Noname manuscript No.
(will be inserted by the editor)

On the expressive power of linear algebra on graphs

Floris Geerts

Received: date / Accepted: date

Abstract There is a long tradition in understanding graphs by investigating their adjacency matrices by means of linear algebra. Similarly, logic-based graph query languages are commonly used to explore graph properties. In this paper, we bridge these two approaches by regarding linear algebra as a graph query language.

More specifically, we consider MATLANG, a matrix query language recently introduced, in which some basic linear algebra functionality is supported. We investigate the problem of characterising the equivalence of graphs, represented by their adjacency matrices, for various fragments of MATLANG. That is, we are interested in understanding when two graphs cannot be distinguished by posing queries in MATLANG on their adjacency matrices.

Surprisingly, a complete picture can be painted of the impact of each of the linear algebra operations supported in MATLANG on their ability to distinguish graphs. Interestingly, these characterisations can often be phrased in terms of spectral and combinatorial properties of graphs.

Furthermore, we also establish links to logical equivalence of graphs. In particular, we show that MATLANG-equivalence of graphs corresponds to equivalence by means of sentences in the three-variable fragment of first-order logic with counting. Equivalence with regards to a smaller MATLANG fragment is shown to correspond to equivalence by means of sentences in the two-variable fragment of this logic.

Keywords Linear algebra · Graphs · Query languages

Floris Geerts
Department of Mathematics and Computer Science, University of Antwerp, Middelheimlaan 1, B-2020 Antwerp, Belgium
Tel.: +32-03-2653907
Fax: +32-03-2653777
E-mail: floris.geerts@uantwerpen.be

1 Introduction

Motivated by the importance of linear algebra for machine learning on big data [8, 9, 15, 55, 63] there is a current interest in languages that combine matrix operations with relational query languages in database systems [26, 43, 49, 50, 53]. Such hybrid languages raise many interesting questions from a database theoretical point of view. It seems natural, however, to first consider query languages for matrices alone. These are the focus of this paper.

More precisely, we continue the investigation of the expressive power of the matrix query language MATLANG, recently introduced by Brijder et al. [10, 11], as an analog for matrices of the relational algebra on relations. Intuitively, queries in MATLANG are built up by composing several linear algebra operations commonly found in linear algebra packages. When arbitrary matrices are concerned, it is known that MATLANG is subsumed by aggregate logic with only three non-numerical variables. This implies, among other things, that when evaluated on adjacency matrices of graphs, MATLANG cannot compute the transitive closure of a graph and neither can it express the four-variable query asking if a graph contains a four-clique [10, 11].

In fact, it is implicit in the work by Brijder et al. that when two graphs G and H are indistinguishable by sentences in the three-variable fragment C^3 of first-order logic with counting, denoted by $G \equiv_{\mathsf{C}^3} H$, then their adjacency matrices cannot be distinguished by MATLANG expressions that return scalars, henceforth referred to as sentences in MATLANG. The equivalence with respect to such sentences is denoted by $G \equiv_{\mathsf{MATLANG}} H$. A natural question is whether the converse implication also holds, i.e., does $G \equiv_{\mathsf{MATLANG}} H$ also imply $G \equiv_{\mathsf{C}^3} H$? We answer this question affirmatively.

The underlying proof technique relies on a close connection between C^3-equivalence and the indistinguishability of graphs by the 2-dimensional Weisfeiler-Lehman (2WL) algorithm, a result dating back to the seminal paper by Cai, Fürer and Immerman [13, 44]. Indeed, as we will see, the linear algebra operators supported in MATLANG have sufficient power to simulate the 2WL algorithm. Hence, when $G \equiv_{\mathsf{MATLANG}} H$, then G and H cannot by distinguished by the 2WL algorithm.

This *combinatorial interpretation* of MATLANG-equivalence immediately provides an insight in which graph properties are preserved under MATLANG-equivalence (see e.g., the work by Fürer [29, 30]). For example, when $G \equiv_{\mathsf{MATLANG}} H$, then G and H must be co-spectral (that is, their adjacency matrices have the same multi-set of eigenvalues) and have the same number of s-cycles, for $s \leq 6$, but not necessarily s-cycles for $s > 7$. As observed in the conference version of this paper [31], the case of 7-cycles easily follows from the connection with MATLANG. Indeed, the linear algebra expressions for counting s-cycles, for $s \leq 7$, given in Noga et al. [1] are expressible in MATLANG and hence, 7-cycles are preserved by 2WL-equivalence. This has been recently verified using other techniques by Arvind et al. [3]. Although formulas exist for counting cycles of length greater than 7 [1], they require counting the number of k-cliques, for $k \geq 4$, which is not possible in MATLANG, as observed earlier.

Apart from the logical and spectral/combinatorial characterisation of MATLANG-equivalence, we also point out the correspondence between C^3-equivalence (and thus also 2WL- and MATLANG-equivalence) and *similarity conditions* between adjacency matrices. As observed by Dawar et al. [23, 24], $G \equiv_{\mathsf{C}^3} H$ if and only if there exists a unitary matrix U such that $A_G \cdot U = U \cdot A_H$ and moreover, U induces an algebraic isomorphism between the so-called coherent algebras of A_G and A_H. Here, A_G and A_H denote the adjacency matrices of G and H, respectively. We recall that a unitary matrix U is a complex matrix whose inverse is its complex conjugate tranpose U^*. Coherent algebras and their isomorphisms are detailed later in the paper.

All combined, we have a logical, combinatorial and similarity-based characterisation of MATLANG-equivalence. Surprisingly, similar characterisations hold also for *fragments of* MATLANG. We define fragments of MATLANG by allowing only certain linear algebra operations in our expressions. Such fragments are denoted by $\mathsf{ML}(\mathcal{L})$, with $\mathcal{L}$ the list of allowed operations. The corresponding notion of equivalence of graphs G and H will be denoted by $G \equiv_{\mathsf{ML}(\mathcal{L})} H$. That is, $G \equiv_{\mathsf{ML}(\mathcal{L})} H$ if any sentence in $\mathsf{ML}(\mathcal{L})$ results in the same scalar when evaluated on A_G and A_H. We investigate equivalence for all sensible MATLANG fragments. Our results are, as follows:

For starters, we consider the fragment $\mathsf{ML}(\cdot, \mathsf{tr})$ that allows for matrix multiplication ($\cdot$) and trace (tr) computation (i.e., taking the sum of the diagonal elements of a matrix). Then, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr})} H$ if and only if G and H are co-spectral, or equivalently, they have the same number of closed walks of any length, or $A_G \cdot O = O \cdot A_H$ for some orthogonal matrix O. We recall that an orthogonal matrix O is a matrix over the real numbers such that its inverse coincides with the transpose matrix O^{t} (Section 5).

Another small fragment, $\mathsf{ML}(\cdot, {}^*, \mathbb{1})$, allows for matrix multiplication, conjugate transposition (*) and the use of the vector $\mathbb{1}$, consisting of all ones. Then, $G \equiv_{\mathsf{ML}(\cdot,{}^*,\mathbb{1})} H$ if and only if G and H are co-main (roughly speaking, they are co-spectral only for special "main" eigenvalues), or equivalently, they have the same number of (not necessarily closed) walks of any length, or $A_G \cdot Q = Q \cdot A_H$ for some doubly quasi-stochastic matrix Q. A doubly quasi-stochastic matrix Q is a matrix over the real numbers such that every of its columns and rows sums up to one (Section 6).

When allowing both tr and $\mathbb{1}$, equivalence of graphs relative to $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1})$ coincides, not surprisingly, to the graphs being both co-spectral and co-main, or equivalently, having the same number of closed and non-closed walks of any length, or such that $A_G \cdot O = O \cdot A_H$, for an orthogonal doubly quasi-stochastic matrix O (Section 6).

More interesting is the fragment $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})$, which also allows for the operation $\mathsf{diag}(\cdot)$ that turns a vector into a diagonal matrix with that vector on its diagonal. For this fragment we can tie equivalence to indistinguishability by the 1-dimensional Weisfeiler-Lehman (1WL) algorithm (or colour refinement). This is known to coincide with the graphs having a common equitable partition, or the existence of a doubly stochastic matrix S such that $A_G \cdot S = S \cdot A_H$ (a.k.a. as a fractional isomorphism), or C^2-equivalence. Here, C^2 denotes the two-variable fragment of first-order logic with counting. We recall that a doubly stochastic matrix is a doubly quasi-stochastic matrix whose entries are all non-negative (Section 7).

In the former fragment, replacing the operator $\mathsf{diag}(\cdot)$ with an operator ($\odot_v$) which pointwise multiplies vectors results in the same distinguishing power. By contrast, the combination of tr and the ability to pointwise multiply vectors results in a stronger notion of equivalence. That is, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\odot_v)} H$ if and only if G and H are co-spectral and indistinguishable by 1WL. Also in this case, $A_G \cdot O = O \cdot A_H$ for an orthogonal matrix O that, in addition, needs to preserve equitable partitions. We define this preservation condition later in the paper (Section 8).

For the larger fragment $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag})$, no elegant combinatorial characterisation is obtained. Nevertheless, for equivalent graphs G and H, $A_G \cdot O = O \cdot A_H$ where O is an orthogonal matrix that can be block-structured according to the equitable partitions. This is a stronger notion than the preservation of equitable partitions. Graphs equivalent with respect to this fragment have, for example, the same number of spanning trees. This is not necessarily true for all previous fragments (Section 7).

Finally, as we already mentioned, equivalence relative to $\mathsf{MATLANG}$ is shown to correspond to C^3-equivalence and 2WL-equivalence. We additionally refine the similarity-based characterisation given by Dawar et al. [23,24] so that it compares more easily to the similarity notions obtained for all previous fragments. Furthermore, we show that pointwise multiplication of matrices (the Schur-Hadamard product) is crucial in this setting (Section 9).

Each of these fragments can be extended with addition and scalar multiplication at no increase in distinguishing power. It is also shown when fragments can be extended to accommodate for *arbitrary* pointwise function applications, on scalars, vectors or matrices. We furthermore exhibit example graphs *separating* all fragments.

For many of our characterisations we rely on the rich literature on spectral graph theory [12,17,18,19,32,39,61,68] and the study on the equivalence by the Weisfeiler-Lehman algorithms and fixed-variable fragments of first-order logic with counting [23, 24,25,35,44,60,65,66,69]. We describe the relevant results in these papers in the course of the paper. We also refer to work by Fürer [29,30] for more examples of connections to graph invariants and to Dawar et al. [23,24] for connections between logic, combinatorial and spectral invariants.

In some sense, we provide a unifying view of various existing results in the literature by grouping them according to the operators supported in $\mathsf{MATLANG}$. We remark that, recently, another unifying approach has been put forward by Dell et al. [25]. In that work, one considers indistinguishability of graphs in terms of *homomorphism vectors*. That is, one defines $\mathsf{HOM}_{\mathcal{F}}(G) := (\mathsf{Hom}(F,G))_{F\in\mathcal{F}}$ for some class $\mathcal{F}$ of graphs, where $\mathsf{Hom}(F,G)$ is the number of homomorphisms from F to G. Then G and H are indistinguishable for some class $\mathcal{F}$ of graphs when $\mathsf{HOM}_{\mathcal{F}}(G) = \mathsf{HOM}_{\mathcal{F}}(H)$. When $\mathcal{F}$ consists of all cycles, this notion of equivalence corresponds to $\mathsf{ML}(\cdot,\mathsf{tr})$-equivalence (recall the closed walk characterisation of the latter); when $\mathcal{F}$ consists of all paths, we have a correspondence with $\mathsf{ML}(\cdot,{}^*,\mathbb{1})$-equivalence (recall the walk characterisation of the latter); when $\mathcal{F}$ consists of trees, G and H are equivalent for the 1WL-algorithm and thus also for C^2 and $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$, and finally, when $\mathcal{F}$ consists of all graphs of tree-width at most 2, G and H are equivalent for the 2WL-algorithm and thus also for C^3 and $\mathsf{MATLANG}$. Our results can thus be regarded as a re-interpretation of the results in Dell et al. [25] in terms of $\mathsf{MATLANG}$.

We also remark that C^k-equivalence, for $k \geq 4$, can be characterised in terms of solutions to linear problems which resemble similarity-based characterisations [4,36, 54]. We leave it to future work to identify which additional linear algebra operators to include in $\mathsf{MATLANG}$ such that C^k-equivalence can be captured, for $k \geq 4$.

Although we made links to logics such as C^2 and C^3, the connection between $\mathsf{MATLANG}$, rank logics and fixed-point logics with counting, as studied in the context of the descriptive complexity of linear algebra [21,20,22,34,37,42], is yet to be explored. Similarly for connections to logic-based graph query languages [2,6].

2 Background

We denote the set of real numbers by $\mathbb{R}$ and the set of complex numbers by . The set of $m\times n$-matrices over the real (resp., complex) numbers is denoted by $\mathbb{R}^{m\times n}$ (resp., $^{m\times n}$). Vectors are elements of $\mathbb{R}^{m\times 1}$ (or $^{m\times 1}$). The entries of an $m\times n$-matrix A are denoted by A_{ij}, for $i=1,\dots,m$ and $j=1,\dots,n$. The entries of a vector v are denoted by v_i, for $i=1,\dots,m$. We often identify $\mathbb{R}^{1\times 1}$ with $\mathbb{R}$, and $^{1\times 1}$ with . The following classes of matrices are of interest in this paper: square matrices (elements in $\mathbb{R}^{n\times n}$ or $^{n\times n}$), symmetric matrices (such that $A_{ij}=A_{ji}$ for all i and j), *doubly stochastic* matrices ($A_{ij}\in\mathbb{R}$, $A_{ij}\geq 0$, $\sum_{j=1}^{n}A_{ij}=1$ and $\sum_{i=1}^{m}A_{ij}=1$ for all i and j), *doubly quasi-stochastic* matrices ($A_{ij}\in\mathbb{R}$, $\sum_{j=1}^{n}A_{ij}=1$ and $\sum_{i=1}^{m}A_{ij}=1$ for all i and j), and *orthogonal* matrices ($O\in\mathbb{R}^{n\times n}$, $O^{\mathrm{t}}\cdot O=I=O\cdot O^{\mathrm{t}}$, where O^{t} denotes the transpose of O obtained by switching rows and columns, $\cdot$ denotes matrix multiplication, and I is the identity matrix in $\mathbb{R}^{n\times n}$).

We only need a couple of notions of linear algebra. We refer to the textbook by Axler [5] for more background. An *eigenvalue* of a matrix A is a scalar λ in for which there is a non-zero vector v satisfying $A\cdot v=\lambda v$. Such a vector is called an *eigenvector* of A for eigenvalue λ. The *eigenspace* of an eigenvalue is the vector space obtained as the span of a maximal set of linear independent eigenvectors for this eigenvalue. Here, the *span* of a set of vectors just refers to the set of all linear combinations of vectors in that set. A set of vectors is linear independent if no vector in that set can be written as a linear combination of other vectors. The *dimension* of an eigenspace is the minimal number of eigenvectors that span the eigenspace.

We will only consider undirected graphs without self-loops. Let $G=(V,E)$ be such a graph with vertices $V=\{1,\dots,n\}$ and unordered edges $E\subseteq\{\{i,j\}\mid i,j\in V\}$. The *order* of G is simply the number of vertices. Then, the *adjacency matrix* of a graph G of order n, denoted by A_G, is an $n\times n$-matrix whose entries $(A_G)_{ij}$ are set to 1 if and only if $\{i,j\}\in E$, all other entries are set to 0. The matrix A_G is a symmetric real matrix with zeroes on its diagonal. The *spectrum* of an undirected graph can be represented as $\mathsf{spec}(G)=\begin{pmatrix}\lambda_1 & \lambda_2 & \cdots & \lambda_p\\ m_1 & m_2 & \cdots & m_p\end{pmatrix}$, where $\lambda_1<\lambda_2<\cdots<\lambda_p$ are the distinct real eigenvalues of the adjacency matrix A_G of G, and where $m_1,m_2,\dots,m_p$ denote the dimensions of the corresponding eigenspaces. Two graphs are said to be *co-spectral* if they have the same spectrum. We introduce other relevant notions throughout the paper. Recall that a *walk of length* k in a graph $G=(V,E)$ is a sequence $(v_0,v_1,\dots,v_k)$ of vertices of G such that consecutive vertices are adjacent in G, i.e., $(v_{i-1},v_i)\in E$ for all $i=1,\dots,k$. Furthermore, a *closed walk* is a walk that starts in and ends at the same vertex. Closed walks of length 0 correspond, as usual, to vertices in G.

3 Matrix query languages

As described in Brijder et al. [10], matrix query languages can be formalised as compositions of linear algebra operations. Intuitively, a linear algebra operation takes a number of matrices as input and returns another matrix. Examples of operations are matrix multiplication, conjugate transposition, computing the trace, just to name a few. By closing such operations under composition "matrix query languages" are formed. More specifically, for linear algebra operations $\mathsf{op}_1,\dots,\mathsf{op}_k$ the corresponding matrix query language is denoted by $\mathsf{ML}(\mathsf{op}_1,\dots,\mathsf{op}_k)$ and consists of expressions formed by the following grammar:

$$e := X \mid \mathsf{op}_1\big(e_1,\dots,e_{p_1}\big) \mid \cdots \mid \mathsf{op}_k\big(e_1,\dots,e_{p_k}\big),$$

where X denotes a *matrix variable* which serves to indicate the input to expressions and p_i denotes the number of inputs required by operation op_i. We focus on the case when only a single matrix variable X is present. The treatment of multiple variables is left for future work.

The semantics of an expression $e(X)$ in $\mathsf{ML}(\mathsf{op}_1,\dots,\mathsf{op}_k)$ is defined inductively, relative to an *assignment* ν of X to a matrix $\nu(X)\in{}^{m\times n}$, for some dimensions m and n. We denote by $e\big(\nu(X)\big)$ the result of evaluating $e(X)$ on $\nu(X)$. As expected, we define $\mathsf{op}_i(e_1(X),\dots,e_{p_i}(X))(\nu(X)) := \mathsf{op}_i\big(e_1(\nu(X)),\dots,e_{p_i}(\nu(X))\big)$ for linear algebra operation op_i. In Table 3.1 we list the operations constituting the basic matrix query language MATLANG, introduced in Brijder et al. [10]. In the table we also show their semantics. We note that restrictions on the dimensions are in place to ensure that operations are well-defined. Using a simple type system one can formalise a notion of well-formed expressions which guarantees that the semantics of such expressions is well-defined. We refer to Brijder et al. [10] for details. We only consider well-formed expressions from here on.

Remark 3.1 The list of operations in Table 3.1 differs slightly from the list presented in Brijder et al. [10]: We explicitly mention scalar multiplication ($\times$), addition ($+$), and the trace operation (tr), all of which can be expressed in MATLANG. Hence, MATLANG and $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply}[f],f\in\Omega)$ are equivalent.

4 Expressive power of matrix query languages

As mentioned in the introduction, we are interested in the expressive power of matrix query languages. In analogy with indistinguishability notions used in logic, we consider *sentences* in our matrix query languages. We define an expression $e(X)$ in $\mathsf{ML}(\mathsf{op}_1,\dots,\mathsf{op}_k)$ to be a *sentence* if $e(\nu(X))$ returns a 1×1-matrix (i.e., a scalar) for any assignment ν of the matrix variable X in $e(X)$. We note that the type system of MATLANG allows to easily check whether an expression in $\mathsf{ML}(\mathcal{L})$ is a sentence (see Brijder et al. [10] for more details). Having defined sentences, a notion of equivalence naturally follows.

Definition 4.1 Two matrices A and B in ${}^{m\times n}$ are said to be $\mathsf{ML}(\mathsf{op}_1,\dots,\mathsf{op}_k)$-*equivalent*, denoted by $A \equiv_{\mathsf{ML}(\mathsf{op}_1,\dots,\mathsf{op}_k)} B$, if and only if $e(A)=e(B)$ for all sentences $e(X)$ in $\mathsf{ML}(\mathsf{op}_1,\dots,\mathsf{op}_k)$.

conjugate transposition ($\mathsf{op}(e) = e^*$)		
$e(v(X)) = A \in^{m\times n}$	$e(v(X))^* = A^* \in^{n\times m}$	$(A^*)_{ij} = A^*_{ji}$
one-vector ($\mathsf{op}(e) = \mathbb{1}(e)$)		
$e(v(X)) = A \in^{m\times n}$	$\mathbb{1}(e(v(X)) = \mathbb{1} \in^{m\times 1}$	$\mathbb{1}_i = 1$
diagonalization of a vector ($\mathsf{op}(e) = \mathsf{diag}(e)$)		
$e(v(X)) = A \in^{m\times 1}$	$\mathsf{diag}(e(v(X)) = \mathsf{diag}(A) \in^{m\times m}$	$\mathsf{diag}(A)_{ii} = A_i$, $\mathsf{diag}(A)_{ij} = 0,\ i \neq j$
matrix multiplication ($\mathsf{op}(e_1, e_2) = e_1 \cdot e_2$)		
$e_1(v(X)) = A \in^{m\times n}$ $e_2(v(X)) = B \in^{m\times o}$	$e_1(v(X)) \cdot e_2(v(X)) = C \in^{m\times o}$	$C_{ij} = \sum_{k=1}^{n} A_{ik} \times B_{kj}$
matrix addition ($\mathsf{op}(e_1, e_2) = e_1 + e_2$)		
$e_i(v(X)) = A^{(i)} \in^{m\times n}$	$e_1(v)(X) + e_2(v(X)) = B \in^{m\times n}$	$B_{ij} = A^{(1)}_{ij} + A^{(2)}_{ij}$
scalar multiplication ($\mathsf{op}(e) = c \times e$, $c \in$)		
$e(v(X)) = A \in^{m\times n}$	$c \times e(v(X)) = B \in^{m\times n}$	$B_{ij} = c \times A_{ij}$
trace ($\mathsf{op}(e) = \mathsf{tr}(e)$)		
$e(v(X)) = A \in^{m\times m}$	$\mathsf{tr}(e(v(X)) = c \in$	$c = \sum_{i=1}^{m} A_{ii}$
pointwise function application ($\mathsf{op}(e_1, \ldots, e_p) = \mathsf{apply}[f](e_1, \ldots, e_p)$), $f :^p \to \in \Omega$		
$e_i(v(X)) = A^{(i)} \in^{m\times n}$	$\mathsf{apply}[f](e_1(v(X)), \ldots, e_p(v(X))) = B \in^{m\times n}$	$B_{ij} = f(A^{(1)}_{ij}, \ldots, A^{(p)}_{ij})$

Table 3.1 Linear algebra operations (supported in MATLANG [10]) and their semantics. In the first operation, * denotes complex conjugation. In the last operation, $\Omega = \bigcup_{k>0} \Omega_k$, where Ω_k consists of functions $f :^k \to$.

In other words, equivalent matrices cannot be distinguished by sentences in the matrix query language under consideration. One could imagine defining equivalence with regards to arbitrary expressions, i.e., expressions in MATLANG that are not necessarily sentences. Such a notion would be too strong, however. Indeed, requiring that $e(A) = e(B)$ for arbitrary expressions $e(X)$ would imply that $A = B$ (just consider $e(X) := X$)) and then the story ends.

We aim to *characterise* equivalence of matrices for various matrix query languages. We will, however, not treat this problem in full generality and instead only consider equivalence of *adjacency matrices of undirected graphs*. We leave the generalisation to directed graphs and to arbitrary matrices for future work. Definition 4.1, when applied to adjacency matrices naturally result in the following notion of *equivalence of graphs*.

Definition 4.2 Two graphs G and H of the same order are said to be $\mathsf{ML}(\mathsf{op_1}, \ldots, \mathsf{op_k})$-*equivalent*, denoted by $G \equiv_{\mathsf{ML}(\mathsf{op_1}, \ldots, \mathsf{op_k})} H$, if and only if their adjacency matrices are $\mathsf{ML}(\mathsf{op_1}, \ldots, \mathsf{op_k})$-equivalent.

In the following sections we consider equivalence of graphs for various fragments, starting from simple fragments only supporting a couple of linear algebra operations, up to the full MATLANG matrix query language.

5 Expressive power of the matrix query language $\mathsf{ML}(\cdot, \mathsf{tr})$

The smallest fragment, in terms of the number of supported operations, that we consider is $\mathsf{ML}(\cdot, \mathsf{tr})$, i.e., the matrix query language in which only matrix multiplication and the trace operation are supported. This is a very restrictive fragment since the only sentences that one can formulate are of the form (i) $\#\mathsf{cwalk}_k(X){:=}\mathsf{tr}(X^k)$, where X^k stands for the k^{th} power of X, i.e., X multiplied k times with itself, and (ii) products of such sentences. We note that, when evaluated on an adjacency matrix A_G, $\#\mathsf{cwalk}_k(A_G)$ is equal to the number of closed walks of length k in G. Indeed, an entry $(A_G^k)_{v,w}$ of the k^{th} power A_G^k of adjacency matrix A_G can be easily seen to correspond to the number of walks from v to w of length k in G. Hence, $\#\mathsf{cwalk}_k(A_G)=\mathsf{tr}(A_G^k)=\sum_{v\in V}(A_G^k)_{vv}$ indeed corresponds to the number of closed walks of length k in G.

The following (folklore) characterisations are known to hold.

Proposition 5.1 *Let G and H be two graphs of the same order. The following statements are equivalent:*

(1) G and H have the same number of closed walks of length k, for all $k\geq 0$;
(2) $\mathsf{tr}(A_G^k)=\mathsf{tr}(A_H^k)$ for all $k\geq 0$;
(3) G and H are co-spectral; and
(4) there exists an orthogonal matrix O such that $A_G\cdot O=O\cdot A_H$.

Proof For a proof of the equivalences (1) $\Leftrightarrow$ (2) $\Leftrightarrow$ (3) we refer to Proposition 1 in [23] (although these equivalences appeared in the literature many times before). The equivalence (3) $\Leftrightarrow$ (4) is also known (see e.g., Theorem 9-12 in [59]). ◻

Example 5.1 The graphs G_1 () and H_1 () are the smallest pair (in terms of number of vertices) of non-isomorphic co-spectral graphs of the same order (see e.g., Figure 6.2 in [16]). From the previous proposition we then know that G_1 and H_1 have the same number of closed walks of any length. We note that the isolated vertex in G_1 ensures that G_1 and H_1 have the same number of vertices (and thus the same number of closed walks of length 0). ◻

As expected, sentences in $\mathsf{ML}(\cdot,\mathsf{tr})$ can only extract information from adjacency matrices related to the number of closed walks in graphs. More precisely, we can add to Proposition 5.1 a fifth equivalent condition based on $\mathsf{ML}(\cdot,\mathsf{tr})$-equivalence:

Proposition 5.2 *For two graphs G and H of the same order, $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr})}H$ if and only if G and H have the same number of closed walks of any length.*

Proof By definition, if $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr})}H$, then $e(A_G)=e(A_H)$ for any sentence $e(X)$ in $\mathsf{ML}(\cdot,\mathsf{tr})$. This holds in particular for the sentences $\#\mathsf{cwalk}_k(X){:=}\mathsf{tr}(X^k)$ in $\mathsf{ML}(\cdot,\mathsf{tr})$, for $k\geq 1$. Hence, G and H have indeed the same number of closed walks of length k, for $k\geq 1$. Furthermore, since G and H are of the same order and $A_G^0=A_H^0=I$ (by convention), G and H have also the same number of closed walks of length 0.

For the converse, if G and H have the same number of closed walks of any length, then the previous proposition tells that $A_G\cdot O=O\cdot A_H$ for some orthogonal matrix O. We next claim that when $A_G\cdot O=O\cdot A_H$ holds for some orthogonal matrix O, then $e(A_G)=e(A_H)$ for any sentence $e(X)$ in $\mathsf{ML}(\cdot,\mathsf{tr})$. In fact, this claim will follow from the more general Lemmas 5.1 and 5.2 below. We separate these Lemmas from the current proof since we also need them later in the paper. ◻

We thus see that yet another interpretation of $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr})} H$ can be given in terms of the homomorphism vectors mentioned in the Introduction. That is, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr})} H$ if and only if $\mathsf{HOM}_{\mathcal{F}}(G) = \mathsf{HOM}_{\mathcal{F}}(H)$ where $\mathcal{F}$ is the set of all cycles [25].

As mentioned in the proof of Proposition 5.2, we still need to show that if $A_G \cdot O = O \cdot A_H$ holds for some orthogonal matrix O, then $e(A_G) = e(A_H)$ for any sentence $e(X)$ in $\mathsf{ML}(\cdot,\mathsf{tr})$. In more generality, we refer to the existence of a (not necessarily orthogonal) matrix T such that $A_G \cdot T = T \cdot A_H$ holds, by saying that A_G and A_H are T-*similar*. We also need the notion of T-similarity for vectors and scalars, as is defined next.

Definition 5.1 Let T be a matrix in $^{n\times n}$. Two matrices A and B in $^{n\times n}$ are called T-*similar* if $A \cdot T = T \cdot B$. Two vectors A and B in $^{n\times 1}$ are T-*similar* if $A = T \cdot B$. Similarly, two vectors A and B in $^{1\times n}$ are T-*similar* if $A \cdot T = B$. Finally, if A and B are scalars in , then A and B are T-similar if $A = B$ (i.e., T-similarity of scalars is simply equality).

In $\mathsf{ML}(\cdot,\mathsf{tr})$ we allow matrix multiplication and the trace operation. We first show that T-similarity is preserved by matrix multiplication for *any* matrix T.

Lemma 5.1 *Let A_G and A_H be two adjacency matrices of the same dimensions. Let $e_1(X)$ and $e_2(X)$ be two expressions in $\mathsf{ML}(\mathcal{L})$ for any $\mathcal{L}$. If $e_i(A_G)$ and $e_i(A_H)$ are T-similar, for $i = 1,2$, for an arbitrary matrix T, then $e_1(A_G) \cdot e_2(A_G)$ is also T-similar to $e_1(A_H) \cdot e_2(A_H)$ (provided, of course, that the multiplication is well-defined).*

Proof The proof consists of a simple case analysis depending on the dimensions of $e_1(A_G)$ and $e_2(A_G)$ (or equivalently, the dimensions of $e_1(A_H)$ and $e_2(A_H)$) and by using the definition of T-similarity. We refer for the proof to the appendix. ☐

When considering the trace operation, we observe that T-similarity is preserved by the trace operation for any *invertible* matrix T.

Lemma 5.2 *Let A_G and A_H be two adjacency matrices of the same dimensions. Let $e_1(X)$ be an expression in $\mathsf{ML}(\mathcal{L})$ for any $\mathcal{L}$. If $e_1(A_G)$ and $e_1(A_H)$ are T-similar for an invertible matrix T, then $\mathsf{tr}(e_1(A_G))$ and $\mathsf{tr}(e_1(A_H))$ are also T-similar.*

Proof Let $e(X) := \mathsf{tr}(e_1(X))$. By assumption, $e_1(A_G) \cdot T = T \cdot e_1(A_H)$ for an invertible matrix T in case that $e_1(A_G)$ is an $n \times n$-matrix, and $e_1(A_G) = e_1(A_H)$ in case that $e_1(A_G)$ is a sentence. In the latter case, clearly also $e(A_G) = \mathsf{tr}(e_1(A_G)) = \mathsf{tr}(e_1(A_H)) = e(A_H)$. In the former case, we use the property that $\mathsf{tr}(T^{-1} \cdot A \cdot T) = \mathsf{tr}(A)$ for any matrix A and invertible matrix T (see e.g., Chapter 10 in [5] for a proof of this property). Hence, we have that $e(A_G) = \mathsf{tr}(e_1(A_G)) = \mathsf{tr}(T^{-1} \cdot e_1(A_G) \cdot T) = \mathsf{tr}(T^{-1} \cdot T \cdot e_1(A_H)) = \mathsf{tr}(I \cdot e_1(A_H)) = \mathsf{tr}(e_1(A_H)) = e(A_H)$ holds, as desired. ☐

We remark that Lemmas 5.1 and 5.2 hold for *any* fragment $\mathsf{ML}(\mathcal{L})$.

The claim at the end of the proof of Proposition 5.2, i.e., O-similarity of A_G and A_H indeed implies that $e(A_G) = e(A_H)$ for any sentence $e(X) \in \mathsf{ML}(\cdot,\mathsf{tr})$, now easily follows by induction on the structure of expressions, Indeed, since orthogonal matrices are invertible, Lemmas 5.1 and 5.2 imply that when $e_1(A_G)$ and $e_1(A_H)$, and $e_2(A_G)$ and $e_2(A_H)$ are O-similar for an orthogonal matrix O, then also $e_1(A_G) \cdot e_2(A_G)$ and

$e_1(A_H)\cdot e_2(A_H)$ are O-similar, and $\mathsf{tr}(e_1(A_G))$ and $\mathsf{tr}(e_1(A_H))$ are O-similar (i.e., equal). Hence, when A_G and A_H are O-similar, $e(A_G)$ and $e(A_H)$ are O-similar for any sentence $e(X)\in\mathsf{ML}(\cdot,\mathsf{tr})$. That is, $e(A_G)=e(A_H)$ for any sentence in $\mathsf{ML}(\cdot,\mathsf{tr})$.

5.1 Adding operations to $\mathsf{ML}(\cdot,\mathsf{tr})$ without increasing its distinguishing power

We conclude this section by investigating how much more $\mathsf{ML}(\cdot,\mathsf{tr})$ can be extended whilst preserving the characterisation given in Proposition 5.2. Some more general observations will be made in this context, which will be used for other fragments later in the paper as well.

First, we consider the extension with scalar multiplication ($\times$) and addition ($+$).

Lemma 5.3 *Let $\mathsf{ML}(\mathcal{L})$ be any matrix query language fragment. Let $e_1(X)$ and $e_2(X)$ be two expressions in $\mathsf{ML}(\mathcal{L})$ and consider two graphs G and H of the same order. Then, if $e_1(A_G)$ and $e_1(A_H)$, and $e_2(A_G)$ and $e_2(A_H)$ are T-similar for some matrix T, then also $e_1(A_G)+e_2(A_G)$ and $e_1(A_H)+e_2(A_H)$ are T-similar, and $a\times e_1(A_G)$ and $a\times e_1(A_H)$ are T-similar for any scalar $a\in C$.*

Proof This is an immediate consequence of the definition of T-similarity and that matrix multiplication is a bilinear operation, i.e., $(a\times A+b\times B)\cdot(c\times C+d\times D)=(a\times c)\times(A\cdot C)+(a\times d)\times(A\cdot D)+(b\times c)\times(B\cdot C)+(b\times d)\times(B\cdot D)$, for scalars a, b, c, $d\in$ and matrices or vectors A,B,C and D. □

We next consider complex conjugate transposition (*).

Lemma 5.4 *Let $\mathsf{ML}(\mathcal{L})$ be any matrix query language fragment. Let $e(X)$ be an expression in $\mathsf{ML}(\mathcal{L})$ and consider two graphs G and H of the same order. Then, if $e(A_G)$ and $e(A_H)$ are T-similar, and $e(A_H)$ and $e(A_G)$ are T^*-similar for some matrix T, then also $(e(A_G))^*$ and $(e(A_H))^*$ are T-similar, and $(e(A_H))^*$ and $(e(A_G))^*$ are T^*-similar.*

Proof We distinguish between a number of cases, depending on the dimensions of $e(A_G)$ (and hence also of $e(A_H)$). Suppose that $e(A_G)$ returns an $n\times n$-matrix. Then, by assumption $e(A_G)\cdot T=T\cdot e(A_H)$ and $e(A_H)\cdot T^*=T^*\cdot e(A_G)$. It then follows, using that the operation * is an involution ($(A^*)^*=A$) and $(A\cdot B)^*=B^*\cdot A^*$, that

$$(e(A_G))^*\cdot T=(T^*\cdot e(A_G))^*=(e(A_H)\cdot T^*)^*=T\cdot(e(A_H))^*,$$

and similarly,

$$(e(A_H))^*\cdot T^*=(T\cdot e(A_H))^*=(e(A_G)\cdot T)^*=T^*\cdot(e(A_G))^*.$$

Furthermore, when $e(A_G)$ is an $n\times 1$-vector, we have by assumption that $e(A_G)=T\cdot e(A_H)$ and $e(A_H)=T^*\cdot e(A_G)$. Hence, $(e(A_G))^*\cdot T=(T^*\cdot e(A_G))^*=(e(A_H))^*$ and $(e(A_H))^*\cdot T^*=(T\cdot e(A_H))^*=(e(A_G))^*$. Similarly, when $e(A_G)$ is a $1\times n$-vector, one can verify that $((e(A_G))^*=T\cdot(e(A_H))^*$ and $(e(A_H))^*=T^*\cdot(e(A_G))^*$. Finally, if $e(A_G)$ is a sentence then clearly $(e(A_G))^*=(e(A_H))^*$. □

We next consider pointwise function applications. Later in the paper we show that pointwise function applications on vectors or matrices do add expressive power. By contrast, when such function applications are *only allowed on scalars* they do not add any expressive power. More precisely, let $f :^k \to$ be a function in Ω. We denote by $\mathsf{apply}_\mathsf{s}[f](e_1,\ldots,e_k)$ the application of f on $e_1(X),\ldots,e_k(X)$ when each $e_i(X)$ is a sentence.

Lemma 5.5 *Let* $\mathsf{ML}(\mathcal{L})$ *be any matrix query language fragment. Consider two graphs* G *and* H *of the same order and sentences* $e_1(X), e_2(X),\ldots,e_k(X)$ *in* $\mathsf{ML}(\mathcal{L})$*. Let* $f :^k \to$ *be a function in* Ω*. Suppose that for each* $i = 1,\ldots,k$, $e_i(A_G) = e_i(A_H)$ *(i.e., they are* T*-similar for any matrix* T*). Then also* $\mathsf{apply}_\mathsf{s}[f](e_1(A_G),\ldots,e_k(A_G)) = \mathsf{apply}_\mathsf{s}[f](e_1(A_H),\ldots,e_k(A_H))$ *(i.e., they are* T*-similar as well).*

Proof This is straightforward since the result of a function $f :^k \to$ is fully determined by its input values. □

Given these lemmas, we can infer that the characterisation given in Proposition 5.2 remains to hold for $\mathsf{ML}(\cdot,\mathsf{tr},+,\times,^*,\mathsf{apply}_\mathsf{s}[f], f\in\Omega)$-equivalence.

Corollary 5.1 *For two graphs* G *and* H *of the same order,* $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr})} H$ *if and only if* $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},+,\times,^*,\mathsf{apply}_\mathsf{s}[f],f\in\Omega)} H$. □

Proof We only need to show that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr})} H$ implies $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},+,\times,^*,\mathsf{apply}_\mathsf{s}[f],f\in\Omega)} H$. By Proposition 5.2, there exists an orthogonal matrix O such that $A_G \cdot O = O \cdot A_H$. Furthermore, we have that $O^* \cdot A_G = (A_G \cdot O)^* = (O \cdot A_H)^* = A_H \cdot O^*$ since A_G and A_H are symmetric real matrices. Hence, A_H and A_G are O^*-similar. We also, importantly, observe that O^* is an orthogonal matrix as well. Lemmas 5.1 and 5.2 then imply that $e(A_G)$ and $e(A_H)$ are O-similar, and $e(A_H)$ and $e(A_G)$ are O^*-similar for any expression $e(X)$ in $\mathsf{ML}(\cdot,\mathsf{tr})$. Furthermore, Lemmas 5.3, 5.4 and 5.5 imply that addition, scalar multiplication, complex conjugate transposition and pointwise function applications on scalars preserve O and O^*-similarity. This in turn implies that $e(A_G) = e(A_H)$ for any sentence $e(X) \in \mathsf{ML}(\cdot, \mathsf{tr},+,\times,^*,\mathsf{apply}_\mathsf{s}[f], f\in\Omega)$. □

As a consequence, the graphs G_1 () and H_1 () from Example 5.1 cannot be distinguished by sentences in $\mathsf{ML}(\cdot,\mathsf{tr},+,\times,^*,\mathsf{apply}_\mathsf{s}[f], f\in\Omega)$. As we will see later, including any other operation from Table 3.1, such as $\mathbb{1}(\cdot)$, $\mathsf{diag}(\cdot)$ or pointwise function applications on vectors or matrices, allows us to distinguish G_1 and H_1.

6 The impact of the $\mathbb{1}(\cdot)$ operation

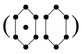
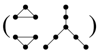
The $\mathbb{1}(\cdot)$ operation, which returns the all-ones vector $\mathbb{1}$[1], allows to extract other information from graphs than just the number of closed walks. Indeed, consider the sentences

$$\#\mathsf{walk}_k(X) := (\mathbb{1}(X))^* \cdot X^k \cdot \mathbb{1}(X) \text{ and } \#\mathsf{walk}'_k(X) := \mathsf{tr}(X^k \cdot \mathbb{1}(X)),$$

in $\mathsf{ML}(\cdot,^*,\mathbb{1})$ and $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$, respectively. When applied on adjacency matrix A_G of a graph G, $\#\mathsf{walk}_k(A_G)$ (and also $\#\mathsf{walk}'_k(A_G)$) returns the number of (not

[1] We use $\mathbb{1}$ to denote the all-ones *vector* (of appropriate dimension) and use $\mathbb{1}(\cdot)$ (with brackets) for the corresponding one-vector *operation*.

necessarily closed) walks in G of length k. In relation to the previous section, co-spectral graphs have the same number of closed walks of any length, yet do not necessarily have the same number of walks of any length. Similarly, graphs with the same number of walks of any length are not necessarily co-spectral.

Example 6.1 It can be verified that the co-spectral graphs G_1 () and H_1 () of Example 5.1 have 16 versus 20 walks of length 2, respectively. As a consequence, $\mathsf{ML}(\cdot,{}^*,\mathbb{1})$ and $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$ can distinguish G_1 from H_1 by means of the sentences $\#\mathsf{walk}_2(X)$ and $\#\mathsf{walk}'_2(X)$, respectively. By contrast, the graphs G_2 () and H_2 () are not co-spectral, yet have the same number of walks of any length. It is easy to see that G_2 and H_2 are not co-spectral (apart from verifying that their spectra are different): H_2 has 12 closed walks of length 3 (because of the triangles), whereas G_2 has no closed walks of length 3. As a consequence, $\mathsf{ML}(\cdot,\mathsf{tr})$ (and thus also $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$) can distinguish G_2 and H_2. We argue below that G_2 and H_2 have the same number of walks of any length and show that $\mathsf{ML}(\cdot,{}^*,\mathbb{1})$ cannot distinguish G_2 and H_2. □

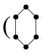
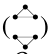
The previous example illustrates the key difference between $\mathsf{ML}(\cdot,{}^*,\mathbb{1})$ and $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$. The former can only detect differences in the number of walks of certain lengths, the latter can detect differences in both the number of walks and closed walks of certain lengths.

Graphs sharing the same number of walks of any length have been investigated before in spectral graph theory [17, 18, 39, 61]. To state a spectral characterisation, the so-called *main spectrum* of a graph needs to be considered. The main spectrum of a graph is the set of eigenvalues whose eigenspace is not orthogonal to the $\mathbb{1}$ vector. More formally, consider an eigenvalue λ and its corresponding eigenspace, represented by a matrix V whose columns are eigenvectors of λ that span the eigenspace of λ. Then, the *main angle* β_λ of λ's eigenspace is $\frac{1}{\sqrt{n}}\|V^{\mathsf{t}}\cdot\mathbb{1}\|_2$, where $\|\cdot\|_2$ is the Euclidean norm. The *main eigenvalues* are now simply those eigenvalues with a non-zero main angle. Furthermore, two graphs are said to be *co-main* if they have the same set of main eigenvalues and corresponding main angles. Intuitively, the importance of the orthogonal projection on $\mathbb{1}$ stems from the observation that $\#\mathsf{walk}_k(A_G)$ can be expressed as $\sum_i \lambda_i^k \beta_{\lambda_i}^2$ where the λ_i's are the distinct eigenvalues of A_G. Clearly, only those eigenvalues λ_i for which β_{λ_i} is non-zero matter when computing $\#\mathsf{walk}_k(A_G)$. This results in the following characterisation.

Proposition 6.1 (Theorem 1.3.5 in Cvetković et al. [19]) *Two graphs G and H of the same order are co-main if and only if they have the same number of walks of length k, for every $k \geq 0$.* □

Furthermore, the following proposition follows implicitly from the proof of Theorem 3 in van Dam et al. [68]. This proposition is also explicitly proved more recently in Theorem 1.2 in Dell et al. [25] in the context of distinguishing graphs by means of homomorphism vectors $\mathsf{HOM}_{\mathcal{F}}(G)$ and $\mathsf{HOM}_{\mathcal{F}}(H)$ where $\mathcal{F}$ consists of all paths.

Proposition 6.2 *Two graphs G and H of the same order have the same number of walks of length k, for every $k \geq 0$, if and only if there is a doubly quasi-stochastic matrix Q such that $A_G \cdot Q = Q \cdot A_H$.* □

Example 6.2 (Continuation of Example 6.1) Consider the subgraph G_3 () of G_2 and the subgraph H_3 () of H_2. It is readily verified that there exists a doubly quasi-stochastic matrix Q such that $A_{G_3} \cdot Q = Q \cdot A_{H_3}$. Indeed, $A_{G_3} \cdot Q$ is equal to

$$\begin{bmatrix} 0&0&1&0&0&1\\ 0&0&0&1&1&0\\ 1&0&0&0&1&0\\ 0&1&0&0&0&1\\ 0&1&1&0&0&0\\ 1&0&0&1&0&0 \end{bmatrix} \cdot \begin{bmatrix} 0&0&\frac{1}{2}&0&0&\frac{1}{2}\\ 0&0&0&\frac{1}{2}&\frac{1}{2}&0\\ \frac{1}{2}&0&0&0&\frac{1}{2}&0\\ 0&\frac{1}{2}&0&0&0&\frac{1}{2}\\ 0&\frac{1}{2}&\frac{1}{2}&0&0&0\\ \frac{1}{2}&0&0&\frac{1}{2}&0&0 \end{bmatrix} = \begin{bmatrix} 0&0&\frac{1}{2}&0&0&\frac{1}{2}\\ 0&0&0&\frac{1}{2}&\frac{1}{2}&0\\ \frac{1}{2}&0&0&0&\frac{1}{2}&0\\ 0&\frac{1}{2}&0&0&0&\frac{1}{2}\\ 0&\frac{1}{2}&\frac{1}{2}&0&0&0\\ \frac{1}{2}&0&0&\frac{1}{2}&0&0 \end{bmatrix} \cdot \begin{bmatrix} 0&0&1&0&0&1\\ 0&0&0&1&1&0\\ 1&0&0&0&1&0\\ 0&1&0&0&0&1\\ 0&1&1&0&0&0\\ 1&0&0&1&0&0 \end{bmatrix},$$

which is equal to $Q \cdot A_{H_3}$. Hence by Proposition 6.2, G_3 and H_3 have the same number of walks on any length. □

Just as for the fragment $\mathsf{ML}(\cdot, \mathsf{tr})$ (Proposition 5.2), it turns out that sentences in $\mathsf{ML}(\cdot, {}^*, \mathbb{1})$ can only extract information from adjacency matrices related to the number of walks in graphs. More precisely,

Proposition 6.3 *Let G and H be two graphs of the same order. Then, $G \equiv_{\mathsf{ML}(\cdot,{}^*,\mathbb{1})} H$ if and only if G and H have the same number of walks of any length.*

Proof It is straightforward to show that $G \equiv_{\mathsf{ML}(\cdot,{}^*,\mathbb{1})} H$ implies that G and H must have the same number of walks of any length. This follows from the same argument as given in the proof of Proposition 5.2. For the converse, we use the characterisation given in Proposition 6.2. That is, if G and H have the same number of walks of any length, then there exists a doubly quasi-stochastic matrix Q such that $A_G \cdot Q = Q \cdot A_H$. In other words, A_G and A_H are Q-similar. We then show that when A_G and A_H are Q-similar, for a doubly quasi-stochastic matrix Q, then $e(A_G) = e(A_H)$ for all sentences $e(X)$ in $\mathsf{ML}(\cdot, {}^*, \mathbb{1})$. We here rely on a more general result (Lemma 6.1 below), which states that T-similarity is preserved by the operation $\mathbb{1}(\cdot)$ provided that T is a quasi-stochastic matrix T, i.e., $T \cdot \mathbb{1} = \mathbb{1}$. We again separate this Lemma from the current proof because we need it also later in the paper. This suffices to conclude that expressions in $\mathsf{ML}(\cdot, {}^*, \mathbb{1})$ preserve Q-similarity. Indeed, to deal with complex conjugate transposition, we note that $A_G \cdot Q = Q \cdot A_H$ implies that $A_H \cdot Q^* = (Q \cdot A_H)^* = (A_G \cdot Q)^* = Q^* \cdot A_G$ since A_G and A_H are symmetric real matrices. Furthermore, since Q is a real matrix and quasi doubly-stochastic, also $Q^* \cdot \mathbb{1} = \mathbb{1}$ holds. That is, Q^* is a (doubly) quasi-stochastic matrix as well. Hence, Lemmas 5.1 and 6.1 imply that Q-similarity and Q^*-similarity are preserved by matrix multiplication and the one-vector operation. Combined with Lemma 5.4, we may indeed conclude that Q-similarity and Q^*-similarity is also preserved by complex conjugate transposition. Hence, by induction on the structure of expressions, $e(A_G) = e(A_H)$ for any sentence $e(X) \in \mathsf{ML}(\cdot, {}^*, \mathbb{1})$. □

We now show that T-similarity is preserved under the one-vector operation for any quasi-stochastic matrix T. In fact, since the result of $\mathbb{1}(\cdot)$ is only dependent on the dimensions of the input, we have do not even need the T-similarity assumption on the inputs.

Lemma 6.1 *Let A_G and A_H be two adjacency matrices of the same dimensions. Let $e_1(X)$ be an expression in $\mathsf{ML}(\mathcal{L})$ for any $\mathcal{L}$. Then, $\mathbb{1}(e_1(A_G))$ and $\mathbb{1}(e_1(A_H))$ are T-similar for any quasi-stochastic matrix T.*

Proof The proof is straightforward. Let $e(X) := \mathbb{1}(e_1(X))$. We distinguish between the following cases, depending on the dimensions of $e_1(A_G)$. If $e_1(A_G)$ is an $n\times n$-matrix or $n\times 1$-vector, then $e(A_G) = e(A_H) = \mathbb{1}$ and $e(A_G) = \mathbb{1} = T\cdot\mathbb{1} = T\cdot e(A_H)$. Furthermore, if $e_1(A_G)$ is a $1\times n$-vector or sentence, then $e(A_G) = e(A_H) = [1]$ and thus these agree and are T-similar. □

We next turn our attention to $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$. We know from Propositions 5.1 and 5.2 that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})} H$ implies that G and H are co-spectral. Combined with Proposition 6.1 and the fact that the sentence $\#\mathsf{walk}'_k(X)$ counts the number of walks of length k, we have that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})} H$ implies that G and H are co-spectral and co-main. The following is known about such graphs.

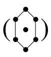
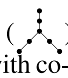
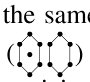
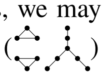
Proposition 6.4 (Corollary to Theorem 2 in Johnson and Newman [46]) *Two co-spectral graphs G and H of the same order are co-main if and only if there exists an orthogonal matrix O such that $A_G\cdot O = O\cdot A_H$ and $O\cdot\mathbb{1} = \mathbb{1}$.* □

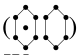
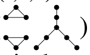
In other words, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})} H$ implies the existence of an orthogonal matrix O such that $O\cdot\mathbb{1} = \mathbb{1}$ (i.e., O is also quasi-stochastic) and $A_G\cdot O = O\cdot A_H$. We can now use Lemmas 5.1, 5.2 and 6.1 to show the converse. Indeed, these lemmas combined tell us that $A_G\cdot O = O\cdot A_H$ implies that $e(A_G) = e(A_H)$ for any sentence $e(X)$ in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$. As a consequence:

Proposition 6.5 *For two graphs G and H of the same order, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})} H$ if and only if G and H have the same number of closed walks and the same number of walks of any length if and only if $A_G\cdot O = O\cdot A_H$ for an orthogonal matrix O such that $O\cdot\mathbb{1} = \mathbb{1}$.* □

We can also phrase $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$-equivalence in terms of homomorphism vectors. That is, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})} H$ if and only if $\mathsf{HOM}_{\mathcal{F}}(G) = \mathsf{HOM}_{\mathcal{F}}(H)$, where $\mathcal{F}$ now consists of all cycles and paths. This complements the results in Dell et al. [25].

As a note aside, an alternative characterisation to Proposition 6.4 (Theorem 3 in van Dam et al. [68]) is that G and H are co-spectral and co-main if and only if both G and H *and* their complement graphs $\bar{G}$ and $\bar{H}$ are co-spectral. Here, the complement graph $\bar{G}$ of G is the graph with adjacency matrix given by $J - A_G - I$, where J is the all-ones matrix, and similarly for $\bar{H}$.

Example 6.3 (Continuation of Example 6.1) Consider the subgraph G_4 () of G_2 and the subgraph H_4 () of H_2. These are known to be the smallest non-isomorphic co-spectral graphs with co-spectral complements (see e.g., Figure 4 in [38]). From the previous remark it then follows that G_4 and H_4 have the same number of (closed) walks of any length. These graphs are thus indistinguishable by sentences in $\mathsf{ML}(\cdot,{}^*,\mathbb{1})$ and $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$. Combined with our earlier observation in Example 6.2 that also G_3 and H_3 have the same number of walks, we may conclude that the disjoint unions $G_2 = G_3 \cup G_4$ () and $H_2 = H_3 \cup H_4$ () have the same number of walks of any length, as anticipated in Example 6.1. □

We remark that as a consequence of Propositions 6.3 and 6.5, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})} H$ implies that $G \equiv_{\mathsf{ML}(\cdot,^*,\mathbb{1})} H$. We already mentioned in Example 6.1 that the graphs G_2 () and H_2 () show that the converse does not hold.

We conclude again by observing that addition, scalar multiplication and pointwise function application on scalars can be added to $\mathsf{ML}(\cdot,^*,\mathbb{1})$ and $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$ at no increase in expressiveness. Similarly, conjugate transposition can be included in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$.

Corollary 6.1 *Let G and H be two graphs of the same order. Then,*

(1) $G \equiv_{\mathsf{ML}(\cdot,^*,\mathbb{1},+,\times,\mathsf{apply_s}[f],f\in\Omega)} H$ if and only if $G \equiv_{\mathsf{ML}(\cdot,^*,\mathbb{1})} H$; and
(2) $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},+,\times,\mathsf{apply_s}[f],f\in\Omega)} H$ if and only if $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})} H$.

Proof (1) We only need to show that $G \equiv_{\mathsf{ML}(\cdot,^*,\mathbb{1})} H$ implies $G \equiv_{\mathsf{ML}(\cdot,^*,\mathbb{1},+,\times,\mathsf{apply_s}[f],f\in\Omega)} H$. We have that $G \equiv_{\mathsf{ML}(\cdot,^*,\mathbb{1})} H$ implies $A_G \cdot Q = Q \cdot A_H$ for a doubly quasi-stochastic matrix Q (Proposition 6.3). Furthermore, in the proof of Proposition 6.3 we have shown that $A_H \cdot Q^* = Q^* \cdot A_G$ where Q^* is again a doubly quasi-stochastic matrix. Lemmas 5.1, 5.3, 5.4, 5.5 and 6.1 imply that Q-similarity and Q^*-similarity are preserved by all operations in $\mathsf{ML}(\cdot,^*,\mathbb{1},+,\times,\mathsf{apply_s}[f], f\in\Omega)$.

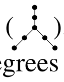
(2) We only need to show that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})} H$ implies $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},+,\times,\mathsf{apply_s}[f],f\in\Omega)} H$. We have that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})} H$ implies $A_G \cdot O = O \cdot A_H$ for an orthogonal quasi-stochastic matrix O (Proposition 6.5). We observe that $A_H \cdot O^* = O^* \cdot A_G$ and furthermore, $O^* \cdot \mathbb{1} = O^* \cdot O \cdot \mathbb{1} = \mathbb{1}$. Hence, O^* is an orthogonal quasi-stochastic matrix as well. Lemmas 5.1, 5.2, 5.3, 5.4, 5.5 and 6.1, imply that O-similarity and O^*-similarity are preserved by all operations in $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},+,\times,\mathsf{apply_s}[f], f\in\Omega)$.

In both cases, we can therefore conclude, by induction on the structure of expressions, that for any sentence $e(X)$, $e(A_G)$ and $e(A_H)$ are similar and hence, $e(A_G) = e(A_H)$. □

As we will see later, including any other operation from Table 3.1, such as $\mathsf{diag}(\cdot)$ or pointwise function applications on vectors or matrices, allows us to distinguish G_4 and H_4. We recall from Example 6.3 that these graphs cannot be distinguished by sentences in $\mathsf{ML}(\cdot,^*,\mathbb{1})$ and $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$.

7 The impact of the diag(·) operation

We next consider the operation $\mathsf{diag}(\cdot)$ which takes a vector as input and returns the diagonal matrix with the input vector on its diagonal. The smallest fragments in which vectors (and sentences) can be defined are $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1})$ and $\mathsf{ML}(\cdot,^*,\mathbb{1})$. Therefore, in this section we consider equivalence with regards to $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$ and $\mathsf{ML}(\cdot,^*,\mathbb{1},\mathsf{diag})$. Using $\mathsf{diag}(\cdot)$ we can again extract new information from graphs, as is illustrated in the following example.

Example 7.1 Consider graphs G_4 () and H_4 (). In G_4 we have vertices of degrees 0 and 2, and in H_4 we have vertices of degrees 1, 2 and 3. We will count the number of vertices of degree 3. Given that we know that 3 is an upper bound on the degrees of vertices in G_4 and H_4, we consider the sentence $\#3\mathsf{degr}(X)$ given by

$$\left(\frac{1}{6}\right)\times(\mathbb{1}(X))^{*}\cdot\big(\mathsf{diag}(X\cdot\mathbb{1}(X)-0\times\mathbb{1}(X))\cdot\mathsf{diag}(X\cdot\mathbb{1}(X)-1\times\mathbb{1}(X))$$
$$\cdot\,\mathsf{diag}(X\cdot\mathbb{1}(X)-2\times\mathbb{1}(X))\big)\cdot\mathbb{1}(X),$$

in which we, for convenience, allow addition and scalar multiplications. Each of the subexpressions $\mathsf{diag}(X\cdot\mathbb{1}(X)-d\times\mathbb{1}(X))$, for $d=0,1$ and 2, sets the diagonal entry corresponding to vertex v to 0 when v has degree d. By taking the product of these diagonal matrices, entries that are set to 0 will remain zero in the resulting diagonal matrix. This implies that the only non-zero diagonal entries are those corresponding to vertices of degree different from 0, 1 and 2. In other words, only for vertices of degree 3 the diagonal entries carry a non-zero value, i.e., the value $6=(3-0)(3-1)(3-2)$. By appropriately rescaling by the factor $\frac{1}{6}$, the diagonal entries for the degree three vertices are set to 1, and then summed up. Hence, $\#3\mathsf{degr}(X)$ indeed counts the number vertices of degree three when evaluated on adjacency matrices of graphs with vertices of maximal degree 3. Since $\#3\mathsf{degr}(A_{G_4})=[0]\neq[1]=\#3\mathsf{degr}(A_{H_4})$ we can distinguish G_4 and H_4. We can obtain similar expressions for $\#d\mathsf{degr}(X)$ for arbitrary d, provided that we know the maximal degree of vertices in the graph. The way that these expressions are constructed is similar to the so-called Schur-Wielandt Principle indicating how to extract entries from a matrix that hold specific values by means of pointwise multiplication of matrices (see e.g., Proposition 1.4 in [58]). Here, we do not have pointwise matrix multiplication available but since we extract information from vectors, pointwise multiplication of vectors is simulated by normal matrix multiplication of diagonal matrices with the vectors on their diagonals. □

The use of the diagonal matrices and their products as in our example sentence $\#3\mathsf{degr}(X)$ can also be generalised to obtain information about so-called *iterated degrees* of vertices in graphs, e.g., to identify and/or count vertices that have a number of neighbours each of which have neighbours of specific degrees, and so on. Such iterated degree information is closely related to *equitable partitions* and *fractional isomorphisms* of graphs (see e.g., Chapter 6 in [62]). We phrase our results in terms of equitable partitions instead of iterated degree sequences.

7.1 Equitable partitions

Formally, an *equitable partition* $\mathcal{V}=\{V_1,\ldots,V_\ell\}$ of G is partition of the vertex set V of G such that for all $i,j=1,\ldots,\ell$ and $v,v'\in V_i$, $\mathsf{deg}(v,V_j)=\mathsf{deg}(v',V_j)$. Here, $\mathsf{deg}(v,V_j)$ is the number of vertices in V_j that are adjacent to v. In other words, an equitable partition is such that the graph is regular within each part, i.e., all vertices in a part have the same degree, and is bi-regular between any two different parts, i.e., the number of edges between any two vertices in two different parts is constant. A graph always has a *trivial* equitable partition: simply treat each vertex as a part by its own. Most interesting is the *coarsest* equitable partition of a graph, i.e., the *unique* equitable partition for which any other equitable partition of the graph is a refinement thereof [62]. The conditions underlying equitable partitions can be equivalently stated in terms of adjacency matrices and indicator vectors describing the partition. More precisely, any partition $\mathcal{V} = \{V_1, \dots, V_\ell\}$ of V can be represented by ℓ indicator vectors $\mathbb{1}_{V_1}, \dots, \mathbb{1}_{V_\ell}$ such that: (i) $(\mathbb{1}_{V_i})_v = 1$ if $v \in V_i$ and $(\mathbb{1}_{V_i})_v = 0$ if $v \notin V_i$, for $i = 1, \dots, \ell$. We observe that $\mathbb{1} = \sum_{i=1}^{\ell} \mathbb{1}_{V_i}$ due to $\mathcal{V}$ being a partition. Then, $\mathcal{V}$ is an equitable partition if and only if for all $i, j = 1, \dots, \ell$,

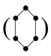
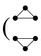
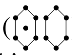
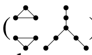
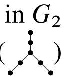
$$\mathsf{diag}(\mathbb{1}_{V_i}) \cdot A_G \cdot \mathbb{1}_{V_j} = \mathsf{deg}(v, V_j) \times \mathbb{1}_{V_i},$$

for some (arbitrary) vertex $v \in V_i$.

Two graphs G and H are said to have a *common* equitable partition if there exists an equitable partition $\mathcal{V} = \{V_1, \dots, V_\ell\}$ of G and an equitable partition $\mathcal{W} = \{W_1, \dots, W_\ell\}$ of H such that (a) the sizes of the parts agree, i.e., $|V_i| = |W_i|$ for each $i = 1, \dots, \ell$, and (b) $\mathsf{deg}(v, V_j) = \mathsf{deg}(w, W_j)$ for any $v \in V_i$ and $w \in W_i$ and any $i, j = 1, \dots, \ell$. We note that, due to condition (b), the trivial partition of graphs do not always result in a common equitable partition. In other words, not every two graphs have a common equitable partition. Proposition 7.1 below characterises when two graphs do have a common equitable partition. Furthermore, when two graphs have a common equitable partition they also have a common coarsest equitable partition (see e.g., Theorem 6.5.1 in [62]).

Equitable partitions naturally arise as the result of the *colour refinement procedure* [7, 35, 69], also known as the 1-dimensional Weisfeiler-Lehman algorithm, used as a subroutine in graph isomorphism solvers. Furthermore, there is a close connection to the study of *fractional isomorphisms* of graphs [62, 65], as already mentioned in the Introduction. We recall: two graphs G and H are said to be fractional isomorphic if there exists a doubly stochastic matrix S such that $A_G \cdot S = S \cdot A_H$. Furthermore, a logical characterisation of graphs with a common equitable partition exists, as is stated next.

Proposition 7.1 (Theorem 1 in Tinhofer [65], Section 4.8 in Immerman and Lander [44]) *Let G and H be two graphs of the same order. Then, G and H are fractional isomorphic if and only if G and H have a common equitable partition if and only if $G \equiv_{\mathsf{C}^2} H$.* □

Example 7.2 The matrix linking the adjacency matrices of G_3 () and H_3 () in Example 6.2 is in fact a doubly stochastic matrix (all its entries are either 0 or $\frac{1}{2}$). Hence, G_3 and H_3 have a common equitable partition, which in this case consists of a single part consisting of all vertices. By contrast, graphs G_2 () and H_2 () do not have a common equitable partition. Indeed, fractional isomorphic graphs must have the same multiset of degrees, i.e., the same multiset consisting of the degrees of vertices (Proposition 6.2.6 in [62]), which does not hold for G_2 and H_2. Indeed, we note that there is an isolated vertex in G_2 but not in H_2. For the same reason, G_1 () and H_1 (), and G_4 () and H_4 () are not fractional isomorphic. □

To relate equitable partitions to $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag})$- and $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})$-equivalence, we show that the presence of $\mathsf{diag}(\cdot)$ allows us to formulate a number of expressions, denoted by $\mathsf{eqpart}_i(X)$, for $i = 1, \dots, \ell$, that together extract the *coarsest equitable partition* from a given graph. By applying these expressions on A_G and A_H, one can

Algorithm 1: Computing the coarsest equitable partition based on algorithm GDCR [48].

Input : Adjacency matrix A_G of G of dimension $n \times n$.
Output : Indicator vectors of coarsest equitable partition of G.

1 Let $B^{(0)} := \mathbb{1}$;
2 Let $i = 1$;
3 **while** $i \leq n$ **do**
4 Let $M^{(i)} := A_G \cdot B^{(i-1)}$;
5 Let $\mathcal{V}^{(i)} := \{V_1^{(i)}, \dots, V_{\ell_i}^{(i)}\}$ a partition such that $v, w \in V_j^{(i)}$ if and only if $M_{v*}^{(i)} = M_{w*}^{(i)}$;
6 Let $B^{(i)} := [\mathbb{1}_{V_1^{(i)}}, \dots, \mathbb{1}_{V_{\ell_i}^{(i)}}]$;
7 Let $i = i + 1$;
8 Return $B^{(n)}$.

use sentences to detect whether these partitions witness that G and H have a common equitable partition. In this subsection, $\mathcal{L}$ can be either $\{\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag}\}$ or $\{\cdot, {}^*, \mathbb{1}, \mathsf{diag}\}$.

Proposition 7.2 *Let G and H be two graphs of the same order. Then, $G \equiv_{\mathsf{ML}(\mathcal{L})} H$ implies that G and H have a common equitable partition.*

Proof We first show that $\mathsf{ML}(\mathcal{L})$ has sufficient power to compute the coarsest equitable partition of a given graph G. In fact, we use addition and scalar multiplication in order to compute these partitions. We denote by $\mathcal{L}^+$ the extension of $\mathcal{L}$ with $+$ and $\times$. When it comes to the equivalence of graphs, it does not matter whether we consider $\mathsf{ML}(\mathcal{L})$- or $\mathsf{ML}(\mathcal{L}^+)$-equivalence [2]. Indeed, expressions in $\mathsf{ML}(\mathcal{L}^+)$ only use linear (or bilinear) operations, i.e., the operations supported in $\mathcal{L}$ and $+$ and $\times$. This implies that any sentence in $\mathsf{ML}(\mathcal{L}^+)$ can be written as a linear combination of sentences in $\mathsf{ML}(\mathcal{L})$. As a consequence, $G \equiv_{\mathsf{ML}(\mathcal{L})} H$ implies $G \equiv_{\mathsf{ML}(\mathcal{L}^+)} H$. Since $G \equiv_{\mathsf{ML}(\mathcal{L}^+)} H$ trivially implies $G \equiv_{\mathsf{ML}(\mathcal{L})} H$, we have that $G \equiv_{\mathsf{ML}(\mathcal{L}^+)} H$ if and only if $G \equiv_{\mathsf{ML}(\mathcal{L})} H$. So, we may indeed use expressions in $\mathsf{ML}(\mathcal{L}^+)$ instead of $\mathsf{ML}(\mathcal{L})$

To compute the indicator vectors of an equitable partition, we implement the algorithm GDCR for finding this partition [48]. We recall this algorithm (in a slightly different form than presented in Kersting et al. [48]) in Algorithm 1. In a nutshell, the algorithm takes as input A_G, the adjacency matrix of G, and returns a matrix whose columns hold indicator vectors that represent the coarsest equitable partition of G.

The algorithm starts, on line 1, by creating a partition consisting of a single part containing all vertices, represented by the indicator vector $\mathbb{1}$, and stored in vector $B^{(0)}$. Then, in the i^{th} step, the current partition is represented by ℓ_{i-1} indicator vectors $\mathbb{1}_{V_1^{(i-1)}}, \dots, \mathbb{1}_{V_{\ell_{i-1}}^{(i-1)}}$ which constitute the columns of matrix $B^{(i-1)}$. The refinement of this partition is then computed in two steps. First, the matrix $M^{(i)} := A_G \cdot B^{(i-1)}$ (line 4) is computed; Second, each $\mathbb{1}_{V_j^{(i-1)}}$ is refined by putting vertices v and w in the same part if and only if they have the same rows in $M^{(i)}$, i.e., when $M_{v*}^{(i)} = M_{w*}^{(i)}$ holds (line 5). The corresponding partition $\mathcal{V}^{(i)}$ is then represented by, say ℓ_i, indicator vectors and stored as the columns of $B^{(i)}$ (line 6). This is repeated until no further refinement

[2] We remark that we cannot rely yet on the similarity preservation Lemma 5.3 to show that $G \equiv_{\mathsf{ML}(\mathcal{L}^+)} H$ if and only if $G \equiv_{\mathsf{ML}(\mathcal{L})} H$. Indeed, at this point we do not know yet for what kind of matrices T, T-similarity is preserved by the $\mathsf{diag}(\cdot)$-operation. This will only be settled in Lemma 7.1 later in this section.

of the partition is obtained. At most n iterations are needed. The correctness of the algorithm is established in [48]. That is, the resulting indicator vectors represent indeed the coarsest equitable partition of G.

We next detail how a run of the algorithm can be simulated using expressions in $\mathsf{ML}(\mathcal{L}^+)$. Let us fix the adjacency matrix A_G. The initialisation step is easy: We compute $B^{(0)}$ by means of the expression $b^{(0)}(X) := \mathbb{1}(X)$. Clearly, $B^{(0)} = b^{(0)}(A_G)$. Next, suppose by induction that we have ℓ_{i-1} expressions $b_1^{(i-1)}(X),\dots,b_{\ell_{i-1}}^{(i-1)}(X)$ such that when these expressions are evaluated on A_G, they return the indicator vectors stored in the columns of $B^{(i-1)}$. That is, $\mathbb{1}_{V_j^{(i-1)}} = b_j^{(i-1)}(A_G)$ for all $j = 1,\dots,\ell_{i-1}$. We next show how the i^{th} iteration is simulated.

We first compute the ℓ_{i-1} vectors stored in the columns of $M^{(i)}$ (line 4). We compute these column vectors one at a time. To this aim, we consider expressions

$$m_j^{(i)}(X) := X \cdot b_j^{(i-1)}(X), \quad \text{for } j = 1,\dots,\ell_{i-1}.$$

Clearly, $m_j^{(i)}(A_G) = M_{*,j}^{(i)}$, as desired.

A bit more challenging is the computation of the refined partition in $\mathcal{V}^{(i)}$ (line 5) since we need to inspect all columns $M_{*,j}^{(i)}$ and identify rows on which all these columns agree, as explained above. It is here that the $\mathsf{diag}(\cdot)$ operation plays a crucial role. Moreover, to compute this refined partition we need to know all values occurring in $M^{(i)}$. The expressions below depend on these values and hence on the input adjacency matrix (i.e., different inputs may lead to different values in $M^{(i)}$).

Let $D_j^{(i)}$ be the set of values occurring in the column vector $M_{*,j}^{(i)}$, for $j = 1,\dots,\ell_{i-1}$. We compute, by means of an $\mathsf{ML}(\mathcal{L}^+)$ expression, an indicator vector which identifies the rows in $M_{*,j}^{(i)}$ that hold a specific value $c \in D_j^{(i)}$. This expression is similar to the one used in Example 7.1 to extract vertices of degree 3 from the degree vector. More precisely, we consider expressions

$$\mathbb{1}_{=c}^{(i),j}(X) = \left(\frac{1}{\prod_{c' \in D_j^{(i)}, c' \neq c}(c - c')}\right) \times \left(\Big(\prod_{c' \in D_j^{(i)}, c' \neq c} \mathsf{diag}\big(m_j^{(i)}(X) - c' \times \mathbb{1}(X)\big)\Big) \cdot \mathbb{1}(X)\right),$$

for the current iteration i, column j in $M^{(i)}$, and value $c \in D_j^{(i)}$. The correctness of these expressions follows from a similar explanation as given in Example 7.1. Given these expressions, one can now easily obtain an indicator vector identifying all rows in $M^{(i)}$ that hold a specific value combination $(c_1,\dots,c_{\ell_{i-1}})$ in their columns, where each $c_j \in D_j^{(i)}$, as follows:

$$\mathbb{1}_{=(c_1,\dots,c_{\ell_{i-1}})}^{(i)}(X) = \mathsf{diag}(\mathbb{1}_{=c_1}^{(i),1}(X)) \cdot \dots \cdot \mathsf{diag}(\mathbb{1}_{=c_{\ell_{i-1}}}^{(i),\ell_{i-1}}(X)) \cdot \mathbb{1}(X).$$

That is, we simply take the boolean conjunction of all indicator vectors $\mathbb{1}_{=c_j}^{(i),j}(X)$, for $j = 1,\dots,\ell_{i-1}$. We note that $\mathbb{1}_{=(c_1,\dots,c_{\ell_{i-1}})}^{(i)}(A_G)$ may return the zero vector, i.e., when $(c_1,\dots,c_{\ell_{i-1}})$ does not occur as a row in $M^{(i)}$. We only need value combinations that occur. Suppose that there are ℓ_i distinct value combinations $(c_1,\dots,c_{\ell_{i-1}})$ for which $\mathbb{1}_{=(c_1,\dots,c_{\ell_{i-1}})}^{(i)}(A_G)$ returns a non-zero indicator vector. We denote by $b_1^{(i)}(X),\dots,b_{\ell_i}^{(i)}(X)$ the corresponding expressions of the form $\mathbb{1}_{=(c_1,\dots,c_{\ell_{i-1}})}^{(i)}(X)$. It should be clear that $b_1^{(i)}(A_G),\dots,b_{\ell_i}^{(i)}(A_G)$ are indicator vectors corresponding to the refined partition $\mathcal{V}^{(i)}$ as stored in $B^{(i)}$. This concludes the simulation of the i^{th} iteration of the algorithm.

Finally, after the n^{th} iteration we define

$$\mathsf{eqpart}_i(X) := b_i^{(n)}(X),$$

for $i = 1, \ldots, \ell_n$. In the following, we denote ℓ_n by ℓ. We remark once more that all expressions defined above depend on the input A_G, as their definitions rely on the values occurring in the matrices $M^{(i)}$ computed along the way.

Recall that we want to show that if $G \equiv_{\mathsf{ML}(\mathcal{L}^+)} H$ holds, then G and H have a common equitable partition. To this aim we show that vectors $\mathsf{eqpart}_i(A_H)$, for $i = 1, \ldots, \ell$, correspond to an equitable partition of H and that this partition, together with the one for G represented by $\mathsf{eqpart}_i(A_G)$, for $i = 1, \ldots, \ell$, show that G and H have a common equitable partition.

The challenge is to check all this by means of sentences in $\mathsf{ML}(\mathcal{L}^+)$. Below, we provide the description for sentences in $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag}, +, \times)$. We note, however, that a minor modification of these sentences suffices such that they belong to $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, +, \times)$. Hence, the proof works for $\mathsf{ML}(\mathcal{L})^+$-sentences.

Indeed, in the sentences below we will use conjugate transposition. In particular we only use it to sum up entries in a vector. That is, when conjugate transposition is used, it is in the form of $(\mathbb{1}(X))^* \cdot e(X)$ for some expression $e(X)$ which evaluates to a (column) vector. It now suffices to consider the expression $\mathsf{tr}(\mathsf{diag}(e(X)))$ instead to turn the $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag}, +, \times)$-sentences into $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, +, \times)$-sentences.

With this in mind, we refer to the sentences below simply as $\mathsf{ML}(\mathcal{L}^+)$-sentences where $\mathcal{L}$ can be either $\{\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag}\}$ or $\{\cdot, {}^*, \mathbb{1}, \mathsf{diag}\}$. We will need the following sentences.

1. For each $i = 1, \ldots, \ell$, we first check whether $\mathsf{eqpart}_i(A_H)$ is also a binary vector containing the same number of 1's as $\mathsf{eqpart}_i(A_G)$. We note that, by construction of the expression $\mathsf{eqpart}_i(X)$, $\mathsf{eqpart}_i(A_H)$ returns a real vector. To check whether every entry in $\mathsf{eqpart}_i(A_H)$ is either 0 or 1, we show that all of its entries must satisfy the equation $x(x-1) = 0$. To this aim, we consider the $\mathsf{ML}(\mathcal{L}^+)$ sentence
$$\mathsf{binary\_diag}(X) := (\mathbb{1}(X))^* \cdot \big((X \cdot X - X) \cdot (X \cdot X - X)\big) \cdot \mathbb{1}(X).$$
We claim that if X is assigned a diagonal real matrix, say Δ, then $\mathsf{binary\_diag}(\Delta) = [0]$ if and only if Δ is a *binary* diagonal matrix.
Indeed, if Δ is a binary diagonal matrix, then $\Delta \cdot \Delta = \Delta$, $\Delta \cdot \Delta - \Delta = Z$, where Z is the zero matrix, and hence $\mathsf{binary\_diag}(\Delta) = \mathbb{1}^{\mathsf{t}} \cdot Z \cdot Z \cdot \mathbb{1} = [0]$. Conversely, suppose that $\mathsf{binary\_diag}(\Delta) = [0]$. We observe that $(\Delta \cdot \Delta - \Delta) \cdot (\Delta \cdot \Delta - \Delta)$ is a diagonal matrix with squared real numbers on its diagonal. Hence, $\mathsf{binary\_diag}(\Delta) = [0]$ implies that the sum of the (squared real) diagonal elements in $\Delta \cdot \Delta - \Delta$ is 0. This in turn implies that every element on the diagonal in $\Delta \cdot \Delta - \Delta$ must be zero. Hence, every element on Δ's diagonal must satisfy the equation $x^2 - x = 0$, implying that either $x = 0$ or $x = 1$. As a consequence, Δ is a binary diagonal matrix.
We observe that $\mathsf{binary\_diag}(\mathsf{diag}(\mathsf{eqpart}_i(A_G))) = [0]$ since $\mathsf{eqpart}_i(A_G)$ returns an indicator vector. Then, $G \equiv_{\mathsf{ML}(\mathcal{L}^+)} H$ implies that the equality
$$\mathsf{binary\_diag}\big(\mathsf{diag}(\mathsf{eqpart}_i(A_G))\big) = [0] = \mathsf{binary\_diag}\big(\mathsf{diag}(\mathsf{eqpart}_i(A_H))\big),$$

must hold, for all $i=1,\ldots,\ell$. Hence, the matrices $\mathsf{diag}(\mathsf{eqpart}_i(A_H))$ are indeed binary and so are its diagonal elements described by $\mathsf{eqpart}_i(A_H)$, as desired.
In addition, we also need to check whether $\mathsf{eqpart}_i(A_H)$ has the same number of entries set to 1 as $\mathsf{eqpart}_i(A_G)$. For this, it suffices to consider the sentence $(\mathbb{1}(X))^* \cdot \mathsf{eqpart}_i(X)$. Clearly, $G \equiv_{\mathsf{ML}(\mathcal{L}^+)} H$ implies that $\mathbb{1}^* \cdot \mathsf{eqpart}_i(A_G) = \mathbb{1}^* \cdot \mathsf{eqpart}_i(A_H)$, for $i=1,\ldots,\ell$. Hence, $\mathsf{eqpart}_i(A_H)$ and $\mathsf{eqpart}_i(A_G)$ contain the same number of ones.

2. We next verify that all indicator vectors $\mathsf{eqpart}_i(A_H)$ combined form a partition of the vertex set of H. To verify this partition condition, we check whether for any two different $i,j=1,\ldots,\ell$, the entries in the vectors $\mathsf{eqpart}_i(A_H)$ and $\mathsf{eqpart}_j(A_H)$ holding a 1 are distinct. This is done by observing that for binary diagonal matrices Δ_1 and Δ_2, $\Delta_1 \cdot \Delta_2$ holds on its diagonal the conjunction of the binary vectors on the diagonals of Δ_1 and Δ_2, respectively. If we want to test that all positions in which Δ_1 and Δ_2 carry value 1 are different, $\Delta_1 \cdot \Delta_2$ should be the zero matrix Z. It now suffices to consider the following sentences
$$\mathsf{partition\_test}_{i,j}(X) := (\mathbb{1}(X))^* \cdot \mathsf{diag}(\mathsf{eqpart}_i(X)) \cdot \mathsf{diag}(\mathsf{eqpart}_j(X)) \cdot \mathbb{1}(X),$$
for $i,j=1,\ldots,\ell$ and $i \neq j$. Then, because $\mathsf{partition\_test}_{ij}(A_G) = [0]$ we have that $G \equiv_{\mathsf{ML}(\mathcal{L}^+)} H$ implies that for $i,j=1,\ldots,\ell$, $i \neq j$,
$$\mathsf{partition\_test}_{ij}(A_G) = [0] = \mathsf{partition\_test}_{ij}(A_H).$$
Hence, the indicator vectors $\mathsf{eqpart}_i(A_H)$, for $i=1,\ldots,\ell$, are all pairwise disjoint. Furthermore, we know that the vectors $\mathsf{eqpart}_i(A_G)$ form a partition. Since we have just shown that $\mathsf{eqpart}_i(A_H)$ and $\mathsf{eqpart}_i(A_G)$ contain the same number of ones, the disjointness of the vectors $\mathsf{eqpart}_i(A_H)$ implies that these also correspond to a partition of the vertex set of H.

To conclude the proof, we argue that the partition $\mathcal{W} = \{W_1,\ldots,W_\ell\}$ of H, represented by the indicator vectors $\mathsf{eqpart}_i(A_H)$, is an equitable partition. Moreover, consider the equitable partition $\mathcal{V} = \{V_1,\ldots,V_\ell\}$ of G, represented by the indicator vectors $\mathsf{eqpart}_i(A_G)$. We show that G and H have a common equitable partition, given by $\mathcal{V}$ and $\mathcal{W}$. We observe that we already know that $|V_i| = |W_i|$ for every $i=1,\ldots,\ell$. To show that G and H have a common equitable partition, it suffices to show that for any $i,j=1,\ldots,\ell$, $\mathsf{deg}(v,V_j) = \mathsf{deg}(w,W_j)$ for any $v \in V_i$ and any $w \in W_i$.

3. As already mentioned at the beginning of this section, we can rephrase "being equitable" in linear algebra terms. In particular, we know that for any $i,j=1,\ldots,\ell$,
$$\mathsf{diag}(\mathsf{eqpart}_i(A_G)) \cdot A_G \cdot \mathsf{diag}(\mathsf{eqpart}_j(A_G)) \cdot \mathbb{1} - \deg(v,V_j) \times \mathsf{eqpart}_i(A_G)$$
returns the zero vector, where v is an arbitrary vertex in V_i, the part corresponding to the indicator vector $\mathsf{eqpart}_i(A_G)$. We want to check whether the same condition holds for A_H. We therefore consider the expression $\mathsf{equi\_test}(X)$, given by
$$\mathsf{diag}\Big(\mathsf{diag}(\mathsf{eqpart}_i(X)) \cdot X \cdot \mathsf{diag}(\mathsf{eqpart}_j(X)) \cdot \mathbb{1}(X) - \deg(v,V_j) \times \mathsf{eqpart}_i(X)\Big)$$
and check whether, when evaluated on A_H, the obtained diagonal matrix is the zero matrix. This would imply that also
$$\mathsf{diag}(\mathsf{eqpart}_i(A_H)) \cdot A_H \cdot \mathsf{diag}(\mathsf{eqpart}_j(A_H)) \cdot \mathbb{1} - \deg(v,V_j) \times \mathsf{eqpart}_i(A_H)$$

returns the zero vector. As a consequence, $\mathcal{W}$ is an equitable partition of H and furthermore, $\mathsf{deg}(w, W_j) = \mathsf{deg}(v, V_j)$, for any $i, j = 1, \dots, \ell$ and vertices $v \in V_i$ and $w \in W_i$. In other words, G and H do indeed have a common equitable partition. It rests us only to show that we can check, by means of sentences, whether a diagonal matrix is the zero matrix. We use the sentence

$$\mathsf{zerotest\_diag}(X) := (\mathbb{1}(X))^* \cdot X \cdot X \cdot \mathbb{1}(X),$$

for this purpose. A similar argument as for the expression $\mathsf{binary\_diag}(X)$ shows that the $\mathsf{zerotest\_diag}(X)$ expression returns [0] on diagonal real matrices if and only if the diagonal matrix is the zero matrix. We here again use that a sum of squares equals zero if and only if each summand is zero. Since we have that $G \equiv_{\mathsf{ML}(\mathcal{L}^+)} H$, $\mathsf{zerotest\_diag}(\mathsf{equi\_test}(A_G)) = [0] = \mathsf{zerotest\_diag}(\mathsf{equi\_test}(A_H))$, as desired.

As mentioned at the beginning of the proof, the $\mathsf{ML}(\mathcal{L}^+)$ sentences obtained can all be written as a linear combination of sentences in $\mathsf{ML}(\mathcal{L})$. So, we may indeed conclude that $\mathsf{ML}(\mathcal{L})$-equivalence of G and H implies that these graphs have a common equitable partition. □

7.2 Characterisation of $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})$-equivalence

We first consider the characterisation of $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})$-equivalence. We have just shown that two $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})$-equivalent graphs have a common equitable partition. The converse also holds, as will be shown next.

Proposition 7.3 *Let G and H be two graphs of the same order. If G and H have a common equitable partition, then $e(A_G) = e(A_H)$ for every sentence $e(X)$ in* $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})$.

Proof By assumption, G and H have a common equitable partition. As a consequence, they also have a common (unique) coarsest equitable partition (see e.g., Theorem 6.5.1 in [62]). Let $\mathcal{V} = \{V_1, \dots, V_\ell\}$ and $\mathcal{W} = \{W_1, \dots, W_\ell\}$ be the common coarsest equitable partitions of G and H, respectively. As before, we denote by $\mathbb{1}_{V_i}$ and $\mathbb{1}_{W_i}$, for $i = 1, \dots, \ell$, the corresponding indicator vectors. We know from Proposition 7.1 that there exists a doubly stochastic matrix S such that $A_G \cdot S = S \cdot A_H$. As previously observed, also $A_H \cdot S^* = S^* \cdot A_G$ holds, where S^* is again doubly stochastic. Then, Lemmas 5.1, 5.4 and 6.1 imply that S-similarity and S^*-similarity are preserved by matrix multiplication, complex conjugate transposition, and the one-vector operation. To conclude that $G \equiv_{\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})} H$ holds, we verify that the $\mathsf{diag}(\cdot)$ operations also preserves S- and S^*-similarity. We rely on a more general result (Lemma 7.1 below), which states that T-similarity, for a matrix T, is preserved by the $\mathsf{diag}(\cdot)$ operation provided that T is doubly quasi-stochastic and *compatible* with the common coarsest equitable partitions of G and H. We again separate this lemma from the current proof because we need it later in the paper. The compatibility condition refers to a *block structure* condition on matrices. More precisely, if G and H have a common equitable partition, then consider the common coarsest equitable partitions described by the indicator vectors $\mathbb{1}_{V_i}$ and $\mathbb{1}_{W_i}$ for G and H, respectively. A matrix T is now said to be *compatible* with respect to $\mathbb{1}_{V_i}$ and $\mathbb{1}_{W_i}$, for $i=1,\ldots,\ell$, if

$$\mathsf{diag}(\mathbb{1}_{V_i})\cdot T = T\cdot \mathsf{diag}(\mathbb{1}_{W_i}),$$

for $i=1,\ldots,\ell$. That is, T has a block structure determined by the partitions and only has non-zero blocks for blocks corresponding to the same parts in the equitable partitions. When considering the doubly stochastic matrix S such that $A_G\cdot S = S\cdot A_H$ holds, the matrix S can be assumed to be compatible in the above sense. To see this, we recall from the proof of Theorem 6.5.1 in [62] that we can take S to be such that for $i\neq j$, $\mathsf{diag}(\mathbb{1}_{V_i})\cdot S\cdot \mathsf{diag}(\mathbb{1}_{W_j})$ is the $|V_i|\times|W_j|$ zero matrix, and for $i=j$, $\mathsf{diag}(\mathbb{1}_{V_i})\cdot S\cdot \mathsf{diag}(\mathbb{1}_{W_i})$ is the square $|V_i|\times|W_i|$ matrix in which all entries are equal to $\frac{1}{|V_i|}$.

As a consequence, if $e_1(A_G)$ and $e_1(A_H)$ are S-similar, then Lemma 7.1 implies that $\mathsf{diag}(e_1(A_G))$ and $\mathsf{diag}(e_1(A_H))$ are S-similar. We also note that $\mathsf{diag}(\mathbb{1}_{W_i})\cdot S^* = S^*\cdot \mathsf{diag}(\mathbb{1}_{V_i})$. So S^* is compatible with $\mathbb{1}_{W_i}$ and $\mathbb{1}_{V_i}$. An inductive argument then shows that $e(A_G)$ and $e(A_H)$ are S-similar (and thus equal) for any sentence $e(X)$ in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$, as desired. ◻

To show that similarity, by means doubly quasi-stochastic matrices that are compatible with respect to the common coarsest equitable partitions, is indeed preserved by the $\mathsf{diag}(\cdot)$ operation, requires a bit more work than our previous similarity preservation results. More precisely, we need that vectors obtained by evaluating expressions in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$ can be written in a canonical way in terms of the indicator vectors representing the common coarsest equitable partitions of the graphs. We state this requirement for general matrix query languages, as follows.

Let $\mathsf{ML}(\mathcal{L})$ be a matrix query language. Let G be a graph with equitable partition $\mathcal{V}=\{V_1,\ldots,V_\ell\}$ and let $\mathbb{1}_{V_1},\ldots,\mathbb{1}_{V_\ell}$ be the corresponding indicator vectors. We say that $\mathsf{ML}(\mathcal{L})$-*vectors are constant on equitable partitions* if for any expression $e(X)\in \mathsf{ML}(\mathcal{L})$ such that $e(A_G)$ is an $n\times 1$-vector, then

$$e(A_G)=\sum_{i=1}^{\ell} a_i\times \mathbb{1}_{V_i} \qquad (7.1)$$

for scalars $a_i\in$. Intuitively, this condition is important for the $\mathsf{diag}(\cdot)$ operation since it takes a vector as input and the linear combination (7.1) allows one to only reason about (linear combinations of) diagonal matrices obtained by the indicator vectors of the equitable partitions. Compatibility implies similarity preservation for such (indicator vector-based) diagonal matrices, which can then be lifted, due to linearity, to similarity of arbitrary diagonal matrices.

Lemma 7.1 *Let G and H be two graphs of the same order which have a common equitable partition. Let $\mathsf{ML}(\mathcal{L})$ be a matrix query language fragment such that $\mathsf{ML}(\mathcal{L})$-vectors are constant on equitable partitions. Let T be a doubly quasi-stochastic matrix which is compatible with the coarsest common equitable partitions of G and H. Let $e(X)$ be an expression in $\mathsf{ML}(\mathcal{L})$. Then, if $e(A_G)$ and $e(A_H)$ are T-similar, then also $\mathsf{diag}(e(A_G))$ and $\mathsf{diag}(e(A_H))$ are T-similar.*

Proof Let $e(X)$ be an expression in $\mathsf{ML}(\mathcal{L})$. Consider now $e'(X):=\mathsf{diag}(e(X))$. We distinguish between two cases, depending on the dimensions of $e(A_G)$. First, if $e(A_G)$

is a sentence then we know by induction that $e(A_G) = e(A_H)$. Hence,

$$e'(A_G) = \mathsf{diag}(e(A_G)) = e(A_G) = e(A_H) = \mathsf{diag}(e(A_H)) = e'(A_H).$$

Next, if $e(A_G)$ is a vector, then we know that $e(A_G) = T \cdot e(A_H)$ and furthermore, since $\mathsf{ML}(\mathcal{L})$-vectors are constant on equitable partitions, that $e(A_G) = \sum_{i=1}^{\ell} a_i \times \mathbb{1}_{V_i}$ and $e(A_H) = \sum_{i=1}^{\ell} b_i \times \mathbb{1}_{W_i}$. We first show that $a_i = b_i$, for $i = 1, \dots, \ell$. Indeed, since $T \cdot \mathbb{1} = \mathbb{1}$ and T is compatible with $\mathbb{1}_{V_i}$ and $\mathbb{1}_{W_i}$, we have that

$$\mathbb{1}_{V_i} = \mathsf{diag}(\mathbb{1}_{V_i}) \cdot \mathbb{1} = \mathsf{diag}(\mathbb{1}_{V_i}) \cdot T \cdot \mathbb{1} = T \cdot \mathsf{diag}(\mathbb{1}_{W_i}) \cdot \mathbb{1} = T \cdot \mathbb{1}_{W_i}.$$

As a consequence, using that $\mathbb{1}_{V_i}^{\mathsf{t}} \cdot \mathbb{1}_{V_j}$ is 0 if $i \neq j$ and $|V_i|$ if $i = j$, we obtain

$$\begin{aligned} a_i \times |V_i| &= \mathbb{1}_{V_i}^{\mathsf{t}} \cdot e(A_G) = \mathbb{1}_{V_i}^{\mathsf{t}} \cdot T \cdot e(A_H) \\ &= \sum_{j=1}^{\ell} b_j \times (\mathbb{1}_{V_i}^{\mathsf{t}} \cdot T \cdot \mathbb{1}_{W_j}) = b_i \times |W_i|, \end{aligned}$$

for all $i = 1, \dots, \ell$. Since $|V_i| = |W_i| \neq 0$, we indeed have that $a_i = b_i$ for all $i = 1, \dots, \ell$.

We may now conclude that

$$\begin{aligned} e'(A_G) \cdot T = \mathsf{diag}(e(A_G)) \cdot T &= \sum_{i=1}^{\ell} a_i \times (\mathsf{diag}(\mathbb{1}_{V_i}) \cdot T) \\ &= \sum_{i=1}^{\ell} a_i \times (T \cdot \mathsf{diag}(\mathbb{1}_{W_i})) = T \cdot \mathsf{diag}(e(A_H)) = T \cdot e'(A_H). \end{aligned}$$

Hence $e'(A_G)$ and $e'(A_H)$ are indeed T-similar. □

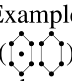
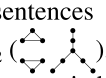
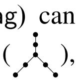
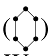
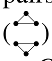
In the context of Proposition 7.3, i.e., to show that the $\mathsf{diag}(\cdot)$ operation preserves S-similarity (and S^*-similarity), we need to verify that $\mathsf{ML}(\cdot, ^*, \mathbb{1}, \mathsf{diag})$-vectors are constant on equitable partitions. We verify this, in the appendix, by induction on the structure of expressions in $\mathsf{ML}(\cdot, ^*, \mathbb{1}, \mathsf{diag})$. The key insight is that the base case for the induction, when $e(X) = X$, holds by the assumption that G and H have a common coarsest equitable partition. In fact, we more generally show the following.

Proposition 7.4 $\mathsf{ML}(\cdot, ^*, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, +, \times, \mathsf{apply_s}[f], f \in \Omega)$*-vectors are constant on equitable partitions.* □

All combined, we obtain the following characterisations.

Theorem 7.1 *Let G and H be two graphs of the same order. Then, $G \equiv_{\mathsf{ML}(\cdot, ^*, \mathbb{1}, \mathsf{diag})} H$ if and only if there is doubly stochastic matrix S such that $A_G \cdot S = S \cdot A_H$ if and only if $G \equiv_{\mathsf{C}^2} H$ if and only if G and H have a common equitable partition.* □

Proof This is a direct consequence of Propositions 7.1, 7.2 and 7.3. □

As a consequence, following Example 7.2, sentences in $\mathsf{ML}(\cdot, ^*, \mathbb{1}, \mathsf{diag})$ can distinguish G_1 () and H_1 (), G_2 () and H_2 (), G_4 () and H_4 (), because all these pairs of graphs do not have a common equitable partition. By contrast, G_3 () and H_3 () cannot be distinguished by sentences in $\mathsf{ML}(\cdot, ^*, \mathbb{1}, \mathsf{diag})$.

We remark that $G \equiv_{\mathsf{ML}(\cdot, ^*, \mathbb{1}, \mathsf{diag})} H$ if and only if $G \equiv_{\mathsf{ML}(\cdot, ^*, \mathbb{1}, \mathsf{diag}, +, \times, \mathsf{apply_s}[f], f \in \Omega)}$ H. This is again a direct consequence of the fact that $G \equiv_{\mathsf{ML}(\cdot, ^*, \mathbb{1}, \mathsf{diag})} H$ implies that

$A_G \cdot S = S \cdot A_H$, $A_H \cdot S^* = S^* \cdot A_G$, and that all operations in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f], f\in\Omega)$ preserve S-similarity and S^*-similarity.

7.3 Characterisation of $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$-equivalence

We next consider $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$-equivalence. We already know a couple of implications when $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ holds. For example, there must exist an orthogonal matrix O such that $O\cdot\mathbb{1}=\mathbb{1}$ and $A_G \cdot O = O \cdot A_H$ (Propositions 6.4 and 6.5). Furthermore, we know that G and H must have a common equitable partition and hence, there exists a doubly stochastic matrix S such that $A_G \cdot S = S \cdot A_H$ (Proposition 7.1). It is tempting to conjecture that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ if and only if there exists an orthogonal doubly stochastic matrix O such that $A_G \cdot O = O \cdot A_H$. This does not hold, however. Indeed, invertible doubly stochastic matrices are necessarily permutation matrices [27]. Then, $A_G \cdot O = O \cdot A_H$ would imply that G and H are isomorphic, contradicting that our fragments cannot go beyond C^3-equivalence [10]. Instead, we have the following characterisation.

Theorem 7.2 *Let G and H be two graphs of the same order. Then the following hold: $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ if and only if G and H have a common equitable partition and $A_G \cdot O = O \cdot A_H$ for some doubly quasi-stochastic orthogonal matrix O which is compatible with the common coarsest equitable partition of G and H.*

Proof To show that the existence of a matrix O, as stated in the Theorem, implies that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$, we argue as before. More precisely, we show that O-similarity is preserved by the operations in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$. This is, however, a direct consequence of Lemmas 5.1, 5.2, 6.1 and 7.1. We remark that Proposition 7.4 guarantees that Lemma 7.1 can be applied. Indeed, Proposition 7.4 implies that $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$-vectors are constant on equitable partitions. We may thus conclude that all expressions in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$ preserve O-similarity. Hence, $e(A_G)=e(A_H)$ for any sentence $e(X)$ in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$.

For the converse direction, we need to show that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ implies that there exists an orthogonal matrix O such that $A_G \cdot O = O \cdot A_H$, and where O satisfies the conditions mentioned in the statement of the Theorem.

The existence of the orthogonal matrix O is shown using Specht's Theorem (see e.g., [45]), which we recall next. Let $\mathcal{A}=\{A_1,\ldots,A_p\}$ and $\mathcal{B}=\{B_1,\ldots,B_p\}$ be two sets of complex matrices that are closed under complex conjugate transposition. The sets $\mathcal{A}$ and $\mathcal{B}$ are called *simultaneously unitary equivalent* if there exists a unitary matrix U such that $A_i \cdot U = U \cdot B_i$, for $i=1,\ldots,p$. Here, a unitary matrix U is such that $U^*\cdot U = U\cdot U^* = I$; it is the complex analogue of a real orthogonal matrix. Specht's Theorem provides a means of checking simultaneous unitary equivalence in terms of *trace identities*. Indeed, Specht's Theorem states that $\mathcal{A}$ and $\mathcal{B}$ are simultaneously unitary equivalent if and only if

$$\mathsf{tr}(w(A_1,\ldots,A_p))=\mathsf{tr}(w(B_1,\ldots,B_p)),$$

for all words $w(x_1,\ldots,x_p)$ over the alphabet $\{x_1,\ldots,x_p\}$. In expression $w(A_1,\ldots,A_p)$ we instantiated x_i with A_i and interpret concatenation in the word w as matrix multiplication; Similarly for $w(B_1,\dots,B_p)$. Specht's Theorem also holds when $\mathcal{A}$ and $\mathcal{B}$ are real matrices and similarity is expressed in terms of orthogonal matrices [45]. The required condition is that $\mathcal{A}$ and $\mathcal{B}$ are closed under transposition. We will rephrase the conditions required for O, i.e., that it is a doubly quasi-stochastic matrix which is compatible with a common equitable partition of G and H, in terms of such trace identities. We note that a similar approach is taken by Thüne [64] in the context of characterising the equivalence of graphs with regards to their 1-dimensional Weisfeiler-Lehman closure.

We start by defining the sets $\mathcal{A}$ and $\mathcal{B}$. Consider the following sets of real symmetric matrices: $\mathcal{A} := \{A_G, J\} \cup \{\mathsf{diag}(\mathbb{1}_{V_i}) \mid i = 1,\dots,\ell\}$ and $\mathcal{B} := \{A_H, J\} \cup \{\mathsf{diag}(\mathbb{1}_{W_i}) \mid i = 1,\dots,\ell\}$, where $\mathbb{1}_{V_i}$ and $\mathbb{1}_{W_i}$ denote the indicator vectors corresponding to the coarsest common equitable partitions in G and H, respectively. We observe that $\mathcal{A}$ and $\mathcal{B}$ are closed under transposition. By the real counterpart of Specht's Theorem we can check whether there exists an orthogonal matrix O such that

$$A_G \cdot O = O \cdot A_H \tag{7.2}$$

$$J \cdot O = O \cdot J \tag{7.3}$$

$$\mathsf{diag}(\mathbb{1}_{V_i}) \cdot O = O \cdot \mathsf{diag}(\mathbb{1}_{W_i}), \tag{7.4}$$

hold, for $i = 1,\dots,\ell$, in terms of trace identities. It is clear that conditions (7.2) and (7.4) express that A_G and A_H must be O-similar and that O must be compatible with the coarsest common equitable partition of G and H. The orthogonality of O is implied by Specht's Theorem. Condition (7.3) ensures that $O \cdot \mathbb{1} = \mathbb{1}$. To see this, we modify the proof of Lemma 4 in Thüne [64], stated for unitary matrices, so that it holds for orthogonal matrices. We first observe that $\mathbb{1}$ is an eigenvector of O. Indeed, $J \cdot O \cdot \mathbb{1} = \mathbb{1} \cdot (\mathbb{1}^{\mathsf{t}} \cdot O \cdot \mathbb{1}) = \alpha \times \mathbb{1}$ with $\alpha = \mathbb{1}^{\mathsf{t}} \cdot O \cdot \mathbb{1}$ and $J \cdot O \cdot \mathbb{1} = O \cdot J \cdot \mathbb{1} = (\mathbb{1}^{\mathsf{t}} \cdot \mathbb{1}) \times O \cdot \mathbb{1}$. In other words, $O \cdot \mathbb{1} = \frac{\alpha}{n} \times \mathbb{1}$ since $\mathbb{1}^{\mathsf{t}} \cdot \mathbb{1} = n$. Furthermore, because $\mathbb{1}^{\mathsf{t}} \cdot O^{\mathsf{t}} \cdot \mathbb{1}$ is a scalar, $\mathbb{1}^{\mathsf{t}} \cdot O^{\mathsf{t}} \cdot \mathbb{1} = (\mathbb{1}^{\mathsf{t}} \cdot O^{\mathsf{t}} \cdot \mathbb{1})^{\mathsf{t}} = \mathbb{1}^{\mathsf{t}} \cdot O \cdot \mathbb{1} = \alpha$. We next show that $\alpha = \pm n$. Indeed, since O is an orthogonal matrix

$$n = \mathbb{1}^{\mathsf{t}} \cdot I \cdot \mathbb{1} = \mathbb{1}^{\mathsf{t}} \cdot O^{\mathsf{t}} \cdot O \cdot \mathbb{1} = \frac{\alpha}{n} \times (\mathbb{1}^{\mathsf{t}} \cdot O^{\mathsf{t}} \cdot \mathbb{1}) = \frac{\alpha^2}{n},$$

and thus $\alpha^2 = n^2$ or $\alpha = \pm n$. Hence, $O \cdot \mathbb{1} = \pm\mathbb{1}$. When $O \cdot \mathbb{1} = \mathbb{1}$, O is already doubly quasi-stochastic. In case that $O \cdot \mathbb{1} = -\mathbb{1}$, we simply replace O by $(-1) \times O$ to obtain that $O \cdot \mathbb{1} = \mathbb{1}$. This rescaling does not impact the validity of conditions (7.2) and (7.4). Hence, O can indeed be assumed to be doubly quasi-stochastic.

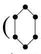
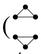
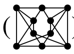
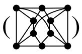
It remains to show that the trace identities implying the existence of an orthogonal O satisfying conditions (7.2), (7.3) and (7.4) can be expressed in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$. For every word $w(x,j,b_1,\dots,b_\ell)$ we consider the sentence

$$e_w(X) := \mathsf{tr}(w(X, \mathbb{1}(X) \cdot (\mathbb{1}(X))^*, \mathsf{diag}(\mathsf{eqpart}_1(X)),\dots,\mathsf{diag}(\mathsf{eqpart}_\ell(X)))),$$

in which variables $x, j, b_1,\dots,b_\ell$ are assigned to matrix variable X, expression $\mathbb{1}(X) \cdot (\mathbb{1}(X))^*$ in $\mathsf{ML}(\cdot,{}^*,\mathbb{1})$, and $\mathsf{diag}(\mathsf{eqpart}_i(X))$, for $i = 1,\dots,\ell$, respectively. Here, the expressions $\mathsf{eqpart}_i(X)$ correspond to the expressions extracting the indicator vectors of the coarsest equitable partition of a graph, as defined in the proof of Proposition 7.2. We recall from that proof that $\mathsf{eqpart}_i(X)$ are defined by using addition and scalar multiplication. As a consequence, the sentences $e_w(X)$ belong to $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times)$. Nevertheless, we next argue that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ im-

plies that $e_w(A_G)=e_w(A_H)$ for every word w. First, we observe that the use of complex conjugate transposition in the sentences $e_w(X)$ is very restricted. Indeed, it only occurs in the form $(\mathbb{1}(X))^*$. So, we may assume that $e_w(X)$ is a sentence in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^\mathsf{t},\mathsf{diag},+,\times)$, where $\mathbb{1}^\mathsf{t}(\cdot)$ is the operation that returns the transpose of $\mathbb{1}(\cdot)$. Second, just as in the proof of Proposition 7.2, we note that the sentences $e_w(X)$ only use multilinear operations, and thus can be written as a linear combination of sentences in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^\mathsf{t},\mathsf{diag})$. As a consequence, $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^\mathsf{t},\mathsf{diag})} H$ implies already that $e_w(A_G)=e_w(A_H)$. It remains to show that $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ implies $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^\mathsf{t},\mathsf{diag})} H$. We prove this in the appendix. Intuitively, in a sentence $e(X)$ in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^\mathsf{t},\mathsf{diag})$ every occurrence of $\mathbb{1}^\mathsf{t}(X)$ appears in a sub-sentence of the form $\mathbb{1}^\mathsf{t}(X)\cdot e'(X)\cdot\mathbb{1}(X)$ where $e'(X)$ does not contain the $\mathbb{1}^\mathsf{t}(\cdot)$ operation. Since we can replace $\mathbb{1}^\mathsf{t}(X)\cdot e'(X)\cdot\mathbb{1}(X)$ by $\mathsf{tr}(\mathsf{diag}(e'(X)\cdot\mathbb{1}(X))$ we can find an equivalent expression for $e(X)$ which does not use $\mathbb{1}^\mathsf{t}(\cdot)$. Hence, $e(X)$ is equivalent to a sentence in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$. Details of this rewriting procedure can be found in the appendix.
□

Note that $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ implies $G\equiv_{\mathsf{ML}(\cdot,^*,\mathbb{1},\mathsf{diag})} H$. The converse does not hold.

Example 7.3 Consider G_3 () and H_3 (). These graphs are fractional isomorphic but are not co-spectral. Hence, $G_3\not\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H_3$ since $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$-equivalence implies co-spectrality. On the other hand, G_5 () and H_5 () are co-spectral regular graphs [67], with co-spectral complements, and whose common equitable partition consists of a single part containing all vertices. In fact, the common equitable partitions of G_5 and H_5 consist of the partitions consisting of all vertices (this holds more generally for any regular graph). Furthermore, since A_{G_5} and A_{H_5} share $\mathbb{1}$ as eigenvector (with eigenvalue 4). We know from before that there exists an orthogonal matrix O such that $A_{G_5}\cdot O=O\cdot A_{H_5}$ and $O\cdot\mathbb{1}=\mathbb{1}$ (this follows from being co-spectral and co-main). Moreover, the compatibility requirement is vacuously satisfied since it requires $\mathsf{diag}(\mathbb{1})\cdot O=O\cdot\mathsf{diag}(\mathbb{1})$. Hence, G_5 and H_5 cannot be distinguished by $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$ by Theorem 7.2. □

We remark that $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ if and only if $G\equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f],f\in\Omega)} H$. This is again a direct consequence of the fact that $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ implies that $A_G\cdot O=O\cdot A_H$, $A_H\cdot O^*=O^*\cdot A_G$, for an orthogonal doubly quasi-stochastic matrix O which is compatible with the coarsest common equitable partitions of G and H, and that all operations in $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f],f\in\Omega)$ preserve O-similarity and O^*-similarity.

7.4 Pointwise function applications on vectors

A crucial ingredient for obtaining characterisations of equivalence in the presence of the $\mathsf{diag}(\cdot)$ operation is that vectors are constant on equitable partitions (Proposition 7.4 and Lemma 7.1). In this way, vectors obtained by evaluating expressions on A_G and A_H are "almost" the same, up to the use of indicator vectors (see equation (7.1)). We next show that this tight relationship among vectors allows us to extend the matrix query languages considered in this section with pointwise function applications on *vectors*. More precisely, we denote by $\mathsf{apply_v}[f]$, for $f \in \Omega$, that we only allow function applications of the form $e(X) := \mathsf{apply_v}[f](e_1(X),\ldots,e_p(X))$ where each $e_i(X)$ returns a vector when evaluated on a matrix.

Proposition 7.5 *Let G and H be two graphs of the same order.*

(1) $G \equiv_{\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})} H$ *if and only if* $G \equiv_{\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f],\mathsf{apply_v}[f],f\in\Omega)} H$.
(2) $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ *if and only if* $G \equiv_{\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f],\mathsf{apply_v}[f],f\in\Omega)} H$.

Proof In view of the previous results, it suffices to show that (1) $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f],f\in\Omega)$-equivalence implies $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f],\mathsf{apply_v}[f],f\in\Omega)$-equivalence; and (2) $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f],f\in\Omega)$-equivalence implies $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f],\mathsf{apply_v}[f],f\in\Omega)$-equivalence. Both implication follow if we can show that $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f]\,\mathsf{apply_v}[f],f\in\Omega)$-vectors are constant on equitable partitions and that $\mathsf{apply_v}[f]$, for $f\in\Omega$, preserves similarity of quasi doubly-stochastic matrices that are compatible with the common coarsest equitable partition of G and H.

For conciseness, let $\mathcal{L}^\dagger$ denote the $\{\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\mathsf{apply_s}[f]\,\mathsf{apply_v}[f],f\in\Omega\}$, i.e., $\mathcal{L}^\dagger$ consists of all operations considered so far. Proposition 7.4 trivially generalizes to $\mathsf{ML}(\mathcal{L}^\dagger)$-vectors. Indeed, it suffices to show consider the case. Let $e(X) := \mathsf{apply_v}[f](e_1(X),\ldots,e_p(X))$, where $e_1(X),\ldots,e_p(X)$ are expressions in $\mathsf{ML}(\mathcal{L}^\dagger)$ such that each $e_i(A_G)$ returns a vector. We may assume by induction that for $i=1,\ldots,p$, $e_i(A_G)=\sum_{j=1}^{\ell} a_j^{(i)}\times\mathbb{1}_{V_i}$ for scalars $a_j^{(i)}\in$, for $j=1,\ldots,\ell$. Since the sets of entries in the indicator vectors holding value 1 are disjoint for any two different indicator vectors and that the vectors on which f is applied have the same constant for every entry in the same part, we have that

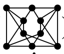
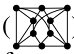
$$e(A_G)=\sum_{i=1}^{\ell}\mathsf{apply_s}[f]\big(a_i^{(1)},\ldots,a_i^{(p)}\big)\times\mathbb{1}_{V_i}.$$

So, indeed, $\mathsf{ML}(\mathcal{L}^\dagger)$-vectors are constant on equitable partitions.

That T-similarity is also preserved by pointwise function applications on vectors now follows easily. Indeed, consider $e(X) := \mathsf{apply_v}[f](e_1(X),\ldots,e_p(X))$. By assumption, $e_i(A_G)=T\cdot e_i(A_H)$ for all $i=1,\ldots,p$. Furthermore, $e_i(A_G)=\sum_{j=1}^{\ell}a_j^{(i)}\times\mathbb{1}_{V_i}$ and $e_i(A_H)=\sum_{j=1}^{\ell}b_j^{(i)}\times\mathbb{1}_{W_i}$. We have seen in the proof of Lemma 7.1 that T-similarity of these vectors implies $a_j^{(i)}=b_j^{(i)}$ for $j=1,\ldots,\ell$ and $i=1,\ldots,p$. As a consequence, $e(A_G)$ is equal to

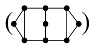
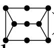
$$\begin{aligned}\mathsf{apply_v}[f](e_1(A_G),\ldots,e_p(A_G)) &= \sum_{i=1}^{\ell}\mathsf{apply_s}[f]\big(a_i^{(1)},\ldots,a_i^{(p)}\big)\times\mathbb{1}_{V_i}\\ &= \sum_{i=1}^{\ell}\mathsf{apply_s}[f]\big(a_i^{(1)},\ldots,a_i^{(p)}\big)\times(T\cdot\mathbb{1}_{W_i})\\ &= T\cdot\mathsf{apply_v}[f](e_1(A_H),\ldots,e_p(A_H)),\end{aligned}$$

which is equal to $T\cdot e(A_H)$, as desired. □

Going back to the graphs G_5 () and H_5 () in Example 7.3, these cannot even be distinguished by sentences in the large fragments in Proposition 7.5. In the next section, we show that by allowing pointwise function applications on matrices (the only operation in Table 3.1 which we did not consider yet), we can distinguish these graphs.

8 The impact of pointwise multiplication on vectors

In the preceding section the main use of the $\mathsf{diag}(\cdot)$ operation related to the construction of the coarsest equitable partition (see e.g., the proof of Proposition 7.2) and more specifically, to the ability to pointwise multiply two vectors (see e.g., Example 7.1). Of course, there is more that one can achieve by means of the $\mathsf{diag}(\cdot)$ operation, especially in combination with the trace operation. In the following, we denote pointwise vector multiplication by the operation $\odot_v$ and investigate how fragments supporting $\odot_v$ differ from those supporting $\mathsf{diag}(\cdot)$.

Example 8.1 Consider the graphs G_6 () and H_6 (). On can verify that these graphs are co-spectral and have a common equitable partition (and thus also have co-spectral complements). Using the diagonal operation we can construct the Laplacian of a graph by simply considering expression $L(X) := (\mathsf{diag}(X \cdot \mathbb{1}(X)) - X$. It is now easy to detect that G_6 and H_6 have Laplacians that are not co-spectral. Indeed, consider the $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, +, \times)$ expression $e_{L,k}(X) := \mathsf{tr}(L(X)^k)$. Then, we can check that $e_{L,3}(A_{G_6}) = 1602 \neq 1618 = e_{L,3}(A_{H_6})$. The relation between co-spectrality and traces of powers of matrices (cfr. Proposition 5.1) holds more generally for symmetric matrices (this follows easily from the real version of Specht's Theorem used in the proof of Theorem 7.2). Hence, we can infer that the Laplacians of G_6 and H_6 are not co-spectral. Another way of verifying this is that G_6 and H_6 have a different number of spanning trees (192 versus 160) and Kirchhoff's matrix tree theorem (see e.g., Proposition 1.3.4 in [12]) implies that graphs with co-spectral Laplacians must have the same number of spanning trees. Hence, G_6 and H_6 can be distinguished by $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, +, \times)$ (and also by sentences in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag})$ since all operations in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, +, \times)$ are linear). Nevertheless, we will see that G_6 and H_6 cannot be distinguished by sentences in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$[3]. More generally, we show that two graphs are $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$-equivalent if and only if they are co-spectral and have a common equitable partition (Proposition 8.4 below). □

In fact, it is for fragments that support the trace and $\mathsf{diag}(\cdot)$ operation that one observes an increase in expressive power compared to fragments supporting the trace and $\odot_v$ operation. Indeed, when considering $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})$, which does not support the trace operation, one can equivalently use $\odot_v$ instead of $\mathsf{diag}(\cdot)$.

Proposition 8.1 *Let G and H be two graphs of the same order. Then, $G \equiv_{\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \odot_v)} H$ if and only if $G \equiv_{\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})} H$.*

[3] It was incorrectly stated in the conference version that pointwise vector multiplication was equally powerful as the $\mathsf{diag}(\cdot)$ operation.

Proof The proof is by a straightforward translation between sentences in the two fragments. Indeed, let $e_1(X)$ and $e_2(X)$ be two expressions in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$ which evaluate to vectors (on input matrices). Then, $e_1(X)\odot_v e_2(X)$ is equivalent to the $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$ expression $\mathsf{diag}(e_1(X))\cdot\mathsf{diag}(e_2(X))\cdot\mathbb{1}(X)$. This implies that we can inductively replace all occurrences of $\odot_v$ in an expression in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\odot_v)$ by expressions in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$. So, every expression $e(X)$ in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\odot_v)$ is equivalent to an expression $e'(X)$ in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$. As a consequence, $G\equiv_{\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})} H$ implies $G\equiv_{\mathsf{ML}(\cdot,{}^*,\mathbb{1},\odot_v)} H$.

For the opposite direction, consider a sentence $e(X)$ in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$. One can assume such a sentence to be of the form $(\mathbb{1}(e_1(X)))^*\cdot e_2(X)\cdot\mathbb{1}(e_3(X))$ for some $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$ expressions $e_1(X)$, $e_2(X)$ and $e_3(X)$. Moreover, we can always replace $\mathbb{1}(e_1(X))$ by either $\mathbb{1}(X)$ or $\mathbb{1}(\mathbb{1}(X)^*)$ (depending on whether $e_1(X)$ evaluates to a matrix or a row vector). Similarly for $\mathbb{1}(e_3(X))$. We can thus assume that only $e_2(X)$ may have occurrences of the $\mathsf{diag}(\cdot)$ operation. We here treat the case when $e(X)=(\mathbb{1}(X))^*\cdot e_{21}(X)\cdot\mathsf{diag}(e_{22}(X))\cdot e_{23}(X)\cdot\mathbb{1}(X)$ for $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$ expressions $e_{21}(X)$, $e_{22}(X)$ and $e_{23}(X)$. The other cases can be dealt with in a similar way. It now suffices to observe that $\mathsf{diag}(e_{22}(X))\cdot e_{23}(X)\cdot\mathbb{1}(X)$ is equivalent to $e_{22}(X)\odot_v(e_{23}(X)\cdot\mathbb{1}(X))$. Hence, we have removed one occurrence of the $\mathsf{diag}(\cdot)$ operation in $e(X)$ and replaced it by an occurrence of $\odot_v$. We can proceed in this way to obtain an expression $e'(X)$ in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\odot_v)$ which is equivalent to $e(X)$. Hence, also $G\equiv_{\mathsf{ML}(\cdot,{}^*,\mathbb{1},\odot_v)} H$ implies $G\equiv_{\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})} H$. □

The above proof fails when the trace operation is present. The reason is that we can have sentences like $e_{L,k}(X)$ in Example 8.1 which are not of the form $(\mathbb{1}(X))^*\cdot e(X)\cdot\mathbb{1}(X)$. For such sentences, the $\mathsf{diag}(\cdot)$ operation cannot be simply replaced by pointwise vector multiplication.

We next consider $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1}^{\mathsf{t}},\mathbb{1},\odot_v)$. Here, we incorporate the $\mathbb{1}^{\mathsf{t}}(\cdot)$ operation, introduced in the proof of Theorem 7.2, in order for the trace operation to also interact with matrices formed by vectors (e.g., one can formulate expressions like $\mathsf{tr}(e_1(X)\cdot(e_2(X))^{\mathsf{t}})$, where $e_1(X)$ and $e_2(X)$ evaluate to vectors). We recall from the proof of Theorem 7.2 that $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ if and only if $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1}^{\mathsf{t}},\mathbb{1},\mathsf{diag})} H$. Using the translation from $\odot_v$ into an expression involving the $\mathsf{diag}(\cdot)$ operation, as in the proof of Proposition 8.1, it then follows that $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ implies $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1}^{\mathsf{t}},\mathbb{1},\odot_v)} H$. We show that the implication from $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1}^{\mathsf{t}},\mathbb{1},\odot_v)} H$ to $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})} H$ does not hold, as anticipated in Example 8.1.

To analyse the distinguishability of graphs by sentences in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1}^{\mathsf{t}},\mathbb{1},\odot_v)$ we follow the same approach as for $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$.

Proposition 8.2 *Let G and H be two graphs of the same order. Then, $G\equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1}^{\mathsf{t}},\mathbb{1},\odot_v)} H$ implies that G and H have a common equitable partition.*

Proof In the proof of Proposition 7.2 we constructed a set Σ of sentences in $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$ such that when $e(A_G)=e(A_H)$ holds, for all $e(X)\in\Sigma$, then G and H must have a common equitable partition. A close inspection of these sentences shows that we only need complex conjugate transposition $({}^*)$ in the form of $(\mathbb{1}(X))^*$. We may thus safely replace $(\mathbb{1}(X))^*$ by $\mathbb{1}^{\mathsf{t}}(X)$ in the sentences in Σ. We next carry out the translation from sentences in Σ, as described in the proof of Proposition 8.1, to replace the occurrences of $\mathsf{diag}(\cdot)$ by $\odot_v$. Let us denote by Σ' the set of $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1}^{\mathsf{t}},\mathbb{1},\odot_v)$ obtained from Σ in this way. Then clearly, when $e'(A_G)=e'(A_H)$ holds for all $e'(X)\in\Sigma'$, we have that G and H have a common equitable partition, as desired. □

Furthermore, we can add pointwise vector multiplication to the list of operations in Proposition 7.4:

Proposition 8.3 $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot_v,\mathsf{diag},+,\times,\mathsf{apply_s}[f],f\in\Omega)$*-vectors are constant on equitable partitions.*

Proof We verify that $\odot_v$ can be added in the appendix. □

It remains to identify an appropriate notion of similarity for pointwise vector multiplication. Let G and H be two graphs that have a common equitable partition. As before, let $\mathcal{V}=\{V_1,\ldots,V_\ell\}$ and $\mathcal{W}=\{W_1,\ldots,W_\ell\}$ be such common partitions of G and H, respectively. The corresponding indicator vectors are denoted by $\mathbb{1}_{V_i}$ and $\mathbb{1}_{W_i}$, for $i=1,\ldots,\ell$, respectively. We say that a matrix T *preserves the coarsest equitable partitions of* G *and* H if $\mathbb{1}_{V_i}=T\cdot\mathbb{1}_{W_i}$ and $T^{\mathsf{t}}\cdot\mathbb{1}_{V_i}=\mathbb{1}_{W_i}$, for $i=1,\ldots,\ell$. We note that this condition is weaker than the compatibility notion used before (see the proof of Lemma 7.1 were we verified the *preservation* of the coarsest common equitable partitions for matrices that are *compatible* with the common coarsest equitable partition).

Lemma 8.1 *Let G and H be two graphs of the same order which have a common equitable partition. Let $\mathsf{ML}(\mathcal{L})$ be a matrix query language such that $\mathsf{ML}(\mathcal{L})$-vectors are constant on equitable partitions. Let T be a matrix which preserves the coarsest equitable partitions of G and H. Let $e_1(X)$ and $e_2(X)$ be expressions in $\mathsf{ML}(\mathcal{L})$ which evaluate to vectors. Then, if $e_1(A_G)$ and $e_1(A_H)$ are T-similar, and $e_2(A_G)$ and $e_2(A_H)$ are T-similar, then also $e_1(A_G)\odot_v e_2(A_G)$ and $e_1(A_H)\odot_v e_2(A_H)$ are T-similar.*

Proof The proof is similar to the proof of Lemma 7.1. Let $e_1(X)$ and $e_2(X)$ be two expressions in $\mathsf{ML}(\mathcal{L})$. Consider now $e'(X):=e_1(X)\odot_v e_2(X)$. We distinguish between three cases, depending on the dimensions of $e(A_G)$. First, if $e(A_G)$ is a sentence then we know by induction that $e_1(A_G)=e_1(A_H)$ and $e_2(A_G)=e_2(A_H)$. Hence,

$$
\begin{aligned}
e'(A_G) &= e_1(A_G)\odot_v e_2(A_G)=e_1(A_G)\cdot e_2(A_G)\\
&= e_1(A_H)\cdot e_2(A_H)=e_1(A_H)\odot_v e_2(A_H)=e'(A_H).
\end{aligned}
$$

Next, if $e_1(A_G)$ and $e_2(A_G)$ are (column) vectors, then we know that $e_1(A_G)=T\cdot e_1(A_H)$ and $e_2(A_G)=T\cdot e_2(A_H)$. We argued in the proof of Lemma 7.1 that when $\mathbb{1}_{V_i}=T\cdot\mathbb{1}_{W_i}$ holds for $i=1,\ldots,\ell$, then since vectors are constant on equitable partitions, $e_1(A_G)=\sum_{i=1}^{\ell}a_i\times\mathbb{1}_{V_i}=\sum_{i=1}^{\ell}a_i\times(T\cdot\mathbb{1}_{W_i})=T\cdot e_1(A_H)$ and $e_2(A_G)=\sum_{i=1}^{\ell}b_i\times\mathbb{1}_{V_i}=\sum_{i=1}^{\ell}b_i\times(T\cdot\mathbb{1}_{W_i})=T\cdot e_2(A_H)$. We may now conclude that

$$
\begin{aligned}
e'(A_G) &= e_1(A_G)\odot_v e_2(A_G)=\sum_{i=1}^{\ell}(a_i\times b_i)\times\mathbb{1}_{V_i}=\sum_{i=1}^{\ell}(a_i\times b_i)\times(T\cdot\mathbb{1}_{W_i})\\
&= T\cdot\Big(\sum_{i=1}^{\ell}(a_i\times b_i)\times\mathbb{1}_{W_i}\Big)=T\cdot(e_1(A_H)\odot_v e_2(A_H))=T\cdot e'(A_H).
\end{aligned}
$$

Hence, $e'(A_G)$ and $e'(A_H)$ are indeed T-similar. The case when $e_1(A_G)$ and $e_2(A_G)$ are row vectors is treated similarly, using that $T^{\mathsf{t}} \cdot \mathbb{1}_{V_i} = \mathbb{1}_{W_i}$, for $i = 1, \dots, \ell$. □

We can now state a characterisation of $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$-equivalence.

Theorem 8.1 *Let G and H be two graphs of the same order. Then, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)} H$ if and only if there exists an orthogonal matrix O which preserves the coarsest equitable partitions of G and H and such that $A_G \cdot O = O \cdot A_H$.*

Proof To show that the existence of a matrix O, as stated in the Theorem, implies $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\mathsf{diag})} H$, we argue as before. More precisely, we show that O-similarity is preserved by the operations in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$. This is, however, a direct consequence of Lemmas 5.1, 5.2, 6.1 and 8.1. We remark that Proposition 8.3 guarantees that Lemma 8.1 can be applied. Indeed, Proposition 8.3 implies that $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$-vectors are constant on equitable partitions. Furthermore, since $\mathbb{1}_{V_i} = O \cdot \mathbb{1}_{W_i}$, for all $i = 1, \dots, \ell$, and $\mathbb{1} = \sum_{i=1}^{\ell} \mathbb{1}_{V_i} = \sum_{i=1}^{\ell} \mathbb{1}_{W_i}$, we have that $\mathbb{1} = O \cdot \mathbb{1}$. Hence, O is doubly quasi-stochastic and Lemma 6.1 applies.

We may thus conclude that all expressions in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$ preserve O-similarity. Hence, $e(A_G) = e(A_H)$ for any sentence $e(X)$ in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$.

For the converse direction, we need to show that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)} H$ implies that there exists an orthogonal matrix O such that $A_G \cdot O = O \cdot A_H$, and where O preserves the coarsest equitable partitions of G and H. This can be shown, just like in the proof of Theorem 7.2, by means trace conditions. In particular, we impose trace conditions such that O satisfies $A_G \cdot O = O \cdot A_H$ and $(\mathbb{1}_{V_i} \cdot \mathbb{1}_{V_i}^{\mathsf{t}}) \cdot O = O \cdot (\mathbb{1}_{W_i} \cdot \mathbb{1}_{W_i}^{\mathsf{t}})$, for $i = 1, \dots, \ell$. These conditions replace conditions (7.3) and (7.4) in the proof of Theorem 7.2. We show in the appendix that this indeed implies that O preserves the coarsest equitable partitions of G and H. As observed in the proof of Theorem 7.2, the trace conditions $e_w(X)$ use expressions $\mathsf{eqpart}_i(X)$ (from the proof of Proposition 7.2 and revised in the proof of Proposition 8.2) which use addition and scalar multiplication. We again observe that addition and linear combination are not needed. Indeed, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)} H$ implies that $e_w(A_G) = e_w(A_H)$ because of the linearity of operations in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$. □

As it turns out, $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$-equivalence precisely captures co-spectral and fractional isomorphic graphs.

Proposition 8.4 *Let G and H be graphs of the same order. Then, $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)} H$ if and only if G and H are co-spectral and have a common equitable partition.*

Proof If $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)} H$, then G and H must have a common equitable partition by Proposition 8.2. Furthermore, we know Proposition 5.1 and Theorem 5.2, that G and H must also be co-spectral. For the converse, we explicitly construct an orthogonal matrix O such that $A_G \cdot O = O \cdot A_H$ and O preserves the coarsest equitable partitions of G and H. Then, Theorem 8.1 implies that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)} H$ holds.

We next construct the matrix O. Let G be of order n and denote by $\mathbb{1}_{V_1}, \dots, \mathbb{1}_{V_\ell}$ the indicator vectors of G's coarsest equitable partition. It is known that, for such indicator vectors, the subspace $U_G = \mathsf{span}(\mathbb{1}_1, \dots, \mathbb{1}_\ell)$ of ^{n} is an A_G-invariant subspace (see e.g., Lemma 5.2 in [14]). In other words, for any $v \in U_G$, $A_G \cdot v \in U_G$.

Furthermore, since A_G is a symmetric matrix, also the orthogonal complement subspace $U_G^{\perp}$ is A_G-invariant (see e.g., Theorem 36 in [47]). Here, $U_G^{\perp}$ consists of all vectors v' in ^{n} that are orthogonal to any vector $v \in U_G$, i.e., such $v^{\mathsf{t}} \cdot v' = 0$ holds. Let us interpret A_G as the linear operator $T_G : {}^n \to {}^n : v \mapsto A_G \cdot v$. This is a diagonalizable operator (because A_G is symmetric) and it is known that the restrictions $T_G|_{U_G}$ and $T_G|_{U_G^{\perp}}$ are also diagonalizable operators (because of the invariance of these two subspaces (see e.g., Corollary 15.9 in [33])). This implies that there exists eigenvectors $v_1, \ldots, v_\ell, v'_1, \ldots, v'_{n-\ell}$ of A_G such that $U_G = \mathsf{span}(v_1, \ldots, v_\ell)$ and $U_G^{\perp} = \mathsf{span}(v'_1, \ldots, v'_{n-\ell})$. Furthermore, if we denote by P_G the matrix with columns $\mathbb{1}_{V_1}, \ldots, \mathbb{1}_{V_\ell}$, then $A_G \cdot P_G = P_G \cdot C$ with C the $\ell \times \ell$-matrix such that $C_{ij} = \deg(v, V_j)$ for $v \in V_i$ (see e.g., Lemma 6.1 in [14]). Also C_{ij} is diagonalizable (this follows from the fact that the characteristic polynomial of C divides that of A_G (see e.g., Theorem 6.2 in [14]) and hence there exists ℓ linearly independent eigenvectors $c_1, \ldots, c_\ell$ of C. It is known that $v_i = P_G \cdot c_i$, for $i = 1, \ldots, \ell$, are independent eigenvectors of A_G. More precisely, if $C \cdot c_i = \lambda_i \times c_i$ then $A_G \cdot (P_G \cdot c_i) = \lambda_i \times (P_G \cdot c_i)$. We may thus assume that U_G is spanned by $P_G \cdot c_1, \ldots, P_G \cdot c_\ell$.

The reasoning above also holds for A_H, i.e., there are eigenvectors $w_1, \ldots, w_\ell, w'_1, \ldots, w'_{n-\ell}$ of A_H such that $U_H = \mathsf{span}(w_1, \ldots, w_\ell)$ and $U_H^{\perp} = \mathsf{span}(w'_1, \ldots, w'_{n-\ell})$. Important to observe here is that since G and H have a common equitable partition, $A_H \cdot P_H = P_H \cdot C$, where P_H is now the matrix with columns $\mathbb{1}_{W_1}, \ldots, \mathbb{1}_{W_\ell}$ and C is the same $\ell \times \ell$-matrix as used above. We may thus assume that U_H is spanned by $P_H \cdot c_1, \ldots, P_H \cdot c_\ell$ and furthermore, $P_G \cdot c_i$ and $P_H \cdot c_i$ are eigenvectors of A_G and A_H, respectively, both belonging to the same eigenvalue λ_i of C.

We next use that G and H are co-spectral. The argument above, combined with co-spectrality, implies that the (multiset) of eigenvalues corresponding to the eigenvectors spanning U_G and U_H are the same. This implies in turn, by co-spectrality, that we may also assume that $A_G \cdot v'_i = \lambda_i \times v'_i$ and $A_H \cdot w'_i = \lambda_i \times w'_i$, for $i = 1, \ldots, n-\ell$, for some eigenvalues λ_i of A_G (and A_H). A final observation is that U_G and U_H are also spanned by $\mathbb{1}_{V_i}, \ldots, \mathbb{1}_{V_\ell}$ and $\mathbb{1}_{W_1}, \ldots, \mathbb{1}_{W_\ell}$, respectively. This implies, that the eigenvectors spanning $U_G^{\perp}$ and $U_H^{\perp}$ are necessarily orthogonal to these indicator vectors.

We define O as the matrix $O_G \cdot O_H^{\mathsf{t}}$, where O_G is the orthonormal matrix consisting of vectors $\frac{1}{n_1}\mathbb{1}_{V_1}, \ldots, \frac{1}{n_\ell}\mathbb{1}_{V_\ell}, v'_1, \ldots, v'_{n-\ell}$ and O_H is the orthonormal matrix consisting of vectors $\frac{1}{n_1}\mathbb{1}_{W_1}, \ldots, \frac{1}{n_\ell}\mathbb{1}_{W_\ell}, w'_1, \ldots, w'_{n-\ell}$, where $n_i = |V_i| = |W_i|$ and were we assume the eigenvectors v'_i and w'_i to be normalized. As a consequence, O is clearly an orthogonal matrix and thus $O \cdot O^{\mathsf{t}} = I = O^{\mathsf{t}} \cdot O$ holds. In view of the construction of the eigenvectors, we have the following more simple expression for O:

$$O = \sum_{j=1}^{\ell} \Big(\frac{1}{n_j} \times (\mathbb{1}_{V_j} \cdot \mathbb{1}_{W_j}^{\mathsf{t}}) \Big) + \sum_{j=1}^{n-\ell} v'_j \cdot (w'_j)^{\mathsf{t}}.$$

We verify the required conditions. To begin with, we note that $O \cdot \mathbb{1}_{W_i} = \mathbb{1}_{V_i}$, for $i = 1, \ldots, \ell$. Indeed, this follows from the fact that $\mathbb{1}_{W_j}^{\mathsf{t}} \cdot \mathbb{1}_{W_i}$ is zero when $i \neq j$ and is $|W_i| = n_i$ when $i = j$. Moreover, $(w'_j)^{\mathsf{t}} \cdot \mathbb{1}_{W_i} = 0$ because of $w'_j \in U_H^{\perp}$, for all $j = 1, \ldots, n-\ell$. Similarly, $\mathbb{1}_{V_i}^{\mathsf{t}} \cdot O = \mathbb{1}_{W_i}^{\mathsf{t}}$, for $i = 1, \ldots, \ell$. Hence, O indeed preserves the coarsest equitable partitions of G and H. It remains to verify that $A_G \cdot O = O \cdot A_H$.

We verify this for both terms in the above expression for O. Since v_i' and w_i' are eigenvectors of A_G and A_H, respectively, belong to the same eigenvalue λ_i, we have for the second term:

$$A_G \cdot \Big(\sum_{j=1}^{n-\ell} v_j' \cdot (w_j')^{\mathsf{t}}\Big) = \sum_{j=1}^{n-\ell} A_G \cdot v_j' \cdot (w_j')^{\mathsf{t}} = \sum_{j=1}^{n-\ell} \lambda_j \times (v_j' \cdot (w_j')^{\mathsf{t}})$$

$$= \sum_{j=1}^{n-\ell} v_j' \cdot (w_j')^{\mathsf{t}} \cdot A_H = \Big(\sum_{j=1}^{n-\ell} v_j' \cdot (w_j')^{\mathsf{t}}\Big) \cdot A_H .$$

For the first term in the expression for O, we consider the matrices

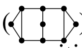
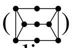
$$B_G = A_G \cdot \Big(\sum_{i=1}^{\ell} \frac{1}{n_j} \times (\mathbb{1}_{V_i} \cdot \mathbb{1}_{W_i}^{\mathsf{t}})\Big) = \sum_{i=1}^{\ell}\sum_{j=1}^{\ell} \big(\frac{1}{n_i} \times \deg(v_i, V_j)\big) \times (\mathbb{1}_{V_j} \cdot \mathbb{1}_{W_i}^{\mathsf{t}})$$

$$B_H = \Big(\sum_{i=1}^{\ell} \frac{1}{n_i} \times (\mathbb{1}_{V_i} \cdot \mathbb{1}_{W_i}^{\mathsf{t}})\Big) \cdot A_H = \sum_{i=1}^{\ell}\sum_{j=1}^{\ell} \big(\frac{1}{n_i} \times \deg(w_i, W_j)\big) \times (\mathbb{1}_{V_i} \cdot \mathbb{1}_{W_j}^{\mathsf{t}}),$$

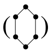
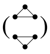
for some (arbitrary) vertices $v_i \in V_i$ and $w_i \in W_i$. We here used that the indicator vectors represent equitable partitions. We now look at the entries in the matrices B_G and B_H. We first observe that $J = \sum_{i,j=1}^{\ell} \mathbb{1}_{V_j} \cdot \mathbb{1}_{W_i}^{\mathsf{t}}$. Hence, for each $p, q \in \{1, \dots, n\}$ we can define $f(p)$ and $f(q)$ as the unique indexes of indicator vectors $\mathbb{1}_{V_{f(p)}}$ and $\mathbb{1}_{W_{f(q)}}$ such that they hold value 1 at position p and q, respectively. Then,

$$(B_G)_{p,q} = \frac{1}{n_{f(p)}} \times \deg(v_{f(p)}, V_{f(q)}) = \frac{1}{n_{f(p)}} \times \deg(w_{f(p)}, W_{f(q)}) = (B_H)_{p,q},$$

because the indicator vectors represent common equitable partitions. Hence, we may indeed conclude that $A_G \cdot O = O \cdot A_H$. □

Example 8.2 We already mentioned that the graphs G_6 () and H_6 () are co-spectral and have a common equitable partition. Proposition 8.4 implies that $G_6 \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)} H_6$, as anticipated. □

We conclude by mentioning that we can extend $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$ with $+$, $\times$, $*$, and pointwise function applications on scalars, without increasing the distinguishing power of the fragments. This can be shown in precisely the same way as for $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag})$. Indeed, we have just seen that $G \equiv_{\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)} H$ implies that $A_G \cdot O = O \cdot A_H$ for some orthogonal matrix O which preserves the coarsest equitable partitions of G and H. Then, also $A_H \cdot O^* = O^* \cdot A_G$ where O^* is again orthogonal and also preserves the coarsest equitable partitions of G and H. It now suffices to observe that all operations in $\mathsf{ML}(\cdot, {}^*, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v, +, \times, \mathsf{apply_s}[f], f \in \Omega)$ preserve O-similarity and O^*-similarity. An inspection of the proof of Proposition 7.5 shows that we can replace the compatibility assumption of O by the preservation of equitable partition condition when using $\odot_v$ instead of $\mathsf{diag}(\cdot)$. Hence, also pointwise function applications on vector preserve O and O^*-similarity and do not add expressive power when included in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$.

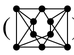
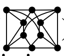
As a consequence, $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$-equivalence and $\mathsf{ML}(\cdot, {}^*, \mathsf{tr}, \mathbb{1}, \odot_v)$-equivalence coincide (we note that we here replace $\mathbb{1}^{\mathsf{t}}(\cdot)$ with $*$). Hence, $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \odot_v)$-equivalence implies $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \odot_v)$-equivalence, since the latter is a smaller fragment than $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\odot_v)$. Proposition 8.1 then implies that $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)$ also implies $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$-equivalence. The reverse implication does not hold. Indeed, we have already seen that G_3 () and H_3 () are two fractionally isomorphic graphs that are not co-spectral. So, these graphs can be distinguished by $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\odot_v)$ but not by $\mathsf{ML}(\cdot,{}^*,\mathbb{1},\mathsf{diag})$.

9 The impact of pointwise functions on matrices

The final operation that we consider is pointwise function applications on *matrices*. In particular, we start by considering the Schur-Hadamard product, which we denote by the binary operator $\odot$, i.e., $(A\odot B)_{ij} = A_{ij}B_{ij}$ for matrices A and B. We show that once two graphs are equivalent with regards to sentences in $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)$, then they will be equivalent with regards to sentences in $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\mathsf{apply}[f],f\in\Omega)$ for *any* pointwise function application $\mathsf{apply}[f]$, be it on scalars, vector or matrices. The latter fragment corresponds to $\mathsf{MATLANG}$, as introduced by Brijder et al. [10] and described in Section 3. From the work by Brijder et al. [10] it implicitly follows that C^3-equivalence implies $\mathsf{MATLANG}$-equivalence. The main result established in this section is that converse implication also holds. That is, $\mathsf{MATLANG}$-equivalence coincides with C^3-equivalence. We first illustrate the additional power that the Schur-Hadamard product provides by means of an example.

Example 9.1 We recall that in expression $\#3\mathsf{degr}(X)$ in Example 7.1, products of diagonal matrices resulted in the ability to zoom in on *vertices* that carry specific degree information. When diagonal matrices are concerned, the product of matrices coincides with pointwise multiplication of the *vectors* on the diagonals. Allowing pointwise multiplication on matrices has the same effect, but now on *edges* in graphs. As an example, suppose that we want to count the number of "triangle paths" in G, i.e., paths $(v_0,\dots,v_k)$ of length k in G such that each edge (v_{i-1},v_i) on the path is part of a triangle. This can be done by expression

$$\#\Delta\mathsf{paths}_k(X) := \mathbb{1}(X)^*\cdot((\mathsf{apply}[f_{>0}](X^2\odot X))^k\cdot\mathbb{1}(X),$$

where $f_{>0}(x)=1$ if $x\neq 0$ and $f_{>0}(x)=0$ otherwise[4]. Indeed, when evaluated on adjacency matrix A_G, $A_G^2\odot A_G$ extracts from A_G^2 only those entries corresponding to paths (u,v,w) of length 2 such that (u,w) is an edge as well, i.e., it identifies edges involved in triangles in G. Then, $\mathsf{apply}[f_{>0}](A_G^2\odot A_G)$ sets all non-zero entries to 1. By considering the kth power of this matrix and summing up all its entries, the number of triangle paths of length k is obtained. It can be verified that for graphs G_5 () and H_5 (), $\#\Delta\mathsf{paths}_2(A_{G_5}) = [160] \neq [132] = \#\Delta\mathsf{paths}_2(A_{H_5})$ and hence, they can be distinguished when the Schur-Hadamard product is available. Recall that all previous fragments could not distinguish between these two graphs. □

In fact, we will use the Schur-Hadamard product to compute *stable edge partitions* of graphs, obtained as the result of the edge colouring algorithm by Weisfeiler-

[4] The use of $\mathsf{apply}[f_{>0}](\cdot)$ is just for convenience. Its application inside sentences can be simulated with operators in $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)$ when evaluated on given adjacency matrices.

Algorithm 2: Computing the stable edge colouring based on algorithm 2-Stab [7].

Input : A graph $G=(V,E)$ of order n.
Output : Stable edge colouring $\chi: V\times V\to C$.
1 Let $\chi:=\chi_0$;
2 Let $C:=\{0,1,2\}$;
3 **repeat**
4   **for** $(v_1,v_2)\in V\times V$ **do**
5   Compute $\mathsf{L}^2(v_1,v_2)$ relative to χ;
6   Replace C by a minimal set of new colours C' and define $\chi': V\times V\to C'$ such that
7   **for** *pairs* (v_1,v_2), (v_1',v_2') *in* $V\times V$ **do**
8     $\chi'(v_1,v_2)=\chi'(v_1',v_2')\Leftrightarrow \mathsf{L}^2(v_1,v_2)=\mathsf{L}^2(v_1',v_2')$
9   Let $C:=C'$;
10   Let $\chi:=\chi'$;
11 **until** $|C|$ does not change;

Lehman [7,13,57,69]. Such partitions can be seen as a generalization of equitable partitions, but now partitioning all pairs of vertices, rather than vertices. Then, similar to the proof of Proposition 7.2, we show that when two graphs are indistinguishable by sentences in $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)$, then they are indistinguishable by edge colouring. It is known from the seminal paper by Cai, Fürer and Immerman [13], that this is equivalent to C^3-equivalence. We next detail these notions.

9.1 Stable edge partitions

The stable edge partition of a graph $G=(V,E)$ arises as the result of applying the edge colouring algorithm by Weisfeiler-Lehman [7,13,57,69], also known as the 2-dimensional Weisfeiler-Lehman algorithm, on G. In Algorithm 2 we provide the pseudo-code of the algorithm 2-Stab, taken from Bastert [7], which implements edge colouring. In a nutshell, the algorithm starts by assigning every vertex pair a colour, and then revises colourings iteratively based on some structural information. When no revision of the colouring occurs, the colouring has stabilized, the algorithm stops and returns the stable colouring. Colourings naturally induce partitions of $V\times V$, by simply grouping together vertex pairs with the same colour. The stable edge partition of G is the partition induced by the stable colouring returned by 2-Stab. The algorithm 2-Stab needs at most n^2 iterations when evaluated on a graph of order n.

More precisely, an (*edge*) *colouring* χ assigns a colour to each vertex pair in $V\times V$, i.e., if we denote by C a set of colours, it is a function $\chi: V\times V\to C$. The partition of $V\times V$ induced by χ is denoted by $\Pi_\chi(G)$ and will be represented by *indicator matrices*, one for each colour $c\in C$. More precisely, for a colour $c\in C$, we denote by E_c the $n\times n$-matrix such that for $v_1,v_2\in V$, $(E_c)_{v_1,v_2}=1$ if $\chi(v_1,v_2)=c$ and $(E_c)_{v_1,v_2}=0$, otherwise. Hence, $\Pi_\chi(G)$ is represented by the indicator matrices E_c, for $c\in C$.

Algorithm 2-Stab starts (on lines 1 and 2) with an initial colouring $\chi_0: V\times V\to \{0,1,2\}$ encoding adjacency, non-adjacency and loop information. More precisely, for vertices $v,w\in V$, $\chi_0(v,v)=2$, $\chi_0(v,w)=1$ if $(v,w)\in E$, and $\chi_0(v,w)=0$ for

$v \neq w$ and $(v,w) \notin E$. Then, 2-Stab adjusts the current colouring in each iteration, as follows.

Suppose that the current colouring is $\chi : V \times V \to C$. Given this colouring, for each pair of vertices $v_1, v_2 \in V$, the so-called *structure list* $\mathsf{L}^2(v_1,v_2)$ is computed (lines 4 and 5). To define these lists, the *structure constants* are needed, which are defined as

$$p^{c,d}_{v_1,v_2} := |\{v_3 \in V \mid \chi(v_1,v_3) = c, \chi(v_3,v_2) = d\}|,$$

for colours c and d in C and vertices v_1 and v_2 in V. These numbers count the number of triangles[5], based on (v_1,v_2) whose other two pairs (v_1,v_3) and (v_3,v_2) have prescribed colours c and d, respectively. Then, in a structure list we simply gather all these constants for a specific vertex pair. That is,

$$\mathsf{L}^2(v_1,v_2) := \{(c,d,p^{c,d}_{v_1,v_2}) \mid p^{c,d}_{v_1,v_2} \neq 0\}.$$

Based on this information, 2-Stab will assign new colours to pairs of vertices (lines 6–8). More precisely, C is replaced by a minimal set of colours C' such that each unique $\mathsf{L}^2(v_1,v_2)$ corresponds precisely to a single colour c' in C'. Hence, the new colouring $\chi' : V \times V \to C'$ will assign (v_1', v_2') the colour c', corresponding to $\mathsf{L}^2(v_1,v_2)$, when $\mathsf{L}^2(v_1,v_2) = \mathsf{L}^2(v_1',v_2')$. It is easily verified that the partition $\Pi_{\chi'}(G)$ is a refinement of $\Pi_{\chi}(G)$, which in turn is a refinement of $\Pi_{\chi_0}(G)$.

Algorithm 2-Stab now replaces χ by χ' and C by C' (lines 9 and 10), and repeats this process until the number of colours remains fixed (line 11). In other words, the corresponding partition is not further refined. The algorithm returns the final (stable) colouring.

The *stable edge partition of* G, denoted by $\Pi(G)$, is now the partition induced by this stable colouring. It is known that $\Pi(G)$ is the unique coarsest partition of $V \times V$ which refines $\Pi_{\chi_0}(G)$ and corresponding to a colouring satisfying the stability condition stated on lines 7 and 8 in Algorithm 2.

Two graphs $G = (V,E)$ and $H = (W,F)$ of the same order are now said to be *indistinguishable by edge colouring*, denoted by $G \equiv_{\mathsf{WL}} H$, if the stable edge partitions $\Pi(G)$ and $\Pi(H)$ of G and H, respectively, are (i) of the form $\Pi(G) = \{E_{c_1}, \dots, E_{c_\ell}\}$ and $\Pi(H) = \{F_{c_1}, \dots, F_{c_\ell}\}$, that is, the parts in the partitions correspond to the same colour; and (ii) the corresponding parts in these partitions have the same size, that is, E_{c_i} and F_{c_i} have the same number of entries carrying the value 1.

In the seminal paper by Cai, Fürer and Immerman [13], the connection with logical indistinguishability was made.

Theorem 9.1 *Let G and H be two graphs of the same order. Then, $G \equiv_{\mathsf{WL}} H$ if and only if $G \equiv_{\mathsf{C}^3} H$.* □

In this section, we complement this correspondence by relating C^3-equivalence to $\mathsf{MATLANG}$-equivalence. More precisely, we show that $G \equiv_{\mathsf{C}^3} H$ if and only if $G \equiv_{\mathsf{MATLANG}} H$. In fact, equivalence with regards to sentences in $\mathsf{ML}(\cdot, {}^*, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, \odot)$ already suffices. We first show that $\mathsf{ML}(\cdot, {}^*, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, \odot)$-equivalence implies indistinguishability by edge colouring.

[5] With a triangle one simply means a triple (v_1,v_2), (v_1,v_3) and (v_2,v_3) of vertex pairs, none of which has to be an edge in G.

Proposition 9.1 *Let G and H be graphs of the same order. Then, $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)} H$ implies that $G \equiv_{\mathsf{WL}} H$.*

Proof We first show that $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\odot)$, where we added addition and scalar multiplication to $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)$, has sufficient power to compute the stable edge partition $\Pi(G)$ of a given graph G. We then construct sentences in $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\odot)$ such that when G and H agree on these sentences, then G and H must be indistinguishable by edge colouring. Finally, we show that we can eliminate addition and scalar multiplication.

The overall proof is similar to the proof of Proposition 7.2, but using indicator matrices (representing the edge partitions) instead of indicator vectors (which represented the vertex partitions), and by relying on the algorithm 2-Stab to compute the stable edge partition of a graph.

Given G, let $\Pi(G)=\{E_{c_1},\dots,E_{c_\ell}\}$ be its stable edge partition. We show that we can construct expressions $\mathsf{stabcol}_{c_i}(X)$ in $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\odot)$, such that $E_{c_i}=\mathsf{stabcol}_{c_i}(A_G)$, for $i=1,\dots,\ell$.

The initialization step of 2-Stab is easy to simulate in $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\odot)$. Indeed, we simply consider expressions $\mathsf{stabcol}_2^{(0)}(X):=\mathsf{diag}(\mathbb{1}(X))$; $\mathsf{stabcol}_1^{(0)}(X):=X$; and $\mathsf{stabcol}_0^{(0)}(X):=\mathbb{1}(X)\cdot(\mathbb{1}(X))^*-X-\mathsf{diag}(\mathbb{1}(X))$. Then, the indicator matrices $\mathsf{stabcol}_0^{(0)}(A_G)$, $\mathsf{stabcol}_1^{(0)}(A_G)$, and $\mathsf{stabcol}_2^{(0)}(A_G)$ represent the initial partition $\Pi_{\chi_0}(G)=\{E_0,E_1,E_2\}$ corresponding to the initial colouring χ_0.

Suppose now that after iteration i, the current set of colours is C and the colouring is $\chi:V\times V\to C$. Assume, by induction, that we have expressions $\mathsf{stabcol}_c^{(i)}(X)$ in $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\odot)$, one for each $c\in C$, such that $\mathsf{stabcol}_c^{(i)}(A_G)$ is an indicator matrix representing the part in the edge partition $\Pi_\chi(G)$, induced by χ, for colour c. Given these, we next construct expressions for the refined partition computed by 2-Stab in the next iteration.

First, for each pair of colours (c,d) in C, we consider the expression

$$P_{c,d}^{(i+1)}(X):=\mathsf{stabcol}_c^{(i)}(X)\cdot\mathsf{stabcol}_d^{(i)}(X).$$

On input A_G, it is readily verified that $P_{c,d}^{(i)}(A_G)$ is a matrix whose entry corresponding to vertices v_1 and v_2 holds the value $p_{v_1,v_2}^{c,d}$. Let $\mathcal{P}_{c,d}^{(i+1)}$ be the set of numbers occurring in $P_{c,d}^{(i+1)}(A_G)$. For each value p in $\mathcal{P}_{c,d}^{(i+1)}$, we now extract an indicator matrix indicating the positions in $P_{c,d}^{(i+1)}(A_G)$ that hold value p.

This can be done using an expression $\mathsf{ind}_{c,d,p}^{(i+1)}(X)$ which works in a similar way as $\#3\mathsf{deg}(X)$ in Example 7.1, but uses the Schur-Hadamard product instead of products of diagonal matrices. The following example illustrates the underlying idea (see also the Schur-Wielandt Principle [58] mentioned before).

Example 9.2 Consider $P_{c,d}=\begin{pmatrix}2&0&3\\1&3&2\\0&2&3\end{pmatrix}$ with $\mathcal{P}_{c,d}=\{0,1,2,3\}$. Suppose that we want to find all entries holding value 3. This can be computed, as follows:

$$\begin{pmatrix}0&0&1\\0&1&0\\0&0&1\end{pmatrix}=\frac{1}{6}\times\left(\begin{pmatrix}2&0&3\\1&3&2\\0&2&3\end{pmatrix}\odot\left(\begin{pmatrix}2&0&3\\1&3&2\\0&2&3\end{pmatrix}-\begin{pmatrix}1&1&1\\1&1&1\\1&1&1\end{pmatrix}\right)\odot\left(\begin{pmatrix}2&0&3\\1&3&2\\0&2&3\end{pmatrix}-\begin{pmatrix}2&2&2\\2&2&2\\2&2&2\end{pmatrix}\right)\right),$$

where $\frac{1}{6}=\frac{1}{3(3-1)(3-2)}$, just as in Example 7.1. □

More generally, to identify positions that hold a specific value in $P^{(i+1)}_{c,d}(A_G)$, we consider the expression $\mathsf{ind}^{(i+1)}_{c,d,p}(X)$ defined by

$$\left(\frac{1}{\prod_{p'\in\mathcal{P}^{(i+1)}_{c,d},p\neq p'}(p-p')}\right)\times\bigodot_{p'\in\mathcal{P}^{(i+1)}_{c,d},p\neq p'}\big(P^{(i+1)}_{c,d}(X)-p'\times(\mathbb{1}(X)\cdot(\mathbb{1}(X))^*)\big).$$

It should be clear from Example 9.2 that $\mathsf{ind}^{(i+1)}_{c,d,p}(A_G)$ indeed results in the desired indicator matrix. We note that the expression $\mathsf{ind}^{(i+1)}_{c,d,p}(X)$ depends on the values in $\mathcal{P}^{(i+1)}_{c,d}$ and hence also depends on A_G.

Let C' be the new set of colours assigned by $\mathsf{2\text{-}Stab}(G)$ during the current iteration. As mentioned earlier, each colour c in C' is in correspondence with $\mathsf{L}^2(v_1,v_2)$ for some vertices v_1 and v_2. Let us pick a colour c in C' and assume that it corresponds to $\mathsf{L}^2(v_1,v_2)=\{(c_1,d_2,p^{c_1,d_1}_{v_1,v_2}),\dots,(c_s,d_s,p^{c_s,d_s}_{v_1,v_2})\}$. We next use $\mathsf{ind}^{(i+1)}_{c,d,p}(X)$ and the Schur-Hadamard product to identify all vertex pairs that are assigned colour c, as follows:

$$\mathsf{stabcol}^{(i+1)}_c(X):=\mathsf{ind}^{(i+1)}_{c_1,d_2,p^{c_1,d_1}_{v_1,v_2}}(X)\odot\cdots\odot\mathsf{ind}^{(i+1)}_{c_s,d_s,p^{c_s,d_s}_{v_1,v_2}}(X).$$

In other words, we use the Schur-Hadamard product to simulate the “conjunction” of the binary matrices representing the vertex pairs (v_1,v_2) having non-zero $p^{c_i,d_i}_{v_1,v_2}$, for $i=1,\dots,s$. It is now easily verified that, on input A_G, $\mathsf{stabcol}^{(i+1)}_c(A_G)$ returns an indicator matrix in which the entries holding a 1 correspond precisely to the pairs $(v_1',v_2')\in V\times V$ such that $\mathsf{L}^2(v_1',v_2')=\mathsf{L}^2(v_1,v_2)$ where $\mathsf{L}^2(v_1,v_2)$ corresponds to colour c. In other words, $\mathsf{stabcol}^{(i+1)}_c(A_G)$ represents the refined edge partition corresponding to the part associated with colour c. We do this for every colour in C'. Clearly, $\mathsf{stabcol}^{(i+1)}_c(A_G)$, for $c\in C'$, represent the refined partition $\Pi_{\chi'}(G)$ corresponding to $\chi':V\times V\to C'$.

We continue in this way until the colouring stabilises. i.e., no further colours are needed. We denote the final set of colours by C and by $\mathsf{stabcol}_c(X)$, for $c\in C$, the $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\mathsf{diag},+,\times,\odot)$ expression computing the parts E_c in $\Pi(G)$. The correctness of these expressions follows from the previous arguments and the correctness of the algorithm 2-Stab.

Just as in the proof of Proposition 7.2, the expressions $\mathsf{stabcol}_c(X)$ depend on A_G since we explicitly used the values occurring in $P^{(i)}_{c,d}(A_G)$ and the colours assigned to vertex pairs during each iteration i of 2-Stab on G. Let $\Pi(H)$ be stable edge partition of H. We next show that $G\equiv_{\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},+,\times,\odot)}H$ implies that $\Pi(H)$ consists of $\mathsf{stabcol}_c(A_H)$, for $c\in C$. Furthermore, we show that the number of ones in $\mathsf{stabcol}_c(A_G)$ and $\mathsf{stabcol}_c(A_H)$ agree for all $c\in C$. Hence, G and H are indistinguishable by edge colouring.

The proof is by induction on the number of iterations of 2-Stab(G) and 2-Stab(H). We denote by $\chi^{(i)}_G:V\times V\to C^{(i)}_G$ and $\chi^{(i)}_H:W\times W\to C^{(i)}_H$ the colouring used in the ith iteration of 2-Stab(G) and 2-Stab(H), respectively. The induction hypothesis is that $G\equiv_{\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},+,\times,\odot)}H$ implies that $C^{(i)}_G=C^{(i)}_H=C^{(i)}$ and furthermore that for each $c\in C^{(i)}$, $\mathsf{stabcol}^{(i)}_c(A_H)$ is an indicator matrix, and all $\mathsf{stabcol}^{(i)}_c(A_H)$ together constitute the edge partition $\Pi_{\chi_H^{(i)}}(H)$. Moreover, we show that for each $c \in C^{(i)}$, $\mathsf{stabcol}_c^{(i)}(A_G)$ and $\mathsf{stabcol}_c^{(i)}(A_H)$ have the same number of ones. This clearly suffices, for if this holds, $\mathsf{stabcol}_c(A_H)$, for $c \in C$, constitute $\Pi(G)$ and $\mathsf{stabcol}_c(A_G)$ and $\mathsf{stabcol}_c(A_H)$ have the same number of ones, for all $c \in C$.

We start by verifying the hypothesis for the base case, i.e., when $i=0$. Clearly, $\chi_G^{(0)}$ and $\chi_H^{(0)}$ use the same colours $C_G^{(0)} = C_H^{(0)} = C^{(0)} = \{0,1,2\}$. By definition of the expressions $\mathsf{stabcol}_c^{(0)}(X)$, all $\mathsf{stabcol}_c^{(0)}(A_H)$ together represent $\Pi_{\chi_H^{(0)}}(H)$. Moreover, by considering the sentences

$$\#\mathsf{ones}_c^{(0)}(X) := (\mathbb{1}(X))^* \cdot \mathsf{stabcol}_c^{(0)}(X) \cdot \mathbb{1}(X),$$

for $c \in C$, $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},+,\times,\odot)} H$ implies that $\#\mathsf{ones}_c^{(0)}(A_G) = \#\mathsf{ones}_c^{(0)}(A_H)$. Hence, $\mathsf{stabcol}_c^{(0)}(A_G)$ and $\mathsf{stabcol}_c^{(0)}(A_H)$ have the same number of ones, as desired.

Suppose, by induction, that $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},+,\times,\odot)} H$ implies that $\chi_G^{(i)} : V \times V \to C_G^{(i)}$ and $\chi_H^{(i)} : W \times W \to C_H^{(i)}$ with $C_G^{(i)} = C_H^{(i)} = C^{(i)}$. Furthermore, the current edge partition $\Pi_{\chi_H^{(i)}}(H)$ of H is represented by $\mathsf{stabcol}_c^{(i)}(A_H)$, for $c \in C^{(i)}$. Furthermore, for each $c \in C^{(i)}$, the number of ones in $\mathsf{stabcol}_c^{(i)}(A_H)$ and $\mathsf{stabcol}_c^{(i)}(A_G)$ agree.

As before, let $\mathcal{P}_{c,d}^{(i+1)}$ be the set of values occurring in $P_{c,d}^{(i+1)}(A_G)$ and consider the expressions $\mathsf{ind}_{c,d,p}^{(i+1)}(X)$ for $c,d \in C^{(i)}$ and $p \in \mathcal{P}_{c,d}^{(i+1)}$. We show that $\mathsf{ind}_{c,d,p}^{(i+1)}(A_H)$ is a binary matrix as well containing the same number of ones as $\mathsf{ind}_{c,d,p}^{(i+1)}(A_G)$. This implies that each value $p \in \mathcal{P}_{c,d}^{(i+1)}$ occurs in $P_{c,d}^{(i+1)}(A_H)$ and moreover, it occurs the same number of times as in $P_{c,d}^{(i+1)}(A_G)$. Hence, the set of values occurring in $P_{c,d}^{(i+1)}(A_H)$ is the same as those occurring in $P_{c,d}^{(i+1)}(A_G)$.

To check that $\mathsf{ind}_{c,d,p}^{(i+1)}(A_H)$ is a binary matrix, we use the sentence

$$\mathsf{binary}(X) := (\mathbb{1}(X))^* \cdot \big((X \odot X - X) \odot (X \odot X - X)\big) \cdot \mathbb{1}(X).$$

This sentence will return $[0]$, when given a real matrix as input, if and only if the input matrix is a binary matrix. Indeed, for a binary matrix B, $B \odot B = B$ and hence $B \odot B - B = Z$, where Z is the zero matrix. Since $Z \odot Z = Z$, $\mathsf{binary}(B) = \mathbb{1}^{\mathsf{t}} \cdot Z \cdot \mathbb{1} = [0]$. For the converse, assume that $\mathsf{binary}(B) = [0]$. We observe that each entry in $(B \odot B - B) \odot (B \odot B - B)$ is non-negative value. Indeed, all entries are squares of real numbers. Hence, when $\mathsf{binary}(B) = [0]$, the sum of all these squared entries must be zero. This implies that $B \odot B - B = Z$. This in turn implies that B can only contain 0 or 1 as entries, since these are the only real values satisfying $x^2 - x = 0$. Hence, when $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},+,\times,\odot)} H$ holds, then since all $\mathsf{ind}_{c,d,p}^{(i+1)}(A_G)$, for $c,d \in C^{(i)}$ and $p \in \mathcal{P}_{c,d}^{(i+1)}$, are binary matrices,

$$\mathsf{binary}(\mathsf{ind}_{c,d,p}^{(i+1)}(A_G)) = [0] = \mathsf{binary}(\mathsf{ind}_{c,d,p}^{(i+1)}(A_H)).$$

So indeed, $\mathsf{ind}_{c,d,p}^{(i+1)}(A_H)$ is a binary matrix as well.

The new colours in 2-Stab(G) are assigned based on the structure lists $\mathsf{L}^2(v_1,v_2)$. We show that for every unique structure list $\mathsf{L}^2(v_1,v_2)$ there is a pair of vertices w_1, w_2 in W such that $\mathsf{L}^2(v_1,v_2) = \mathsf{L}^2(w_1,w_2)$. This implies that 2-Stab(H) will use the same colours for refining $\chi_H^{(i)}$ as 2-Stab(G) uses to refine $\chi_G^{(i)}$. Hence, the revised colourings $\chi_G^{(i+1)}: V \times V \to C_G^{(i+1)}$ and $\chi_H^{(i+1)}: W \times W \to C_H^{(i+1)}$ satisfy indeed that $C_G^{(i+1)} = C_H^{(i+1)} = C^{(i+1)}$.

Consider a structure list $\mathsf{L}^2(v_1, v_2)$ and assume that it corresponds to a new colour $c \in C_G^{(i+1)}$. We know that $\mathsf{stabcol}_c^{(i+1)}(A_G)$ returns the indicator matrix indicating which vertex pairs in $V \times V$ have this structure list (colour c). The expression $\mathsf{stabcol}_c^{(i+1)}(X)$ consists of the Schur-Hadamard product of $\mathsf{ind}_{c,d,p}^{(i+1)}(X)$ for every (c, d, p) in $\mathsf{L}^2(v_1, v_2)$. We have shown above that $\mathsf{ind}_{c,d,p}^{(i+1)}(A_G)$ and $\mathsf{ind}_{c,d,p}^{(i+1)}(A_H)$ contain the same number of ones, meaning that there are vertex pairs $(w_1, w_2) \in W \times W$ for which $p_{w_1,w_2}^{c,d} = p = p_{v_1,v_2}^{c,d}$. Furthermore, in a similar way as above, we can show that $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},+,\times,\odot)} H$ implies that $\mathsf{stabcol}_c^{(i+1)}(A_H)$ is a binary matrix which consists of the same number of ones as $\mathsf{stabcol}_c^{(i+1)}(A_G)$. So, 2-Stab$(H)$ needs the same set of colours $C_G^{(i+1)}$ as 2-Stab(H) in the refinement phase. Hence, we can take $C_G^{(i+1)} = C_H^{(i+1)} = C^{(i+1)}$. By construction, $\mathsf{stabcol}_c^{(i+1)}(A_H)$ and $\mathsf{stabcol}_{c'}^{(i+1)}(A_H)$ do not have a common entry holding value 1, for each distinct pair of colours $c, c' \in C^{(i+1)}$. We note that the number of entries holding value 1 in all $\mathsf{stabcol}_c^{(i+1)}(A_H)$ combined sum up n^2. Indeed, this holds for $\mathsf{stabcol}_c^{(i+1)}(A_G)$ and we have just shown that $\mathsf{stabcol}_c^{(i+1)}(A_H)$ consists of the same number of ones as $\mathsf{stabcol}_c^{(i+1)}(A_G)$. Hence, $\mathsf{stabcol}_c^{(i+1)}(A_H)$ also represent a partition of $W \times W$, i.e., $\Pi_{\chi_H^{(i+1)}}(H)$, satisfying our induction hypothesis.

To conclude the proof we observe that all operations used in the sentences in $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},+,\times,\odot)$ in the inductive argument are linear operations. We can therefore write all sentences as linear combinations of sentences in $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)$. Hence, when $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)} H$ holds, then G and H will agree on all linear combination of sentences in $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)$. In other words, $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)} H$ implies that G and H are indistinguishable by edge colouring. ☐

We are now ready to show our main result.

Theorem 9.2 *Let G and H be two graphs of the same order, then $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)} H$ if and only if $G \equiv_{\mathsf{MATLANG}} H$ if and only if $G \equiv_{\mathsf{C}^3} H$.*

Proof We show that $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)} H$ implies $G \equiv_{\mathsf{C}^3} H$, and that $G \equiv_{\mathsf{C}^3} H$ implies $G \equiv_{\mathsf{MATLANG}} H$. Since $\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)$ is a smaller fragment than $\mathsf{MATLANG}$, $G \equiv_{\mathsf{MATLANG}} H$ clearly implies $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)} H$.

We assume first that $G \equiv_{\mathsf{ML}(\cdot,^*,\mathsf{tr},\mathbb{1},\mathsf{diag},\odot)} H$ holds. Then, the previous proposition implies that $G \equiv_{\mathsf{WL}} H$. Combined with Theorem 9.1, this implies that $G \equiv_{\mathsf{C}^3} H$. Next, we assume that $G \equiv_{\mathsf{C}^3} H$ holds. We show that this implies that $G \equiv_{\mathsf{MATLANG}} H$. In Proposition 4.2 in Brijder et al. [10] it was shown that for every sentence $e(X)$ in $\mathsf{MATLANG}$ there exists an equivalent formula $\varphi_e(z)$ in the relational calculus with aggregates which uses only three "base variables". We will not recall the syntax of this calculus formally (see [51] for a full definition) but only recall that in this calculus, we have base variables and numerical variables. Base variables can be bound to base columns of relations, and compared for equality. Numerical variables can be bound to numerical columns, and can be equated to function applications and aggregates. The free variable z in $\varphi_e(z)$ is a numeric variable since a scalar is returned by $e(X)$.

We now make the connection between matrices, on which MATLANG expressions are evaluated, and such typed relations, on which calculus expressions are evaluated. More specifically, a matrix A is encoded as a ternary relation $\mathsf{Rel}(A)$ where two base columns are reserved for the indices of the matrix and the numerical column holds the value in each entry (vectors and scalars are represented analogously). It is now understood that the equivalence of $e(X)$ and $\varphi_e(z)$ means that $e(A_G)$ and the evaluation of $\varphi_e(z)$ on $\mathsf{Rel}(A_G)$ results in the same scalar. Let $c = e(A_G) \in$ and consider the calculus sentence $\psi_e := \exists z\, \varphi_e(z) \wedge z = c$. Following the arguments in the proof of Proposition 4.4. in [10], which in turn rely on standard translation techniques (see e.g., [41,51]), one can show that ψ_e can be equivalently expressed by a sentence ψ_e' in $C^3_{\infty\omega}$ [56], i.e., in infinitary counting logic with three distinct (untyped) variables over binary relations. These binary relations encode graphs in a standard way by simply storing the edge relation. It is known that $G \equiv_{C^3_{\infty\omega}} H$ if and only if $G \equiv_{\mathsf{C}^3} H$ [40]. By assumption $G \equiv_{\mathsf{C}^3} H$ and hence $G \equiv_{C^3_{\infty\omega}} H$. This implies that $\psi_e'(G) = \psi_e'(H)$ since ψ_e' is a sentence in $C^3_{\infty\omega}$. Hence, also ψ_e evaluates to true on both $\mathsf{Rel}(A_G)$ and $\mathsf{Rel}(A_H)$, and $\varphi_e(z)$ returns the value c on both $\mathsf{Rel}(A_G)$ and $\mathsf{Rel}(A_H)$. As a consequence, also $e(A_H) = c$ and $e(A_G) = e(A_H)$. Since this argument works for any MATLANG sentence $e(X)$, we have that $G \equiv_{\mathsf{MATLANG}} H$. □

We conclude by providing an algebraic characterisation of MATLANG-equivalence based on an result by Dawar et al [23]. To state this result, we need the notion of coherent algebra (see e.g., [28]). The *coherent algebra* $\mathfrak{C}(A_G)$ associated with A_G is the smallest complex matrix algebra containing A_G, I, and J and which is closed under the Schur-Hadamard product. Similarly for A_H. The algebras $\mathfrak{C}(A_G)$ and $\mathfrak{C}(A_H)$ are said to be algebraically isomorphic if there is bijection $\iota : \mathfrak{C}(A_G) \to \mathfrak{C}(A_H)$ which is an algebra morphism which in addition satisfies: $\iota(J) = J$, $\iota(A^*) = (\iota(A))^*$ and $\iota(A \odot B) = \iota(A) \odot \iota(B)$, for all matrices $A, B \in \mathfrak{C}(A_G)$.

Proposition 9.2 (Proposition 7 in Dawar et al. [23]) *Let G and H be two graphs of the same order. Then, $G \equiv_{\mathsf{C}^3} H$ if and only if there exists an algebraic isomorphism $\iota : \mathfrak{C}(A_G) \to \mathfrak{C}(A_H)$ such that $\iota(A_G) = \iota(A_H)$.* □

This correspondence can be made a bit more precise and in line with our previous characterizations.

Proposition 9.3 *Let G and H be two graphs of the same order, then $G \equiv_{\mathsf{MATLANG}} H$ if and only if there exists an orthogonal matrix O such that $E_c \cdot O = O \cdot F_c$, for $c \in C$, where E_c and F_c, for $c \in C$, constitute the stable edge partitions $\Pi(G)$ and $\Pi(H)$, of G and H, respectively. (Here, C denotes the set of colours used by the colourings that induce the partitions).*

Proof We know from Proposition 9.1 that $G \equiv_{\mathsf{MATLANG}} H$ implies that $G \equiv_{\mathsf{WL}} H$. Moreover, we can compute $\Pi(G)$ and $\Pi(H)$ by means of the expressions $\mathsf{stabcol}_c(X)$ in MATLANG. Let $C = \{c_1, \dots, c_\ell\}$ be the set of colours used in these partitions. Just as in the proof of Theorem 7.2, we consider sentences $e_w(X) := \mathsf{tr}\big(w(\mathsf{stabcol}_{c_1}(X), \dots, \mathsf{stabcol}_{c_\ell}(X))\big)$ for some word w over ℓ variables. Then, $G \equiv_{\mathsf{MATLANG}} H$ implies that $e_w(A_G) = e_w(A_H)$ for any such word w, and thus by the real version of Specht's Theorem, there exists an orthogonal matrix O such that $\mathsf{stabcol}_c(A_G)\cdot O = O\cdot\mathsf{stabcol}_c(A_H)$ for all $c\in C$, as desired. In the application of Specht's Theorem it is crucial that $\Pi(G)$ and $\Pi(H)$ are closed under transposition. This known to hold (see e.g., [7]).

For the converse, suppose that there exists an orthogonal matrix O such that $E_c\cdot O = O\cdot F_c$, for $c\in C$. We note that this implies that $A_G\cdot O = O\cdot A_H$ since $A_G = \sum_{c\in D} E_c$ and $A_H = \sum_{c\in D} F_c$ for some subset of colours D of C. This follows the fact that the edge colouring algorithm refines the initial colouring, in which edges in are coloured differently than non-edges. So, a color used for an edge in G can only be used for an edge in H, and vice versa. Moreover, it is known that the binary matrices in $\Pi(G)$ and $\Pi(H)$ form a basis for $\mathfrak{C}(A_G)$ and $\mathfrak{C}(A_H)$, respectively. This basis is closed under the Shur-Hadamard product, among other things. If we now consider $\iota:\mathfrak{C}(A_G)\to\mathfrak{C}(A_H): A\mapsto O\cdot A\cdot O^{\mathrm{t}}$, then this is known to be an algebraic isomorphism between $\mathfrak{C}(A_G)$ and $\mathfrak{C}(A_H)$ [28]. Hence, by Proposition 9.2, $G\equiv_{\mathsf{C}^3} H$ and thus also $G\equiv_{\mathsf{MATLANG}} H$ by Theorem 9.2. □

Remark 9.1 The orthogonal matrix O in the statement of Proposition 9.3 can be taken to be compatible with the common equitable partitions of G and H, just as in Theorem 7.2. This follows from the fact that there is a subset K of colours such that $I = \sum_{c\in K} E_c = \sum_{c\in K} F_c$ [7]. Furthermore, the diagonal matrices E_c, for $c\in K$, correspond to $\mathsf{diag}(\mathbb{1}_{V_c})$ for the coarsest equitable partition $\mathcal{V} = \{V_c \mid c\in K\}$ of G. Similarly, for $c\in K$, $F_c = \mathsf{diag}(\mathbb{1}_{W_c})$, for the coarsest equitable partition $\mathcal{W} = \{W_c \mid c\in K\}$ of H [7].

Remark 9.2 The proof of Proposition 9.3 relied on results by Brijder et al [10] and Dawar et al [23] in which connections with C^3-equivalence were made. We can circumvent this by showing that O-similarity, for an orthogonal matrix O such that $E_c\cdot O = O\cdot F_c$ holds for each colour $c\in C$, is preserved by all operations in MATLANG, including arbitrary pointwise functions on matrices. We do not detail this further in this paper, in order to keep the paper of reasonably length (the proof consists of many case analyses in which all previous similarity preserving conditions need to be verified in the context of stable edge partitions). The crucial ingredient in all this is that one can verify that for any expression $e(X)$ in MATLANG, such that $e(A_G)$ returns a matrix, we can write $e(A_G) = \sum_{c\in C} a_c\times E_c$ and $e(A_H) = \sum_{c\in C} a_c\times F_c$. This is generalization $\mathsf{ML}(\mathcal{L})$-vectors being constant on equitable partitions, but now for $\mathsf{ML}(\mathcal{L})$-matrices being constant on stable edge partitions. The ability to rewrite $e(A_G)$ (and $e(A_H)$) in terms of the indicator matrices allows to show, e.g., that O-similarity is preserved by the Schur-Hadamard product and, more generally, by any pointwise function application on matrices.

10 Concluding remarks

We have characterised $\mathsf{ML}(\mathcal{L})$-equivalence for undirected graphs and clearly identified what additional distinguishing power each of the operations has. That natural characterisations can be obtained once more attests that MATLANG is an adequate matrix language. We conclude with some avenues for further investigation.

Although some of the results generalise to directed graphs (with asymmetric adjacency matrices), an extension to the case when queries can have multiple inputs seems do-able but challenging. The generalisation beyond graphs, i.e., for arbitrary matrices, is wide open.

Of interest may also be to connect $\mathsf{ML}(\mathcal{L})$-equivalence to fragments of first-order logic (without counting). A possible line of attack could be to work over the boolean semiring instead of over the complex numbers (see Grohe and Otto [36] for a similar approach). More general semirings could open the way for modelling and querying labeled graphs using matrix query languages.

We also note that $\mathsf{MATLANG}$ was extended in Brijder et al. [10] with an operator inv that computes the inverse of a matrix, if it exists, and returns the zero matrix otherwise. The extension, $\mathsf{MATLANG}+\mathsf{inv}$, was shown to be more expressive than $\mathsf{MATLANG}$. For example, connectedness of graphs can be checked by a single sentence in $\mathsf{MATLANG}+\mathsf{inv}$. Of course, we here consider *equivalence* of graphs. Even when considering a "classical" logic like FO^3, the three-variable fragment of first-order logic, $G \equiv_{\mathsf{FO}^3} H$ implies that G is connected if and only if H is connected. Translated to our setting, for any fragment $\mathsf{ML}(\mathcal{L})$ in which $G \equiv_{\mathsf{ML}(\mathcal{L})} H$ implies that the Laplacian $\mathsf{diag}(A_G \cdot \mathbb{1}) - A_G$ of G is co-spectral with the Laplacian of $\mathsf{diag}(A_H \cdot \mathbb{1}) - A_H$ of H, $G \equiv_{\mathsf{ML}(\mathcal{L})} H$ implies that G is connected if and only if H is connected. It even implies that G and H must have the same number of connected components, as this is determined by the multiplicity of the eigenvalue 0 of the Laplacian [12].

Nevertheless, we can also consider equivalence of graphs relative to $\mathsf{MATLANG}+\mathsf{inv}$. We observe, however, that the inverse of a matrix can be computed using $+$ and $\times$, by the Cayley-Hamilton Theorem [5], given the coefficients of the characteristic polynomial of the adjacency matrix. These coefficients can be computed using $+$, $\times$ and tr. For fragments supporting $\cdot$, $+$, $\times$ and tr, the operator inv thus does not add distinguishing power. It is unclear what the impact is of inv for smaller fragments such as $\mathsf{ML}(\cdot, {}^*, \mathbb{1})$ and $\mathsf{ML}(\cdot, {}^*, \mathbb{1}, \mathsf{diag})$.

To relate our notion of equivalence more closely to the expressiveness questions studied in Brijder et al. [10], it may be interesting to investigate notions of *locality* of $\mathsf{ML}(\mathcal{L})$ expressions, as this underlies the inexpressibility of connectivity of $\mathsf{MATLANG}$ [52]. It would be nice if this can be achieved in purely algebraic terms, without relying on locality notions in logic.

Finally, $\mathsf{MATLANG}$ was also extended with an eigen operator which returns a matrix whose columns consist of eigenvectors spanning the eigenspaces [10]. Since the choice of eigenvectors is not unique, this results in a non-deterministic semantics. We leave it for future work to study the equivalence of graphs relative to *deterministic* fragments supporting the eigen operator, i.e., such that the result of expressions does not depend on the eigenvectors returned. As a starting point one could, for example, force determinism by considering a certain answer semantics. That is, if $e(X)$ is an expression using $\mathsf{eigen}(X)$, one can define $\mathsf{cert}(e(A_G)) := \bigcap_V e(A_G, V)$, where V ranges over all bases of the eigenspaces. Distinguishability with regards to such a certain answer semantics demands further investigation.

Acknowledgements I am indebted to Stuart Anderson for providing cospectral and fractionally isomorphic graphs with non-cospectral Laplacians (graphs G_6 and H_6 in Example 8.1).

References

1. Noga Alon, Raphael Yuster, and Uri Zwick. Finding and counting given length cycles. *Algorithmica*, 17(3):209–223, 1997. https://doi.org/10.1007/BF02523189.
2. Renzo Angles, Marcelo Arenas, Pablo Barceló, Aidan Hogan, Juan Reutter, and Domagoj Vrgoč. Foundations of modern query languages for graph databases. *ACM Comput. Surv.*, 50(5):68:1–68:40, 2017. http://doi.acm.org/10.1145/3104031.
3. Vikraman Arvind, Frank Fuhlbrück, Johannes Köbler, and Oleg Verbitsky. On weisfeiler-leman invariance: Subgraph counts and related graph properties. *CoRR*, abs/1811.04801, 2018. http://arxiv.org/abs/1811.04801.
4. Albert Atserias and Elitza N. Maneva. Sherali-Adams relaxations and indistinguishability in counting logics. *SIAM J. Comput.*, 42(1):112–137, 2013. https://doi.org/10.1137/120867834.
5. Sheldon Axler. *Linear Algebra Done Right*. Springer, third edition, 2015. https://doi.org/10.1007/978-3-319-11080-6.
6. Pablo Barceló. Querying graph databases. In *Proceedings of the 32nd ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS, pages 175–188, 2013. http://doi.acm.org/10.1145/2463664.2465216.
7. Oliver Bastert. *Stabilization procedures and applications*. PhD thesis, Technical University Munich, Germany, 2001. http://nbn-resolving.de/urn:nbn:de:bvb:91-diss2002070500045.
8. Matthias Boehm, Michael W. Dusenberry, Deron Eriksson, Alexandre V. Evfimievski, Faraz Makari Manshadi, Niketan Pansare, Berthold Reinwald, Frederick R. Reiss, Prithviraj Sen, Arvind C. Surve, and Shirsih Tatikonda. SystemML: Declarative machine learning on Spark. *Proceedings of the VLDB Endowment*, 9(13):1425–1436, 2016. https://doi.org/10.14778/3007263.3007279.
9. Matthias Boehm, Berthold Reinwald, Dylan Hutchison, Prithviraj Sen, Alexandre V. Evfimievski, and Niketan Pansare. On optimizing operator fusion plans for large-scale machine learning in SystemML. *Proceedings of the VLDB Endowment*, 11(12):1755–1768, 2018. https://doi.org/10.14778/3229863.3229865.
10. Robert Brijder, Floris Geerts, Jan Van den Bussche, and Timmy Weerwag. On the expressive power of query languages for matrices. In *21st International Conference on Database Theory*, ICDT, pages 10:1–10:17, 2018. https://doi.org/10.4230/LIPIcs.ICDT.2018.10.
11. Robert Brijder, Floris Geerts, Jan Van den Bussche, and Timmy Weerwag. On the expressive power of query languages for matrices. *ACM Trans. on Database Systems*, 2019. To appear.
12. Andries E. Brouwer and Willem H. Haemers. *Spectra of Graphs*. Universitext. Springer, 2012. https://doi.org/10.1007/978-1-4614-1939-6.
13. Jin-Yi Cai, Martin Fürer, and Neil Immerman. An optimal lower bound on the number of variables for graph identification. *Combinatorica*, 12(4):389–410, 1992. https://doi.org/10.1007/BF01305232.
14. Ada Chan and Chris D. Godsil. Symmetry and eigenvectors. In *Graph symmetry (Montreal, PQ, 1996)*, volume 497 of *NATO Adv. Sci. Inst. Ser. C Math. Phys. Sci.*, pages 75–106. Kluwer Acad. Publ., Dordrecht, 1997. https://doi.org/10.1007/978-94-015-8937-6_3.
15. Lingjiao Chen, Arun Kumar, Jeffrey Naughton, and Jignesh M. Patel. Towards linear algebra over normalized data. *Proceedings of the VLDB Endowment*, 10(11):1214–1225, 2017. https://doi.org/10.14778/3137628.3137633.
16. Dragoš M. Cvetković. Graphs and their spectra. *Publikacije Elektrotehničkog fakulteta. Serija Matematika i fizika*, 354/356:1–50, 1971. http://www.jstor.org/stable/43667526.
17. Dragoš M. Cvetković. The main part of the spectrum, divisors and switching of graphs. *Publ. Inst. Math. (Beograd) (N.S.)*, 23(37):31–38, 1978. http://elib.mi.sanu.ac.rs/files/journals/publ/43/6.pdf.
18. Dragoš M. Cvetković, Peter Rowlinson, and Slobodan Simić. *Eigenspaces of Graphs*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1997. https://doi.org/10.1017/CBO9781139086547.
19. Dragoš M. Cvetković, Peter Rowlinson, and Slobodan Simić. *An Introduction to the Theory of Graph Spectra*. London Mathematical Society Student Texts. Cambridge University Press, 2009. https://doi.org/10.1017/CBO9780511801518.

20. Anuj Dawar. On the descriptive complexity of linear algebra. In *Proceedings of the 15th International Workshop on Logic, Language, Information and Computation*, WoLLIC, pages 17–25, 2008. http://dx.doi.org/10.1007/978-3-540-69937-8_2.
21. Anuj Dawar, Martin Grohe, Bjarki Holm, and Bastian Laubner. Logics with rank operators. In *Proceedings of the 24th Annual IEEE Symposium on Logic In Computer Science*, LICS, pages 113–122, 2009. https://doi.org/10.1109/LICS.2009.24.
22. Anuj Dawar and Bjarki Holm. Pebble games with algebraic rules. *Fund. Inform.*, 150(3-4):281–316, 2017. https://doi.org/10.3233/FI-2017-1471.
23. Anuj Dawar, Simone Severini, and Octavio Zapata. Descriptive complexity of graph spectra. In *Proceedings of the 23rd International Workshop on Logic, Language, Information and Computation*, WoLLIC, pages 183–199, 2016. https://doi.org/10.1007/978-3-662-52921-8_12.
24. Anuj Dawar, Simone Severini, and Octavio Zapata. Descriptive complexity of graph spectra. *Annals of Pure and Applied Logic*, 170(9):993 – 1007, 2019. https://doi.org/10.1016/j.apal.2019.04.005.
25. Holger Dell, Martin Grohe, and Gaurav Rattan. Lovász meets Weisfeiler and Lehman. In *45th International Colloquium on Automata, Languages, and Programming*, ICALP, pages 40:1–40:14, 2018. https://doi.org/10.4230/LIPIcs.ICALP.2018.40.
26. Ahmed Elgohary, Matthias Boehm, Peter J. Haas, Frederick R. Reiss, and Berthold Reinwald. Compressed linear algebra for large-scale machine learning. *The VLDB Journal*, pages 1–26, 2017. https://doi.org/10.1007/s00778-017-0478-1".
27. H. K. Farahat. The semigroup of doubly-stochastic matrices. *Proceedings of the Glasgow Mathematical Association*, 7(4):178183, 1966. https://doi.org/10.1017/S2040618500035401.
28. Shmuel Friedland. Coherent algebras and the graph isomorphism problem. *Discrete Applied Mathematics*, 25(1):73–98, 1989. https://doi.org/10.1016/0166-218X(89)90047-4.
29. Martin Fürer. On the power of combinatorial and spectral invariants. *Linear Algebra and its Applications*, 432(9):2373–2380, 2010. https://doi.org/10.1016/j.laa.2009.07.019.
30. Martin Fürer. On the combinatorial power of the Weisfeiler-Lehman algorithm. In Dimitris Fotakis, Aris Pagourtzis, and Vangelis Th. Paschos, editors, *Algorithms and Complexity*, pages 260–271. Springer, 2017. https://doi.org/10.1007/978-3-319-57586-5_22.
31. Floris Geerts. On the expressive power of linear algebra on graphs. In *22nd International Conference on Database Theory*, ICDT, pages 7:1–7:19, 2019. https://doi.org/10.4230/LIPIcs.ICDT.2019.7.
32. Chris Godsil and Gordon F. Royle. *Algebraic Graph Theory*, volume 207 of *Graduate Texts in Mathematics.* Springer, 2001. https://doi.org/10.1007/978-1-4613-0163-9.
33. Alexey L. Gorodentsev. *Algebra I: Textbook for Students of Mathematics*. Springer, 2016. https://doi.org/10.1007/978-3-319-45285-2.
34. Erich Grädel and Wied Pakusa. Rank logic is dead, long live rank logic! In *24th EACSL Annual Conference on Computer Science Logic*, CSL, pages 390–404, 2015. https://doi.org/10.4230/LIPIcs.CSL.2015.390.
35. Martin Grohe, Kristian Kersting, Martin Mladenov, and Erkal Selman. Dimension reduction via colour refinement. In *22th Annual European Symposium on Algorithms*, ESA, pages 505–516, 2014. https://doi.org/10.1007/978-3-662-44777-2_42.
36. Martin Grohe and Martin Otto. Pebble games and linear equations. *The Journal of Symbolic Logic*, 80(3):797–844, 2015. http://doi.org/10.1017/jsl.2015.28.
37. Martin Grohe and Wied Pakusa. Descriptive complexity of linear equation systems and applications to propositional proof complexity. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS, pages 1–12, 2017. https://doi.org/10.1109/LICS.2017.8005081.
38. Willem H. Haemers and Edward Spence. Enumeration of cospectral graphs. *European J. Combin.*, 25(2):199–211, 2004. https://doi.org/10.1016/S0195-6698(03)00100-8.
39. Frank Harary and Allen J. Schwenk. The spectral approach to determining the number of walks in a graph. *Pacific J. Math.*, 80(2):443–449, 1979. https://projecteuclid.org:443/euclid.pjm/1102785717.
40. Lauri Hella. Logical hierarchies in PTIME. *Information and Computation*, 129(1):1–19, 1996. https://doi.org/10.1006/inco.1996.0070.
41. Lauri Hella, Leonid Libkin, Juha Nurmonen, and Limsoon Wong. Logics with aggregate operators. *Journal of the ACM*, 48(4):880–907, 2001. https://doi.org/10.1145/502090.502100.
42. Bjarki Holm. *Descriptive Complexity of Linear Algebra*. PhD thesis, University of Cambridge, 2010.

43. Dylan Hutchison, Bill Howe, and Dan Suciu. LaraDB: A minimalist kernel for linear and relational algebra computation. In *Proceedings of the 4th ACM SIGMOD Workshop on Algorithms and Systems for MapReduce and Beyond*, BeyondMR, pages 2:1–2:10, 2017. http://doi.acm.org/10.1145/3070607.3070608.
44. Neil Immerman and Eric Lander. Describing graphs: A first-order approach to graph canonization. In Alan L. Selman, editor, *Complexity Theory Retrospective: In Honor of Juris Hartmanis on the Occasion of His Sixtieth Birthday*, pages 59–81. Springer, 1990. https://doi.org/10.1007/978-1-4612-4478-3_5.
45. Naihuan Jing. Unitary and orthogonal equivalence of sets of matrices. *Linear Algebra and its Applications*, 481:235–242, 2015. https://doi.org/10.1016/j.laa.2015.04.036.
46. Charles R Johnson and Morris Newman. A note on cospectral graphs. *Journal of Combinatorial Theory, Series B*, 28(1):96 – 103, 1980. https://doi.org/10.1016/0095-8956(80)90058-1.
47. Irving Kaplansky. *Linear Algebra and Geometry. A Second Course*. Chelsea Publishing Company, 1974.
48. Kristian Kersting, Martin Mladenov, Roman Garnett, and Martin Grohe. Power iterated color refinement. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, pages 1904–1910. AAAI Press, 2014. https://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8377.
49. Andreas Kunft, Alexander Alexandrov, Asterios Katsifodimos, and Volker Markl. Bridging the gap: Towards optimization across linear and relational algebra. In *Proceedings of the 3rd ACM SIGMOD Workshop on Algorithms and Systems for MapReduce and Beyond*, BeyondMR, pages 1:1–1:4, 2016. http://doi.acm.org/10.1145/2926534.2926540.
50. Andreas Kunft, Asterios Katsifodimos, Sebastian Schelter, Tilmann Rabl, and Volker Markl. Blockjoin: Efficient matrix partitioning through joins. *Proceedings of the VLDB Endowment*, 10(13):2061–2072, 2017. https://doi.org/10.14778/3151106.3151110.
51. Leonid Libkin. Expressive power of SQL. *Theoretical Computer Science*, 296:379–404, 2003. https://doi.org/10.1016/S0304-3975(02)00736-3.
52. Leonid Libkin. *Elements of Finite Model Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2004. https://doi.org/10.1007/978-3-662-07003-1.
53. Shangyu Luo, Zekai J. Gao, Michael Gubanov, Luis L. Perez, and Christopher Jermaine. Scalable linear algebra on a relational database system. *SIGMOD Rec.*, 47(1):24–31, 2018. http://doi.acm.org/10.1145/3277006.3277013.
54. Peter N. Malkin. Sherali–Adams relaxations of graph isomorphism polytopes. *Discrete Optimization*, 12:73 – 97, 2014. https://doi.org/10.1016/j.disopt.2014.01.004.
55. Hung Q. Ngo, XuanLong Nguyen, Dan Olteanu, and Maximilian Schleich. In-database factorized learning. In *Proceedings of the 11th Alberto Mendelzon International Workshop on Foundations of Data Management and the Web*, AMW, 2017. http://ceur-ws.org/Vol-1912/paper21.pdf.
56. Martin Otto. *Bounded Variable Logics and Counting: A Study in Finite Models*, volume 9 of *Lecture Notes in Logic*. Springer, 1997. https://doi.org/10.1017/9781316716878.
57. Martin Otto. *Bounded Variable Logics and Counting: A Study in Finite Models*, volume 9 of *Lecture Notes in Logic*. Cambridge University Press, 2017. https://doi.org/10.1017/9781316716878.
58. Christian Pech. Coherent algebras. https://doi.org/10.13140/2.1.2856.2248, 2002.
59. Sam Perlis. *Theory of matrices*. Addison-Wesley Press, Inc., Cambridge, Mass., 1952.
60. Motakuri V. Ramana, Edward R. Scheinerman, and Daniel Ullman. Fractional isomorphism of graphs. *Discrete Mathematics*, 132(1-3):247–265, 1994. https://doi.org/10.1016/0012-365X(94)90241-0.
61. Peter Rowlinson. The main eigenvalues of a graph: A survey. *Applicable Analysis and Discrete Mathematics*, 1(2):455–471, 2007. http://www.jstor.org/stable/43666075.
62. Edward R. Scheinerman and Daniel H. Ullman. *Fractional Graph Theory: a Rational Approach to the Theory of Graphs*. John Wiley & Sons, 1997. https://www.ams.jhu.edu/ers/wp-content/uploads/sites/2/2015/12/fgt.pdf.
63. Maximilian Schleich, Dan Olteanu, and Radu Ciucanu. Learning linear regression models over factorized joins. In *Proceedings of the 2016 International Conference on Management of Data*, SIGMOD, pages 3–18, 2016. http://doi.acm.org/10.1145/2882903.2882939.
64. Mario Thüne. *Eigenvalues of matrices and graphs*. PhD thesis, University of Leipzig, 2012.
65. Gottfried Tinhofer. Graph isomorphism and theorems of Birkhoff type. *Computing*, 36(4):285–300, 1986. https://doi.org/10.1007/BF02240204.
66. Gottfried Tinhofer. A note on compact graphs. *Discrete Applied Mathematics*, 30(2):253–264, 1991. https://doi.org/10.1016/0166-218X(91)90049-3.

67. Edwin R. van Dam and Willem H. Haemers. Which graphs are determined by their spectrum? *Linear Algebra and its Applications*, 373:241 – 272, 2003. https://doi.org/10.1016/S0024-3795(03)00483-X.
68. Erwin R. van Dam, Willem H. Haemers, and Jack H. Koolen. Cospectral graphs and the generalized adjacency matrix. *Linear Algebra and its Applications*, 423(1):33–41, 2007. https://doi.org/10.1016/j.laa.2006.07.017.
69. Boris J. Weisfeiler and Andrei A. Lehman. A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsiya*, 2(9):12–16, 1968. https://www.iti.zcu.cz/wl2018/pdf/wl_paper_translation.pdf.

Proof of Lemma 5.1

Lemma 5.1 *Let A_G and A_H be two adjacency matrices of the same dimensions which are T-similar for an arbitrary matrix T. Let $e_1(X)$ and $e_2(X)$ be two expressions in $\mathsf{ML}(\mathcal{L})$ for any $\mathcal{L}$. If $e_i(A_G)$ and $e_i(A_H)$ are T-similar, for $i=1,2$, then $e_1(A_G)\cdot e_2(A_G)$ is also T-similar to $e_1(A_H)\cdot e_2(A_H)$ (provided, of course, that the multiplication is well-defined).*

Proof To show this lemma, we distinguish between the following cases, depending on the dimensions of $e_1(A_G)$ and $e_2(A_G)$ (or equivalently, the dimensions of $e_1(A_H)$ and $e_2(A_H)$). Let $e(X) := e_1(X)\cdot e_2(X)$. Let n be the order of G (and H).

- **(n×n, n×n)**: $e_1(A_G)$ and $e_2(A_G)$ are of dimension $n\times n$. By assumption, $e_1(A_G)\cdot T = T\cdot e_1(A_H)$ and $e_2(A_G)\cdot T = T\cdot e_2(A_H)$. Hence,
$$e(A_G)\cdot T = e_1(A_G)\cdot e_2(A_G)\cdot T = e_1(A_G)\cdot T\cdot e_2(A_H) = T\cdot e_1(A_H)\cdot e_2(A_H) = T\cdot e(A_H).$$
- **(n×n, n×1)**: $e_1(A_G)$ is of dimension $n\times n$ and $e_2(A_G)$ is of dimension $n\times 1$. By assumption, $e_1(A_G)\cdot T = T\cdot e_1(A_H)$ and $e_2(A_G) = T\cdot e_2(A_H)$. Hence,
$$e(A_G) = e_1(A_G)\cdot e_2(A_G) = e_1(A_G)\cdot T\cdot e_2(A_H) = T\cdot e_1(A_H)\cdot e_2(A_H) = T\cdot e(A_H).$$
- **(n×1, 1×n)**: $e_1(A_G)$ is of dimension $n\times 1$ and $e_2(A_G)$ is of dimension $1\times n$. By assumption, $e_1(A_G) = T\cdot e_1(A_H)$ and $e_2(A_G)\cdot T = e_2(A_H)$. Hence,
$$e(A_G)\cdot T = e_1(A_G)\cdot e_2(A_G)\cdot T = e_1(A_G)\cdot e_2(A_H) = T\cdot e_1(A_H)\cdot e_2(A_H) = T\cdot e(A_H)).$$
- **(n×1, 1×1)**: $e_1(A_G)$ is of dimension $n\times 1$ and $e_2(A_G)$ is of dimension 1×1. By assumption, $e_1(A_G) = T\cdot e_1(A_H)$ and $e_2(A_G) = e_2(A_H)$. Hence,
$$e(A_G) = e_1(A_G)\cdot e_2(A_G) = e_1(A_G)\cdot e_2(A_H) = T\cdot e_1(A_H)\cdot e_2(A_H) = T\cdot e(A_H)).$$
- **(1×n, n×n)**: $e_1(A_G)$ is of dimension $1\times n$ and $e_2(A_G)$ is of dimension $n\times n$. By assumption, $e_1(A_G)\cdot T = e_1(A_H)$ and $e_2(A_G)\cdot T = T\cdot e_2(A_H)$. Hence,
$$e(A_G)\cdot T = e_1(A_G)\cdot e_2(A_G)\cdot T = e_1(A_H)\cdot T\cdot e_2(A_H) = e_1(A_H)\cdot e_2(A_H) = e(A_H)).$$
- **(1×n, n×1)**: $e_1(A_G)$ is of dimension $1\times n$ and $e_2(A_G)$ is of dimension $n\times 1$. By assumption, $e_1(A_G)\cdot T = e_1(A_H)$ and $e_2(A_G) = T\cdot e_2(A_H)$. Hence,
$$e(A_G) = e_1(A_G)\cdot e_2(A_G) = e_1(A_G)\cdot T\cdot e_2(A_H) = e_1(A_H)\cdot e_2(A_H) = e(A_H).$$
- **(1×1, 1×n)**: $e_1(A_G)$ is of dimension 1×1 and $e_2(A_G)$ is of dimension $1\times n$. By assumption, $e_1(A_G) = e_1(A_H)$ and $e_2(A_G)\cdot T = e_2(A_H)$. Hence,
$$e(A_G)\cdot T = e_1(A_G)\cdot e_2(A_G)\cdot T = e_1(A_G)\cdot e_2(A_H) = e_1(A_G)\cdot e_2(A_H) = e(A_H).$$
- **(1×1, 1×1)**: $e_1(A)$ and $e_2(A)$ are of dimension 1×1. By assumption, $e_1(A_G) = e_1(A_H)$ and $e_2(A_G) = e_2(A_H)$. Hence, $e(A_G) = e_1(A_G)\cdot e_2(A_G) = e_1(A_H)\cdot e_2(A_H) = e(A_H)$.

This concludes the proof. □

Proof of Proposition 7.4

Proposition 7.4 $\mathsf{ML}(\cdot, {}^*, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, +, \times, \mathsf{apply}_s[f], f\in\Omega)$*-vectors are constant on equitable partitions.*

Proof Let $\mathcal{L}^{\#}$ denote $\{\cdot, {}^*, \mathsf{tr}, \mathbb{1}, \mathsf{diag}, +, \times, \mathsf{apply_s}[f], f \in \Omega\}$. Consider a graph G of order n with equitable partition $\mathcal{V} = \{V_1, \ldots, V_\ell\}$. As before, let $\mathbb{1}_{V_1}, \ldots, \mathbb{1}_{V_\ell}$ be the corresponding indicator vectors. We will show that for any expression $e(X) \in \mathsf{ML}(\mathcal{L}^{\#})$ such that $e(A_G)$ is an $n\times 1$-vector, $e(A_G)$ can be uniquely written in the form $\sum_{i=1}^{\ell} a_i \times \mathbb{1}_{V_i}$ for scalars $a_i \in$.

We show, by induction on the structure of expressions in $\mathsf{ML}(\mathcal{L}^{\#})$, that the following properties hold;

(a) if $e(A_G)$ returns an $n\times n$-matrix, then for any pair $i, j = 1, \ldots, \ell$ there exists a scalars $a_{ij}, b_{ij} \in$ such that

$$\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e(A_G) \cdot \mathbb{1}_{V_j} = a_{ij} \times \mathbb{1}_{V_i} \text{ and } \mathbb{1}_{V_j}^{\mathsf{t}} \cdot e(A_G) \cdot \mathsf{diag}(\mathbb{1}_{V_i}) = b_{ij} \times \mathbb{1}_{V_i}^{\mathsf{t}}$$

(b) if $e(A_G)$ returns an $n\times 1$-vector, then for any $i = 1, \ldots, \ell$, there exists a scalar $a_i \in C$ such that

$$\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e(A_G) = a_i \times \mathbb{1}_{V_i}.$$

Clearly, if (b) holds for every $i = 1, \ldots, \ell$, then, $f_{e,G} : V \to$ is indeed constant on each part in $\mathcal{V}$. We remark these properties can be seen as generalization of the known fact that the vector space spanned by indicator vectors of an equitable partition of G is invariant under multiplication by A_G (See e.g., Lemma 5.2 in [14]). That is, for any linear combination $v = \sum_{i=1}^{\ell} a_i \times \mathbb{1}_{V_i}$ we have that $A \cdot v = \sum_{i=1}^{\ell} b_i \times \mathbb{1}_{V_i}$. In our setting, (a) and (b) imply that $e(A_G) \cdot v$ is again a linear combination of indicator vectors, when $e(A_G)$ returns an $n\times n$-matrix. We next verify properties (a) and (b). We often use that $I = \sum_{i=1}^{\ell} \mathsf{diag}(\mathbb{1}_{V_i})$ and $\mathbb{1} = \sum_{i=1}^{\ell} \mathbb{1}_{V_i}$.

(**base case**) Let $e(X) := X$. The required property is simply a restatement of the being equitable. That is,

$$\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e(A_G) \cdot \mathbb{1}_{V_j} = \mathsf{deg}(v, V_j) \times \mathbb{1}_{V_i},$$

for an arbitrary vertex $v \in V_i$. So, we can take $a_{ij} = \mathsf{deg}(v, V_j)$. Similarly, because we A_G is a symmetric matrix,

$$\mathbb{1}_{V_j}^{\mathsf{t}} \cdot e(A_G) \cdot \mathsf{diag}(\mathbb{1}_{V_i}) = (\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e(A_G) \cdot \mathbb{1}_{V_j})^{\mathsf{t}} = \mathsf{deg}(v, V_j) \times \mathbb{1}_{V_i}^{\mathsf{t}},$$

for an arbitrary vertex $v \in V_i$. So, we can take $a_{ij} = \mathsf{deg}(v, V_j)$.

For condition (a) we only verify that $\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e(A_G) \cdot \mathbb{1}_{V_j} = a_{ij} \times \mathbb{1}_{V_i}$ holds. The verification of $\mathbb{1}_{V_j}^{\mathsf{t}} \cdot e(A_G) \cdot \mathsf{diag}(\mathbb{1}_{V_i}) = b_{ij} \times \mathbb{1}_{V_i}^{\mathsf{t}}$ is entirely similar.

(**multiplication**) Let $e(X) := e_1(X) \cdot e_2(X)$. We distinguish between a number of cases, depending on the dimensions of $e_1(A_G)$ and $e_2(A_G)$. We first check the cases when $e(A_G)$ returns an $n\times n$-matrix and need to show that property (a) holds.

– (**n×n, n×n**): $e_1(A_G)$ and $e_2(A_G)$ are of dimension $n\times n$. By induction, $\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e_1(A_G) \cdot \mathbb{1}_{V_j} = a_{ij} \times \mathbb{1}_{V_j}$ and
$\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e_2(A_G) \cdot \mathbb{1}_{V_j} = b_{ij} \times \mathbb{1}_{V_i}$.
Then, $\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e(A_G) \cdot \mathbb{1}_{V_j}$ is equal to

$$\begin{aligned}
\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e_1(A_G) \cdot e_2(A_G) \cdot \mathbb{1}_{V_j} &= \sum_{k=1}^{\ell} \mathsf{diag}(\mathbb{1}_{V_i}) \cdot e_1(A_G) \cdot \mathsf{diag}(\mathbb{1}_{V_k}) \cdot e_2(A_G) \cdot \mathbb{1}_{V_j} \\
&= \sum_{k=1}^{\ell} b_{kj} \times \big(\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e_1(A_G) \cdot \mathbb{1}_{V_k}\big) = \Big(\sum_{k=1}^{\ell} b_{kj} \times a_{ik}\Big) \times \mathbb{1}_{V_i},
\end{aligned}$$

as desired.

– (**n×1, 1×n**): $e_1(A_G)$ is of dimension $n\times 1$ and $e_2(A_G)$ is of dimension $1\times n$. By induction we have that $\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e_1(A_G) = a_i \times \mathbb{1}_{V_i}$ and $\mathsf{diag}(\mathbb{1}_{V_i}) \cdot (e_2(A_G))^{\mathsf{t}} = b_i \times \mathbb{1}_{V_i}$. Hence, $\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e(A_G) \cdot \mathbb{1}_{V_j}$ is equal to

$$\begin{aligned}
\mathsf{diag}(\mathbb{1}_{V_i}) \cdot e_1(A_G) \cdot e_2(A_G) \cdot \mathbb{1}_{V_j} &= a_i \times (\mathbb{1}_{V_i} \cdot e_2(A_G) \cdot \mathbb{1}_{V_j}) \\
&= \sum_{k=1}^{\ell} a_i \times (\mathbb{1}_{V_i} \cdot e_2(A_G) \cdot \mathsf{diag}(\mathbb{1}_{V_k}) \cdot \mathbb{1}_{V_j}) \\
&= \sum_{k=1}^{\ell} a_i \times (\mathbb{1}_{V_i} \cdot \big(\mathsf{diag}(\mathbb{1}_{V_k}) \cdot (e_2(A_G))^{\mathsf{t}}\big)^{\mathsf{t}} \cdot \mathbb{1}_{V_j}) \\
&= \sum_{k=1}^{\ell} (a_i \times b_k) \times (\mathbb{1}_{V_i} \cdot \mathbb{1}_{V_k}^{\mathsf{t}} \cdot \mathbb{1}_{V_j})
\end{aligned}$$

$$=(a_i \times b_j \times |V_i|)\times \mathbb{1}_{V_i}$$

as desired.

Here we used that $\mathbb{1}_{V_k}^{\mathsf{t}} \cdot \mathbb{1}_{V_j}$ is either 0, in case that $k \neq j$, or $|V_j|$ in case that $j = k$.

We next check that condition (b) holds when $e(A_G)$ returns an $n \times 1$-vector.

– $(\mathbf{n}{\times}\mathbf{n}, \mathbf{n}{\times}\mathbf{1})$: $e_1(A_G)$ is of dimension $n \times n$ and $e_2(A_G)$ is of dimension $n \times 1$. By induction, we have that $\mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G)\cdot \mathbb{1}_{V_j} = a_{ij} \times \mathbb{1}_{V_i}$ and $\mathsf{diag}(\mathbb{1}_{V_i})\cdot e_2(A_G) = b_i \times \mathbb{1}_{V_i}$. Hence, $\mathsf{diag}(\mathbb{1}_{V_i})\cdot e(A_G)$ is equal to

$$\begin{aligned}\mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G)\cdot e_2(A_G) &= \sum_{j=1}^{\ell} \mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G)\cdot \mathsf{diag}(\mathbb{1}_{V_j})\cdot e_2(A_G)\\ &= \sum_{j=1}^{\ell} b_j \times (\mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G)\cdot \mathbb{1}_{V_j}) = \sum_{j=1}^{\ell} (a_{ij}\times b_j)\times \mathbb{1}_{V_i},\end{aligned}$$

as desired.

– $(\mathbf{n}{\times}\mathbf{1}, \mathbf{1}{\times}\mathbf{1})$: $e_1(A_G)$ is of dimension $n \times 1$ and $e_2(A_G)$ is of dimension 1×1. By induction we have that $\mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G) = a_i \times \mathbb{1}_{V_i}$ and $e_2(A_G) = b \in$. Hence,

$$\mathsf{diag}(\mathbb{1}_{V_i})\cdot e(A_G) = \mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G)\cdot e_2(A_G) = (a_i \times b)\times \mathbb{1}_{V_i},$$

as desired.

(**ones vector**) $e(X) := \mathbb{1}(e_1(X))$. We only need to consider the case when $e_1(A_G)$ is an $n \times n$-matrix or $n \times 1$-vector. In both cases, it suffices to observe that $\mathbb{1} = \sum_{i=1}^{\ell} \mathbb{1}_{V_i i}$. Indeed,

$$\mathsf{diag}(\mathbb{1}_{V_i})\cdot e(A_G) = \mathsf{diag}(\mathbb{1}_{V_i})\cdot \mathbb{1} = \mathbb{1}_{V_i}.$$

(**conjugate transpose**) $e(X) := (e_1(X))^*$. If $e_1(A_G)$ returns a $1 \times n$-vector, then $\mathsf{diag}(\mathbb{1}_{V_i})\cdot (e_1(A_G))^{\mathsf{t}} = a_i \times \mathbb{1}_{V_i}$. Hence, $\mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G) = a_i^* \times \mathbb{1}_{V_i}$. If $e_1(A_G)$ returns an $n \times n$-matrix, then by induction, $\mathbb{1}_{V_j}^{\mathsf{t}} \cdot e_1(A_G)\cdot \mathsf{diag}(\mathbb{1}_{V_i}) = b_{ij} \times \mathbb{1}_{V_i}^{\mathsf{t}}$. Hence,

$$\mathsf{diag}(\mathbb{1}_{V_i})\cdot e(A_G)\cdot \mathbb{1}_{V_j} = (\mathbb{1}_{V_j}^{\mathsf{t}} \cdot e_1(A_G)\cdot \mathsf{diag}(\mathbb{1}_{V_i}))^* = b_{ij}^* \times \mathbb{1}_{V_i},$$

as desired.

(**diag operation**) $e(X) := \mathsf{diag}(e_1(X))$ where $e_1(A_G)$ is an $n \times 1$-vector. By induction, $\mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G) = a_i \times \mathbb{1}_{V_i}$. Hence, in view of the linearity of the diagonal operation,

$$\mathsf{diag}(\mathbb{1}_{V_i})\cdot e(A_G)\cdot \mathbb{1}_{V_j} = \sum_{k=1}^{\ell} a_i \times \big(\mathsf{diag}(\mathbb{1}_{V_i})\cdot \mathsf{diag}(\mathbb{1}_{V_k})\cdot \mathbb{1}_{V_j}\big) = a_i \times \mathbb{1}_{V_i},$$

since $\mathsf{diag}(\mathbb{1}_{V_k})\cdot \mathbb{1}_{V_j}$ is $\mathbb{1}_{V_j}$ when $k = j$ and the zero vector otherwise.

(**addition**) $e(X) := e_1(X) + e_2(X)$. Clearly, when condition (a) or (b) hold for $e_1(A_G)$ and $e_2(A_G)$, they remain to hold for $e(A_G)$.

(**scalar multiplication**) $e(X) := a \times e_1(X)$. Clearly, when condition (a) or (b) hold for $e_1(A_G)$, they remain to hold for $e(A_G)$.

(**trace**) $e(X) := \mathsf{tr}(e_1(X))$. Such sub-expressions do not return matrices or vectors.

(**pointwise function applications**) $e(X) := \mathsf{apply}_{\mathsf{s}}[f](e_1(X), \ldots, e_p(X))$ where each $e_i(X)$ is a sentence. Again, such sub-expressions do not return matrices or vectors. □

Continuation of the proof of Theorem 7.2

In the main body of the paper we showed that, by using sentences in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \mathsf{diag})$, one can express trace identities which imply the existence of an orthogonal doubly quasi-stochastic matrix O such that $A_G \cdot O = O \cdot A_H$, and in addition, such that O is compatible with the common coarsest equitable partitions of G and H. Moreover, we sketched an argument indicating that the use of $\mathbb{1}^{\mathsf{t}}(X)$ can be eliminated, and as a consequence, $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathsf{diag})$-equivalence suffices to guarantee the existence of the desired orthogonal matrix. We now detail the elimination procedure. More precisely, we show

by induction on the structure of expressions $e(X)$ in $\mathsf{ML}(\cdot, \mathsf{tr}, \mathbb{1}, \mathbb{1}^{\mathsf{t}}, \mathsf{diag})$, that

– If $e(A_G)$ is an $n \times n$-matrix, then $e(X) \equiv c \times f(X)\cdot \mathbb{1}(X)\cdot e_{\mathsf{tr}}(X)\cdot \mathbb{1}^{\mathsf{t}}(X)\cdot g(X)$;

- If $e(A_G)$ is an $n\times 1$-matrix, then $e(X)\equiv c\times f(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}(X)$;
- If $e(A_G)$ is a $1\times n$-matrix, then $e(X)\equiv c\times e_{\mathsf{tr}}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g(X)$; and
- If $e(A_G)$ is a 1×1-matrix, then $e(X)\equiv c\times e_{\mathsf{tr}}(X)$,

where $c\in$, $f(X)$ and $g(X)$ are expressions in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$ and $e_{\mathsf{tr}}(X)$ is an expression of the form

$$\prod_{i\in K}\mathsf{tr}(h_i(X)),$$

with $h_i(X)$ expressions in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$, for each $i\in K$. In the first case, $\mathbb{1}(X)\cdot e_{\mathsf{tr}}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)$ is optional. This happens when $e(X)$ does not contain the $\mathbb{1}^{\mathsf{t}}(\cdot)$ operation. Furthermore, also $f(X)$, $g(X)$ and the expressions $e_{\mathsf{tr}}(X)$ may be optional. Nevertheless, we can always assume them to be $f(X)=\mathsf{diag}(\mathbb{1}(X))$, $g(X)=\mathsf{diag}(\mathbb{1}(X))$ and $e_{\mathsf{tr}}(X)=\mathbb{1}(\mathsf{tr}(\mathsf{diag}(\mathbb{1}(X))))$. Indeed, these evaluate to the identity matrix and [1], respectively, and hence do not have an effect on the evaluation. In the following we therefore always assume $f(X)$, $g(X)$ and $e_{\mathsf{tr}}(X)$ to be present. Similarly, we assume $\mathbb{1}(X)\cdot e_{\mathsf{tr}}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)$ to be present when $e(A_G)$ returns a matrix, except for the base case. It can easily be shown that the case analysis below carries through when $\mathbb{1}(X)\cdot e_{\mathsf{tr}}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)$ may be absent. As already mentioned in the main body of the paper, the key insight is that we can replace any sub-expression $\mathbb{1}^{\mathsf{t}}(X)\cdot e'(X)\cdot\mathbb{1}(X)$ by $\mathsf{tr}(\mathsf{diag}(e'(X)\cdot\mathbb{1}(X))$ and that $\mathbb{1}^{\mathsf{t}}(X)$ only occurs in such sub-expressions in sentences in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathbb{1}^{\mathsf{t}},\mathsf{diag})$.

(**base case**) $e:=X$. We have that $e(X)\equiv f(X)$ with $f(X):=X$, which is of the desired form.

(**multiplication**) $e(X):=e_1(X)\cdot e_2(X)$. We distinguish between the following cases, depending on the dimensions of $e_1(A_G)$ and $e_2(A_G)$.

- (**n×n, n×n**): $e_1(A_G)$ and $e_2(A_G)$ are of dimension $n\times n$. By induction, $e_1(X)\equiv c_1\times f_1(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(1)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)$ and $e_2(X)\equiv c_2\times f_2(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X)$. This implies that
$$e(X)\equiv(c_1\times c_2)\times f_1(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(1)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)\cdot f_2(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X).$$
Because $\mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)\cdot f_2(X)\cdot\mathbb{1}(X)$ is equivalent to $e_{\mathsf{tr}}(X):=\mathsf{tr}(\mathsf{diag}(g_1(X)\cdot f_2(X)\cdot\mathbb{1}(X)))$, we have
$$e(X)\equiv(c_1\times c_2)\times f_1(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(1)}(X)\cdot e_{\mathsf{tr}}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X),$$
which is of the desired form.
- (**n×n, n×1**): $e_1(A_G)$ is of dimension $n\times n$ and $e_2(A_G)$ is of dimension $n\times 1$. By induction, $e_1(X)\equiv c_1\times f_1(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(1)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)$ and $e_2(X)\equiv c_2\times f_2(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)$. Hence,
$$\begin{aligned}e(X)&\equiv(c_1\times c_2)\times f_1(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(1)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)\cdot f_2(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)\\&\equiv(c_1\times c_2)\times f_1(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(1)}(X)\cdot e_{\mathsf{tr}}(X)\cdot e_{\mathsf{tr}}^{(2)}(X),\end{aligned}$$
where $e_{\mathsf{tr}}(X):=\mathsf{tr}(\mathsf{diag}(g_1(X)\cdot f_2(X)\cdot\mathbb{1}(X)))$ and thus $e(X)$ is equivalent again to an expression of the desired form.
- (**n×1, 1×n**): $e_1(A_G)$ is of dimension $n\times 1$ and $e_2(A_G)$ is of dimension $1\times n$. By induction, $e_1(X)\equiv c_1\times f_1(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(1)}(X)$ and $e_2(X)\equiv c_2\times e_{\mathsf{tr}}^{(2)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X)$. Hence,
$$e(X)\equiv(c_1\times c_2)\times f_1(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(1)}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X),$$
which is of the desired form.
- (**n×1, 1×1**): $e_1(A_G)$ is of dimension $n\times 1$ and $e_2(A_G)$ is of dimension 1×1. By induction, $e_1(X)\equiv c_1\times f_1(X)\cdot\mathbb{1}\cdot e_{\mathsf{tr}}^{(1)}(X)$ and $e_2(X)\equiv c_2\times e_{\mathsf{tr}}^{(2)}(X)$. Hence,
$$e(X)\equiv(c_1\times c_2)\times f_1(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(1)}(X)\cdot e_{\mathsf{tr}}^{(2)}(X),$$
which is already of the desired form.
- (**1×n, n×n**): $e_1(A_G)$ is of dimension $1\times n$ and $e_2(A_G)$ is of dimension $n\times n$. By induction, $e_1(X)\equiv c_1\times e_{\mathsf{tr}}^{(1)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)$ and $e_2(X)\equiv c_2\times f_2(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X)$. As before, this implies that
$$\begin{aligned}e(X)&\equiv(c_1\times c_2)\times e_{\mathsf{tr}}^{(1)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)\cdot f_2(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X)\\&\equiv(c_1\times c_2)\times e_{\mathsf{tr}}^{(1)}(X)\cdot e_{\mathsf{tr}}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X),\end{aligned}$$
where $e_{\mathsf{tr}}(X):=\mathsf{tr}(\mathsf{diag}(g_1(X)\cdot f_2(X)\cdot\mathbb{1}(X)))$.
- (**1×n, n×1**): $e_1(A_G)$ is of dimension $1\times n$ and $e_2(A_G)$ is of dimension $n\times 1$. By induction, $e_1(X)\equiv c_1\times e_{\mathsf{tr}}^{(1)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)$ and $e_2(X)\equiv c_2\times f_2(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(2)}(X)$. Hence,
$$e(X)\equiv(c_1\times c_2)\times e_{\mathsf{tr}}^{(1)}(X)\cdot\mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)\cdot f_2(X)\cdot\mathbb{1}(X)\cdot e_{\mathsf{tr}}^{(2)}(X).$$

As before, let $e_{\mathsf{tr}}(X) := \mathsf{tr}(\mathsf{diag}(g_1(X)\cdot f_2(X)\cdot \mathbb{1}(X)))$. Then,

$$e(X) \equiv (c_1 \times c_2) \times e^{(1)}_{\mathsf{tr}}(X)\cdot e_{\mathsf{tr}}(X)\cdot e^{(2)}_{\mathsf{tr}}(X),$$

as desired.

– $(\mathbf{1{\times}1,1{\times}n})$: $e_1(A_G)$ is of dimension 1×1 and $e_2(A_G)$ is of dimension $1\times n$. By induction, $e_1(X) \equiv c_1 \times e^{(1)}_{\mathsf{tr}}(X)$ and $e_2(X) \equiv c_2 \times e^{(2)}_{\mathsf{tr}}(X)\cdot \mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X)$. Hence,

$$e(X) \equiv (c_1\times c_2)\times e^{(1)}_{\mathsf{tr}}(X)\cdot e^{(2)}_{\mathsf{tr}}(X)\cdot \mathbb{1}^{\mathsf{t}}(X)\cdot g_2(X),$$

which is of the desired form.

– $(\mathbf{1{\times}1,1{\times}1})$: $e_1(A)$ and $e_2(A)$ are of dimension 1×1. By induction, $e_1(X)\equiv c_1\times e^{(1)}_{\mathsf{tr}}(X)$ and $e_2(X)\equiv c_2\times e^{(2)}_{\mathsf{tr}}(X)$. Clearly, this implies that $e(X)\equiv (c_1\times c_2)\times e^{(1)}_{\mathsf{tr}}(X)\cdot e^{(2)}_{\mathsf{tr}}(X)$ which is of the desired form.

(**ones vector**) $e(X) := \mathbb{1}(e_1(X))$. If $e_1(A_G)$ returns an $n\times n$-matrix or $n\times 1$-vector, then $e(X)$ is equivalent to $\mathbb{1}(X)$; if $e_1(A_G)$ returns a $1\times n$-vector or 1×1-matrix, then $e(X)$ is equivalent to $\mathsf{tr}(\mathbb{1}(e_1(X)))$.

(**transposed ones vector**) $e(X) := \mathbb{1}^{\mathsf{t}}(e_1(X))$. If $e_1(A_G)$ returns an $n\times n$-matrix or $n\times 1$-vector, then $e(X)$ is equivalent to $\mathsf{tr}(\mathbb{1}(e_1(X)))$; if $e_1(A_G)$ returns a $1\times n$-vector or 1×1-matrix, then $e(X)$ is equivalent to $\mathbb{1}(X)$.

(**trace**) $e(X) := \mathsf{tr}(e_1(X))$. If $e_1(A_G)$ is a sentence, then $e(X)\equiv e_1(X)$.

If $e_1(A_G)$ is an $n\times n$-matrix, then by induction, $e_1(X)\equiv c\times f_1(X)\cdot \mathbb{1}(X)\cdot e_{\mathsf{tr}}(X)\cdot \mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)$. We observe that

$$\mathsf{tr}(f_1(X)\cdot \mathbb{1}(X)\cdot \mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)) \equiv \mathbb{1}^{\mathsf{t}}(X)\cdot g_1(X)\cdot f_1(X)\cdot \mathbb{1}(X) = \mathsf{tr}(\mathsf{diag}(g_1(X)\cdot f_1(X)\cdot \mathbb{1}(X))).$$

Hence,

$$e(X)\equiv c\times \mathsf{tr}(\mathsf{diag}(g_1(X)\cdot f_1(X)\cdot \mathbb{1}(X)))\cdot e_{\mathsf{tr}}(X),$$

which is of the desired form.

(**diagonalisation**) $e(X) := \mathsf{diag}(e_1(X))$. Here, $e_1(X)$ can only be a 1×1-matrix or an $n\times 1$-vector. In both cases, $e_1(X)$ is equivalent, by induction, to an expression in $\mathsf{ML}(\cdot,\mathsf{tr},\mathbb{1},\mathsf{diag})$. Hence, also $e(X)$ is equivalent to an expression in this fragment. □

Proof of Proposition 8.3

Proposition 8.3 $\mathsf{ML}(\cdot,{}^*,\mathsf{tr},\mathbb{1},\odot_v,\mathsf{diag},+,\times,\mathsf{apply_s}[f], f\in\Omega)$*-vectors are constant on equitable partitions*

Proof Given that we verified this property of all operations except for $\odot_v$ in the proof of Proposition 7.4, we only need to verify that $\odot_v$ can be added to the list of supported operations. We use the same induction hypotheses as in the proof of Proposition 7.4 and verify that these hypotheses remain to hold for $\odot_v$:

(**pointwise vector multiplication**) $e(X) := e_1(X)\odot_v e_2(X)$ where $e_1(X)$ and $e_2(X)$ return vectors. By induction we have that $\mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G) = a_i\times \mathbb{1}_{V_i}$ and $\mathsf{diag}(\mathbb{1}_{V_i})\cdot e_2(A_G) = b_i\times \mathbb{1}_{V_i}$. As a consequence,

$$\begin{aligned}\mathsf{diag}(\mathbb{1}_{V_i})\cdot e(A_G) &= \mathsf{diag}(\mathbb{1}_{V_i})\cdot e_1(A_G)\odot_v e_2(A_G) = a_i\times(\mathbb{1}_{V_i}\odot_v e_2(A_G))\\ &\sum_{j=1}^{\ell} a_i\times(\mathbb{1}_{V_i}\odot_v(\mathsf{diag}(\mathbb{1}_{V_j})\cdot e_2(A_G)) = \sum_{j=1}^{\ell}(a_i\times b_j)\times(\mathbb{1}_{V_i}\odot_v \mathbb{1}_{V_j})\\ &(a_i\times b_i)\times \mathbb{1}_{V_i},\end{aligned}$$

because $\mathbb{1}_{V_i}\odot_v \mathbb{1}_{V_j})$ is either $\mathbb{1}_{V_i}$ when $i=j$, or the zero vector when $i\neq j$. □

Continuation of the proof of Theorem 8.1

In the proof in the main body of the paper we left open the verification that $(\mathbb{1}_{V_i}\cdot \mathbb{1}^{\mathsf{t}}_{V_i})\cdot O = O\cdot(\mathbb{1}_{W_i}\cdot \mathbb{1}^{\mathsf{t}}_{W_i})$, for $i=1,\ldots,\ell$, implies that O preserves the coarsest equitable partitions of G and H. In particular, we need to verify that $\mathbb{1}_{V_i} = O\cdot \mathbb{1}_{W_i}$, for $i=1,\ldots,\ell$. This can be easily shown, just as in the proof of Theorem 7.2 (based on Lemma 4 in Thüne [64]), in which we verified that $J\cdot O = O\cdot J$ implies that $\mathbb{1} = O\cdot \mathbb{1}$.

First, we observe that $(\mathbb{1}_{V_i}\cdot\mathbb{1}_{V_i}^{\mathsf{t}})\cdot O\cdot\mathbb{1}_{W_i}=\mathbb{1}_{V_i}\cdot(\mathbb{1}_{V_i}^{\mathsf{t}}\cdot O\cdot\mathbb{1}_{W_i})=\alpha_i\times\mathbb{1}_{V_i}$ with $\alpha_i=\mathbb{1}_{V_i}^{\mathsf{t}}\cdot O\cdot\mathbb{1}_{W_i}$ and $(\mathbb{1}_{V_i}\cdot\mathbb{1}_{V_i}^{\mathsf{t}})\cdot O\cdot\mathbb{1}_{W_i}=O\cdot(\mathbb{1}_{W_i}\cdot\mathbb{1}_{W_i}^{\mathsf{t}})\cdot\mathbb{1}_{W_i}=(\mathbb{1}_{W_i}^{\mathsf{t}}\cdot\mathbb{1}_{W_i})\times O\cdot\mathbb{1}_{W_i}$. In other words, $O\cdot\mathbb{1}_{W_i}=\frac{\alpha_i}{n_i}\times\mathbb{1}_{V_i}$ where $\mathbb{1}_{W_i}^{\mathsf{t}}\cdot\mathbb{1}_{W_i}=|W_i|=n_i$. Furthermore, because $\mathbb{1}_{V_i}^{\mathsf{t}}\cdot O^{\mathsf{t}}\cdot\mathbb{1}_{W_i}$ is a scalar, $\mathbb{1}_{W_i}^{\mathsf{t}}\cdot O^{\mathsf{t}}\cdot\mathbb{1}_{V_i}=(\mathbb{1}_{V_i}^{\mathsf{t}}\cdot O\cdot\mathbb{1}_{W_i})^{\mathsf{t}}=\mathbb{1}_{V_i}^{\mathsf{t}}\cdot O\cdot\mathbb{1}_{W_i}=\alpha_i$. We next show that $\alpha=\pm n_i$. Indeed, since O is an orthogonal matrix

$$n_i=\mathbb{1}_{V_i}^{\mathsf{t}}\cdot I\cdot\mathbb{1}_{W_i}=\mathbb{1}_{V_i}^{\mathsf{t}}\cdot O^{\mathsf{t}}\cdot O\cdot\mathbb{1}_{W_i}=\frac{\alpha_i}{n_i}\times(\mathbb{1}_{V_i}^{\mathsf{t}}\cdot O^{\mathsf{t}}\cdot\mathbb{1}_{V_i})=\frac{\alpha_i^2}{n_i},$$

and thus $\alpha_i^2=n_i^2$ or $\alpha_i=\pm n_i$. Hence, $O\cdot\mathbb{1}_{W_i}=\pm\mathbb{1}_{V_i}$. We note that $\mathbb{1}=\sum_{i=1}^{\ell}\mathbb{1}_{V_i}=\sum_{i=1}^{\ell}\mathbb{1}_{W_i}$. We now argue that either $\mathbb{1}_{V_i}=O\cdot\mathbb{1}_{W_i}$ for all $i=1,\ldots,\ell$, or $-\mathbb{1}_{V_i}=O\cdot\mathbb{1}_{W_i}$ for all $i=1,\ldots,\ell$. Indeed, suppose that we have $\mathbb{1}_{V_i}=O\cdot\mathbb{1}_{W_i}$ for $i\in K\subset\{1,\ldots,\ell\}$ and $-\mathbb{1}_{V_i}=O\cdot\mathbb{1}_{W_i}$ for $i\in\bar{K}=\{1,\ldots,\ell\}\setminus K$, for some non-empty subset K of $\{1,\ldots,\ell\}$. Then $\sum_{i\in K}\mathbb{1}_{V_i}=O\cdot\big(\sum_{i\in K}\mathbb{1}_{W_i}\big)$ and hence since $\sum_{i\in\bar{K}}\mathbb{1}_{V_i}=\mathbb{1}-\sum_{i\in K}\mathbb{1}_{V_i}$ and $\sum_{i\in\bar{K}}\mathbb{1}_{W_i}=\mathbb{1}-\sum_{i\in K}\mathbb{1}_{W_i}$,

$$\sum_{i\in\bar{K}}\mathbb{1}_{V_i}=O\cdot\big(\sum_{i\in\bar{K}}\mathbb{1}_{W_i}\big).$$

This contradicts that $-\sum_{i\in\bar{K}}\mathbb{1}_{V_i}=O\cdot\big(\sum_{i\in\bar{K}}\mathbb{1}_{W_i}\big)$. Hence, when $\mathbb{1}_{V_i}=O\cdot\mathbb{1}_{W_i}$ for all $i=1,\ldots,\ell$, O satisfies the desired property already. Otherwise, when $-\mathbb{1}_{V_i}=O\cdot\mathbb{1}_{W_i}$ for all $i=1,\ldots,\ell$, we simply replace O by $(-1)\times O$ to obtain that $O\cdot\mathbb{1}_{W_i}=\mathbb{1}_{V_i}$. This rescaling does not impact that $A_G\cdot O=O\cdot A_H$ and we can thus indeed conclude that O preserves the coarsest equitable partitions of G and H. ◻