Similarity matrices for colored graphs *

Paul Van Dooren   Catherine Fraikin

Abstract

In this paper, we extend the notion of similarity matrix, which has been used to define similarity between nodes of two graphs, to the case of colored graphs, where the coloring is either on the nodes or on the edges of both graphs. The proposed method tries to find the optimal matching between the nodes or edges of both graphs but only performs the comparison when their colors are the same. The proposed cost function nevertheless uses the connectivity between all nodes and edges of both graphs. We then also show how to extend this to the notion of low rank similarity matrix, by defining it as a constrained optimization problem.

1 Introduction

Graphs are a powerful tool for many practical problems such as pattern recognition, shape analysis, image processing and data mining. A fundamental task in this context is that of graph matching. The notion of similarity between nodes of two graphs has been introduced in [1] and it was shown there that it had a significant potential for such applications. In this paper we propose various extensions of this definition, analyze their properties and propose algorithms for computing them.

We first introduce the notion of similarity matrix of two colored (node or edge) graphs and show how it extends the ideas introduced in [1] and [7]. This extension can be viewed as a modified graph matching method to compare two graphs

*This paper presents research results of the Belgian Programme on Interuniversity Attraction Poles, initiated by the Belgian Federal Science Policy Office, and a grant Action de Recherche Concertée (ARC) of the Communauté Française de Belgique. The scientific responsibility rests with its authors.

Received by the editors February 2009.

Communicated by A. Bultheel.

2000 *Mathematics Subject Classification* : 05C50, 05C85, 15A18, 68R10.

Key words and phrases : Algorithms, graph algorithms, graph theory, eigenvalues of graphs.

with nodes that have been subdivided into classes of different type or color. This paper also unifies preliminary results obtained in [4], [5] and presents them more formally. The nodes of the same color in the two graphs are compared to each other, but taking into account the complete interconnection pattern of the graphs. This is applied to graphs with colored nodes or colored edges and we show that these problems can still be solved via an appropriately defined eigenvalue problem, just as in the case of the similarity matrix introduced in [1].

When computing the similarity (or its extensions) of large graphs, the complexity can still be quite high, since one needs to solve an eigenvalue problem of a dimension that is essentially the product of the number of nodes in both graphs. In order to reduce the complexity of this problem, we introduced in [3] projected correlation matrices, which can be viewed as low rank approximations of similarity matrices. We show here that also these ideas can be extended to colored graphs. The computation of these projected correlation matrices is done via the constrained optimization of a certain cost function. The method specializes to the spectral method of Caelli and Kosinov [2] in the case that the graphs to be compared are undirected and contain only one type of nodes. It is also an extension of the method described in [3] which handles the directed graph case for nodes of one type only, which in turn is a low rank approximation of the similarity matrix introduced in [1]. Preliminary results on this subject were already presented in [4], [5] but we unify these results here using an optimization framework that shows how these problems relate to each other. The computational technique that we propose is also very similar to that of the above two methods and is essentially a modified power method with special correction applied at each step of the iteration. Since the basic operation to be performed at each step is that of multiplying certain bases with the adjacency matrices of the two graphs, the basic step of the computational procedure can be implemented at reasonable cost for large sparse graphs.

Notation. In this paper we will consider various linear matrix functions of matrix ℓ-tuples $(Y_1, \ldots, Y_\ell) = \mathcal{M}(X_1, \ldots, X_\ell)$ where $X_i, Y_i \in \Re^{m_i \times n_i}$. We associate with this vector space the following inner product

$$\langle (Y_1, \ldots, Y_\ell), (X_1, \ldots, X_\ell) \rangle := \sum_i trace(Y_i^T X_i)$$

and the corresponding norm

$$\|(X_1, \ldots, X_\ell)\| := \langle (X_1, \ldots, X_\ell), (X_1, \ldots, X_\ell) \rangle^{\frac{1}{2}}.$$

For $\ell = 1$ (single matrices $X, Y \in \Re^{m \times n}$) this is known as the Frobenius inner product and the Frobenius norm. For $\ell = n = 1$ (single vectors $x, y \in \Re^m$) this is the standard vector inner product and the so-called 2-norm. We use $vec(X)$ to denote the vector containing the successive columns of X, which implies that $\|vec(X)\| = \|X\|$. Similarly $vec(X_1, \ldots, X_\ell)$ denotes the vector of successive columns of the ℓ-tuple $(X_1, \ldots, X_\ell)$, and we also have that $\|vec(X_1, \ldots, X_\ell)\| = \|(X_1, \ldots, X_\ell)\|$. When rewriting linear functions of matrix ℓ-tuples into their *vec* form, we will need the Kronecker product $B \otimes A$ of two matrices $B \in \Re^{m \times n}$ and $A \in \Re^{s \times t}$, which is a real matrix of dimension $ms \times nt$ with subblocks $B_{i,j}A$ for

$i = 1, \ldots m$ and $j = 1, \ldots n$. For properties of Kronecker products, we refer to [6]. We will also use $\mathbf{1}_d$ to denote a d-vector of all ones and $\mathbf{1}_{m,n}$ to denote a $m \times n$ matrix of all ones.

2 The similarity matrix with type constraints

In [1] one uses an extremal solution of the non-negative matrix equation

$$\rho\, S = \mathcal{M}(S) := ASB^T + A^T SB \tag{1}$$

to define the similarity matrix S between two graphs G_A and G_B with node adjacency matrices A and B. This is actually an eigenvector equation, since the right hand side is a linear map $\mathcal{M}(S)$ in the matrix S. This is made more explicit when rewriting the equation using the vector form $vec(S)$, which stands for a vector containing the successive columns of the matrix S :

$$\rho\, vec(S) = Mvec(S) := (B \otimes A + B^T \otimes A^T)vec(S). \tag{2}$$

Note that the matrix $M := (B \otimes A + B^T \otimes A^T)$ is a natural matrix representation of the linear map $\mathcal{M}(\cdot)$ and that it is symmetric and non-negative, which means that the non-negative vector $vec(S)$ is a Perron vector of M, corresponding to the Perron root (i.e. the spectral radius) $\rho = \max_{\lambda x = Mx} |\lambda|$, and that S is a Perron vector of $\mathcal{M}$ with the Perron root $\rho = \max_{\lambda S = \mathcal{M}(S)} |\lambda|$. Since $\mathcal{M}$ is symmetric, its eigenvalues are real and hence it can have only two extremal eigenvalues, ρ and possibly $-\rho$. On the other hand, $\mathcal{M}^2$ is also non-negative and its only extremal eigenvalue is ρ^2, but its geometric multiplicity can be larger that 1. Let $\mathcal{P}_\rho$ be the orthogonal projector on the space of eigenvectors of $\mathcal{M}^2$ with eigenvalue ρ^2, then $\mathcal{P}_\rho$ is also a non-negative map (satisfying $\rho^2\mathcal{P}_\rho = \mathcal{M}^2\mathcal{P}_\rho$) and any matrix $S = \mathcal{P}_\rho(S_0)$, with S_0 non-negative, will then be a non-negative solution of $\rho^2 S = \mathcal{M}^2(S)$. We make the definition of the similarity matrix S unique by choosing the non-negative solution corresponding to $S_0 = \mathbf{1}_{m,n}$:

$$\boxed{\rho^2\mathcal{P}_\rho = \mathcal{M}^2\mathcal{P}_\rho, \quad S := \mathcal{P}_\rho(\mathbf{1}_{m,n})/\|\mathcal{P}_\rho(\mathbf{1}_{m,n})\|.} \tag{3}$$

Note that when $\mathcal{M}$ does not have an extremal eigenvalue $-\rho$, $\mathcal{P}_\rho$ is also a the projector onto the eigenspace of $\mathcal{M}$ corresponding to the eigenvalue ρ (i.e. $\rho\mathcal{P}_\rho = \mathcal{M}\mathcal{P}_\rho$), and S will then be an extremal solution of (1). It was shown in [1] that the even iterates of the following recurrence

$$S_0 = \mathbf{1}_{m,n}, \quad S_{k+1} = \mathcal{M}(S_k)/||\mathcal{M}(S_k)||,\ k = 0, 1, 2, \ldots \tag{4}$$

converges to the unique solution (3). This iteration is in fact the power method applied to the symmetric matrix M defined above, as can be seen from the *vec* form of the above recurrence :

$$vec(S_0) = \mathbf{1}, \quad vec(S_{k+1}) = Mvec(S_k)/||Mvec(S_k)||,\ k = 0, 1, 2, \ldots \tag{5}$$

and its even iterates will converge to the unique normalized projection of the vector $\mathbf{1}_{mn}$ on the eigenspace of M^2 corresponding to the Perron root ρ^2. The odd iterates will converge to the solution of the problem

$$\rho^2 \mathcal{P}_\rho = \mathcal{M}^2 \mathcal{P}_\rho, \quad S := \mathcal{P}_\rho(\mathcal{M}\mathbf{1}_{m,n})/\|\mathcal{P}_\rho(\mathcal{M}\mathbf{1}_{m,n})\|$$

which is also non-negative, but can be shown to have a smaller 1-norm than the previous one (see [1]). Notice that the matrix iteration (4) is usually more economical to implement (see [1] for details).

One way to interpret the above matrix equations is to say that (1) expresses an equilibrium of the following implicit relation

the similarity between node i of G_A and node j of G_B is large
if the similarity of their respective children and parents is large.

Indeed, rewriting one iteration of (4) as follows:

$$S_{k+1} = (AS_kB^T + A^TS_kB)/||(AS_kB^T + A^TS_kB)|| \tag{6}$$

one sees that its (i, j) element replaces $s_{i,j}$ (this is the similarity between node i of G_A and node j of G_B) by a scalar times the (i, j) element of $ASB^T + A^TSB$ (this is in fact the sum of the similarities of the children and parents of node i of G_A and node j of G_B). The same can be said about the even iterates and hence about the similarity matrix defined in (3). The eigenvector equation (3) thus expresses an equilibrium of the above implicit relation.

Let us now extend this to graphs with ℓ different types of nodes (one can think of them as having different colors). We will assume that the nodes in both graphs are relabeled such that those of color 1 come first, then those of color 2 etc. The corresponding adjacency matrices can thus be partitioned as follows

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1\ell} \\ A_{21} & A_{22} & \dots & A_{2\ell} \\ \vdots & \vdots & \ddots & \vdots \\ A_{\ell 1} & A_{\ell 2} & \dots & A_{\ell\ell} \end{bmatrix}, \quad B = \begin{bmatrix} B_{11} & B_{12} & \dots & B_{1\ell} \\ B_{21} & B_{22} & \dots & B_{2\ell} \\ \vdots & \vdots & \ddots & \vdots \\ B_{\ell 1} & B_{\ell 2} & \dots & B_{\ell\ell} \end{bmatrix}$$

where the blocks $A_{i,j} \in \Re_+^{m_i \times m_j}$ and $B_{i,j} \in \Re_+^{n_i \times n_j}$ describe the edges between nodes of type i to nodes of type j in both A and B.

In this extension we only want to compare nodes of the same color in both graphs, which means that we define similarity matrices S_{ii}, $i = 1, ..., \ell$ of respective dimensions $m_i \times n_i$, which we could put in a block-diagonal similarity matrix

$$S = \begin{bmatrix} S_{11} & 0 & \dots & 0 \\ 0 & S_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & S_{\ell\ell} \end{bmatrix}. \tag{7}$$

If we rewrite equation (1) for such a constrained block diagonal matrix, we obtain

$$\rho\, S = \Pi(ASB^T + A^TSB) \tag{8}$$

where Π is the projector of a matrix to a block diagonal form of the type (7). Since the right hand side is again a linear map $\mathcal{M}(S)$ on the linear space of ℓ-tuples $S = (S_{11}, \ldots, S_{\ell\ell})$, this is still an eigenvector equation and it in fact expresses the implicit relation

the similarity between two nodes of the same color in G_A and G_B is large if the similarity of their respective children and parents is large.

We should point out here that the connections between nodes of different colors are still used in this relation since the children and parents may change color.

In order to make this more explicit we take $\ell = 2$ (two types of nodes) and thus consider two graphs with partitioned adjacency matrices

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$

with $A_{i,j} \in \Re^{m_i \times m_j}$ and $B_{i,j} \in \Re^{n_i \times n_j}$, $i = 1, 2$ and $j = 1, 2$. Equation (8) then specializes to

$$\begin{aligned} \rho\, S_{11} &= A_{11} S_{11} B_{11}^T + A_{11}^T S_{11} B_{11} + A_{12} S_{22} B_{12}^T + A_{21}^T S_{22} B_{21} \\ \rho\, S_{22} &= A_{21} S_{11} B_{21}^T + A_{12}^T S_{11} B_{12} + A_{22} S_{22} B_{22}^T + A_{22}^T S_{22} B_{22} \end{aligned} \tag{9}$$

which in *vec* notation becomes

$$\rho \begin{bmatrix} vec(S_{11}) \\ vec(S_{22}) \end{bmatrix} = M \begin{bmatrix} vec(S_{11}) \\ vec(S_{22}) \end{bmatrix} \tag{10}$$

where

$$M := \begin{bmatrix} B_{11} \otimes A_{11} + B_{11}^T \otimes A_{11}^T & B_{12} \otimes A_{12} + B_{21}^T \otimes A_{21}^T \\ B_{21} \otimes A_{21} + B_{12}^T \otimes A_{12}^T & B_{22} \otimes A_{22} + B_{22}^T \otimes A_{22}^T \end{bmatrix} \tag{11}$$

is the matrix representation of the linear map $\mathcal{M}(\cdot)$. The diagonal blocks in the matrix M are related to links staying within the node sets of the same color, while the off diagonal blocks refer to links with nodes of another color.

Notice that the matrix M is again symmetric for this new problem and hence we can define a unique solution of the equation (8) via (3) and compute it since the even iterates of the vector recurrence (4) will converge to it. The general recurrence is given by

$$S_{k+1} = \Pi(A S_k B^T + A^T S_k B) / ||\Pi(A S_k B^T + A^T S_k B)|| \tag{12}$$

and for the special case of two blocks considered above, this specializes to

$$\begin{aligned} Z_1 &= A_{11} S_{11} B_{11}^T + A_{11}^T S_{11} B_{11} + A_{12} S_{22} B_{12}^T + A_{21}^T S_{22} B_{21} \\ Z_2 &= A_{21} S_{11} B_{21}^T + A_{12}^T S_{11} B_{12} + A_{22} S_{22} B_{22}^T + A_{22}^T S_{22} B_{22} \\ (S_{11}, S_{22})_+ &= (Z_1, Z_2) / \|(Z_1, Z_2)\|, \end{aligned} \tag{13}$$

where the subscript $._+$ refers to the next iteration value. Notice that this iteration is more economical than the full matrix iteration (6), since only the diagonal blocks are carried along in the calculations. Imposing color constraints makes the problem thus simpler. A simple special case of such a constraint would be to choose a separate color for one particular node in both G_A and G_B. These two nodes would then only be compared to each other and would simplify (but also affect) the calculation of the similarity of the remaining nodes.

Example 1 The following small example illustrates this. It consists of two types of nodes (those with full lines and those with dashed lines) both arranged in a circular ring of five nodes. The constrained similarity matrix for graph A and graph B (Figure 1) is then given by

$$S = \begin{bmatrix} 0.43 & 0 & 0 & 0 & 0 \\ 0.26 & 0.27 & 0 & 0 & 0 \\ 0 & 0.43 & 0 & 0 & 0 \\ 0 & 0 & 0.43 & 0.26 & 0 \\ 0 & 0 & 0 & 0.27 & 0.43 \end{bmatrix}. \tag{14}$$

As one could expect, the highest similarity score (0.43) is obtained for the pairs

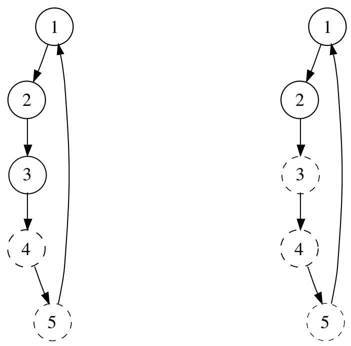

Figure 1: Graph A and graph B with two types of nodes (full and dashed)

of nodes that are at the transition between two types of nodes, i.e. the pairs (1,1), (3,2), (4,3) and (5,5).

3 Similarity between colored edges

In [7] a definition is given for the similarity between edges of two graphs G_A and G_B. This similarity measure is based on the following observation:

the similarity between an edge in G_A and an edge in G_B is large
if the similarity between their respective source and terminal nodes is large.

In [7] this is transformed into matrix equations as follows. Let $G_A(E_A, V_A)$ be a graph with m_A elements in the edge set E_A and n_A elements in the node set V_A. Let $s_A(i)$ denote the source node of edge i in this graph, and let $t_A(i)$ denote the terminal node of edge i. The adjacency structure of the graph can then described by a pair of $n_A \times m_A$ matrices, the *source-edge matrix* A_S and the *terminal-edge matrix* A_T, defined as follows:

$$
\begin{aligned}
[A_S]_{ij} &= 1 \text{ if } s_A(j) = i \quad (0 \text{ otherwise}), & (15)\\
[A_T]_{ij} &= 1 \text{ if } t_A(j) = i \quad (0 \text{ otherwise}). & (16)
\end{aligned}
$$

The node-node adjacency matrix A of the graph G_A is an $n_A \times n_A$ matrix which is then given by

$$A := A_S A_T^T$$

while the edge-edge adjacency matrix A_e is an $m_A \times m_A$ matrix given by

$$A_e := A_S^T A_T$$

which says when one edge originates at the terminal node of another edge, i.e.

$$\left[A_S^T A_T\right]_{ij} = 1 \text{ if } s_A(i) = t_A(j) \quad (0 \text{ otherwise}). \tag{17}$$

The diagonal element $[A_S^T A_T]_{ii} = 1$ of this matrix is 1 if edge i is a self-loop. Notice also that $D_{A_S} = A_S A_S^T$ and $D_{A_T} = A_T A_T^T$ are diagonal matrices with respectively the *out-degree* and *in-degree* of node i in the ith diagonal entry.

The matrices $H_{A_S} := A_S^T A_S$ and $H_{A_T} := A_T^T A_T$ can also be viewed as edge-edge adjacency matrices, which tell us when two edges start or terminate at the same node, respectively, i.e.

$$
\begin{aligned}
\left[H_{A_S}\right]_{ij} &= 1 \text{ if } s_A(i) = s_A(j) \quad (0 \text{ otherwise}), & (18)\\
\left[H_{A_T}\right]_{ij} &= 1 \text{ if } t_A(i) = t_A(j) \quad (0 \text{ otherwise}). & (19)
\end{aligned}
$$

These matrices are symmetric and have 1's on the diagonal.

In [7] one then introduces a node similarity matrix X and an edge similarity matrix Y, which are linked via the matrix equations

$$
\begin{aligned}
\sigma X &= A_S Y B_S^T + A_T Y B_T^T, & (20)\\
\sigma Y &= A_S^T X B_S + A_T^T X B_T. & (21)
\end{aligned}
$$

The right hand side is a linear transformation $\mathcal{M}(X, Y)$ of the pair of matrices (X, Y) and this is therefore again an eigenvector equation, as is easily seen from its *vec* form:

$$\sigma \begin{bmatrix} vec(X) \\ vec(Y) \end{bmatrix} = \begin{bmatrix} 0 & B_S \otimes A_S + B_T \otimes A_T \\ B_S^T \otimes A_S^T + B_T^T \otimes A_T^T & 0 \end{bmatrix} \begin{bmatrix} vec(X) \\ vec(Y) \end{bmatrix}. \tag{22}$$

This is clearly again a symmetric map, which can be rewritten as

$$\sigma \begin{bmatrix} vec(X) \\ vec(Y) \end{bmatrix} = \begin{bmatrix} 0 & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} vec(X) \\ vec(Y) \end{bmatrix}, \quad G := B_S \otimes A_S + B_T \otimes A_T. \tag{23}$$

The similarity matrix pair (X, Y) can be uniquely defined using (3) and computed using the even iterates of (4) which here takes the following form:

$$(X, Y)_+ = \mathcal{M}(X, Y) / \|\mathcal{M}(X, Y)\|, \quad \mathcal{M}(X, Y) := (\mathcal{G}(Y), \mathcal{G}^T(X)), \tag{24}$$

where $\mathcal{G}(Y) := A_S Y B_S^T + A_T Y B_T^T, \mathcal{G}^T(X) := A_S^T X B_S + A_T^T X B_T$ are the linear maps corresponding to (23). Moreover, the even iterates of (24) are also the iterates of the normalized map $\mathcal{M}^2(X,Y)$ which decouples into:

$$X_+ = \mathcal{G}(\mathcal{G}^T(X))/\|\mathcal{G}(\mathcal{G}^T(X))\|, \quad Y_+ = \mathcal{G}^T(\mathcal{G}(Y))/\|\mathcal{G}^T(\mathcal{G}(Y))\|, \tag{25}$$

but with a scaling that differs from the joint scaling of (24).

Remark 1. *Note also that one can obtain from* (20) *the decoupled equations :*

$$\sigma^2 X = \mathcal{G}(\mathcal{G}^T(X)) = A_S(A_S^T X B_S + A_T^T X B_T)B_S^T + A_T(A_S^T X B_S + A_T^T X B_T)B_T^T \tag{26}$$

$$\sigma^2 Y = \mathcal{G}^T(\mathcal{G}(Y)) = A_S^T(A_S Y B_S^T + A_T Y B_T^T)B_S + A_T^T(A_S Y B_S^T + A_T Y B_T^T)B_T \tag{27}$$

which can be rewritten as

$$\sigma^2 X = AXB^T + A^T XB + D_{A_S} X D_{B_S} + D_{A_T} X D_{B_T}, \tag{28}$$

$$\sigma^2 Y = A_e Y B_e^T + A_e^T Y B_e + H_{A_S} Y H_{B_S} + H_{A_T} Y H_{B_T}. \tag{29}$$

The equation for X can thus be seen as an enhanced similarity matrix equation since the first two terms are the adjacency matrix terms appearing also in (1)*. The additional terms amplify the scores of nodes that are highly connected. In the equation for Y the first two terms are also edge-edge adjacency matrix terms while the second terms involve positive semi-definite matrices with ones on diagonal. Notice that the decoupled equations could be normalized independently of each other. But this will only yield a relative scaling between the matrices as a whole, and not between the elements within each matrix.*

Let us now try to rewrite this for graphs with edges of ℓ different colors. The edges can be relabeled such that those of the same color are adjacent in the source-edge and terminal-edge matrices:

$$A_S = \begin{bmatrix} A_{S_1} & \dots & A_{S_\ell} \end{bmatrix}, \quad A_T = \begin{bmatrix} A_{T_1} & \dots & A_{T_\ell} \end{bmatrix},$$

$$B_S = \begin{bmatrix} B_{S_1} & \dots & B_{S_\ell} \end{bmatrix}, \quad B_T = \begin{bmatrix} B_{T_1} & \dots & B_{T_\ell} \end{bmatrix}.$$

The blocks $A_{S_i}, A_{T_i} \in \Re^{n_a \times m_{a_i}}$ and $B_{S_i}, B_{T_i} \in \Re^{n_b \times m_{b_i}}$, $i = 1, ..., \ell$ thus correspond to edges of the same type, where m_{a_i} and m_{b_i}, represent the number of edges of type i in G_A and G_B respectively.

The edge similarity matrix has to be block-diagonal because we compare only edges of the same type. This matrix has thus a block diagonal structure with blocks Y_{ii} of dimension $m_{a_i} \times m_{b_i}$:

$$Y = \begin{bmatrix} Y_{11} & 0 & \dots & 0 \\ 0 & Y_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Y_{\ell\ell} \end{bmatrix}. \tag{30}$$

If we rewrite equation (20) for such a constrained block diagonal matrix, we obtain

$$\sigma X = A_S Y B_S^T + A_T Y B_T^T, \tag{31}$$

$$\sigma Y = \Pi(A_S^T X B_S + A_T^T X B_T), \tag{32}$$

where Π is the projector of a matrix to a block diagonal form of the type (30). Since the right hand side is again a linear map $\mathcal{M}(X, Y_{11}, \ldots, Y_{\ell\ell})$ in the $(\ell+1)$-tuple $(X, Y_{11}, \ldots, Y_{\ell\ell})$ this is still an eigenvector equation and it in fact expresses the implicit relation

the similarity between two edges of the same color in G_A and G_B is large if the similarity of their respective source and terminal nodes is large.

We make this again explicit for the two color case ($\ell = 2$) and thus consider two graphs with partitioned matrices

$$A_S = \left[\begin{array}{cc} A_{S_1} & A_{S_2} \end{array}\right], \quad A_T = \left[\begin{array}{cc} A_{T_1} & A_{T_2} \end{array}\right],$$

$$B_S = \left[\begin{array}{cc} B_{S_1} & B_{S_2} \end{array}\right], \quad B_T = \left[\begin{array}{cc} B_{T_1} & B_{T_2} \end{array}\right],$$

and

$$Y = \left[\begin{array}{cc} Y_{11} & 0 \\ 0 & Y_{22} \end{array}\right].$$

Equation (20) in partitioned form becomes

$$\begin{aligned} \sigma X &= A_{S_1} Y_{11} B_{S_1}^T + A_{S_2} Y_{22} B_{S_2}^T + A_{T_1} Y_{11} B_{T_1}^T + A_{T_2} Y_{22} B_{T_2}^T, \\ \sigma Y_{11} &= A_{S_1}^T X B_{S_1} + A_{T_1}^T X B_{T_1}, \\ \sigma Y_{22} &= A_{S_2}^T X B_{S_2} + A_{T_2}^T X B_{T_2}, \end{aligned} \tag{33}$$

which is an eigenvector equation $\sigma(X, Y_{11}, Y_{22}) = \mathcal{M}(X, Y_{11}, Y_{22})$. By applying the *vec* operator, we indeed get an explicit eigenvalue problem

$$\sigma \left[\begin{array}{c} vec(X) \\ vec(Y_{11}) \\ vec(Y_{22}) \end{array}\right] = M \left[\begin{array}{c} vec(X) \\ vec(Y_{11}) \\ vec(Y_{22}) \end{array}\right] \tag{34}$$

with

$$M := \left[\begin{array}{cc} 0 & G \\ G^T & 0 \end{array}\right], \quad G := [B_{S_1} \otimes A_{S_1} + B_{T_1} \otimes A_{T_1},\ B_{S_2} \otimes A_{S_2} + B_{T_2} \otimes A_{T_2}]$$

which shows again that the linear map $\mathcal{M}$ is symmetric. One can then again apply (3) to define a unique solution, and use the even iterates of (4) to construct an algorithm to converge to that solution. That iteration essentially consists of normalizing the right hand sides of (33) to define the updating matrix triple

$$(X, Y_{11}, Y_{22})_+ = \mathcal{M}(X, Y_{11}, Y_{22}) / \|\mathcal{M}(X, Y_{11}, Y_{22})\|.$$

The even iterates will again decompose into the (separately scaled) iterations:

$$X_+ = \mathcal{G}(\mathcal{G}^T(X)) / \|\mathcal{G}(\mathcal{G}^T(X))\|, \quad (Y_{11}, Y_{22})_+ = \mathcal{G}^T(\mathcal{G}(Y_{11}, Y_{22})) / \|\mathcal{G}^T(\mathcal{G}(Y_{11}, Y_{22}))\|.$$

Remark 2. *One can also consider the combination of colored nodes and colored edges. The matrices X and Y will in this case both be block diagonal, possibly with a different number of blocks. In order to simplify the discussion, we develop this for the case that* *there are two types of nodes and two types of edges. The matrices X and Y then have the form*

$$X = \begin{bmatrix} X_{11} & 0 \\ 0 & X_{22} \end{bmatrix}, \quad Y = \begin{bmatrix} Y_{11} & 0 \\ 0 & Y_{22} \end{bmatrix}$$

and the source-edge and terminal-edge matrices for graphs A and B then become:

$$A_S = \begin{bmatrix} A_{S_{11}} & A_{S_{12}} \\ A_{S_{21}} & A_{S_{22}} \end{bmatrix}, \ A_T = \begin{bmatrix} A_{T_{11}} & A_{T_{12}} \\ A_{T_{21}} & A_{T_{22}} \end{bmatrix},$$

$$B_S = \begin{bmatrix} B_{S_{11}} & B_{S_{12}} \\ B_{S_{21}} & B_{S_{22}} \end{bmatrix}, \ B_T = \begin{bmatrix} B_{T_{11}} & B_{T_{12}} \\ B_{T_{21}} & B_{T_{22}} \end{bmatrix}.$$

The iteration matrix still has the same structure as for the case of colored edges:

$$M := \begin{bmatrix} 0 & G \\ G^T & 0 \end{bmatrix}, \ G := \begin{bmatrix} B_{S_{11}} \otimes A_{S_{11}} + B_{T_{11}} \otimes A_{T_{11}} & B_{S_{12}} \otimes A_{S_{12}} + B_{T_{12}} \otimes A_{T_{12}} \\ B_{S_{21}} \otimes A_{S_{21}} + B_{T_{21}} \otimes A_{T_{21}} & B_{S_{22}} \otimes A_{S_{22}} + B_{T_{22}} \otimes A_{T_{22}} \end{bmatrix}$$

which is why we do not analyze this in more detail.

Example 2 The two graphs of Figure 2 are very similar in the sense that node 3 of graph A has been replaced by three *identical* nodes 3, 4 and 5 in graph B. This is also detected by the node and edge similarity matrices for these two graphs, which are given by

$$X = \begin{bmatrix} 0.22 & 0 & 0 & 0 & 0 \\ 0 & 0.89 & 0 & 0 & 0 \\ 0 & 0 & 0.22 & 0.22 & 0.22 \end{bmatrix}, \qquad Y = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0.5 \end{bmatrix}. \tag{35}$$

Notice that the matrix Y is block diagonal, indicating that both edges (1-2) are

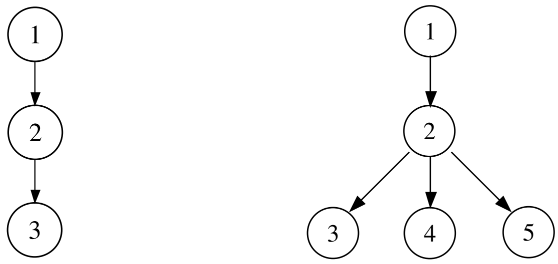

Figure 2: Comparison between the edges of graph A and graph B

only similar to each other, while edge (2-3) in A is equally similar to edges (2-3), (2-4) and (2-5) in B and not at all similar to edge (1-2) in B. Clearly we would have obtained the same results with the colored edge graphs A and B of Figure 3 since now we are *imposing* the matrix Y to be block diagonal. Clearly the same can be said about the combined colored node and edge graphs in Figure 3 since then we are imposing the block diagonal structure in both X and Y. Let us point out here that matrices X and Y are equal for all three figures only because we impose a global normalization rather than a normalization of individual blocks in these matrices.

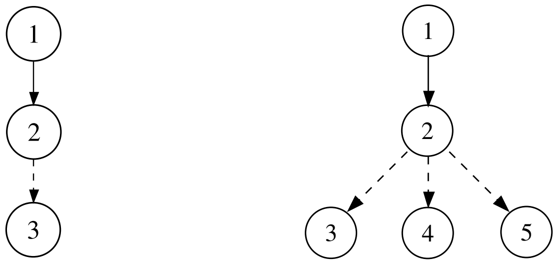

Figure 3: Comparison between the colored edges of graph A and graph B

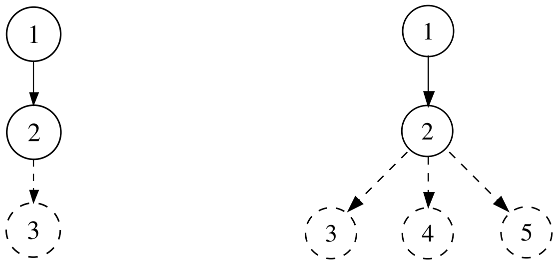

Figure 4: Comparison between the colored edges and nodes of graph A and graph B

4 Similarity matrices as an optimization problem

In this section we interpret the eigenvalue problems presented in the previous section as optimization problems, for which the eigenvalue problems yield the stationary points. The reason for doing that is that we will then see how to extend these problems to a more general setting. In particular we will show in the next section how we can cast low rank approximation problems for similarity matrices as constrained optimization problems.

Every (extremal) eigenvalue equation $\lambda S = \mathcal{M}(S)$ can be viewed as a stationary point of a quadratic optimization problem on a normed vectorspace :

$$\max_{\langle S,S\rangle=1} \langle S, \mathcal{M}(S)\rangle. \tag{36}$$

Here S can be a vector $x \in \Re^m$, a matrix $X \in \Re^{m\times n}$, or an ℓ-tuple of matrices $X_i \in \Re^{m_i\times n_i}$. Introducing a Lagrange multiplier λ for the norm constraint $\langle S, S\rangle = 1$, one obtains the unconstrained problem

$$\max_S L(S,\lambda) := \max_S \langle S, \mathcal{M}(S)\rangle + \lambda(1 - \langle S, S\rangle) \tag{37}$$

and deriving the Lagrangian $L(S,\lambda)$ versus S and λ then yields the first order optimality conditions

$$\nabla_S L(S,\lambda) = \mathcal{M}(S) - \lambda S = 0, \quad \nabla_\lambda L(S,\lambda) = 1 - \langle S, S\rangle = 0. \tag{38}$$

The connection with the previous sections is obvious, except that we considered there the squared map $\mathcal{M}^2$. If $\mathcal{M}$ has no eigenvalue equal to minus its spectral radius then these problems become identical. Since this is almost always the case (it corresponds to the absence of even cycles in the underlying graphs), we will make this assumption in the rest of the paper. The more general treatment would require to consider the squared map $\mathcal{M}^2$ (which is also linear) and all equations would just become harder to read.

If we now apply these ideas to the four problems discussed in the previous section, we find equivalent optimization problems. The combination of colored edges and nodes is similar and will not be developed further here.

1. The similarity matrix S in (1) is an extremal point of the optimization problem

$$\max_{\|S\|=1} 2\langle S, ASB^T\rangle = \max_{\|S\|=1} \langle S, ASB^T + A^TSB\rangle. \tag{39}$$

The first-order optimality condition is indeed (1) with Lagrange multiplier $\lambda = \rho$.

2. The constrained similarity matrix pair (S_{11}, S_{22}) in (9) is the solution of the optimization problem

$$\max_{\|(S_{11},S_{22})\|=1} 2\langle (S_{11}, S_{22}), (A_{11}S_{11}B_{11}^T + A_{12}S_{22}B_{12}^T, A_{22}S_{22}B_{22}^T + A_{21}S_{11}B_{21}^T)\rangle. \tag{40}$$

The Lagrange multiplier in the first-order optimality condition is again $\lambda = \rho$.

3. The coupled node-edge similarity matrix pair (X, Y) (20) is the solution of the optimization problem (with $\lambda = \sigma$)

$$\max_{\|(X,Y)\|=1} \langle (X, Y), (A_SYB_S^T + A_TYB_T^T, A_S^TXB_S + A_T^TXB_T)\rangle. \tag{41}$$

4. The coupled node-edge similarity matrix triple (X, Y_{11}, Y_{22}) (33) is the solution of the optimization problem (with $\lambda = \sigma$)

$$\begin{aligned}\max\nolimits_{\|(X,Y_{11},Y_{22})\|=1} \langle (X, Y_{11}, Y_{22}), (&A_{S_1}Y_{11}B_{S_1}^T + A_{T_1}Y_{11}B_{T_1}^T + A_{S_2}Y_{22}B_{S_2}^T + A_{T_2}Y_{22}B_{T_2}^T,\\ &A_{S_1}^TXB_{S_1} + A_{T_1}^TXB_{T_1}, A_{S_2}^TXB_{S_2} + A_{T_2}^TXB_{T_2})\rangle.\end{aligned}$$

5 Optimization of projected matrices

The extensions presented in this paper are given without proof. The basic ideas of the modified optimization problems and their stationary points, the iterative algorithm and their proofs of convergence can also be found in [3], [4] and [5]. In this section we make the same assumptions about the underlying linear map $\mathcal{M}$ as in the previous section.

5.1 Projected similarity

In [3] it is suggested to compare two large square matrices A and B, possibly of different dimension n_A and n_B, via projections of the matrices on a k dimensional subspace where $k \le \min(n_A, n_B)$. The $k \times k$ projected matrices A_U and B_V are obtained as follows

$$A_U := U^TAU, \quad U^TU = I_k, \quad B_V := V^TBV, \quad V^TV = I_k$$

and the isometries U and V are obtained from the optimization of the following cost function

$$\max_{U^TU=V^TV=I_k} 2\langle A_U, B_V\rangle = \max_{U^TU=V^TV=I_k} 2\langle U^TAU, V^TBV\rangle. \tag{42}$$

The first-order optimality conditions can be derived from the Lagrangian

$$L(U,V,F,G) = \langle U^TAU, V^TBV\rangle + \langle U^TA^TU, V^TB^TV\rangle + \langle F, I-U^TU\rangle + \langle G, I-V^TV\rangle \tag{43}$$

where F and G are symmetric matrices of Lagrange multipliers for the isometry constraints. Partial gradients of L with respect to (U,V) lead to the following first order optimality conditions

$$\begin{aligned} \nabla_U L &= AU\left(V^TB^TV\right) + A^TU\left(V^TBV\right) - UF = 0 \\ \nabla_V L &= BV\left(U^TA^TU\right) + B^TV\left(U^TAU\right) - VG = 0 \end{aligned}$$

or equivalently

$$\begin{aligned} UF &= AU(V^TB^TV) + A^TU(V^TBV) \\ VG &= BV(U^TA^TU) + B^TV(U^TAU) \end{aligned}$$

and of course the constraints $U^TU = V^TV = I$. It is shown in [3] that $F = G$ and can be chosen equal to a diagonal matrix Λ, which finally yields necessary and sufficient conditions for a stationary point of the optimization problem

$$U\Lambda V^T = SVD_k(AUV^TB^T + A^TUV^TB), \quad U^TU = V^TV = I_k, \tag{44}$$

where $SVD_k(Z)$ is the best rank k approximation of the matrix Z, and can be obtained by truncating the singular values decomposition of Z.

This last equation is quite similar to the equation (1) introduced in the definition of the similarity matrix of two graphs G_A and G_B, except for the restriction to a rank k matrix. The connection is even clearer when looking at the above optimization problem (42), which can be rewritten as

$$\max_{U^TU=V^TV=I_k} 2\langle UV^T, AUV^TB^T\rangle. \tag{45}$$

Clearly the role of S in (39) is replaced here by the rank k matrix UV^T with a normalization now imposed on the factors U and V. For more comments on how U and V are actually used to compare the individual nodes of the two graphs, we refer to [3],[2].

In [3] it is also proved that the following iteration converges to such a stationary point under very mild conditions :

$$U_+\Sigma V_+^T = SVD_k(AUV^TB^T + A^TUV^TB + sUV^T). \tag{46}$$

The shift s has to be chosen such that $\Sigma = \Lambda + sI_k$ is positive definite. The above iteration then defines an iteration on the product $(UV^T)_+ = \phi(UV^T)$ that is compatible with (46) and converges to a stationary point of (45).

5.2 Colored projected similarity

Let us revisit the node similarity matrix problem with two types of nodes and with the following partitioned adjacency matrices:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}.$$

The blocks $A_{i,j} \in \Re^{m_i \times m_j}$ and $B_{i,j} \in \Re^{n_i \times n_j}$, $i = 1,2$ and $j = 1,2$, describe the edges between nodes of type i to nodes of type j in both A and B. Since the projections U and V should not mix nodes of different types, we constrain them to have a block diagonal form:

$$U = \begin{bmatrix} U_1 & 0 \\ 0 & U_2 \end{bmatrix}, \quad V = \begin{bmatrix} V_1 & 0 \\ 0 & V_2 \end{bmatrix}, \quad UV^T = \begin{bmatrix} U_1V_1^T & 0 \\ 0 & U_2V_2^T \end{bmatrix}, \tag{47}$$

where U_i has dimension $m_i \times k$ and $U_i^TU_i = I_k$, V_i has dimension $m_i \times k$ and $V_i^TV_i = I_k$. The optimization problem (42) then becomes

$$\max_{U_i^TU_i=V_i^TV_i=I_k} 2\langle (U_1V_1^T, U_2V_2^T), (A_{11}U_1V_1^TB_{11}^T + A_{12}U_2V_2^TB_{12}^T, A_{22}U_2V_2^TB_{22}^T + A_{21}U_1V_1^TB_{21}^T)\rangle \tag{48}$$

which again is very similar to (40) with S_{ii} replaced by the product $U_iV_i^T$. The first-order conditions can be derived from the Lagrangian $L(U_1, U_2, V_1, V_2, F_1, F_2, G_1, G_2)$:

$$\begin{aligned} \langle U_1^TA_{11}U_1, V_1^TB_{11}V_1\rangle &+ \langle U_1^TA_{12}U_2, V_1^TB_{12}V_2\rangle + \\ \langle U_2^TA_{21}U_1, V_2^TB_{21}V_1\rangle &+ \langle U_2^TA_{22}U_2, V_2^TB_{22}V_2\rangle + \\ \langle F_1, (I - U_1^TU_1)\rangle &+ \langle F_2, (I - U_2^TU_2)\rangle + \\ \langle G_1, (I - V_1^TV_1)\rangle &+ \langle G_2, (I - V_2^TV_2)\rangle \end{aligned} \tag{49}$$

where F_i and G_i are symmetric matrices of Lagrange multipliers for the orthogonality constraints. By setting the partial gradients of this Lagrangian to zero, the first order conditions are found to be :

$$\begin{aligned} U_1F_1 &= [A_{11}U_1V_1^TB_{11}^T + A_{11}^TU_1V_1^TB_{11} + A_{12}U_2V_2^TB_{12}^T + A_{21}^TU_2V_2^TB_{21}]V_1, \\ V_1G_1 &= [B_{11}V_1U_1^TA_{11}^T + B_{11}^TV_1U_1^TA_{11} + B_{12}V_2U_2^TA_{12}^T + B_{21}^TV_2U_2^TA_{21}]U_1, \\ U_2F_2 &= [A_{22}U_2V_2^TB_{22}^T + A_{22}^TU_2V_2^TB_{22} + A_{21}U_1V_1^TB_{21}^T + A_{12}^TU_1V_1^TB_{12}]V_2, \\ V_2G_2 &= [B_{22}V_2U_2^TA_{22}^T + B_{22}^TV_2U_2^TA_{22} + B_{21}V_1U_1^TA_{21}^T + B_{12}^TV_1U_1^TA_{12}]U_2, \end{aligned}$$

where $F_i = G_i$ are symmetric matrices that can be chosen equal to a diagonal matrix Λ_i. We finally obtain from the combination of these equations

$$U_1\Lambda_1V_1^T = SVD_k(A_{11}U_1V_1^TB_{11}^T + A_{11}^TU_1V_1^TB_{11} + A_{12}U_2V_2^TB_{12}^T + A_{21}^TU_2V_2^TB_{21}), \tag{50}$$
$$U_2\Lambda_2V_2^T = SVD_k(A_{22}U_2V_2^TB_{22}^T + A_{22}^TU_2V_2^TB_{22} + A_{21}U_1V_1^TB_{21}^T + A_{12}^TU_1V_1^TB_{12}). \tag{51}$$

This yields again iteration functions $(U_iV_i^T)_+ = \phi_i(U_iV_i^T)$, $i = 1,2$ that are based on the following rank k approximations

$$U_{1+}\Sigma_1V_{1+}^T = SVD_k(A_{11}U_1V_1^TB_{11}^T + A_{11}^TU_1V_1^TB_{11} + A_{12}U_2V_2^TB_{12}^T + A_{21}^TU_2V_2^TB_{21} + sU_1V_1^T),$$
$$U_{2+}\Sigma_2V_{2+}^T = SVD_k(A_{22}U_2V_2^TB_{22}^T + A_{22}^TU_2V_2^TB_{22} + A_{21}U_1V_1^TB_{21}^T + A_{12}^TU_1V_1^TB_{12} + sU_2V_2^T),$$

which converge to a stationary point $(U_1V_1^T, U_2V_2^T)$ of (48) provided the two diagonal matrices $\Sigma_i := \Lambda_i + sI_k$ are positive.

5.3 Projected node-edge similarity

We now consider the projected version of the node-edge similarity matrices introduced in (20) and for which the corresponding optimization problem is given in (41). The matrices X and Y are here approximated by $U_1V_1^T$ and $U_2V_2^T$, respectively, where $U_1 \in \Re^{n_a \times k}$, $V_1 \in \Re^{n_b \times k}$, $U_2 \in \Re^{m_a \times k}$ and $V_2 \in \Re^{m_b \times k}$. The matrices U_1, U_2, V_1 and V_2 are obtained from the optimization problem

$$\max_{U_i^TU_i=V_i^TV_i=I_k} \frac{1}{2}\langle (U_1V_1^T, U_2V_2^T), (A_SU_2V_2^TB_S^T + A_TU_2V_2^TB_T^T, A_S^TU_1V_1^TB_S + A_T^TU_1V_1^TB_T)\rangle.$$

Optimality conditions can be derived from the Lagrangian $L(U_1, V_1, U_2, V_2, F_1, G_1, F_2, G_2)$, given by

$$\begin{array}{c} \langle U_1^TA_SU_2, V_1^TB_SV_2\rangle \;+\; \langle U_1^TA_TU_2, V_1^TB_TV_2\rangle \;+ \\ \langle F_1, (I - U_1^TU_1)\rangle \;+\; \langle G_1, (I - V_1^TV_1)\rangle \;+\; \langle F_2, (I - U_2^TU_2)\rangle \;+\; \langle G_2, (I - V_2^TV_2)\rangle \end{array} \tag{52}$$

where F_i and G_i are symmetric matrices of Lagrange multipliers that are equal and can be chosen diagonal : $F_i = G_i = \Lambda_i,\ i = 1, 2$. The first order conditions are then found to be :

$$\begin{aligned} U_1\Lambda_1V_1^T &= A_SU_2V_2^TB_S^T + A_TU_2V_2^TB_T^T, \\ U_2\Lambda_2V_2^T &= A_S^TU_1V_1^TB_S + A_T^TU_1V_1^TB_T, \end{aligned} \tag{53}$$

which again leads to a simple updating scheme $(U_iV_i^T)_+ = \phi_i(U_iV_i^T),\ i = 1, 2$ given by

$$\begin{aligned} U_{1+}\Sigma_1V_{1+}^T &= SVD_k(A_SU_2V_2^TB_S^T + A_TU_2V_2^TB_T^T + sU_1V_1^T), \\ U_{2+}\Sigma_2V_{2+}^T &= SVD_k(A_S^TU_1V_1^TB_S + A_T^TU_1V_1^TB_T + sU_2V_2^T), \end{aligned}$$

which converge under mild conditions to a stationary point $(U_1V_1^T, U_2V_2^T)$ of the above optimization problem provided the diagonal matrices $\Sigma_i := \Lambda_i + sI_k$ are positive.

5.4 Colored projected node-edge similarity

As explained in (31), the extension to graphs with different types of edges requires a partitioning of the edge-node matrices :

$$A_S = \left[\ A_{S_1} \quad A_{S_2}\ \right], \quad A_T = \left[\ A_{T_1} \quad A_{T_2}\ \right],$$

$$B_S = \left[\ B_{S_1} \quad B_{S_2}\ \right], \quad B_T = \left[\ B_{T_1} \quad B_{T_2}\ \right],$$

where the edges of the same type $i = 1, 2$ correspond to the same blocks in all four matrices. The matrices X, Y_1 and Y_2 are now approximated by low rank matrices :

$$X = U_1 V_1^T, \quad Y = \begin{bmatrix} U_2 V_2^T & 0 \\ 0 & U_3 V_3^T \end{bmatrix},$$

where the isometries U_i and V_i matrices have all only k columns, and obtained from the optimization problem

$$\begin{array}{c} \max_{U_i^T U_i = V_i^T V_i = I_k} \frac{1}{2} \langle (U_1 V_1^T, U_2 V_2^T, U_3 V_3^T) \\ (A_{S_1} U_2 V_2^T B_{S_1}^T + A_{T_1} U_2 V_2^T B_{T_1}^T + A_{S_2} U_3 V_3^T B_{S_2}^T + A_{T_2} U_3 V_3^T B_{T_2}^T, \\ A_{S_1}^T U_1 V_1^T B_{S_1} + A_{T_1}^T U_1 V_1^T B_{T_1}, A_{S_2}^T U_1 V_1^T B_{S_2} + A_{T_2}^T U_1 V_1^T A_{T_2}) \rangle. \end{array} \tag{54}$$

The first order optimality conditions can be derived from the Lagrangian $L(U_i, V_i, F_i, G_i)$, which is given by

$$\begin{array}{rcl} \langle U_1^T A_{S_1} U_2, V_1^T B_{S_1} V_2 \rangle & + & \langle U_1^T A_{T_1} U_2, V_1^T B_{T_1} V_2 \rangle + \\ \langle U_1^T A_{S_2} U_3, V_1^T B_{S_2} V_3 \rangle + \langle U_1^T A_{T_2} U_3, V_1^T B_{T_2} V_3 \rangle & + & \langle F_1, I - U_1^T U_1 \rangle + \langle G_1, I - V_1^T V_1 \rangle + \\ \langle F_2, I - U_2^T U_2 \rangle + \langle G_2, I - V_2^T V_2 \rangle & + & \langle F_3, I - U_3^T U_3 \rangle + \langle G_3, I - V_3^T V_3 \rangle \end{array}$$

where G_i and F_i are symmetric matrices of Lagrange multipliers which can be chosen equal to a diagonal matrix Λ_i. The conditions then become :

$$\begin{array}{rcl} U_1 \Lambda_1 V_1^T & = & A_{S_1} U_2 V_2^T B_{S_1}^T + A_{T_1} U_2 V_2^T B_{T_1}^T + A_{S_2} U_3 V_3^T B_{S_2}^T + A_{T_2} U_3 V_3^T B_{T_2}^T, \\ U_2 \Lambda_2 V_2^T & = & A_{S_1}^T U_1 V_1^T B_{S_1} + A_{T_1}^T U_1 V_1^T B_{T_1}, \\ U_3 \Lambda_3 V_3^T & = & A_{S_2}^T U_1 V_1^T B_{S_2} + A_{T_2}^T U_1 V_1^T B_{T_2}, \end{array}$$

which again leads to iterations $(U_i V_i^T)_+ = \phi_i(U_i V_i^T)$, $i = 1, 2, 3$ given by

$$\begin{array}{l} U_{1+} \Sigma_1 V_{1+}^T = SVD_k(A_{S_1} U_2 V_2^T B_{S_1}^T + A_{T_1} U_2 V_2^T B_{T_1}^T + A_{S_2} U_3 V_3^T B_{S_2}^T + A_{T_2} U_3 V_3^T B_{T_2}^T + s U_1 V_1^T), \\ U_{2+} \Sigma_2 V_{2+}^T = SVD_k(A_{S_1}^T U_1 V_1^T B_{S_1} + A_{T_1}^T U_1 V_1^T B_{T_1} + s U_2 V_2^T), \\ U_{3+} \Sigma_3 V_{3+}^T = SVD_k(A_{S_2}^T U_1 V_1^T B_{S_2} + A_{T_2}^T U_1 V_1^T B_{T_2} + s U_3 V_3^T), \end{array}$$

and which converge to a stationary point $(U_1 V_1^T, U_2 V_2^T, U_3 V_3^T)$ of (54) provided the diagonal matrices $\Sigma_i := \Lambda_i + s I_k$ are positive.

Example 3 The following small example shows the use of projected similarity matrices. We created two graphs A and B which are equal to each other up to a labeling of nodes and edges (see Fig. 5), i.e. there exist permutation matrices P_n, P_{e1} and P_{e2}, applied to the nodes and two sets of edges of graph A, respectively, in order to yield graph B:

$$(1, 2, 3, 4, 5, 6, 7, 8, 9, 10) \xrightarrow{P_n} (10, 7, 6, 9, 2, 5, 1, 8, 3, 4),$$

$$(1, 2, 3, 4, 5, 6) \xrightarrow{P_{e1}} (3, 2, 5, 1, 6, 4),$$

$$(7, 8, 9, 10, 11, 12, 13) \xrightarrow{P_{e2}} (7, 12, 13, 9, 10, 8, 11).$$

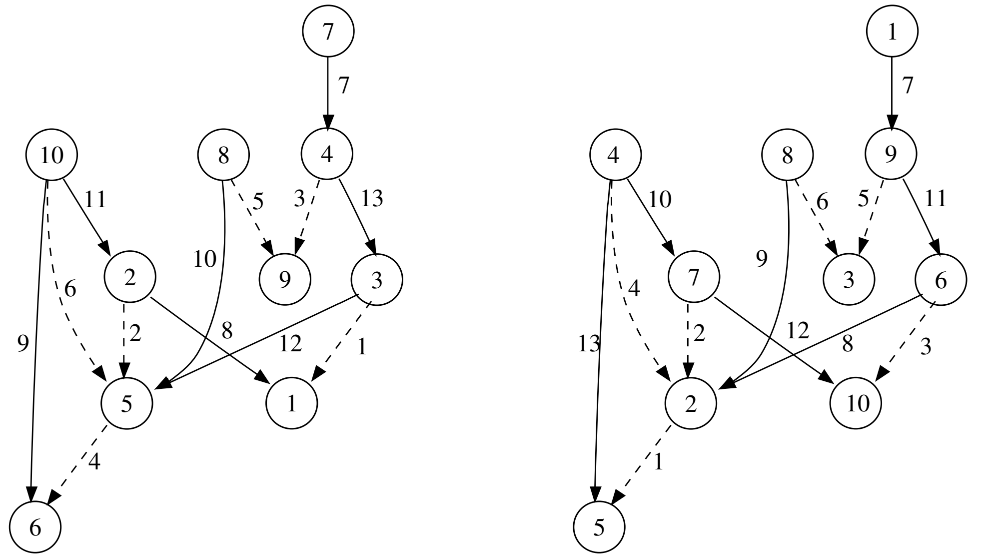

Figure 5: Permuted graphs A and B

Therefore, the source-edge and terminal-edge matrices of A and B must be permutations of each other as well:

$$B_S = P_n \begin{bmatrix} A_{S_1} & A_{S_2} \end{bmatrix} \begin{bmatrix} P_{e1} \\ P_{e2} \end{bmatrix}, \quad B_T = P_n \begin{bmatrix} A_{T_1} & A_{T_2} \end{bmatrix} \begin{bmatrix} P_{e1} \\ P_{e2} \end{bmatrix} \tag{55}$$

and the projected similarity matrices must then be related as follows:

$$V_1 = P_n U_1, \quad U_2 = P_{e1} V_2, \quad U_3 = P_{e2} V_3. \tag{56}$$

Our algorithm managed to reconstruct the exact permutation from the matrices U_i and V_i because these matrices did not contain any repeated rows. In such case, it suffices to order the rows of these matrices in order to reconstruct the permutation. This example is a particular instance of the graph isomorphism problem, for which there is no polynomial time algorithm yet, while our algorithm seems to solve the problem. But when the matrices U_i and V_i have repeated rows, one has to check all possible permutations of these repeated rows, leading to a combinatorial problem. In such a case, one may help the algorithm by assigning a few pairs of nodes for which the correspondence is known. This can be achieved by assigning them a separate color thereby forcing the match. This illustrates the use of separate colors in the similarity matrix problem. We presented in this paper a few academic examples in order to illustrate the concept of similarity matrix for colored graphs, but we refer to [1], [4] for a few larger problems.

6 Concluding remarks

In this paper we presented a number of extensions of the notion of similarity matrix defined in [1] as a tool to compare nodes between two graphs. In [7] this was already extended to compare edges between two graphs. In both papers, the definition was made based on a linear map $\mathcal{M}(S)$ of a particular matrix.

We extended this here to matrix ℓ-tuples and showed how a particular linear map $\mathcal{M}(X_1, \cdots, X_\ell)$ can be used to introduce the notion of similarity matrix between colored graphs, where either the nodes or edges have different "colors" or "types". We then extended this further to allow for low rank similarity matrices, by combining the linear map $\mathcal{M}(X_1, \cdots, X_\ell)$ with projections on low rank ℓ-tuples. It is shown in [3] that this has the additional advantage of making the iterative algorithm particularly efficient since the updating formulas of the iteration will have a complexity that is linear in the dimensions of the graphs, and quadratic in the rank of the matrices. We are currently investigating the convergence of the iterative schemes described in this paper. We hope to be able to prove global linear convergence of the projected iterative schemes to local maxima of the objective function, and numerous experiments have confirmed this.

References

[1] V. Blondel, A. Gajardo, M. Heymans, P. Sennelart and P. Van Dooren, Measure of similarity between graph vertices. Applications to synonym extraction and web searching". *SIAM Review*, 46:647-666, 2004.

[2] T. Caelli and S. Kosinov, An eigenspace projection clustering method for inexact graph matching, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 26(4):515-519, 2004.

[3] C. Fraikin, Y. Nesterov and P. Van Dooren, Optimizing the coupling between two isometric projections of matrices, *SIAM J. Matrix Anal. Appl.*, 30(1):324-345, 2008.

[4] C. Fraikin and P. Van Dooren, *Graph matching with type constraints*, ECC 07, Greece, July 2-5, 2007.

[5] C. Fraikin and P. Van Dooren, Graph matching with type constraints on nodes and edges, *Dagstuhl Seminar N 07071 on Web Information Retrieval and Linear Algebra Algorithms*, Feb. 2007.

[6] A.J. Laub, *Matrix Analysis for Scientists and Engineers*, SIAM Publ., Philadelphia, 2004.

[7] L. Zager and G. Verghese, Graph similarity scoring and matching. *Applied Mathematics Letters*, 21(1):8694, 2008.

Université catholique de Louvain, CESAME
4 Av. G. Lemaître, Louvain-la-Neuve, BELGIUM
email: paul.vandooren@uclouvain.be