this sort of scaling is not useful in numerical linear algebra problems, as it is always more efficient to treat the separate irreducible components independently.*

2.3. Hungarian algorithm. In order to Hungarian scale a matrix $\mathcal{A} \in \mathbb{R}_{\max}^{n \times n}$ we must compute an optimal assignment and Hungarian pair for $\mathcal{A}$. The best known algorithms for this have worst case cost $O\big(n\tau + n^2 \log n\big)$, where τ is the number of finite entries in $\mathcal{A}$ [10] (recall that finite entries are the max-plus equivalent of nonzero entries). However, in practical numerical examples it is found that optimal assignment algorithms such as Kuhn's Hungarian algorithm [9], the successive shortest paths algorithm [17], and the auction algorithm [12] have run-times roughly linear in the number of finite entries in the matrix. It is only for some very special examples that the worst case complexity bound is attained.

Typically the space $\operatorname{col}(\mathcal{H}^\star)$ contains more than one possible scaling, so that different optimal assignment algorithms may return different Hungarian pairs, which result in different scalings that may have different properties. Theorem 2.5 tells us that these different scalings are all related by similarity scalings. Moreover, if we

suppose that $\mathcal{A}$ has been Hungarian scaled and reordered into a Hungarian matrix $\mathcal{H}$, then Theorem 2.5 tells us that for $s \in \mathrm{col}(\mathcal{H}^\star)$, $\mathcal{H}_s = \mathrm{diag}_\infty(-s) \otimes \mathcal{H} \otimes \mathrm{diag}_\infty(s)$ is also a Hungarian matrix (i.e., $\mathcal{H}_s$ is obtained from $\mathcal{H}$ by diagonal similarity scaling). In the remainder of this paper, we consider one possible strategy for choosing the diagonal scaling parameter s, namely max-balancing.

3. Max-balancing. A matrix $A \in \mathbb{C}^{n\times n}$ is *p-balanced* for some $p \in [1,\infty)$ if

$$\sum_{j=1}^{n} |a_{ij}|^p = \sum_{j=1}^{n} |a_{ji}|^p, \qquad i = 1,\dots,n,$$

and ∞-*balanced* if $\max_{1\le j\le n} |a_{ij}| = \max_{1\le j\le n} |a_{ji}|$, $i = 1,\dots,n$. A matrix $A \in \mathbb{C}^{n\times n}$ is max-balanced if for any nontrivial subset $\mathcal{J} \subset \{1,\dots,n\}$ we have

$$\max_{i\in\mathcal{J}, j\notin\mathcal{J}} |a_{ij}| = \max_{i\notin\mathcal{J}, j\in\mathcal{J}} |a_{ij}|; \tag{3.1}$$

see [20]. A matrix being max-balanced is a stronger condition than being balanced in the ∞-norm sense. Indeed, the matrix

$$A = \begin{bmatrix} 0 & 10 & 0 & 0 \\ 10 & 0 & 1 & 0 \\ 0 & 0.1 & 0 & 10 \\ 0 & 0 & 10 & 0 \end{bmatrix},$$

taken from [20], is ∞-balanced but not max-balanced since (3.1) is not satisfied for $\mathcal{J} = \{1,2\}$. Note that Hermitian or symmetric matrices are automatically max-balanced.

3.1. Properties of max-balanced matrices. It is shown in [18] that for any irreducible $A \in \mathbb{C}^{n\times n}$ and $p \in [1,\infty)$ there exists a unique p-balanced matrix B_p diagonally similar to A,

$$B_p = \mathrm{diag}(d_p)^{-1} A\, \mathrm{diag}(d_p), \tag{3.2}$$

where the scaling parameter $d_p \in \mathbb{R}^n_+$ is unique up to scalar multiplication. Schneider and Schneider show that a similar result holds for an irreducible nonnegative matrix and max-balancing. It is trivial to rephrase their result for complex matrices.

THEOREM 3.1 ([20], Corollary 9). *For any irreducible $A \in \mathbb{C}^{n\times n}$ there exists a unique max-balanced matrix M diagonally similar to A,*

$$M = \mathrm{diag}(d_{\max})^{-1} A\, \mathrm{diag}(d_{\max}),$$

where the scaling parameter $d_{\max} \in \mathbb{R}^n_+$ is unique up to scalar multiple.

We define the *Frobenius p-norm* of $A \in \mathbb{C}^{n\times n}$ by

$$\|A\|_{F_p} = \|\mathrm{vec}(A)\|_p = \Big(\sum_{i,j=1}^{n} |a_{ij}|^p \Big)^{\frac{1}{p}}.$$

For any irreducible $A \in \mathbb{C}^{n\times n}$ and $p \in [1,\infty)$, Osborne shows that [18, Lem. 2(iii)]

$$\min_{d\in\mathbb{R}^n_+} \|\mathrm{diag}(d)^{-1} A\, \mathrm{diag}(d)\|_{F_p} = \|B_p\|_{F_p}, \tag{3.3}$$

where B_p is the unique p-balanced matrix diagonally similar to A.

An irreducible matrix $A \in \mathbb{C}^{n\times n}$ may be diagonally similar to a range of different ∞-balanced matrices, but it is diagonally similar to a unique p-balanced scaling B_p with $p \in [1,\infty)$ and a unique max-balanced scaling M. The next result shows that we can think of the max-balanced scaling of A as the limit of its p-balanced scaling in the limit $p \to \infty$.

THEOREM 3.2. *Let A be irreducible, and let M and B_p with $p \in [1,\infty)$ be the max-balanced and p-balanced matrices, respectively, diagonally similar to A. Then* $\lim_{p\to\infty} B_p = M$.

Proof. The function $f : \mathbb{C}^{n\times n} \mapsto \mathbb{R}_+$ defined by

$$f(B) = \max_{\mathcal{I}\subset\{1,\dots,n\}} \Big| \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |b_{ij}| - \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |b_{ji}| \Big|$$

is continuous and $f(B) = 0$ if and only if B is max-balanced. It follows from Theorem 3.1 that if B is a similarity scaling of A and $f(B) = 0$, then $B = M$.

Since B_p is p-balanced, for any nontrivial subset $\mathcal{I} \subset \{1,\dots,n\}$, we have

$$\sum_{i\in\mathcal{I}} \sum_{j=1}^{n} |(B_p)_{ij}|^p = \sum_{i\in\mathcal{I}} \sum_{j=1}^{n} |(B_p)_{ji}|^p,$$

and removing any entries that appear on both sides yields

$$\sum_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ij}|^p = \sum_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ji}|^p.$$

The left-hand side of this expression satisfies

$$\big(\max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ij}| \big)^p \le \sum_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ij}|^p \le n^2 \big(\max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ij}| \big)^p,$$

and similarly for the right-hand side so that

$$n^{-2} \big(\max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ji}| \big)^p \le \big(\max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ij}| \big)^p \le n^2 \big(\max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ji}| \big)^p.$$

Taking logs and dividing by p yields

$$\Big| \max_{i\in\mathcal{I}, j\notin\mathcal{I}} \log |(B_p)_{ij}| - \max_{i\in\mathcal{I}, j\notin\mathcal{I}} \log |(B_p)_{ji}| \Big| \le \frac{2\log n}{p}. \tag{3.4}$$

For all $p \in [1,\infty)$, we have from (3.3) that

$$\max_{1\le i,j\le n} |(B_p)_{ij}| \le \|B_p\|_{F_p} \le \|A\|_{F_p} \le n^2 \max_{1\le i,j\le n} |a_{ij}|.$$

Also, using the fact that for $a, b \in \mathbb{R}_+$, $|a-b| \le |\log a - \log b| \max\{a,b\}$, inequality (3.4) becomes

$$\Big| \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ij}| - \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(B_p)_{ji}| \Big| \le \frac{2n^2 \log n \max_{1\le i,j\le n} |a_{ij}|}{p}.$$

Therefore, $\lim_{p\to\infty} f\big(B_p\big) = 0$ so that $\lim_{p\to\infty} B_p = M$. ∎

For $A \in \mathbb{C}^{n\times n}$ define $\operatorname{sort}\big(\operatorname{vec}(|A|)\big)$ to be the vector containing the absolute values of all of the n^2 entries in A sorted in decreasing order. Now define the *lexicographic partial order* $\prec_L$ on $\mathbb{C}^{n\times n}$ by $A \prec_L B$ if and only if $\operatorname{sort}\big(\operatorname{vec}(|A|)\big) \ne \operatorname{sort}\big(\operatorname{vec}(|B|)\big)$ and the first position i where these two vectors disagree satisfies $\big(\operatorname{sort}(\operatorname{vec}(|A|))\big)_i < \big(\operatorname{sort}\big(\operatorname{vec}(|B|)\big)\big)_i$.

LEMMA 3.3. *Let $A, B \in \mathbb{C}^{n\times n}$. Then $A \prec_L B$ if and only if there exists $p' \in \mathbb{R}$ such that $\|A\|_{F_p} < \|B\|_{F_p}$ for all $p > p'$.*

Proof. If $A \prec_L B$, then there exists i such that $\big(\text{sort}(\text{vec}(|A|))\big)_j = \big(\text{sort}\big(\text{vec}(|B|)\big)\big)_j$ for $j = 1, \dots, i-1$ and $\big(\text{sort}(\text{vec}(|A|))\big)_i < \big(\text{sort}\big(\text{vec}(|B|)\big)\big)_i$. Therefore,

$$\|B\|_{F_p}^p \geq \big(\text{sort}\big(\text{vec}(|B|)\big)\big)_i^p + \sum_{j=1}^{i-1} \big(\text{sort}(\text{vec}(|A|))\big)_j^p,$$

$$\|A\|_{F_p}^p \leq (n-i+1)\big(\text{sort}\big(\text{vec}(|A|)\big)\big)_i^p + \sum_{j=1}^{i-1} \big(\text{sort}(\text{vec}(|A|))\big)_j^p$$

so that $\|A\|_{F_p} < \|B\|_{F_p}$ whenever $(n-i+1)\big(\text{sort}\big(\text{vec}(|A|)\big)\big)_i^p \leq \big(\text{sort}\big(\text{vec}(|B|)\big)\big)_i^p$, which is satisfied for all $p > p'$ with

$$p' = \frac{\log(n-i+1)}{\log\big(\text{sort}(\text{vec}(|B|))_i\big) - \log\big(\text{sort}(\text{vec}(|A|))_i\big)}. \qquad \square$$

The next result by Rothblum, Schneider, and Schneider in [19, Thm. 8] is given in terms of weighted graphs and reweighing potentials. It is trivial to rephrase the result, as we have done, in terms of similarity scaling of complex matrices.

THEOREM 3.4 ([19], Theorem 8). *Let $A \in \mathbb{C}^{n\times n}$ be irreducible, and let M be the unique max-balanced similarity scaling of A. Then*

$$M \prec_L \text{diag}(d)^{-1} A \,\text{diag}(d)$$

for all $d \in \mathbb{R}_+^n$ such that $\text{diag}(d)^{-1} A \,\text{diag}(d) \neq M$.

The following corollary follows immediately from Theorem 3.4. Note the resemblance to (3.3).

COROLLARY 3.5. *Let $A \in \mathbb{C}^{n\times n}$ be irreducible, and let M be the unique max-balanced similarity scaling of A. Then for all $d \in \mathbb{R}_+^n$ such that $\text{diag}(d)^{-1} A \,\text{diag}(d) \neq M$, there exists $p' \in \mathbb{R}_+$ such that for all $p > p'$*

$$\|M\|_{F_p} < \|\text{diag}(d)^{-1} A \,\text{diag}(d)\|_{F_p}.$$

In Example 3.11, we compute the max-balanced Hungarian scaling for the matrix A of Example 2.3. Table 2.1 displays the Frobenius norm of the max-balanced Hungarian scaling of A as well as the Frobenius norms of all of the other Hungarian scalings of A that we considered Example 2.3. Note that the max-balanced Hungarian scaling has the smallest Frobenius norm out of all of these Hungarian scalings. In this example, we see that max-balancing not only minimizes the Frobenius p-norm in the limit as p tends to ∞ but also does a good job of reducing the standard Frobenius 2-norm. This behavior agrees with the findings of our numerical experiments on diagonal dominance presented in section 4.1.

3.2. Properties of max-balanced Hungarian scaled and reordered matrices. A max-balanced similarity scaling preserves the Hungarian property, as we now show.

THEOREM 3.6. *Let $H \in \mathbb{C}^{n\times n}$ be an irreducible Hungarian matrix, and let $d_{\max} \in \mathbb{R}^n$ be such that $M = \text{diag}(d_{\max})^{-1} H \,\text{diag}(d_{\max})$ is the max-balanced scaling of H. Then M is also a Hungarian matrix.*

Proof. Recall that $H \in \mathbb{C}^{n\times n}$ is a Hungarian matrix if and only if $|h_{ij}| \le 1$ and $|h_{ii}| = 1$ for all $i, j = 1, \dots, n$. Similarity scaling has no effect on diagonal entries, so we only need to verify that $|m_{ij}| \le 1$ for all $i, j = 1, \dots, n$. Suppose instead that $|m_{ij}| > 1$ for some i, j. Then $H \prec_L M$, and this contradicts Theorem 3.4. ∎

Therefore, for an irreducible matrix $A \in \mathbb{C}^{n\times n}$, after computing a Hungarian scaling and reordering, $H = P_\pi \mathrm{diag}(d_L) A \,\mathrm{diag}(d_R)$, we can apply a further similarity scaling to obtain the max-balanced Hungarian scaled matrix

$$M = \mathrm{diag}(d_{\max})^{-1} P_\pi \mathrm{diag}(d_L) A \,\mathrm{diag}(d_R) \mathrm{diag}(d_{\max}),$$

which satisfies the conditions $|m_{ij}| \le 1$ and $|m_{ii}| = 1$ for all $i, j = 1, \dots, n$ and

$$\max_{i\in\mathcal{I}, j\notin\mathcal{I}} |m_{ij}| = \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |m_{ji}|$$

for any nontrivial subset $\mathcal{I} \subset \{1, \dots, n\}$. The next theorem says that the max-balanced reordered Hungarian scaling of A is unique up to multiplication on the left by permutation matrices which switch between different choices of optimal assignment.

THEOREM 3.7. *Let $A \in \mathbb{C}^{n\times n}$ be irreducible, and let π_k and (u_k, v_k), $k = 1, 2$, be optimal assignments and Hungarian pairs of $\mathcal{A} = \mathcal{V}(A)$, respectively, so that*

$$H_k = P_{\pi_k} \mathrm{diag}(\exp(-u_k)) A \,\mathrm{diag}(\exp(-v_k)), \quad k = 1, 2,$$

are two possibly distinct reordered Hungarian scalings of A. Then the max-balanced similarity scalings $M_k = \mathrm{diag}({d_{\max}}^{(k)})^{-1} H_k \,\mathrm{diag}({d_{\max}}^{(k)})$ of H_k, $k = 1, 2$, are related by $M_2 = P_\pi M_1$, where $\pi = \pi_1^{-1} \circ \pi_2$.

Proof. First note that $P_\pi M_1$ is a diagonal scaling of M_2 since

$$\begin{aligned} P_\pi M_1 = {} & \left(P_\pi \mathrm{diag}({d_{\max}}^{(2)}) \mathrm{diag}({d_{\max}}^{(1)})^{-1} P_\pi^T\right) \\ & \times \left(P_{\pi_2} \mathrm{diag}(\exp(-u_1 + u_2)) P_{\pi_2}^T\right) M_2 \mathrm{diag}(\exp(-v_1 + v_2)) \mathrm{diag}({d_{\max}}^{(1)}) \mathrm{diag}({d_{\max}}^{(2)})^{-1}. \end{aligned}$$

From Theorem 3.6 we know that the M_k are both Hungarian scaled matrices with $|(M_k)_{ij}| \le 1$ and $|(M_k)_{ii}| = 1$ for all $i, j = 1, \dots, n$. It follows from (2.8) that

$$\{\mathrm{id}, \pi\} \subset \mathrm{oas}(M_1), \quad \{\pi^{-1}, \mathrm{id}\} \subset \mathrm{oas}(P_\pi M_1).$$

Thus $P_\pi M_1$ is a Hungarian scaling of M_2, and by Theorem 2.5 $P_\pi M_1$ must be a similarity scaling of M_2.

We now show that $P_\pi M_1$ is max-balanced. For $\mathcal{I} \subset \{1, \dots, n\}$ suppose that $\pi(\mathcal{I}) = \mathcal{I}$; then since M_1 is max-balanced we have

$$\max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(P_\pi M_1)_{ij}| = \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(M_1)_{ij}| = \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(M_1)_{ji}| = \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(P_\pi M_1)_{ji}|.$$

Now suppose that $\pi(\mathcal{I}) \ne \mathcal{I}$; then there exist $k \in \mathcal{I}$ such that $\pi(k) \notin \mathcal{I}$ and $\ell \notin \mathcal{I}$ such that $\pi(\ell) \in \mathcal{I}$. Since $\{\mathrm{id}, \pi\} \subset \mathrm{oas}(M_1)$ we have $|(M_1)_{ii}| = 1$ for $i = 1, \dots, n$, and since $|(M_1)_{ij}| \le 1$ for all $i, j = 1, \dots, n$ we have

$$\begin{aligned} \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(P_\pi M_1)_{ij}| &= |(P_\pi M_1)_{k\pi(k)}| = |(M_1)_{\pi(k)\pi(k)}| = 1, \\ \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |(P_\pi M_1)_{ji}| &= |(P_\pi M_1)_{\ell\pi(\ell)}| = |(M_1)_{\pi(\ell)\pi(\ell)}| = 1. \end{aligned}$$

Thus $\max_{i\in\mathcal{I},j\notin\mathcal{I}} |(P_\pi M_1)_{ij}| = \max_{i\in\mathcal{I},j\notin\mathcal{I}} |(P_\pi M_1)_{ji}|$ for any nontrivial subset $\mathcal{I}$ so that $P_\pi M_1$ is max-balanced.

Finally, by Theorem 3.1, since $P_\pi M_1$ is a similarity scaling of M_2 and they are both max-balanced, we must have $M_2 = P_\pi M_1$. ∎

For $A \in \mathbb{C}^{n\times n}$, define the following measure of diagonal dominance:

$$\Delta_p(A) = \sqrt[p]{\sum_{i=1}^{n} \frac{\sum_{j\neq i} |a_{ij}|^p}{|a_{ii}|^p}}, \tag{3.5}$$

with $\Delta_p(A) = +\infty$ if $a_{ii} = 0$ for any $i = 1, \ldots, n$. Since we are only working with irreducible matrices, the case where both the numerator and denominator in (3.5) are zero can be ignored. This measure is a special case of (1.2) that compares the p-norm of the off-diagonal elements to the diagonal element for each row and then amalgamates their scores into a single score for the whole matrix by taking the p-norm of the individual row scores.

For $A, B \in \mathbb{C}^{n\times n}$ we also define the ordering $\prec_\Delta$ by $A \prec_\Delta B$ if and only if there exists p' such that for all $p \geq p'$ we have $\Delta_p(A) < \Delta_p(B)$. The ordering $A \prec_\Delta B$ implies that A is more diagonally dominant than B when viewed through the p-norm for sufficiently large p. Note that if A and B have identical constant diagonal entries, i.e., if there exists $\alpha \in \mathbb{C}$ such that $a_{ii} = b_{ii} = \alpha$ for all $i = 1, \ldots, n$, then $A \prec_\Delta B$ if and only if $A \prec_L B$, where $\prec_L$ is the lexicographic partial order introduced before Theorem 3.4. However, if A and B do not have identical constant diagonal entries, then the orderings $\prec_\Delta$ and $\prec_L$ are not equivalent.

The next theorem shows that max-balanced Hungarian scaled and reordered matrices are optimal with respect to the ordering $\prec_\Delta$. In other words, they are the most diagonally dominant diagonal scaling and reordering of A, with respect to the measure Δ_p, as p tends to ∞.

THEOREM 3.8. *Let $A \in \mathbb{C}^{n\times n}$ be irreducible, and let*

$$M = P_\pi \mathrm{diag}\big(\exp(-u)\big) A \,\mathrm{diag}\big(\exp(-v)\big)$$

be a max-balanced Hungarian scaling and reordering of A. Then for any permutation $\varpi \in \Pi(n)$ and any nonsingular diagonal matrices $D_1, D_2 \in \mathbb{R}^{n\times n}$, we have

$$M \prec_\Delta B, \qquad B = P_\varpi D_1 A D_2, \tag{3.6}$$

unless $\varpi \in \mathrm{oas}(\mathcal{A})$ and $B = \mathrm{diag}(t) P_{\pi^{-1}\circ\varpi} M$ for some $t \in \mathbb{R}^n$, in which case B is a row scaling of a max-balanced Hungarian scaling of A. Moreover,

$$M \prec_\Delta B \text{ and } M^T \prec_\Delta B^T, \qquad B = P_\varpi D_1 A D_2, \tag{3.7}$$

unless $\varpi \in \mathrm{oas}(\mathcal{A})$ and $B = \alpha P_{\pi^{-1}\circ\varpi} M$ for some $\alpha \in \mathbb{R}$, in which case B is a scalar multiple of a max-balanced Hungarian scaling of A.

Proof. Since M is a Hungarian scaled and reordered matrix, we have $|m_{ij}| \leq |m_{ii}|$ for all $i, j = 1, \ldots, n$ and therefore $\Delta_p(M) \leq (n^2 - n)^{1/p}$, where the upper bound converges to 1 as p tends to ∞.

First suppose that there exist i, j such that $|b_{ij}| > |b_{ii}|$; then $\Delta_p(B) \geq |b_{ij}|/|b_{ii}| > 1$ and therefore we have the result $M \prec_\Delta B$.

By irreducibility each row of B must contain a nonzero entry. Now suppose that $|b_{ij}| \leq |b_{ii}|$ for all $i, j = 1, \ldots, n$, and let $b \in \mathbb{R}^n$ be the diagonal of B. It follows

that each entry of b must be nonzero. Then $H = \operatorname{diag}(b)^{-1}B$ satisfies $|h_{ij}| \le 1$ and $|h_{ii}| = 1$ for all $i, j = 1, \dots, n$. Therefore, H is a Hungarian scaling and reordering of A and ϖ must be an optimal assignment of $\mathcal{A}$, where $\mathcal{A} = \mathcal{V}(A)$.

Since ϖ is an optimal assignment of $\mathcal{A}$, it follows from arguments made in the proof of Theorem 3.7 that the matrix

$$M' = P_{\pi^{-1}\circ\varpi}M = P_{\varpi}\operatorname{diag}\big(\exp(-u)\big)A\operatorname{diag}\big(\exp(-v)\big) \tag{3.8}$$

is also a max-balanced Hungarian scaling and reordering of A. From (3.7) and (3.8) it is also clear that M' is a diagonal scaling of B; i.e., $B = \operatorname{diag}(s)M'\operatorname{diag}(f)$ for some $f, s \in \mathbb{R}^n$.

Using the fact that $|m'_{ii}| = |m_{ii}| = 1$ for all $i = 1, \dots, n$, we have

$$\big(\Delta_p(B)\big)^p = \sum_{i=1}^{n} \frac{\sum_{j\neq i} |m'_{ij}s_if_j|^p}{|m'_{ii}s_if_i|^p} = \sum_{i=1}^{n}\sum_{j\neq i}\left|m'_{ij}\frac{f_j}{f_i}\right|^p = \|\operatorname{diag}(f)^{-1}M'\operatorname{diag}(f)\|_{F_p}^p - n$$

and $\big(\Delta_p(M)\big)^p = \|M'\|_{F_p}^p - n$, where we have used the fact that $\|M'\|_{F_p}^p = \|M\|_{F_p}^p$, which follows from $M' = P_{\pi^{-1}\circ\varpi}M$. Corollary 3.5 states that there exists $p' > 0$ such that for all $p > p'$ we have $\|M'\|_{F_p}^p < \|\operatorname{diag}(f)^{-1}M'\operatorname{diag}(f)\|_{F_p}^p$ unless

$$\operatorname{diag}(f)^{-1}M'\operatorname{diag}(f) = M'. \tag{3.9}$$

In the case that $\|M'\|_{F_p}^p < \|\operatorname{diag}(f)^{-1}M'\operatorname{diag}(f)\|_{F_p}^p$, we clearly have the result $M \prec_\Delta B$. Next we will deal with the case when $\|M'\|_{F_p}^p \ge \|\operatorname{diag}(f)^{-1}M'\operatorname{diag}(f)\|_{F_p}^p$, i.e., when (3.9) holds. Suppose that $f_i \neq f_j$ for some $i, j \in \{1, \dots, n\}$. Then by irreducibility of M there exists a sequence $\sigma(1), \sigma(2), \dots, \sigma(\ell)$ with $\sigma(1) = i$ and $\sigma(\ell) = j$, such that $m_{\sigma(k),\sigma(k+1)} \neq 0$ for $k = 1, \dots, \ell - 1$. Since $f_{\sigma(1)} \neq f_{\sigma(\ell)}$, there must be at least one $k \in \{1, \dots, \ell - 1\}$ such that $f_{\sigma(k)} \neq f_{\sigma(k+1)}$ and we have

$$\big(\operatorname{diag}(f)^{-1}M'\operatorname{diag}(f)\big)_{\sigma(k),\sigma(k+1)} = m'_{\sigma(k),\sigma(k+1)}f_{\sigma(k+1)}/f_{\sigma(k)} \neq m'_{\sigma(k),\sigma(k+1)},$$

which violates condition (3.9). Therefore, f_i is independent of i and scaling the columns by f is equivalent to scaling the whole matrix by the scalar f_1 so that

$$B = f_1\operatorname{diag}(s)M' = \operatorname{diag}(t)P_{\pi^{-1}\circ\varpi}M,$$

where $t = f_1 s$.

For the second part of the proof, note that comparing the transposed matrices B^T and M^T is equivalent to working with the columnwise version of Δ_p. However, we cannot simply take the transpose of (3.6) as it will not be compatible with the presupposed form $B = P_\varpi D_1 A D_2$, which requires the permutation matrix to act on the rows and not the columns. Instead, following the same argument as above, we find that $M^T \prec_\Delta B^T$ with $B = P_\varpi D_1 A D_2$ unless $\varpi \in \operatorname{oas}(\mathcal{A})$ and $B^T = \operatorname{diag}(t_{col})M^T P_{\varpi^{-1}\circ\pi}$ for some $t_{col} \in \mathbb{R}^n$, in which case B^T is a row scaling of the transpose of a max-balanced Hungarian scaling of A. Now if $M \prec_\Delta B$ and $M^T \prec_\Delta B^T$, then there exist $t_{row}, t_{col} \in \mathbb{R}^n$ such that

$$B = \operatorname{diag}(t_{row})P_{\pi^{-1}\circ\varpi}M, \quad B^T = \operatorname{diag}(t_{col})M^T P_{\varpi^{-1}\circ\pi},$$

which implies $\operatorname{diag}(t_{row})\big(P_{\pi^{-1}\circ\varpi}M\big) = \big(P_{\pi^{-1}\circ\varpi}M\big)\operatorname{diag}(t_{col})$, and since $P_{\pi^{-1}\circ\varpi}M$ is irreducible, this is only possible if $(t_{row})_i$ and $(t_{col})_i$ are the same constants that do not depend on i. In this case $B = \alpha P_{\pi^{-1}\circ\varpi}M$, where $\alpha = (t_{row})_1$. ∎

3.3. Max-balancing algorithm. Schneider and Schneider's description of the max-balancing algorithm in [20] is purely in terms of the precedence graph of the matrix. Our description of the algorithm is in terms of max-plus matrices.

A max-plus matrix $\mathcal{A} \in \mathbb{R}_{\max}^{n\times n}$ is max-balanced if for any nontrivial subset $\mathcal{J} \subset \{1,\dots,n\}$ we have

$$\max_{i\in\mathcal{J},j\notin\mathcal{J}} a_{ij} = \max_{i\notin\mathcal{J},j\in\mathcal{J}} a_{ij}.$$

Hence $A \in \mathbb{C}^{n\times n}$ is max-balanced if and only if $\mathcal{A} = \mathcal{V}(A) \in \mathbb{R}_{\max}^{n\times n}$ is max-balanced.

To describe the max-balancing algorithm, we need the notion of *subeigenvectors* for max-plus matrices. For $\mathcal{A} \in \mathbb{R}_{\max}^{n\times n}$ and $\beta \in \mathbb{R}_{\max}$, a vector $\chi \in \mathbb{R}_{\max}^{n}$ with at least one finite entry satisfying $\mathcal{A} \otimes \chi \le \beta \otimes \chi$ is called a subeigenvector of $\mathcal{A}$ associated with β. Subeigenvectors will be used to define the max-balancing similarity scaling so they should have finite entries. The existence of subeigenvectors with finite entries is addressed in the next lemma (see [2, Thm. 1.6.18(a)]). Here $\lambda_{\max}(\mathcal{A})$ is the maximum cycle mean of $\mathcal{A}$ defined in (2.9).

LEMMA 3.9. *Let $\mathcal{A} \in \mathbb{R}_{\max}^{n\times n}$ and $\beta \in \mathbb{R}_{\max}$. Then $\mathcal{A} \otimes x \le \beta \otimes x$ has a finite solution $x \in \mathbb{R}^n$ if and only if $\beta \ge \lambda_{\max}(\mathcal{A})$ and $\beta > -\infty$.*

We say that an elementary cycle C is *critical* in the precedence graph of $\mathcal{A}$ if $\mathrm{w}(C)/\mathrm{l}(C) = \lambda_{\max}(\mathcal{A})$. We are now ready to describe the max-balancing algorithm.

ALGORITHM 3.10 (max-balancing). *Given an irreducible matrix $\mathcal{A} \in \mathbb{R}_{\max}^{n\times n}$, this algorithm returns $d_{\max} \in \mathbb{R}^n$ such that $\mathrm{diag}_\infty(-d_{\max}) \otimes \mathcal{A} \otimes \mathrm{diag}_\infty(d_{\max})$ is max-balanced.*

1 Set $\mathcal{A} = \mathcal{V}(A)$, $t = 1$, $m_0 = n$, $f_1 = \mathrm{id}$.
2 Let $\mathcal{A}_1 \in \mathbb{R}_{\max}^{n\times n}$ be such that $(\mathcal{A}_1)_{ij} = a_{ij}$ if $i \ne j$ and $(\mathcal{A}_1)_{ii} = -\infty$.
3 Compute $\beta_1 := \lambda_{\max}(\mathcal{A}_1)$ with critical cycle C_1.
4 Compute a subeigenvector $s_1 \in \mathbb{R}^n$ of $\mathcal{A}_1$ associated with β_1.
5 Let $m_1 := m_0 + 1 -$ number of vertices in C_1.
6 while $m_t > 1$
7 $\quad t = t + 1$
8 $\quad \mathcal{S}_t = \mathrm{diag}_\infty(-s_{t-1}) \otimes \mathcal{A}_{t-1} \otimes \mathrm{diag}_\infty(s_{t-1})$
9 $\quad$ Let $f_t : \{1,\dots,m_{t-2}\} \mapsto \{1,\dots,m_{t-1}\}$ be such that $f_t(i) = f_t(j)$ if and only if i and j are both vertices of C_{t-1}. Let $\mathcal{A}_t \in \mathbb{R}_{\max}^{m_{t-1}\times m_{t-1}}$ be such that $(\mathcal{A}_t)_{\ell p} = \begin{cases} -\infty \text{ if } \ell = p, \\ \max\{(\mathcal{S}_t)_{ij} :\ f_t(i) = \ell, f_t(j) = p\} \text{ otherwise.} \end{cases}$
10 $\quad$ Compute $\beta_t := \lambda_{\max}(\mathcal{A}_t)$ with critical cycle C_t.
11 $\quad$ Compute a subeigenvector $s_t \in \mathbb{R}^{m_t}$ of $\mathcal{A}_t$ associated with β_t.
12 $\quad m_t = m_{t-1} + 1 -$ number of nodes in C_t
13 end
14 $d_{\max} = s_1(f_1) + s_2(f_2 \circ f_1) + \cdots + s_t(f_t \circ \cdots \circ f_1)$.

Note that since diagonal similarities do not affect diagonal entries of the matrix they are applied to, there is no harm in setting the diagonal entries of $\mathcal{A}$ to $-\infty$ in line 2 of Algorithm 3.10. We say that the matrix $\mathcal{A}_t$ on line 9 is a *contraction* of $\mathcal{S}_t$ with respect to the projection f_t, which we denote by $\mathcal{A}_t = \mathrm{contr}(\mathcal{S}_t, f_t)$. Since the diagonal entries of the matrices $\mathcal{A}_t$ are equal to $-\infty$, the number of nodes in the critical cycles C_t is always strictly larger than 1, so the size of the matrix $\mathcal{A}_t$ decreases at each step. It is then easy to see that the algorithm terminates after at most n steps.

That the cycle means β_t are all finite follows from the fact that any contraction

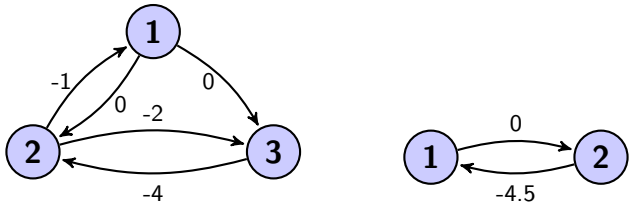


FIG. 3.1. (a) *is the precedence graph of* $\mathcal{A}_1$ *in* (3.10)*, and* (b) *is that of* $\mathcal{A}_2$ *in* (3.11).

of an irreducible matrix is also an irreducible matrix so that while $m_t > 1$ the graph $\Gamma(\mathcal{A}_t)$ contains at least one cycle of finite weight and therefore $\beta_t > -\infty$. Hence, by Lemma 3.9, the subeigenvectors s_t exist and have finite entries.

On line 14, $s_\ell(g_\ell)$ with $g_\ell = f_\ell \circ \cdots \circ f_1$ is a vector of length n such that $\big(s_\ell(g_\ell)\big)_i = (s_\ell)_{g_\ell(i)}$, $\ell = 1, \ldots, t$, t being the number of steps required for the max-balancing algorithm to terminate. Schneider and Schneider [20, Thm. 6] show that the vector $d_{\max}$ returned by Algorithm 3.10 defines the diagonal similarity scaling which max-balances $\mathcal{A}$. The max-balancing scaling of $A \in \mathbb{C}^{n\times n}$ is then given by

$$A_{d_{\max}} = \operatorname{diag}\big(\exp(-d_{\max})\big) A \operatorname{diag}\big(\exp(d_{\max})\big).$$

Young, Tarjan, and Orlin [22] show that the max-balancing algorithm can be implemented with $O\big(n\tau + n^2 \log n\big)$ operations, τ being the number of finite entries in $\mathcal{A}$.

EXAMPLE 3.11. *Let us use Algorithm* 3.10 *to max-balance* $\mathcal{A} = \mathcal{H}_d = \mathcal{V}(H_d)$*, where* H_d *is one of the max-plus Hungarian scaled matrices of Example* 2.3.
$t = 1$. *We start by setting the diagonal entries of* $\mathcal{A}$ *to* $-\infty$ *to give*

$$\mathcal{A}_1 = \begin{bmatrix} -\infty & 0 & 0 \\ -1 & -\infty & -2 \\ -\infty & -4 & -\infty \end{bmatrix}. \tag{3.10}$$

The precedence graph $\Gamma(\mathcal{A}_1)$ *is shown in Figure* 3.1(*a*)*. The maximum cycle mean* β_1*, a critical cycle* C_1*, and a subeigenvector* s_1 *for* $\mathcal{A}_1$ *associated with* β_1 *are given by* $\beta_1 = -0.5$*,* $C_1 = \{(1,2),(2,1)\}$*, and* $s_1 = [0, -0.5, 0]^T$ *so that* $m_1 = 2$.
$t = 2$. *We compute*

$$\mathcal{S}_2 = \operatorname{diag}_\infty(-s_1) \otimes \mathcal{A}_1 \otimes \operatorname{diag}_\infty(s_1) = \begin{bmatrix} -\infty & -0.5 & 0 \\ -0.5 & -\infty & -1.5 \\ -\infty & -4.5 & -\infty \end{bmatrix}.$$

Next we set $f_2(1) = f_2(2) = 1$*,* $f_2(3) = 2$ *so that*

$$\mathcal{A}_2 = \begin{bmatrix} -\infty & \max\{0, -1.5\} \\ \max\{-\infty, -4.5\} & -\infty \end{bmatrix} = \begin{bmatrix} -\infty & 0 \\ -4.5 & -\infty \end{bmatrix}. \tag{3.11}$$

The precedence graph $\Gamma(\mathcal{A}_2)$ *is shown in Figure* 3.1(*b*)*. The maximum cycle mean, critical cycle, and subeigenvector for* $\mathcal{H}_2$ *are given by* $\beta_2 = -2.25$*,* $C_2 = \{(1,2),(2,1)\}$*, and* $s_2 = [0, -2.25]^T$ *so that* $m_2 = 2 - 2 + 1 = 1$ *and*

the algorithm terminates. The max-balancing scaling parameter $d_{\max}$ *is then given by* $d_{\max} = s_1 + s_2(f_2) = [0, -0.5, 0]^T + [0, 0, -2.25]^T = [0, -0.5, -2.25]^T$, *which results in the max-balanced Hungarian scaled max-plus matrix*

$$z\mathcal{H}_{d_{\max}} = \operatorname{diag}_\infty(-d_{\max}) \otimes \mathcal{A} \otimes \operatorname{diag}_\infty(d_{\max}) = \begin{bmatrix} 0 & -0.5 & -2.25 \\ -0.5 & 0 & -3.75 \\ -\infty & -2.25 & 0 \end{bmatrix}.$$

For the matrices $A, H \in \mathbb{C}^{n\times n}$ *of Example* 2.3, *max-balancing leads to the max-balanced Hungarian scaled matrix*

$$H_{d_{\max}} = \operatorname{diag}\big(\exp(-d_{\max})\big) H \operatorname{diag}\big(\exp(d_{\max})\big) = \begin{bmatrix} 1 & \exp(-\frac{1}{2}) & \exp(-\frac{9}{4}) \\ \exp(-\frac{1}{2}) & 1 & \exp(-\frac{15}{4}) \\ 0 & \exp(-\frac{9}{4}) & 1 \end{bmatrix}.$$

Table 2.1 *shows that* $H_{d_{\max}}$ *has the smallest norm and condition number amongst all of the Hungarian scaled matrices obtained so far from* A. *Note that* $H_{d_{\max}}$ *is diagonally dominant by row and by column.*

4. Numerical results for linear system scalings. In this section we report on the performance of a variety of scaling and reordering methods applied as a preprocessing treatment before calling direct and iterative solvers. Computations were performed using MATLAB and UMFPACK [4]. Our 114 test matrices are from the SuiteSparse Matrix Collection [5].[1] We select all real irreducible matrices of dimension 100 or greater having numerical symmetry less than or equal to 0.9 for which the max-balancing scaling could be computed within 30 minutes. The largest matrix in our set has dimension 62424.

To each matrix, and for each scaling method, we first apply the optimal assignment permutation computed by a MEX interface to the HSL code MC64 [14]. The MC64 code also provides a Hungarian pair, which we use to form the Hungarian scaled and reordered matrix $H = D_1 A D_2 P$. Given H, we can then apply the max-balancing scaling via the similarity transform $D_s^{-1} H D_s$, where $D_s = \operatorname{diag}(s)$ is nonsingular. To compute the max-balancing vector s, we use a MEX interface to our own C++ implementation of Algorithm 3.10. We stress here that our max-balancing code is not optimized. However, we give some statistics on the time to compute the scaling here. The fastest scaling was computed in 0.013 seconds (`gre_115`, $n = 115$), while the slowest scaling took 1300 seconds (`cage11`, $n = 39082$). Although the median time to compute the max-balancing scaling is 3.2 seconds, the mean is 110 seconds; this indicates that for most matrices computing the max-balancing scaling is fast, but for a small number of matrices it is slow. In order to compute a max-balancing scaling for an arbitrary sparse matrix in a time commensurate with solving a linear system, we may need a new approach dealing with these rare but difficult problems. We note that the HSL code MC64 [14] is far from a basic implementation of the Hungarian algorithm as it contains several heuristics designed to speed up the computation. We anticipate that a similar approach could be taken to speeding up Algorithm 3.10.

We compare these two Hungarian scalings to the iterative equilibration algorithm proposed by Knight, Ruiz, and Uçar [15] using the recommended settings, namely one step of ∞-norm scaling followed by three steps of 1-norm scaling.[2] The abbreviations for the different scalings and permutation options are listed in Table 4.1.

[1] Formerly known as the University of Florida Sparse Matrix Collection.

[2] The ScalingSuit MATLAB implementation of the KRU scaling is available at http://perso.ens-lyon.fr/bora.ucar/codes.html.

TABLE 4.1
Abbreviations for different scaling options. In all cases the optimal assignment ordering is applied.

	Scaling
O	unscaled
KRU	Knight, Ruiz, and Uçar [15]
H	MC64 Hungarian
MB	max-balanced Hungarian

We make use of performance profiles [6] that allow us to easily display, for all matrices in the test set, how the scalings affect a performance measure like the condition number. To obtain the performance profile, we first define the performance ratio for scaling k on a given matrix to be the ratio of the performance for scaling k to that of the best performing scaling, out of all of the scaling methods being compared, for that matrix. Throughout, we assume that the performance measure of interest is one for which a smaller number is better. The monotonically increasing function $f_{\text{scale}}(\alpha)$, $\alpha \in [1, \infty)$, then measures the proportion of matrices for which the performance ratio for a particular scaling is at most α. Plotting the curves $f_{\text{scale}}(\alpha)$ against α for the different scalings gives a performance profile that shows which scaling performs best or joint-best ($\alpha = 1$) and which scalings are near-best (small α). Additionally, $\lim_{\alpha\to\infty}$ indicates when a scaling fails (say, to produce L and U factors without pivoting) on a matrix for which at least one other scaling does not fail.

4.1. Matrix properties. To assess the row diagonal dominance of a matrix $A \in \mathbb{R}^{n\times n}$, we measure $\Delta_p(A)$ as in (3.5). The smaller the score, the more diagonally dominant the matrix. Since all of the matrices in our test set are irreducible, it is not possible for any of them to have a score of zero when the optimal assignment ordering has been applied.

For each matrix in the test set we record $\Delta_p(A)$ for each of the scaling methods and for $p = 1, 2, 16$. See Figure 4.1(a)–(c). Without scaling many matrices are far from diagonally dominant, but the KRU and H methods are both close to the best method for the vast majority of matrices. Although their performance is nearly identical for $p = 1$ and $p = 2$, the H method outperforms the KRU method for $p = 16$. Theorem 3.8 states that for any matrix $A \in \mathbb{R}^{n\times n}$, there exists $p' > 0$, such that for all $p \geq p'$, the MB method will be optimal. Thus, for larger values of p, the MB method will outperform all of the other methods. The figure shows that MB outperforms all of the other methods even for the smaller values of p, although the number of problems for which it is best is larger for $p = 16$. Note that we also measured column diagonal dominance (the row diagonal dominance measure applied to A^T); the results were similar and so have been omitted.

We additionally note that applying the optimal assignment scaling has a significant impact on diagonal dominance. If the optimal assignment permutation is not applied, there are 19 matrices with zeros on the diagonal. When methods O and KRU are used without first applying the optimal assignment permutation, both suffer from many fails and are the worst-performing methods.

Figure 4.1(d) shows the estimated condition number of the scaled matrices, using the MATLAB function `condest`. All of the scaling methods consistently outperform method O. The KRU and MB methods have similar performance. The H method is also close to these two methods but is slightly weaker at achieving near-best condition numbers, and for some problems it is far from the best scaling.

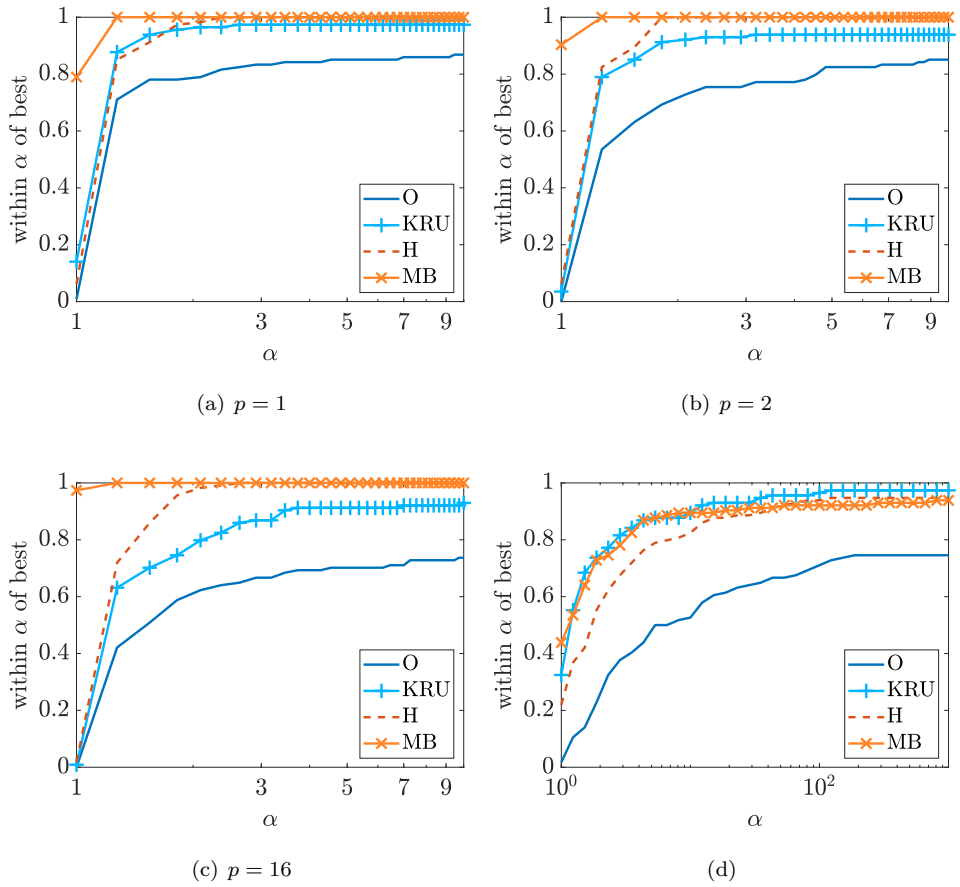
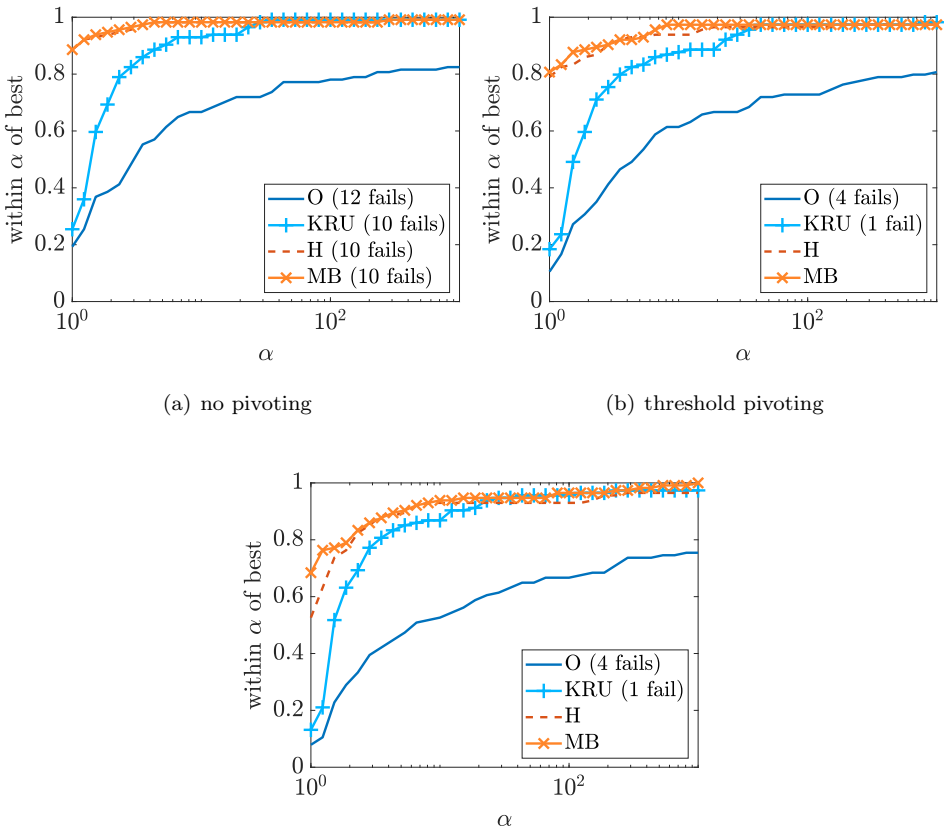


FIG. 4.1. (a)–(c) *Performance profile of the row diagonal dominance factor* Δ_p *in* (3.5) *for* $p = 1, 2, 16$. (d) *Performance profile of the estimated condition numbers.*

4.2. Gaussian elimination. In this subsection we examine the effect of the scalings on the performance of Gaussian elimination. We use UMFPACK with the MATLAB interface to compute the LU factors with the symmetric pivoting strategy (to prevent column reordering) and the CHOLMOD fill-reducing ordering option. Otherwise default settings are applied. Three different pivoting options are tested: no pivoting, threshold pivoting with default tolerances, and partial pivoting. We denote by a fail any matrix for which the estimated condition number (using the MATLAB function `condest`) is larger than 10^{15}.

Figure 4.2 shows the condition number of the upper triangular factor U computed during Gaussian elimination. When pivoting is not used, the O method results in very poorly conditioned U factors for several of the problems. The three scaling methods have similar numbers of fails, and the H and MB methods have nearly identical performance; they are typically optimal or near-optimal. With threshold pivoting the pattern is the same except that there are fewer fails, with the H and MB methods computing a reasonably conditioned U factor for all matrices. With partial pivoting the results are much the same as with threshold pivoting, although the MB method



(a) no pivoting

(b) threshold pivoting

(c) partial pivoting

FIG. 4.2. *Performance profile of the estimated condition number of U.*

TABLE 4.2

Number of problems for which solver B requires at least 50 *more off-diagonal pivots than solver A for threshold pivoting.*

	Solver B			
Solver A	O	KRU	H	MB
O	—	5	1	2
KRU	17	—	13	1
H	17	13	—	3
MB	25	15	15	—

is optimal for slightly more matrices.

We count the number of off-diagonal pivots (as tabulated by UMFPACK) used in Gaussian elimination. Performance profiles are not appropriate for displaying this data as there are certain problems and scalings for which no off-diagonal pivots are required, so we make use of tables instead. Tables 4.2 and 4.3 show the number of problems for which one solver requires at least 50 fewer off-diagonal pivots than the other solvers. From this we see that when threshold pivoting is applied, method O is the weakest, only winning over another method eight times. KRU and H have similar

TABLE 4.3
Number of problems for which solver B requires at least 50 *more off-diagonal pivots than solver A for partial pivoting.*

	Solver B			
Solver A	O	KRU	H	MB
O	—	11	6	7
KRU	12	—	11	7
H	32	33	—	8
MB	37	37	19	—

performances, with MB being the best performing method. When partial pivoting is used, the ordering of the methods is the same, but the differences between them become more pronounced. Although the H method is closer to the MB method, the MB method wins over the H method 19 times and loses to it only eight times.

Tables 4.4, 4.5, and 4.6 show the factorization times for the nine matrices for which factorization took longer than 0.25 seconds for all scalings and pivoting strategies. For each matrix we record the average time over 10 runs, and the minimum time out of these 10. We mark with a dash factorizations which ended in breakdown. Table 4.7 also shows the number of row interchanges needed for these factorizations.

Without pivoting the time taken to compute the factorization depends only on the pattern of the matrix. Thus we expect the different scalings to have very similar runtimes because the methods will return matrices with the same pattern. The reason for the breakdown of the factorizations of `bbmat` is not so clear, but this highlights that although the optimal assignment ordering and Hungarian scaling improves the robustness and quality of LU factorizations in many cases, it is not guaranteed to do so.

The effects that determine the time taken to compute a factorization with pivoting are more complex. Performing lots of row interchanges will add to the computation time but may also affect the density of the LU factors. However, in general we find that with threshold pivoting there is still very little difference between the scalings, with the exception of the KRU and MB scalings for `bbmat`, and the KRU scaling for `cage11`. With partial pivoting, method MB wins, being within 5% of the fastest time for seven out of nine problems.

The number of pivots required was generally reduced significantly by applying the optimal assignment permutation. For example, when neither the optimal assignment permutation nor a diagonal scaling is applied to the matrix `Ill_Stokes`, threshold pivoting requires 4948 pivots. However, after applying the optimal assignment ordering, the factorization can be computed without pivoting.

4.3. Iterative solvers. In this subsection we examine the effect of scaling on the performance of iterative solvers with incomplete LU (ILU) preconditioners. For each test matrix A we take the scaled and reordered matrix $S = P_R D_R A D_C P_C$ and then compute ILU factors LU for S using the MATLAB function `ilu` with options

```
setup.type='ilutp'; setup.droptol=0.01;
```

which performs threshold ILU with partial pivoting and a drop tolerance of 0.01. Combining the ILU factors with the scaling and permutation matrices results in the preconditioner

$$M = (D_R^{-1} P_R^{-1} L)(U P_C^{-1} D_C^{-1}).$$

Next we solve the linear system $Ax = b$ using right-preconditioned GMRES and left-

TABLE 4.4

Average factorization time with minimum factorization time in parentheses. Factors are computed without pivoting. Numbers in bold represent average times that are within 5% of the lowest time for that problem.

Name	O	KRU	H	MB
Ill_Stokes	**0.29 (0.29)**	**0.29 (0.29)**	**0.29 (0.29)**	**0.29 (0.29)**
bbmat	—	—	—	—
cage11	**3.02 (3.01)**	**3.08 (3.07)**	**3.08 (3.01)**	**3.09 (3.07)**
ns3Da	**0.65 (0.65)**	**0.66 (0.65)**	**0.66 (0.65)**	**0.65 (0.64)**
psmigr_2	**0.61 (0.61)**	**0.61 (0.61)**	**0.61 (0.61)**	**0.61 (0.61)**
psmigr_3	**0.55 (0.55)**	**0.55 (0.55)**	**0.55 (0.55)**	**0.56 (0.55)**
raefsky3	**0.26 (0.26)**	**0.26 (0.26)**	**0.26 (0.26)**	**0.26 (0.25)**
venkat01	**0.37 (0.36)**	**0.37 (0.37)**	**0.37 (0.37)**	**0.37 (0.36)**
wang4	**0.28 (0.28)**	**0.28 (0.28)**	**0.28 (0.28)**	**0.28 (0.28)**

TABLE 4.5

Average factorization time with minimum factorization time in parentheses. Factors are computed with threshold pivoting. Numbers in bold represent average times that are within 5% of the lowest time for that problem.

Name	O	KRU	H	MB
Ill_Stokes	**0.29 (0.29)**	**0.29 (0.29)**	**0.30 (0.29)**	**0.29 (0.29)**
bbmat	**1.22 (1.19)**	1.28 (1.28)	**1.20 (1.19)**	1.28 (1.27)
cage11	**3.07 (3.01)**	3.43 (3.02)	**3.06 (3.01)**	**3.05 (3.01)**
ns3Da	**0.65 (0.65)**	**0.66 (0.65)**	**0.66 (0.65)**	**0.65 (0.64)**
psmigr_2	**0.61 (0.61)**	**0.61 (0.61)**	**0.61 (0.61)**	**0.61 (0.61)**
psmigr_3	**0.55 (0.54)**	**0.55 (0.55)**	**0.55 (0.55)**	**0.55 (0.55)**
raefsky3	**0.26 (0.25)**	**0.26 (0.26)**	**0.26 (0.26)**	**0.25 (0.25)**
venkat01	**0.37 (0.36)**	**0.37 (0.37)**	**0.37 (0.36)**	**0.37 (0.36)**
wang4	**0.28 (0.28)**	**0.28 (0.28)**	**0.28 (0.28)**	**0.28 (0.28)**

TABLE 4.6

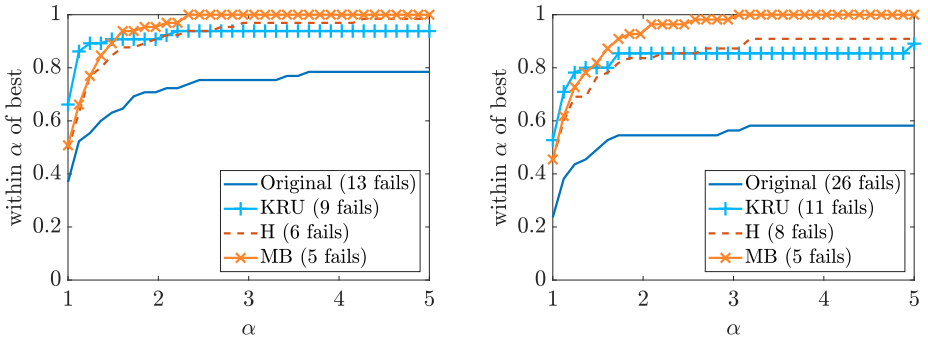
Average factorization time with minimum factorization time in parentheses. Factors are computed with partial pivoting. Numbers in bold represent average times that are within 5% of the lowest time for that problem.

Name	O	KRU	H	MB
Ill_Stokes	7.09 (7.07)	5.31 (5.30)	5.57 (5.55)	**2.43 (2.41)**
bbmat	11.69 (11.61)	5.99 (5.92)	10.07 (10.04)	**3.91 (3.89)**
cage11	**3.09 (3.08)**	**3.03 (3.01)**	**3.01 (3.00)**	**3.01 (3.01)**
ns3Da	1.44 (1.42)	**1.25 (1.25)**	2.10 (2.10)	1.82 (1.82)
psmigr_2	1.67 (1.67)	1.70 (1.69)	1.96 (1.96)	**1.15 (1.15)**
psmigr_3	**0.55 (0.54)**	**0.55 (0.55)**	**0.55 (0.55)**	**0.56 (0.55)**
raefsky3	1.30 (1.30)	**0.33 (0.33)**	0.53 (0.53)	0.65 (0.65)
venkat01	**0.38 (0.37)**	**0.37 (0.37)**	**0.37 (0.37)**	**0.37 (0.36)**
wang4	**0.28 (0.28)**	0.30 (0.30)	**0.28 (0.28)**	**0.28 (0.28)**

TABLE 4.7

The number of row interchanges used by UMFPACK when threshold and partial pivoting are used.

Name	Threshold				Partial			
	O	KRU	H	MB	O	KRU	H	MB
Ill_Stokes	0	0	0	0	7215	4162	3789	3003
bbmat	149	112	132	128	8881	5314	6729	2988
cage11	0	0	0	0	0	0	0	0
ns3Da	2	2	2	2	1661	1468	1352	1165
psmigr_2	16	13	5	6	913	903	905	714
psmigr_3	0	0	0	0	2	2	0	0
raefsky3	0	0	0	0	1551	608	601	1166
venkat01	0	0	0	0	16425	179	13135	186
wang4	0	0	0	0	0	7	0	2



(a) GMRES (b) BiCGSTAB

FIG. 4.3. *Performance profiles of number of iterations needed for convergence.*

preconditioned BiCGSTAB, where b is chosen so that the exact solution is a vector of ones. We use the MATLAB functions `gmres` (without restarts) and `bicgstab`, with a tolerance of 10^{-6}, and a maximum of $\min\{n, 1000\}$ iterations. If either method fails to converge below the tolerance within the maximum number of iterations, then we record a fail. Many of the problems in the test set are solved very easily, so we omit any matrix for which the O method converges in fewer than 10 iterations. This leaves 65 problems for GMRES and 55 problems for BiCGStab.

Figure 4.3 shows the number of iterations needed for the different scaling strategies. All of the scaling methods significantly outperform method O when GMRES is used. Method MB outperforms method H by a small margin. Method KRU is slightly better than method MB at producing very low numbers of iterations but is less reliable, resulting in four more fails. The pattern is the same for BiCGSTAB except that the advantage of the KRU method for low numbers of iterations is smaller and the advantage of the MB method for reliability is greater, with four fewer fails than KRU.

5. Conclusion. We have introduced max-balanced Hungarian scaling, which is applied to a matrix $A \in \mathbb{C}^{n\times n}$ in two stages. First we apply a Hungarian scaling and optimal assignment reordering $H = PD_1AD_2$, such that $|h_{ij}| \le 1$ and $|h_{ii}| = 1$ for $i = 1, \dots, n$. The permutation matrix P and diagonal matrices D_1, D_2 can be obtained using standard algorithms such as the HSL code MC64 [14]. The second stage is to apply a max-balancing similarity scaling $M = S^{-1}HS$, such that for all $\mathcal{I} \subset \{1, \dots, n\}$ we have

$$\max_{i\in\mathcal{I}, j\notin\mathcal{I}} |m_{ij}| = \max_{i\in\mathcal{I}, j\notin\mathcal{I}} |m_{ji}|. \tag{5.1}$$

The diagonal matrix S can be obtained using Algorithm 3.10, which was first introduced by Schneider and Schneider [20], with a more efficient implementation given by Young, Tarjan, and Orlin [22].

In Theorem 3.6 we proved that max-balancing preserves the properties of a Hungarian scaling, so that M satisfies $|m_{ij}| \le 1$ and $|m_{ii}| = 1$ for $i = 1, \dots, n$ as well as (5.1). In Theorem 3.8 we proved that M is the most diagonally dominant matrix out of all possible scalings and reorderings of A, when viewed through the p-norm for sufficiently large p, up to multiplication by permutation matrices that switch between

optimal assignments and multiplication by a scalar.

The experiments in section 4 demonstrate the improved diagonal dominance brought about by max-balanced Hungarian scaling. The most notable difference between max-balanced Hungarian scaling and plain Hungarian scaling is the number of row interchanges used during Gaussian elimination with partial pivoting; max-balancing significantly reduces the number of row interchanges needed for many of the test problems. Max-balancing also tends to reduce the number of iterations required for convergence of GMRES or BiCGSTAB with an ILU preconditioner.

Comparing the whole suite of scaling and reordering methods considered, we see that Hungarian scaling, max-balanced Hungarian scaling, and KRU scaling all significantly reduce the matrix condition number. Applying any one of these scalings together with the optimal assignment permutation significantly reduces the need for row interchanges with threshold or partial pivoting. The three methods have roughly the same performance under threshold pivoting, but max-balanced Hungarian scaling has a clear lead with partial pivoting. The condition number of the U factor of the matrix is significantly reduced by Hungarian scaling and max-balanced Hungarian scaling, although there does not appear to be any extra advantage to using max-balanced Hungarian scaling here. The effect of scaling on factorization time is more complicated, with surprising behavior such as extra pivoting sometimes reducing the factorization time. However, there is some evidence that max-balanced Hungarian scaling tends to give the fastest factorization when using partial pivoting. All of the scaling methods significantly reduce the number of iterations required for convergence of GMRES or BiCGSTAB with an ILU preconditioner. The KRU method was strongest at producing close to optimal iteration numbers, but the MB method was more reliable, resulting in the fewest fails.

As discussed in the introduction, Hungarian scaling has been shown to be a beneficial preprocessing treatment for solving linear systems, and this has been attributed to the fact that the Hungarian scaled and reordered matrix tends to be more diagonally dominant. We have shown that max-balanced Hungarian scaling results in a matrix which is optimally diagonally dominant. So according to the rule that diagonal dominance is beneficial to solving linear systems, the max-balanced Hungarian scaling ought to be the optimal preprocessing treatment. In our numerical experiments we have seen that the Hungarian scaling returned by the HSL code MC64 [14] has a very similar performance to max-balanced Hungarian scaling. So we must conclude that, although the algorithm used by MC64 is oblivious to the fact that there is typically a range of possible Hungarian scalings for a given matrix A, it tends to return one that is close in performance to the optimal scaling. The only test where MC64 was significantly outperformed was in the number of row interchanges required during Gaussian elimination with partial pivoting, so we should encourage the use of max-balanced Hungarian scaling when minimizing this quantity is the objective. But to be able to compute a max-balanced scaling for an arbitrary sparse matrix in a time commensurate with solving a linear system, we may need to develop new heuristics aiming at speeding up the computation, as was done in the efficient MC64 implementation of the Hungarian scaling.

Acknowledgment. Thanks are given to the reviewers for their valuable comments and suggestions, which considerably improved the original manuscript.

REFERENCES

[1] M. Benzi, J. C. Haws, and M. Tůma, *Preconditioning highly indefinite and nonsymmetric matrices*, SIAM J. Sci. Comput., 22 (2000), pp. 1333–1353, https://doi.org/10.1137/S1064827599361308.
[2] P. Butkovič, *Max-Linear Systems: Theory and Algorithms*, Springer-Verlag, London, 2010.
[3] P. Butkovič and H. Schneider, *Applications of max algebra to diagonal scaling of matrices*, Electron. J. Linear Algebra, 13 (2005), pp. 262–273.
[4] T. A. Davis, *Algorithm* 832: *UMFPACK V*4.3*—an unsymmetric-pattern multifrontal method*, ACM Trans. Math. Software, 30 (2004), pp. 196–199.
[5] T. A. Davis and Y. Hu, *The University of Florida sparse matrix collection*, ACM Trans. Math. Software, 38 (2011), pp. 1–25.
[6] E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., 91 (2002), pp. 201–213.
[7] I. S. Duff and J. Koster, *The design and use of algorithms for permuting large entries to the diagonal of sparse matrices*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 889–901, https://doi.org/10.1137/S0895479897317661.
[8] I. S. Duff and J. Koster, *On algorithms for permuting large entries to the diagonal of a sparse matrix*, SIAM J. Matrix Anal. Appl., 22 (2001), pp. 973–996, https://doi.org/10.1137/S0895479899358443.
[9] A. Frank, *On Kuhn's Hungarian Method—A Tribute from Hungary*, Technical report TR-2004-14, Egerváry Research Group, Budapest, 2004, http://www.cs.elte.hu/egres.
[10] M. L. Fredman and R. E. Tarjan, *Fibonacci heaps and their uses in improved network optimization algorithms*, J. Assoc. Comput. Mach., 34 (1987), pp. 596–615.
[11] J. D. Hogg and J. A. Scott, *The Effects of Scalings on the Performance of a Sparse Symmetric Indefinite Solver*, Report RAL-TR-2008-007, Atlas Centre, Rutherford Appleton Laboratory, Didcot, Oxon, UK, 2008.
[12] J. D. Hogg and J. A. Scott, *On the use of suboptimal matchings for scaling and ordering sparse symmetric matrices*, Numer. Linear Algebra Appl., 22 (2015), pp. 648–663.
[13] J. Hook and F. Tisseur, *Incomplete LU preconditioner based on max-plus approximation of LU factorization*, SIAM J. Matrix Anal. Appl., 38 (2017), pp. 1160–1189, https://doi.org/10.1137/16M1094579.
[14] HSL, *A Collection of Fortran Codes for Large Scale Scientific Computation*, http://www.hsl.rl.ac.uk/.
[15] P. A. Knight, D. Ruiz, and B. Uçar, *A symmetry preserving algorithm for matrix scaling*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 931–955, https://doi.org/10.1137/110825753.
[16] M. Olschowka and A. Neumaier, *A new pivoting strategy for Gaussian elimination*, Linear Algebra Appl., 240 (1996), pp. 131–151.
[17] J. B. Orlin and Y. Lee, *Quickmatch—A Very Fast Algorithm for the Assignment Problem*, Working paper 3547-93, Sloan School of Management, Massachusetts Institute of Technology (MIT), Cambridge, MA, 1993, http://EconPapers.repec.org/RePEc:mit:sloanp:2460.
[18] E. E. Osborne, *On pre-conditioning of matrices*, J. Assoc. Comput. Mach., 7 (1960), pp. 338–345.
[19] U. Rothblum, H. Schneider, and M. Schneider, *Characterizations of max-balanced flows*, Discrete Appl. Math., 39 (1992), pp. 241–261.
[20] H. Schneider and M. H. Schneider, *Max-balancing weighted directed graphs and matrix scaling*, Math. Oper. Res., 16 (1991), pp. 208–222.
[21] S. Sergeev, H. Schneider, and P. Butkovič, *On visualization scaling, subeigenvectors and Kleene stars in max algebra*, Linear Algebra Appl., 431 (2009), pp. 2395–2406.
[22] N. E. Young, R. E. Tarjan, and J. B. Orlin, *Faster parametric shortest path and minimum balance algorithms*, Networks, 1 (1991), pp. 205–221.