Concurrent Alignment of Multiple Anonymized Social Networks with Generic Stable Matching

Jiawei Zhang, Qianyi Zhan and Philip S. Yu

Abstract Users nowadays are normally involved in multiple (usually more than two) online social networks simultaneously to enjoy more social network services. Some of the networks that users are involved in can share common structures either due to the analogous network construction purposes or because of the similar social network characteristics. However, the social network datasets available in research are usually pre-anonymized and accounts of the shared users in different networks are mostly isolated without any known connections. In this paper, we want to identify such connections between the shared users' accounts in multiple social networks (which are called the anchor links), and the problem is formally defined as the M-NASA (Multiple Anonymized Social Networks Alignment) problem. M-NASA is very challenging to address due to (1) the lack of known anchor links to build models, (2) the studied networks are anonymized, where no users' personal profile or attribute information is available, and (3) the "*transitivity law*" and the "*one-to-one property*" based constraints on anchor links. To resolve these challenges, a novel two-phase network alignment framework UMA (Unsupervised Multi-network Alignment) is proposed in this paper. Extensive experiments conducted on multiple real-world partially aligned social networks demonstrate that UMA can perform very well in solving the M-NASA problem.

This paper is an extended version of PNA: Partial Network Alignment with Generic Stable Matching accepted by IEEE IRI 2015 [32].

J. Zhang (✉) · P.S. Yu
University of Illinois at Chicago, Chicago, IL, USA
e-mail: jzhan9@uic.edu

Q. Zhan
National Laboratory for Novel Software Technology,
Nanjing University, Nanjing, China
e-mail: zhanqianyi@gmail.com

P.S. Yu
Institute for Data Science, Tsinghua University, Beijing, China
e-mail: psyu@cs.uic.edu

T. Bouabana-Tebibel and S.H. Rubin (eds.), *Theoretical Information Reuse and Integration*, Advances in Intelligent Systems and Computing 446,
DOI 10.1007/978-3-319-31311-5_8

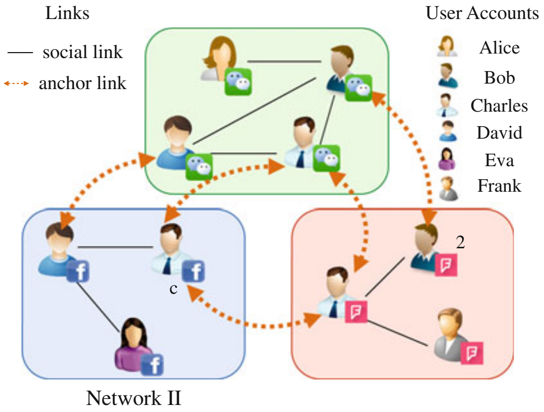
Keywords Partial network alignment · Multiple heterogeneous social networks · Data mining

1 Introduction

As proposed in [13], people nowadays are normally involved in multiple (usually *more than two*) social networks simultaneously to enjoy more social network services. Many of these networks can share common structure information (e.g., friendship connections) due to either the analogous network establishing purposes or because of similar network characteristics. Meanwhile, social network data available for research is usually anonymized for privacy concerns [2], where users' personal profile and attribute information (e.g., names, hometown, gender and age) is either removed or replaced with meaningless unique identifiers, and the accounts of the shared users in these anonymized social networks are mostly isolated without any correspondence relationships. In this paper, we want to study the "*Multiple Anonymized Social Networks Alignment*" (M-NASA) problem to identify such correspondence relationships between the shared users' accounts across multiple anonymized social networks.

By following terminology definitions used in existing aligned networks studies [13, 37], social networks sharing common users are defined as "*partially aligned networks*", where the shared users are named as "*anchor users*" [37] and the correspondence relationships between anchor users' accounts in different networks are called "*anchor links*" [13]. The M-NASA problem studied in this paper aims at identifying the anchor links among multiple anonymized social networks. To help illustrate the M-NASA problem more clearly, we also give an example in Fig. 1, which involves 3 different social networks (i.e., networks I, II and III). Users in these 3 networks are all anonymized and their names are replaced with randomly

Fig. 1 An example of multiple anonymized partially aligned social networks



generated identifiers. Each pair of these 3 anonymized networks can actually share some common users, e.g., "David" participates in both networks I and II simultaneously, "Bob" is using networks I and III concurrently, and "Charles" is involved in all these 3 networks at the same time. Besides these shared anchor users, in these 3 partially aligned networks, some users are involved in one single network only (i.e., the non-anchor users [37]), e.g., "Alice" in network I, "Eva" in network II and "Frank" in network III. The M-NASA problem studied in this paper aims at discovering the anchor links (i.e., the dashed bi-directional orange lines) connecting anchor users across these 3 social networks.

The M-NASA problem is of great importance for online social networks, as it can be the prerequisite for various cross-site social network services, e.g., cross-network link transfer [37], inter-network community detection [34], and viral marketing across networks [31]. With the information transferred from developed social networks, link prediction models proposed in [37] can overcome the *cold-start problem* effectively; constrained by the anchor links, community detection across aligned networks can refine the community structures of each social network mutually [10, 34]; via the anchor users, information can diffuse not only within but also across networks which will lead to broader impact and activate more users in viral marketing [31].

Besides its importance, the M-NASA problem is a novel problem and totally different from existing works, e.g., (1) *supervised anchor link inference across social networks* [13], which focuses on inferring the anchor links between *two* social networks with a supervised learning model; (2) *network matching* [12, 18], which explores various heuristics to match *two* networks based the known existence probabilities of potential correspondence relationships; (3) *entity resolution* [4], which aims at discovering multiple references to the same entity in *one single database* with a relational clustering algorithm; and (4) *cross-media user identification* [30], which matches users between *two* networks based on various node attribute information generated by users' social activities.

M-NASA differs from all these related works in various aspects: (1) M-NASA is a general multi-network alignment problem and can be applied to align either two [13] or more than two social networks; (2) M-NASA is an *unsupervised* network alignment problem and requires no known anchor links (which are also extremely expensive to obtain in the real world); (3) no extra heuristics will be needed and used in the M-NASA problem; and (4) no information about the potential anchor links nor their existence probabilities is required; and (5) social networks studied in M-NASA are anonymized and involve structure information only but no attribute information.

Besides these easily distinguishable distinctions mentioned above, another significant difference of M-NASA from existing *two* network alignment problems is due to the "*transitivity law*" that anchor links follow. In traditional set theory [15], a relation $\mathcal{R}$ is defined to be a *transitive relation* in domain $\mathcal{X}$ iff $\forall a, b, c \in \mathcal{X}, (a, b) \in \mathcal{R} \wedge (b, c) \in \mathcal{R} \rightarrow (a, c) \in \mathcal{R}$. If we treat the union of user account sets of all these social networks as the target domain $\mathcal{X}$ and treat anchor links as the relation $\mathcal{R}$, then anchor links depict a "*transitive relation*" among users across networks. We can take the networks shown in Fig. 1 as an example. Let u be a user involved in networks I, II and III simultaneously, whose accounts in these networks are u^I, u^{II} and u^{III} respectively. If anchor links (u^I, u^{II}) and (u^{II}, u^{III}) are identified in aligning networks (I, II) and networks (II, III) respectively (i.e., u^I, u^{II} and u^{III} are discovered to be the same user), then anchor link (u^I, u^{III}) should also exist in the alignment result of networks (I, III) as well. In the M-NASA problem, we need to guarantee the inferred anchor links can meet the *transitivity law*.

In addition to its importance and novelty, the M-NASA problem is very difficult to solve due to the following challenges:

- *unsupervised network alignment*: No existing anchor links are available between pairs of social networks in the M-NASA problem and inferring anchor links between social networks in an unsupervised manner is very challenging.
- *anonymized network alignment*: Networks studied in this paper are all pre-anonymized, where no attribute information indicating users' personal characteristics exists. It makes the M-NASA problem much tougher to address.
- *transitivity law preservation and utilization*: Anchor links among social networks follow the "transitivity law". How to (1) preserve such a property of anchor links, and (2) utilize such a property to improve the multiple networks partial alignment is still an open problem in this context so far.
- *one-to-one constraint on anchor links*: Anchor links have an inherent *one-to-one* constraint [13], i.e., each user can have at most one account in each social network, which will pose extra challenges on solving the M-NASA problem. (The case that users have multiple accounts in one network can be resolved with method introduced in [27], where these duplicated accounts can be aggregated in advance to form one unique virtual account and the constraint on anchor links connecting these virtual accounts will still be "one-to-one".)

To solve the M-NASA problem, a novel network alignment framework UMA (Unsupervised Multi-network Alignment) is proposed in this paper. UMA addresses the M-NASA problem with two steps: (1) unsupervised transitive anchor link inference across multi-networks, and (2) transitive multi-network matching to maintain the constraints on anchor links. In step (1), UMA infers sets of potential anchor links with unsupervised learning techniques by minimizing the *friendship inconsistency* and preserving the *alignment transitivity* property across networks. In step (2), UMA keeps the one-to-one constraint on anchor links by selecting those with high confidence scores but no blocking pairs, while maintaining the *matching transitivity* property at the same time. The above mentioned new concepts will be introduced in Sect. 3.

The rest of this paper is organized as follows. In Sect. 2, we define some important concepts and the M-NASA problem. Method UMA will be introduced in Sect. 3 and evaluated in Sect. 4. Finally, we introduce the related works in Sect. 5 and conclude this paper in Sect. 6.

2 Problem Formulation

In this section, we will follow the definitions of "*aligned networks*" and "*anchor links*" proposed in [37], which are introduced as follows.

Definition 1 (*Anonymized Social Network*) An anonymized social network can be represented as graph $G = (\mathcal{U}, \mathcal{E})$, where $\mathcal{U}$ denotes the set of users in the network and $\mathcal{E}$ represents the *social links* among users. Users' profile and attribute information in G has all been deleted to protect individuals' privacy.

Definition 2 (*Multiple Aligned Social Networks*) Multiple aligned social networks can be represented as $\mathcal{G} = ((G^{(1)}, G^{(2)}, \dots, G^{(n)}), (\mathcal{A}^{(1,2)}, \mathcal{A}^{(1,3)}, \dots, \mathcal{A}^{(n-1,n)}))$, where $G^{(i)}, i \in \{1, 2, \dots, n\}$ represents an anonymized social network and $\mathcal{A}^{(i,j)}, i, j \in \{1, 2, \dots, n\}$ denotes the set of undirected *anchor links* between networks $G^{(i)}$ and $G^{(j)}$.

Definition 3 (*Anchor Links*) Given two social networks $G^{(i)}$ and $G^{(j)}$, link $(u^{(i)}, v^{(j)})$ is an *anchor link* between $G^{(i)}$ and $G^{(j)}$ iff $(u^{(i)} \in \mathcal{U}^{(i)}) \wedge (v^{(j)} \in \mathcal{U}^{(j)}) \wedge (u^{(i)}$ and $v^{(j)}$ are accounts of the same user), where $\mathcal{U}^{(i)}$ and $\mathcal{U}^{(j)}$ are the user sets of $G^{(i)}$ and $G^{(j)}$ respectively.

Social networks studied in this paper are all partially aligned [37] and the formal definitions of the concepts like "anchor users", "non-anchor users", "full alignment", "partial alignment" are available in [37].

Based on the above definitions, the M-NASA problem can be formulated as follows:

The M-NASA Problem: Given the n isolated anonymized social networks $\{G^{(1)}, G^{(2)}, \dots, G^{(n)}\}$, the M-NASA problem aims at discovering the anchor links among these n networks, i.e., the anchor link sets $\mathcal{A}^{(1,2)}, \mathcal{A}^{(1,3)}, \dots, \mathcal{A}^{(n-1,n)}$. Networks $G^{(1)}, G^{(2)}, \dots, G^{(n)}$ are partially aligned and the constraint on anchor links in $\mathcal{A}^{(1,2)}, \mathcal{A}^{(1,3)}, \dots, \mathcal{A}^{(n-1,n)}$ is *one-to-one*, which also need to follow the *transitivity law*.

3 Proposed Method

Based on observation about the "transitivity property" of anchor links, in this section, we will introduce the framework UMA to address the M-NASA problem: in Sect. 3.1, we formulate the unsupervised pairwise network alignment based on friendship connection information as an optimization problem; integrated multi-network alignment will be introduced in Sect. 3.2, where an extra constraint called *alignment transitivity* penalty is added to the objective function; the joint optimization function will be solved in Sect. 3.3 by relaxing its constraints, and the redundant non-existing anchor links introduced by such relaxation will be pruned with *transitive network matching* in Sect. 3.4.

3.1 Unsupervised Pairwise Network Alignment

Anchor links between any two given networks $G^{(i)}$ and $G^{(j)}$ actually define an *one-to-one* mapping (of users and social links) between $G^{(i)}$ and $G^{(j)}$. To evaluate the quality of different inferred mapping (i.e., the inferred anchor links), we introduce the concepts of cross-network *Friendship Consistency/Inconsistency* in this paper. The optimal inferred anchor links are those which can maximize the *Friendship Consistency* (or minimize the *Friendship Inconsistency*) across networks.

For any anonymized social network $G = (\mathcal{U}, \mathcal{E})$, the social connections among users in it can be represented with the *social adjacency matrix*.

Definition 4 (*Social Adjacency Matrix*) Given network $G = (\mathcal{U}, \mathcal{E})$, its *social adjacency matrix* can be represented with binary matrix $\mathbf{S} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{U}|}$ and entry $\mathbf{S}(l, m) = 1$ iff the corresponding social link $(u_l, u_m) \in \mathcal{E}$, where u_l and u_m are users in G.

Based on the above definition, given two partially aligned social networks $G^{(i)} = (\mathcal{U}^{(i)}, \mathcal{E}^{(i)})$ and $G^{(j)} = (\mathcal{U}^{(j)}, \mathcal{E}^{(j)})$, we can represent their corresponding *social adjacency* matrices to be $\mathbf{S}^{(i)} \in \mathbb{R}^{|\mathcal{U}^{(i)}| \times |\mathcal{U}^{(i)}|}$ and $\mathbf{S}^{(j)} \in \mathbb{R}^{|\mathcal{U}^{(j)}| \times |\mathcal{U}^{(j)}|}$ respectively.

Meanwhile, let $\mathcal{A}^{(i,j)}$ be the set of undirected anchor links to be inferred connecting networks $G^{(i)}$ and $G^{(j)}$, based on which, we can construct the corresponding *binary transitional matrix* $\mathbf{T}^{(i,j)}$ between networks $G^{(i)}$ and $G^{(j)}$, where users corresponding to rows and columns of $\mathbf{T}^{(i,j)}$ are of the same order as those of $\mathbf{S}^{(i)}$ and $\mathbf{S}^{(j)}$ respectively.

Definition 5 (*Binary Transitional Matrix*) Given anchor link set $\mathcal{A}^{(i,j)} \subset \mathcal{U}^{(i)} \times \mathcal{U}^{(j)}$ between networks $G^{(i)}$ and $G^{(j)}$, the *binary transitional matrix* from $G^{(i)}$ to $G^{(j)}$ can be represented as $\mathbf{T}^{(i,j)} \in \{0, 1\}^{|\mathcal{U}^{(i)}| \times |\mathcal{U}^{(j)}|}$, where $\mathbf{T}^{(i,j)}(l, m) = 1$ iff link $(u_l^{(i)}, u_m^{(j)}) \in \mathcal{A}^{(i,j)}$, $u_l^{(i)} \in \mathcal{U}^{(i)}$, $u_m^{(j)} \in \mathcal{U}^{(j)}$.

The *binary transitional matrix* from $G^{(j)}$ to $G^{(i)}$ can be defined in a similar way, which can be represented as $\mathbf{T}^{(j,i)} \in \{0, 1\}^{|\mathcal{U}^{(j)}| \times |\mathcal{U}^{(i)}|}$, where $(\mathbf{T}^{(i,j)})^\top = \mathbf{T}^{(j,i)}$ as the anchor links between $G^{(i)}$ and $G^{(j)}$ are undirected. Considering that anchor links have an inherent *one-to-one* constraint, each row and each column of the *binary transitional matrices* $\mathbf{T}^{(i,j)}$ and $\mathbf{T}^{(j,i)}$ should have at most one entry filled with 1, which will constrain the inference space of potential *binary transitional matrices* $\mathbf{T}^{(i,j)}$ and $\mathbf{T}^{(j,i)}$ greatly.

Binary transitional matrix $\mathbf{T}^{(i,j)}$ defines a mapping of users from network $G^{(i)}$ to $G^{(j)}$, i.e., $\mathbf{T}^{(i,j)} : \mathcal{U}^{(i)} \to \mathcal{U}^{(j)}$. Besides the user nodes, the social links in network $G^{(i)}$ can also be projected to network $G^{(j)}$ via the binary transitional matrices $\mathbf{T}^{(i,j)}$ and $\mathbf{T}^{(j,i)}$: the *social adjacency matrix* $\mathbf{S}^{(i)}$ being mapped from $G^{(i)}$ to $G^{(j)}$ can be represented as $\mathbf{T}^{(j,i)}\mathbf{S}^{(i)}\mathbf{T}^{(i,j)}$ (i.e., $(\mathbf{T}^{(i,j)})^\top\mathbf{S}^{(i)}\mathbf{T}^{(i,j)}$). Furthermore, considering social networks $G^{(i)}$ and $G^{(j)}$ share significant community structure overlaps, the friendship connections mapped from $G^{(i)}$ to $G^{(j)}$ (i.e., $(\mathbf{T}^{(i,j)})^\top\mathbf{S}^{(i)}\mathbf{T}^{(i,j)}$) should be consistent with those in $G^{(j)}$ (i.e., $\mathbf{S}^{(j)}$), which can be quantified as the following cross-network *friendship consistency* formally [14].

Definition 6 (*Friendship Consistency/Inconsistency*) The *friendship consistency* between network $G^{(i)}$ and $G^{(j)}$ introduced by the cross-network mapping $\mathbf{T}^{(i,j)}$ is defined as number of shared social links between those mapped from $G^{(i)}$ and the social links in $G^{(j)}$ originally.

Meanwhile, we can define the *friendship inconsistency* as the number of non-shared social links between those mapped from $G^{(i)}$ and those in $G^{(j)}$. Based on the inferred *anchor transitional matrix* $\mathbf{T}^{(i,j)}$, the introduced *friendship inconsistency* between matrices $(\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)}$ and $\mathbf{S}^{(j)}$ can be represented as:

$$\left\| (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} - \mathbf{S}^{(j)} \right\|_F^2,$$

where $\|\cdot\|_F$ denotes the Frobenius norm. And the optimal *binary transitional matrix* $\bar{\mathbf{T}}^{(i,j)}$, which can lead to the minimum *friendship inconsistency* can be represented as

$$\begin{aligned} \bar{\mathbf{T}}^{(i,j)} &= \arg\min\nolimits_{\mathbf{T}^{(i,j)}} \left\| (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} - \mathbf{S}^{(j)} \right\|_F^2 \\ s.t. \quad & \mathbf{T}^{(i,j)} \in \{0, 1\}^{|\mathcal{U}^{(i)}| \times |\mathcal{U}^{(j)}|}, \\ & \mathbf{T}^{(i,j)} \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1}, \\ & (\mathbf{T}^{(i,j)})^\top \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1}, \end{aligned}$$

where the last two equations are added to maintain the *one-to-one* constraint on anchor links and $\mathbf{X} \preccurlyeq \mathbf{Y}$ iff $\mathbf{X}$ is of the same dimensions as $\mathbf{Y}$ and every entry in $\mathbf{X}$ is no greater than the corresponding entry in $\mathbf{Y}$.

3.2 Transitive Integrate Network Alignment

Isolated network alignment can work well in addressing the alignment problem of two social networks. However, in the M-NASA problem studied in this paper, multiple (more than two) social networks are to be aligned simultaneously. Besides minimizing the *friendship inconsistency* between each pair of networks, the *transitivity* property of anchor links also needs to be preserved in the transitional matrices inference.

The *transitivity* property should holds for the alignment of any n networks, where the minimum of n is 3. To help illustrate the *transitivity property* more clearly and simplify the descriptions of the model, we will use 3 network alignment as an example to introduce the M-NASA problem, which can be easily generalized to the case of n networks alignment. Let $G^{(i)}$, $G^{(j)}$ and $G^{(k)}$ be 3 social networks to be aligned concurrently. To accommodate the alignment results and preserve the *transitivity* property, we introduce the following *alignment transitivity penalty*:

Definition 7 (*Alignment Transitivity Penalty*) Let $\mathbf{T}^{(i,j)}$, $\mathbf{T}^{(j,k)}$ and $\mathbf{T}^{(i,k)}$ be the inferred binary transitional matrices from $G^{(i)}$ to $G^{(j)}$, from $G^{(j)}$ to $G^{(k)}$ and from $G^{(i)}$ to $G^{(k)}$ respectively among these 3 networks. The *alignment transitivity penalty* $C(\{G^{(i)}, G^{(j)}, G^{(k)}\})$ introduced by the inferred transitional matrices can be quantified as the number of inconsistent social links being mapped from $G^{(i)}$ to $G^{(k)}$ via two different alignment paths $G^{(i)} \rightarrow G^{(j)} \rightarrow G^{(k)}$ and $G^{(i)} \rightarrow G^{(k)}$, i.e.,

$$C(\{G^{(i)}, G^{(j)}, G^{(k)}\}) = \left\| (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} - (\mathbf{T}^{(i,k)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,k)} \right\|_F^2 .$$

Alignment transitivity penalty is a general penalty concept and can be applied to n networks $\{G^{(1)}, G^{(2)}, \ldots, G^{(n)}\}, n \geq 3$ as well, which can be defined as the summation of penalty introduced by any three networks in the set, i.e.,

$$C(\{G^{(1)}, G^{(2)}, \ldots, G^{(n)}\}) = \sum_{\forall \{G^{(i)}, G^{(j)}, G^{(k)}\} \subset \{G^{(1)}, G^{(2)}, \ldots, G^{(n)}\}} C(\{G^{(i)}, G^{(j)}, G^{(k)}\}).$$

The optimal *binary transitional matrices* $\bar{\mathbf{T}}^{(i,j)}$, $\bar{\mathbf{T}}^{(j,k)}$ and $\bar{\mathbf{T}}^{(k,i)}$ which can minimize friendship inconsistency and the *alignment transitivity penalty* at the same time can be represented to be

$$\begin{aligned}
\bar{\mathbf{T}}^{(i,j)}, \bar{\mathbf{T}}^{(j,k)}, \bar{\mathbf{T}}^{(k,i)} = \arg\min\nolimits_{\mathbf{T}^{(i,j)}, \mathbf{T}^{(j,k)}, \mathbf{T}^{(k,i)}} & \left\| (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} - \mathbf{S}^{(j)} \right\|_F^2 \\
& + \left\| (\mathbf{T}^{(j,k)})^\top \mathbf{S}^{(j)} \mathbf{T}^{(j,k)} - \mathbf{S}^{(k)} \right\|_F^2 + \left\| (\mathbf{T}^{(k,i)})^\top \mathbf{S}^{(k)} \mathbf{T}^{(k,i)} - \mathbf{S}^{(i)} \right\|_F^2 \\
& + \alpha \cdot \left\| (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} - \mathbf{T}^{(k,i)} \mathbf{S}^{(i)} (\mathbf{T}^{(k,i)})^\top \right\|_F^2 \\
s.t.\ & \mathbf{T}^{(i,j)} \in \{0,1\}^{|\mathcal{U}^{(i)}| \times |\mathcal{U}^{(j)}|}, \mathbf{T}^{(j,k)} \in \{0,1\}^{|\mathcal{U}^{(j)}| \times |\mathcal{U}^{(k)}|} \\
& \quad \mathbf{T}^{(k,i)} \in \{0,1\}^{|\mathcal{U}^{(k)}| \times |\mathcal{U}^{(i)}|} \\
& \quad \mathbf{T}^{(i,j)} \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1}, (\mathbf{T}^{(i,j)})^\top \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1}, \\
& \quad \mathbf{T}^{(j,k)} \mathbf{1}^{|\mathcal{U}^{(k)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1}, (\mathbf{T}^{(j,k)})^\top \mathbf{1}^{|\mathcal{U}^{(j)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(k)}| \times 1}, \\
& \quad \mathbf{T}^{(k,i)} \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(k)}| \times 1}, (\mathbf{T}^{(k,i)})^\top \mathbf{1}^{|\mathcal{U}^{(k)}| \times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(i)}| \times 1},
\end{aligned}$$

where parameter α denotes the weight of the alignment transitivity penalty term, which is set as 1 by default in this paper.

3.3 Relaxation of the Optimization Problem

The above objective function aims at obtaining the *hard* mappings among users across different networks and entries in all these *transitional matrices* are binary, which can lead to a fatal drawback: *hard assignment* can be neither possible nor realistic for networks with star structures as proposed in [14] and the hard subgraph isomorphism [16] is NP-hard.

To overcome such a problem, we propose to relax the binary constraint of entries in transitional matrices to allow them to be real values within range [0, 1]. Each entry in the transitional matrix represents a probability, denoting the confidence of certain user-user mapping across networks. Such a relaxation can make the *one-to-one* constraint no longer hold (which will be addressed with transitive network matching in the next subsection) as multiple entries in rows/columns of the transitional matrix can have non-zero values. To limit the existence of non-zero entries in the transitional matrices, we replace the one-to-one constraint, e.g.,

$$\mathbf{T}^{(k,i)}\mathbf{1}^{|\mathcal{U}^{(i)}|\times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(k)}|\times 1}, (\mathbf{T}^{(k,i)})^\top \mathbf{1}^{|\mathcal{U}^{(k)}|\times 1} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(i)}|\times 1}$$

with *sparsity constraints*

$$\left\|\mathbf{T}^{(k,i)}\right\|_0 \leq t$$

instead, where term $\|\mathbf{T}\|_0$ denotes the L_0 norm of matrix $\mathbf{T}$, i.e., the number of non-zero entries in $\mathbf{T}$, and t is a small positive number to limit the non-zero entries in the matrix (i.e., the sparsity). Furthermore, in this paper, we propose to add term $\|\mathbf{T}\|_0$ to the minimization objective function, as it can be hard to determine the value of t in the constraint.

Based on the above relaxations, we can obtain the new objective function (available in the Appendix), which involves 3 variables $\mathbf{T}^{(i,j)}$, $\mathbf{T}^{(j,k)}$ and $\mathbf{T}^{(k,i)}$ simultaneously, obtaining the joint optimal solution for which at the same time is very hard and time consuming. We propose to address the above objective function by fixing two variables and updating the other variable alternatively with gradient descent method [1]. As proposed in [14], if during the alternating updating steps, the entries of the transitional matrices become invalid (i.e., values less than 0 or greater than 1), we apply the projection technique introduced in [14] to project (1) negative entries to 0, and (2) entries greater than 1 to 1 instead. With these processes, the updating equations of matrices $\mathbf{T}^{(i,j)}$, $\mathbf{T}^{(j,k)}$, $\mathbf{T}^{(k,i)}$ at step $t+1$ are given as follows

$$\mathbf{T}^{(i,j)}(t+1) = \mathbf{T}^{(i,j)}(t) - \eta^{(i,j)} \frac{\partial \mathcal{L}\left(\mathbf{T}^{(i,j)}(t), \mathbf{T}^{(j,k)}(t), \mathbf{T}^{(k,i)}(t), \beta, \gamma, \theta\right)}{\partial \mathbf{T}^{(i,j)}},$$

$$\mathbf{T}^{(j,k)}(t+1) = \mathbf{T}^{(j,k)}(t) - \eta^{(j,k)} \frac{\partial \mathcal{L}\left(\mathbf{T}^{(i,j)}(t+1), \mathbf{T}^{(j,k)}(t), \mathbf{T}^{(k,i)}(t), \beta, \gamma, \theta\right)}{\partial \mathbf{T}^{(j,k)}},$$

$$\mathbf{T}^{(k,i)}(t+1) = \mathbf{T}^{(k,i)}(t) - \eta^{(k,i)} \frac{\partial \mathcal{L}\left(\mathbf{T}^{(i,j)}(t+1), \mathbf{T}^{(j,k)}(t+1), \mathbf{T}^{(k,i)}(t), \beta, \gamma, \theta\right)}{\partial \mathbf{T}^{(k,i)}}.$$

Such an iteratively updating process will stop when all *transitional matrices* converge. In the updating equations, $\eta^{(i,j)}$, $\eta^{(j,k)}$ and $\eta^{(k,i)}$ are the gradient descent steps in updating $\mathbf{T}^{(i,j)}$, $\mathbf{T}^{(j,k)}$ and $\mathbf{T}^{(k,i)}$ respectively. The Lagrangian function of the objective function is available in the Appendix.

Meanwhile, considering that $\|\cdot\|_0$ is not differentiable because of its discrete values [29], we will replace the $\|\cdot\|_0$ with the $\|\cdot\|_1$ instead (i.e., the sum of absolute values of all entries). Furthermore, as all the negative entries will be projected to 0, the L_1 norm of transitional matrix $\mathbf{T}$ can be represented as $\left\|\mathbf{T}^{(k,i)}\right\|_1 = \mathbf{1}^\top \mathbf{T}^{(k,i)} \mathbf{1}$ (i.e., the sum of all entries in the matrix). In addition, the Frobenius norm $\|\mathbf{X}\|_F^2$ can be represented with trace $\text{Tr}(\mathbf{X}\mathbf{X}^\top)$. The partial derivatives of function $\mathcal{L}$ with regard to $\mathbf{T}^{(i,j)}$, $\mathbf{T}^{(j,k)}$, and $\mathbf{T}^{(k,i)}$ are given in the Appendix.

3.4 Transitive Generic Stable Matching

Based on the transitive integrated network alignment introduced in the previous sections, we can obtain the confidence scores among users across networks, which can be used to construct user's partner preference list across networks. For instance, if the score of link $(u^{(i)}, v^{(j)})$ is greater than that of link $(u^{(i)}, w^{(j)})$ between networks $G^{(i)}$ and $G^{(j)}$, then we can user $u^{(i)}$ prefers $v^{(j)}$ to $w^{(j)}$.

However, due to the constraint relaxation, the *one-to-one* constraint on the inferred anchor links can no longer hold. In this section, we propose to apply the *transitive network matching* algorithm to help prune the redundant non-existing anchor links introduced by the constraint relaxation.

In this section, we will first briefly talk about the traditional stable matching for two networks, then we will introduce the generic stable matching for two networks. Finally, we will introduce transitive generic stable matching for multiple networks.

3.4.1 Traditional Stable Matching

Meanwhile, as proposed in [13], the *one-to-one* constraint of anchor links across *fully aligned social networks* can be met by pruning extra potential *anchor link candidates* with *traditional stable matching*. In this subsection, we will introduce the concept of traditional *stable matching* briefly.

We first use a toy example in Fig. 2 to illustrate the main idea of our solution. Suppose in Fig. 2a we are given the ranking scores from the transitive integrated network alignment. We can see in Fig. 2b that link prediction methods with a fixed threshold may not be able to predict well, because the predicted links do not satisfy the constraint of one-to-one relationship. Thus one user account in the source network can be linked with multiple accounts in the target network. In Fig. 2c, *weighted maximum matching* methods can find a set of links with maximum sum of weights. However, it is worth noting that the input scores are uncalibrated, so maximum

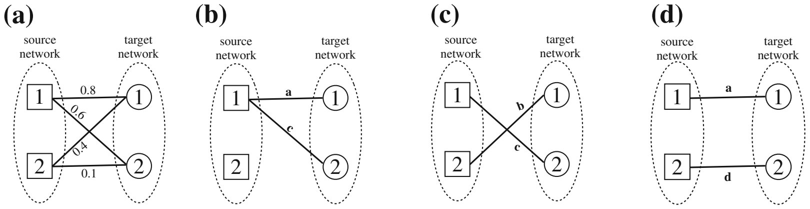


Fig. 2 An example of anchor link inference by different methods. **a** is the input, ranking scores. **b–d** are the results of different methods for anchor link inference. **a** Input scores. **b** Link prediction. **c** Max weight(1:1). **d** UMA(1:1)

weight matching may not be a good solution for anchor link prediction problems. The input scores only indicate the ranking of different user pairs, i.e., the preference relationship among different user pairs.

Here we say 'node x prefers node y over node z', if the score of pair (x, y) is larger than the score of pair (x, z). For example, in Fig. 2c, the weight of pair a, i.e., Score $(a) = 0.8$, is larger than Score $(c) = 0.6$. It shows that user u_i (the first user in the source network) *prefers* v_i over v_j. The problem with the prediction result in Fig. 2c is that, the pair (u_i, v_i) should be more likely to be an anchor link due to the following reasons: (1) u_i prefers v_i over v_j; (2) v_i also prefers u_i over u_j.

Given the user sets $\mathcal{U}^{(1)}$ and $\mathcal{U}^{(2)}$ of two *partially aligned social networks* $G^{(1)}$ and $G^{(2)}$, each user in $\mathcal{U}^{(1)}$(or $\mathcal{U}^{(2)}$) has his preference over users in $\mathcal{U}^{(2)}$(or $\mathcal{U}^{(1)}$). Term $v_j P_{u_i}^{(1)} v_k$ is used to denote that $u_i \in \mathcal{U}^{(1)}$ prefers v_j to v_k for simplicity, where $v_j, v_k \in \mathcal{U}^{(2)}$ and $P_{u_i}^{(1)}$ is the preference operator of $u_i \in \mathcal{U}^{(1)}$. Similarly, we can use term $u_i P_{v_j}^{(2)} u_k$ to denote that $v_j \in \mathcal{U}^{(2)}$ prefers u_i to u_k in $\mathcal{U}^{(1)}$ as well.

Definition 8 (*Matching*) Mapping $\mu : \mathcal{U}^{(1)} \cup \mathcal{U}^{(2)} \rightarrow \mathcal{U}^{(1)} \cup \mathcal{U}^{(2)}$ is defined to be a *matching* iff (1) $|\mu(u_i)| = 1, \forall u_i \in \mathcal{U}^{(1)}$ and $\mu(u_i) \in \mathcal{U}^{(2)}$; (2) $|\mu(v_j)| = 1, \forall v_j \in \mathcal{U}^{(2)}$ and $\mu(v_j) \in \mathcal{U}^{(1)}$; (3) $\mu(u_i) = v_j$ iff $\mu(v_j) = u_i$.

Definition 9 (*Blocking Pair*) A pair (u_i, v_j) is a a *blocking pair* of matching μ if u_i and v_j prefers each other to their mapped partner, i.e., $(\mu(u_i) \neq v_j) \wedge (\mu(v_j) \neq u_i)$ and $(v_j P_{u_i}^{(1)} \mu(u_i)) \wedge (u_i P_{v_j}^{(2)} \mu(v_j))$.

Definition 10 (*Stable Matching*) Given a matching μ, μ is *stable* if there is no *blocking pair* in the matching results [8].

We propose to formulate the anchor link prediction problem as a stable matching problem between user accounts in source network and accounts in target network. Assume that we have two sets of unlabeled user accounts, i.e., $\mathcal{U}^{(1)} = \{u_1, u_2, \ldots, u_{|\mathcal{U}^{(1)}|}\}$ in source network and $\mathcal{U}^{(2)} = \{v_1, v_2, \ldots, v_{|\mathcal{U}^{(2)}|}\}$ in target network. Each u_i has a ranking list or preference list $P(u_i)$ over all the user accounts in target network ($v_i \in \mathcal{U}^{(2)}$) based upon the input scores of different pairs. For example, in Fig. 2a, the preference list of node u_i is $P(u_i) = (v_i, v_j)$, indicating that node v_i is preferred by u_i over v_j. The preference list of node u_j is also $P(u_j) = (v_i, v_j)$.

Similarly, we also build a preference list for each user account in the target network. In Fig. 2a, $P(v_i) = P(v_j) = (u_i, u_j)$.

3.4.2 Generic Stable Matching

Stable matching based method proposed in [13] can only work well in *fully aligned social networks*. However, in the real world, few social networks are fully aligned and lots of users in social networks are involved in one network only, i.e., *non-anchor users*, and they should not be connected by any anchor links. However, traditional *stable matching* method cannot identify these *non-anchor users* and remove the predicted *potential anchor links* connected with them. To overcome such a problem, we will introduce the *generic stable matching* to identify the *non-anchor users* and prune the anchor link results to meet the *one-to-one* constraint.

In UMA, we introduce a novel concept, *self matching*, which allows users to be mapped to themselves if they are discovered to be *non-anchor users*. In other words, we will identify the *non-anchor users* as those who are mapped to themselves in the final matching results.

Definition 11 (*Self Matching*) For the given two partially aligned networks $G^{(1)}$ and $G^{(2)}$, user $u_i \in \mathcal{U}^{(1)}$, can have his preference $P^{(1)}_{u_i}$ over users in $\mathcal{U}^{(2)} \cup \{u_i\}$ and u_i preferring u_i himself denotes that u_i is an *non-anchor user* and prefers to stay unconnected, which is formally defined as *self matching*.

Users in one social network will be matched with either partners in other social networks or themselves according to their preference lists (i.e., from high preference scores to low preference scores). Only partners that users prefer over themselves will be *accepted* finally, otherwise users will be matched with themselves instead.

Definition 12 (*Acceptable Partner*) For a given *matching* $\mu : \mathcal{U}^{(1)} \cup \mathcal{U}^{(2)} \rightarrow \mathcal{U}^{(1)} \cup \mathcal{U}^{(2)}$, the mapped partner of users $u_i \in \mathcal{U}^{(1)}$, i.e., $\mu(u_i)$, is *acceptable* to u_i iff $\mu(u_i) P^{(1)}_{u_i} u_i$.

To cut off the partners with very low *preference scores*, we propose the *partial matching strategy* to obtain the promising partners, who will participate in the matching finally.

Definition 13 (*Partial Matching Strategy*) The *partial matching strategy* of user $u_i \in \mathcal{U}^{(1)}$, i.e., $Q^{(1)}_{u_i}$, consists of the first K the *acceptable partners* in u_i's preference list $P^{(1)}_{u_i}$, which are in the same order as those in $P^{(1)}_{u_i}$, and u_i in the $(K + 1)$th entry of $Q^{(1)}_{u_i}$. Parameter K is called the *partial matching rate* in this paper.

An example is given at the last plot of Fig. 3, where to get the top 2 promising partners for the user, we place the user himself at the 3rd cell in the preference list. All the remaining potential partners will be cut off and only the top 3 users will participate in the final matching.

Based on the concepts of *self matching* and *partial matching strategy*, we define the concepts of *partial stable matching* and *generic stable matching* as follow.

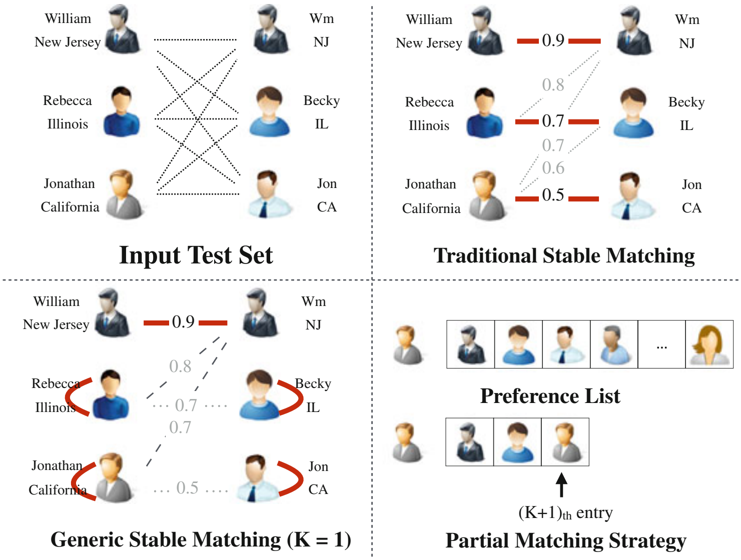


Fig. 3 Partial network alignment with pruning

Definition 14 (*Partial Stable Matching*) For a given *matching* μ, μ is (1) *rational* if $\mu(u_i) Q^{(1)}_{u_i} u_i, \forall u_i \in \mathcal{U}^{(1)}$ and $\mu(v_j) Q^{(2)}_{v_j} v_j, \forall v_j \in \mathcal{U}^{(2)}$, (2) *pairwise stable* if there exist no *blocking pairs* in the matching results, and (3) *stable* if it is both *rational* and *pairwise stable*.

Definition 15 (*Generic Stable Matching*) For a given *matching* μ, μ is a *generic stable matching* iff μ is a *self matching* or μ is a *partial stable matching*.

As example of *generic stable matching* is shown in the bottom two plots of Fig. 3. *Traditional stable matching* can prune most non-existing anchor links and make sure the results can meet *one-to-one* constraint. However, it preserves the anchor links (Rebecca, Becky) and (Jonathan, Jon), which are connecting *non-anchor users*. In *generic stable matching* with parameter $K = 1$, users will be either connected with their most preferred partner or stay *unconnected*. Users "William" and "Wm" are matched as link (William, Wm) has the highest score. "Rebecca" and "Jonathan" will prefer to stay *unconnected* as their most preferred partner "Wm" is connected with "William" already. Furthermore, "Becky" and "Jon" will stay *unconnected* as their most preferred partner "Rebecca" and "Jonathan" prefer to stay *unconnected*. In this way, *generic stable matching* can further prune the non-existing anchor links (Rebecca, Becky) and (Jonathan, Jon).

The *truncated generic stable matching* results can be achieved with the *Generic Gale-Shapley* algorithm as given in Algorithm 1.

Algorithm 1 Generalized Gale-Shapley Algorithm

```
Input: user sets of aligned networks: $\mathcal{U}^{(1)}$ and $\mathcal{U}^{(2)}$.
       classification results of potential anchor links in $\mathcal{L}$
       known anchor links in $\mathcal{A}^{(1,2)}$
       truncation rate $K$
Output: a set of inferred anchor links $\mathcal{L}'$
Initialize the preference lists of users in $\mathcal{U}^{(1)}$ and $\mathcal{U}^{(2)}$ with predicted existence probabilities of links in $\mathcal{L}$ and known anchor links in $\mathcal{A}^{(1,2)}$, whose existence probabilities are 1.0
construct the truncated strategies from the preference lists
Initialize all users in $\mathcal{U}^{(1)}$ and $\mathcal{U}^{(2)}$ as free
$\mathcal{L}' = \emptyset$
while $\exists$ free $u_i^{(1)}$ in $\mathcal{U}^{(1)}$ and $u_i^{(1)}$'s truncated strategy is non-empty do
    Remove the top-ranked account $u_j^{(2)}$ from $u_i^{(1)}$'s truncated strategy
    if $u_j^{(2)}$==$u_i^{(1)}$ then
        $\mathcal{L}' = \mathcal{L}' \cup \{(u_i^{(1)}, u_i^{(1)})\}$
        Set $u_i^{(1)}$ as stay unconnected
    else
        if $u_j^{(2)}$ is free then
            $\mathcal{L}' = \mathcal{L}' \cup \{(u_i^{(1)}, u_j^{(2)})\}$
            Set $u_i^{(1)}$ and $u_j^{(2)}$ as occupied
        else
            $\exists u_p^{(1)}$ that $u_j^{(2)}$ is occupied with.
            if $u_j^{(2)}$ prefers $u_i^{(1)}$ to $u_p^{(1)}$ then
                $\mathcal{L}' = (\mathcal{L}' - \{(u_p^{(1)}, u_j^{(2)})\}) \cup \{(u_i^{(1)}, u_j^{(2)})\}$
                Set $u_p^{(1)}$ as free and $u_i^{(1)}$ as occupied
            end if
        end if
    end if
end while
```

3.4.3 Transitive Generic Stable Matching

To ensure the network matching results can meet the "*transitivity law*", in matching networks ($G^{(i)}$, $G^{(j)}$), ($G^{(j)}$, $G^{(k)}$) and ($G^{(k)}$, $G^{(i)}$), we need to consider the results globally. For instance,when matching these 3 networks, we can match networks ($G^{(j)}$, $G^{(k)}$) with Algorithm 1, which is identical to the regular pairwise network matching problem. Next, we can match networks ($G^{(i)}$, $G^{(j)}$). If we identify ($u^{(i)}$, $v^{(j)}$) and ($v^{(j)}$, $w^{(k)}$) should be matched between networks ($G^{(i)}$, $G^{(j)}$) and ($G^{(j)}$, $G^{(k)}$) respectively, we will follow the following strategy to either pre-add ($w^{(k)}, u^{(i)}$) to the alignment result between networks ($G^{(k)}$, $G^{(i)}$) or separate pair ($u^{(i)}$, $v^{(j)}$) and set $u^{(i)}$ and $v^{(j)}$ as self-occupied:

- *case 1*: Given that ($v^{(j)}$, $w^{(k)}$) is matched between networks ($G^{(j)}$, $G^{(k)}$), if users ($u^{(i)}$, $v^{(j)}$) is paired together between networks ($G^{(i)}$, $G^{(j)}$), and $u^{(i)}$ and $w^{(k)}$ are either free or self-occupied, then we will add ($w^{(k)}, u^{(i)}$) to the result between networks ($G^{(k)}$, $G^{(i)}$).
- *case 2*: Given that ($v^{(j)}$, $w^{(k)}$) is matched between networks ($G^{(j)}$, $G^{(k)}$), if users ($u^{(i)}$, $v^{(j)}$) is paired together between networks ($G^{(i)}$, $G^{(j)}$), but either $u^{(i)}$ or $w^{(k)}$ has been matched with other users when matching networks ($G^{(k)}$, $G^{(i)}$), then we

will set users $u^{(i)}$ and $v^{(j)}$ to be self-occupied in the results between networks $(G^{(i)}, G^{(j)})$.

Next, we can match networks $G^{(k)}$, $G^{(i)}$ by following very similar strategies. For each user pair $(w^{(k)}, u^{(i)})$ to be matched (excluding the pre-added ones), we check the matching statuses of users $w^{(k)}$ and $u^{(i)}$ in the matching of $(G^{(i)}, G^{(j)})$ and $(G^{(j)}, G^{(k)})$:

- *case 1*: if $w^{(k)}$ and $u^{(i)}$ are both paired with other users in matching $(G^{(i)}, G^{(j)})$ and $(G^{(j)}, G^{(k)})$, and their partners are the same user actually, then we will add $(w^{(k)}, u^{(i)})$ into the alignment result of networks $(G^{(k)}, G^{(i)})$;
- *case 2*: if $w^{(k)}$ and $u^{(i)}$ are both paired with other users in matching $(G^{(i)}, G^{(j)})$ and $(G^{(j)}, G^{(k)})$, but their partners are different users, then we will set $w^{(k)}$ and $u^{(i)}$ as free/self-occupied and continue the matching process of networks $(G^{(k)}, G^{(i)})$;
- *case 3*: if one user (e.g., $w^{(k)}$) is matched with one user (e.g., $v^{(j)}$) but the other one (i.e., $u^{(i)}$) is set as self-occupied in matching $(G^{(i)}, G^{(j)})$ and $(G^{(j)}, G^{(k)})$, then we check the status of $v^{(j)}$ in matching $(G^{(j)}, G^{(k)})$. If $v^{(j)}$ is paired with another user, then we will set $w^{(k)}$ and $u^{(i)}$ as free/self-occupied and continue the matching process of networks $(G^{(k)}, G^{(i)})$;
- *case 4*: if $v^{(j)}$ is also set as self-occupied in matching networks $(G^{(j)}, G^{(k)})$, then we will add pair $(v^{(j)}, w^{(k)})$ into the matching result of networks $(G^{(j)}, G^{(k)})$ and add pair $(w^{(k)}, u^{(i)})$ into the alignment result of networks $(G^{(k)}, G^{(i)})$.

Finally, we can achieve the matching results among networks $G^{(i)}$, $G^{(j)}$ and $G^{(k)}$ respectively.

4 Experiments

To examine the effectiveness of UMA in addressing the M-NASA problem, extensive experiments on real-world multiple partially aligned social networks will be done in this section. Next, we will introduce the dataset used in the experiments in Sect. 4.1 and give brief descriptions about the experiment settings in Sect. 4.2. Experiment results and detailed analysis will be given in Sects. 4.3 and 4.4.

4.1 Dataset Description

Nowadays, Question-and-Answer (Q&A) websites are becoming a new platform for people to share knowledge, where individuals can conveniently post their questions online and get first-hand replies very quickly. A large number of Q&A sites have sprung out overnight, e.g., Stack Overflow,[1] Super User,[2] Programmers,[3] Quora.[4] Stack Overflow, Super User and Programmers are all Q&A sites constructed for exchanging knowledge about computer science and share large number of common users, which are used as the partially aligned networks $G^{(i)}$, $G^{(j)}$ and $G^{(k)}$ respectively in the experiments.

We crawled the multiple partially aligned Q&A networks during November 2014–January 2015 and the complete information of $10,000$ users in Stack Overflow, Super User and Programmers Q&A sites respectively. The anchor links (i.e., the ground truth) between pairs of these Q&A networks are obtained by crawling their homepages in these sites respectively, where users' IDs in all these networks they participate in are listed. For example, at site,[5] we can have access to all the Q&A sites IDs that Jon Skeet owns, which can be used to extract the ground truth anchor links across networks. Among these 3 networks, the number of shared anchor users (1) between Stack Overflow and Super User is $3,677$, (2) between Stack Overflow and Programmers is $2,626$, (3) between Super User and Programmers is $1,953$. Users in Q&A sites can answer questions which are of their interests. Considering that users don't have social links in these Q&A sites, we will create social connections among users if they have every answered the same question in the past. Answering common questions in Q&A sites denotes that they may share common interests as well as common expertise in certain areas.

4.2 Experiment Settings

In the experiments, anchor links between users across networks are used for validation only and are not involved in building models. Considering that the network alignment method introduced in this paper is based on the social link information only, isolated users with no social connections in each network are sampled and removed. Based on the social links among users, we infer the optimal transitional matrices between pairs of networks by minimizing the *friendship inconsistency* as well as the alignment transitivity penalty. Alternative updating method is used to solve the joint objective function, where the transitional matrices are initialized with method introduced in [14]. All users in each network are partitioned into 10 bins according to their social degrees, where initial anchor links are assumed to exist between users belonging to the corresponding bins between pairs of networks, e.g., users in bin 1 of Stack Overflow and those in bin 1 of Programmers. The initial values of entries corresponding to these anchor links in transitional matrices are calculated with the *relative degree distance*

[1] http://stackoverflow.com.

[2] http://superuser.com.

[3] http://programmers.stackexchange.com.

[4] http://www.quora.com.

[5] http://stackexchange.com/users/11683/jon-skeet?tab=accounts.

based on their social degrees, e.g., $rdd(u_l^{(i)}, u_m^{(j)}) = \left(1 + \frac{|deg(u_l^{(i)}) - deg(u_m^{(j)})|}{(deg(u_l^{(i)}) + deg(u_m^{(j)}))/2}\right)^{-1}$, where $deg(u)$ denotes the social degree of user u in the networks. Based on the inferred transitional matrices, anchor links with the highest scores but can meet the *one-to-one* constraint and *transitivity law* are selected with the method introduced in Sect. 3.4, which can output both the confidence scores and their inferred labels.

Comparison Methods: Considering that social networks studied in this paper (1) contain only social link information, and (2) no known anchor links exist between networks, therefore, neither inter-network user resolution method MOBIUS [30] built with various user attribute information nor supervised network alignment method MNA [13] can be applied to address the M-NASA problem. To show the advantages of UMA, we compare UMA with many other baseline methods, including both state-of-art network alignment methods as well as extended traditional methods, which are all unsupervised network alignment methods based on the link information only. All the comparison methods used in the experiments are listed as follows.

- *Unsupervised Multi-network Alignment*: Method UMA introduced in this paper can align multiple partially networks concurrently, which include two steps: (1) transitive network alignment, and (2) transitive network matching. Anchor links inferred by UMA can maintain both *one-to-one* constraint and *transitivity property*.
- *Integrated Network Alignment* (INA): To show that transitive network matching can improve the alignment results, we introduce another method named INA, which is identical to the first step of UMA but without the matching step. Anchor links inferred by INA cannot maintain the *one-to-one* constraint nor *transitivity law* property.
- *Pairwise Network Alignment*: Big-Align is a state-of-art unsupervised network alignment method proposed in [14] for aligning pairwise networks. When applied to the multiple-network case, Big-Align can only align networks pair by pair. What's more, the output of Big-Align cannot maintain the *one-to-one* constraint nor *transitivity property* of anchor links. We also use Big-Align as a baseline method to show the advantages of the multiple-network alignment framework UMA introduced in this paper.
- *Pairwise Alignment + Pairwise Matching*: We also extend Big-Align [14] and introduce another baseline method Big-Align-PM, which can further prune the redundant non-existing anchor links with pairwise network stable matching proposed in [13] to guarantee the inferred anchor links can meet the *one-to-one* constraint.
- *Relative Degree Distance (RDD) based Alignment*: The transitional matrix initialization method RDD [14] is compared as another baseline methods, which calculate the confidence scores of potential anchor links with the degree information of users.
- *Relative PageRank based Alignment*: Traditional PageRank method is mainly proposed for calculating the correlation rank scores of a webpage to the given query. In addition, we also extend the traditional PageRank method and propose a new

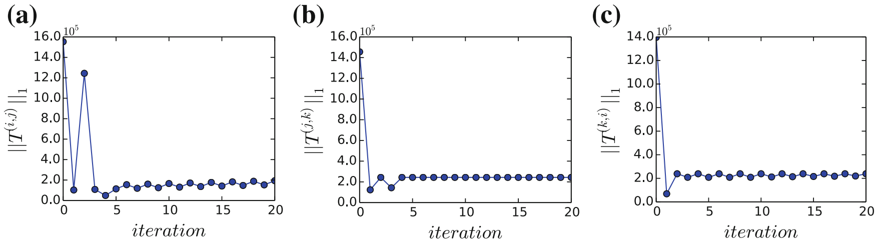


Fig. 4 L_1 norm of transitional matrices at each iteration. **a** Matrix $\mathbf{T}^{(i,j)}$. **b** Matrix $\mathbf{T}^{(j,k)}$. **c** Matrix $\mathbf{T}^{(k,i)}$

method RPR to infer potential anchor links. For a potential anchor link $(u_l^{(i)}, u_m^{(j)})$, RPR calculates the reciprocal of the relative pagerank scores between $u_l^{(i)}, u_m^{(j)}$ as its existence confidence, i.e., $|pagerank(u_l^{(i)}) - pagerank(u_m^{(j)})|^{-1}$.

Evaluation Metrics:
To evaluate the performance of different comparison methods, various commonly used evaluation metrics are applied. All these comparison methods (in INA, the selected anchor links are assigned with scores 1, while those not selected are assigned with scores 0) can output confidence scores of potential anchor links, which are evaluated by metrics AUC and Precision@100.

4.3 Convergence Analysis

To solve the objective function in Sect. 3.3, alternative updating method is applied to infer the optimal transitional matrices across networks. To demonstrate that the matrix updating equation can converge within a limited iterations, we calculate the L_1 norms (i.e., the sum of all entries' absolute value) of transitional matrices $\mathbf{T}^{(i,j)}$, $\mathbf{T}^{(j,k)}$ and $\mathbf{T}^{(k,i)}$ at each iteration, which are available in Fig. 4. As shown in the plots, after a few iterations (about 5 iterations), the L_1 norm of these transitional matrices will converge quickly with minor fluctuations around certain values, which demonstrates that the derived equation updating can converge very well in updating the transitional matrices.

4.4 Experiment Results

The experiment results of all these comparison methods are available in Fig. 5, where performance of all these comparison methods in Fig. 5 are evaluated by AUC and Precision@100 respectively.

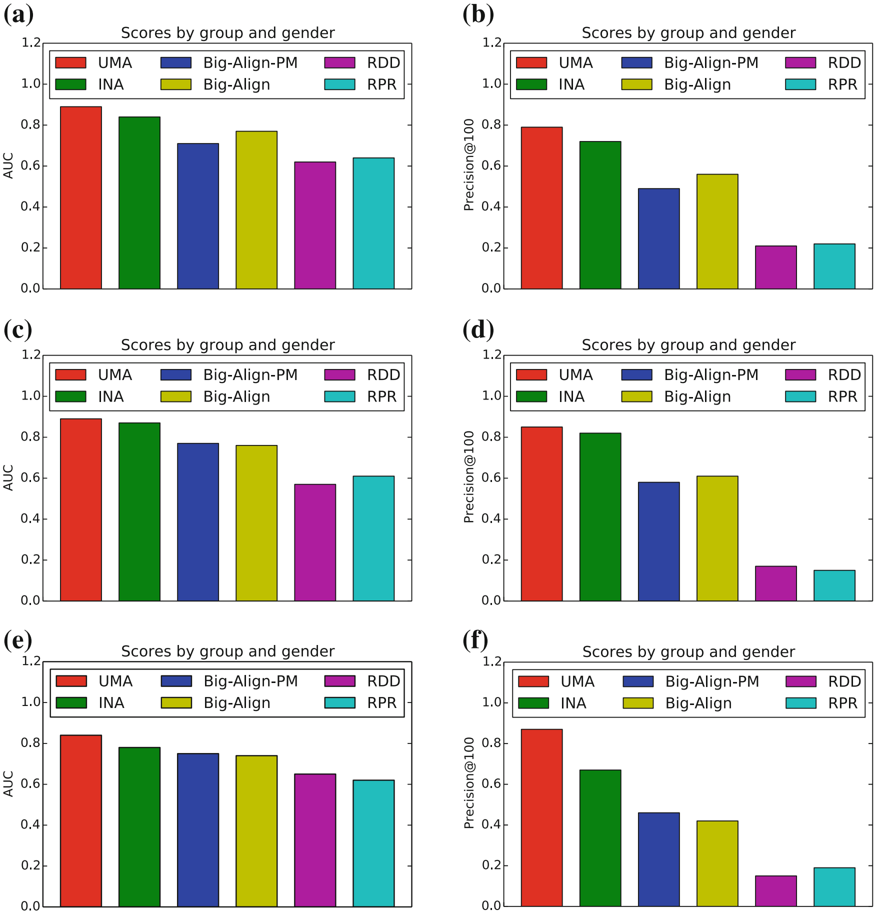


Fig. 5 Performance comparison of different methods evaluated by AUC and Precision@100. **a** AUC $(G^{(i)}, G^{(J)})$. **b** Precision @ 100 $(G^{(i)}, G^{(J)})$. **c** AUC $(G^{(j)}, G^{(k)})$. **d** Precision @ 100 $(G^{(j)}, G^{(k)})$. **e** AUC $(G^{(k)}, G^{(i)})$. **f** Precision @ 100 $(G^{(k)}, G^{(i)})$

In Fig. 5, we show the alignment results achieved by all the 6 comparison methods between network pairs $(G^{(i)}, G^{(j)})$, $(G^{(j)}, G^{(k)})$ and $(G^{(k)}, G^{(i)})$. As shown in the plots, UMA performs much better than all the other comparison methods with great advantages in predicting the anchor links between all these networks pairs. For instance, in Fig. 5a, the AUC obtained by UMA is 0.89, which is about 4 % larger than INA and over 13 % larger than the other comparison methods; in Fig. 5f, the Precision@100 achieved by UMA is 0.87, which is over 25 % higher than that of INA, almost the double of that gained by Big-Align and Big-Align-PM, and even 4–5 times of that obtained by RDD and RPR.

By comparing UMA and INA, method UMA consisting of transitive integrated network alignment and transitive network matching performs better, which demonstrates the effectiveness of the transitive network matching step in pruning redundant non-existing anchor links.

Compared with the isolated pairwise network alignment method BIG-ALIGN, the fact that INA achieves better performance justifies that aligning multiple networks simultaneously by incorporating the alignment transitivity penalty into the objective function can identify better anchor links than pairwise isolated network alignment.

By comparing BIG-ALIGN-PM and BIG-ALIGN, the pairwise network matching step can help improve the prediction results of anchor links between networks $(G^{(k)}, G^{(i)})$ but has no positive effects (even has negative effects) on the anchor links between other network pairs, e.g., network pairs $(G^{(i)}, G^{(j)})$ and $(G^{(j)}, G^{(k)})$. However, the effective of the transitive network matching method applied in UMA has been proved in the comparison of UMA and INA. It may show that transitive network matching exploiting the transitivity law performs much better than the pairwise network matching method.

For completeness, we also compare UMA with extensions of traditional methods RDD and RPR and the advantages of UMA over these methods are very obvious.

5 Related Works

Graph alignment is an important research problem in graph studies [6] and dozens of papers have been published on this topic in the past decades. Depending on specific disciplines, the studied graphs can be social networks in data mining [13] protein-protein interaction (PPI) networks and gene regulatory networks in bioinformatics [11, 17, 23, 24], chemical compound in chemistry [26], data schemas in data warehouse [19], ontology in web semantics [7], graph matching in combinatorial mathematics [18], as well as graphs in computer vision and pattern recognition [3, 5].

In bioinformatics, the network alignment problem aims at predicting the best mapping between two biological networks based on the similarity of the molecules and their interaction patterns. By studying the cross-species variations of biological networks, network alignment problem can be applied to predict conserved functional modules [21] and infer the functions of proteins [20]. Graemlin [9] conducts pairwise network alignment by maximizing an objective function based on a set of learned parameters. Some works have been done on aligning multiple network in bioinformatics. IsoRank proposed in [25] can align multiple networks greedily based on the pairwise node similarity scores calculated with spectral graph theory. IsoRankN [17] further extends IsoRank by exploiting a spectral clustering scheme.

In recent years, with rapid development of online social networks, researchers' attention starts to shift to the alignment of social networks. A comprehensive survey about recent works on heterogeneous social networks, including the recent network alignment works, is available in [22]. Enlightened by the homogeneous network alignment method in [28], Koutra et al. [14] propose to align two bipartite graphs with a fast alignment algorithm. Zafarani et al. [30] propose to match users across social networks based on various node attributes, e.g., username, typing patterns and language patterns etc. Kong et al. formulate the heterogeneous social network alignment problem as an anchor link prediction problem. A two-step supervised method MNA is proposed in [13] to infer potential anchor links across networks with heterogeneous information in the networks. However, social networks in the real world are mostly partially aligned actually and lots of users are not anchor users. Zhang et al. have proposed the partial network alignment methods based on supervised learning setting and PU learning setting in [32, 33] respectively. Existing social network alignment paper mostly focus on aligning two social networks, Zhang et al. [35] introduce a multiple network concurrent alignment framework to align multiple social networks simultaneously. Besides the common users shared by different social networks, many other categories of information entities, e.g., movies, geo-locations, and products, can also be shared by different movie-related networks, location based social networks, and e-commerce sites respectively. Zhang et al. are the first to introduce the partial co-alignment of social network, and propose a sophisticated network co-alignment framework in [36].

6 Conclusion

In this paper, we have studied the *multiple anonymized social network alignment* (M-NASA) problem to infer the anchor links across multiple anonymized online social networks simultaneously. An effective two-step multiple network alignment framework UMA has been proposed to address the M-NASA problem. The anchor links to be inferred follow both *transitivity law* and *one-to-one* property, under the constraint of which, UMA matches multiple anonymized networks by minimizing the *friendship inconsistency* and selects anchor links which can lead to the maximum confidence scores across multiple anonymized social networks based on the generic stable matching method. In this paper, we take 3 Q&A networks as an example to introduce both the method and conduct the experiments. In our future works, we will generalize the proposed model to multiple networks of diverse categories.

Acknowledgments This work is supported in part by NSF through grants III-1526499, CNS-1115234, and OISE-1129076, Google Research Award, and the Pinnacle Lab at Singapore Management University.

Appendix: New Objective Function

Based on the above relaxations used in Sect. 3.3, the new objective function can be represented as

$$\begin{aligned}
&\bar{\mathbf{T}}^{(i,j)}, \bar{\mathbf{T}}^{(j,k)}, \bar{\mathbf{T}}^{(k,i)} \\
&= \arg\min\nolimits_{\mathbf{T}^{(i,j)},\mathbf{T}^{(j,k)},\mathbf{T}^{(k,i)}} \left\|(\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} - \mathbf{S}^{(j)}\right\|_F^2 \\
&+ \left\|(\mathbf{T}^{(j,k)})^\top \mathbf{S}^{(j)} \mathbf{T}^{(j,k)} - \mathbf{S}^{(k)}\right\|_F^2 + \left\|(\mathbf{T}^{(k,i)})^\top \mathbf{S}^{(k)} \mathbf{T}^{(k,i)} - \mathbf{S}^{(i)}\right\|_F^2 \\
&+ \alpha \cdot \left\|(\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} - \mathbf{T}^{(k,i)} \mathbf{S}^{(i)} (\mathbf{T}^{(k,i)})^\top\right\|_F^2 \\
&+ \beta \cdot \left\|\mathbf{T}^{(i,j)}\right\|_0 + \gamma \cdot \left\|\mathbf{T}^{(j,k)}\right\|_0 + \theta \cdot \left\|\mathbf{T}^{(k,i)}\right\|_0 \\
&s.t.\ \mathbf{0}^{|\mathcal{U}^{(i)}|\times|\mathcal{U}^{(j)}|} \preccurlyeq \mathbf{T}^{(i,j)} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(i)}|\times|\mathcal{U}^{(j)}|}, \\
&\quad \mathbf{0}^{|\mathcal{U}^{(j)}|\times|\mathcal{U}^{(k)}|} \preccurlyeq \mathbf{T}^{(j,k)} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(j)}|\times|\mathcal{U}^{(k)}|}, \\
&\quad \mathbf{0}^{|\mathcal{U}^{(k)}|\times|\mathcal{U}^{(i)}|} \preccurlyeq \mathbf{T}^{(k,i)} \preccurlyeq \mathbf{1}^{|\mathcal{U}^{(k)}|\times|\mathcal{U}^{(i)}|}.
\end{aligned}$$

The Lagrangian function of the objective function can be represented as

$$\begin{aligned}
&\mathcal{L}(\mathbf{T}^{(i,j)}, \mathbf{T}^{(j,k)}, \mathbf{T}^{(k,i)}, \beta, \gamma, \theta) = \left\|(\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} - \mathbf{S}^{(j)}\right\|_F^2 \\
&+ \left\|(\mathbf{T}^{(j,k)})^\top \mathbf{S}^{(j)} \mathbf{T}^{(j,k)} - \mathbf{S}^{(k)}\right\|_F^2 + \left\|(\mathbf{T}^{(k,i)})^\top \mathbf{S}^{(k)} \mathbf{T}^{(k,i)} - \mathbf{S}^{(i)}\right\|_F^2 \\
&+ \alpha \cdot \left\|(\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} - \mathbf{T}^{(k,i)} \mathbf{S}^{(i)} (\mathbf{T}^{(k,i)})^\top\right\|_F^2 \\
&+ \beta \cdot \left\|\mathbf{T}^{(i,j)}\right\|_0 + \gamma \cdot \left\|\mathbf{T}^{(j,k)}\right\|_0 + \theta \cdot \left\|\mathbf{T}^{(k,i)}\right\|_0 .
\end{aligned}$$

The partial derivatives of function $\mathcal{L}$ with regard to $\mathbf{T}^{(i,j)}$, $\mathbf{T}^{(j,k)}$, and $\mathbf{T}^{(k,i)}$ will be:

$$\begin{aligned}
&(1)\frac{\partial \mathcal{L}\left(\mathbf{T}^{(i,j)}, \mathbf{T}^{(j,k)}, \mathbf{T}^{(k,i)}, \beta, \gamma, \theta\right)}{\partial \mathbf{T}^{(i,j)}} \\
&= 2 \cdot \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} (\mathbf{T}^{(i,j)})^\top (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)} \\
&+ 2 \cdot (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)} (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \\
&+ 2\alpha \cdot \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} (\mathbf{T}^{(j,k)})^\top \\
&+ 2\alpha \cdot (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} (\mathbf{T}^{(j,k)})^\top \\
&- 2 \cdot \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} (\mathbf{S}^{(j)})^\top - 2 \cdot (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)} \mathbf{S}^{(j)} \\
&- 2\alpha \cdot (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} \mathbf{T}^{(k,i)} \mathbf{S}^{(i)} (\mathbf{T}^{(k,i)})^\top (\mathbf{T}^{(j,k)})^\top \\
&- 2\alpha \cdot \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} \mathbf{T}^{(k,i)} (\mathbf{S}^{(i)})^\top (\mathbf{T}^{(k,i)})^\top (\mathbf{T}^{(j,k)})^\top - \beta \cdot \mathbf{1}\mathbf{1}^\top. \\
&(2)\frac{\partial \mathcal{L}\left(\mathbf{T}^{(i,j)}, \mathbf{T}^{(j,k)}, \mathbf{T}^{(k,i)}, \beta, \gamma, \theta\right)}{\partial \mathbf{T}^{(j,k)}} \\
&= 2 \cdot \mathbf{S}^{(j)} \mathbf{T}^{(j,k)} (\mathbf{T}^{(j,k)})^\top (\mathbf{S}^{(j)})^\top \mathbf{T}^{(j,k)} \\
&+ 2 \cdot (\mathbf{S}^{(j)})^\top \mathbf{T}^{(j,k)} (\mathbf{T}^{(j,k)})^\top \mathbf{S}^{(j)} \mathbf{T}^{(j,k)} \\
&+ 2\alpha \cdot (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} \\
&+ 2\alpha \cdot (\mathbf{T}^{(i,j)})^\top (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)} \mathbf{T}^{(i,j)} \mathbf{T}^{(j,k)} \\
&- 2 \cdot \mathbf{S}^{(j)} \mathbf{T}^{(j,k)} (\mathbf{S}^{(k)})^\top - 2 \cdot (\mathbf{S}^{(j)})^\top \mathbf{T}^{(j,k)} \mathbf{S}^{(k)}
\end{aligned}$$

$$
\begin{aligned}
&- 2\alpha \cdot (\mathbf{T}^{(i,j)})^\top (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)}\mathbf{T}^{(j,k)}\mathbf{T}^{(k,i)}\mathbf{S}^{(i)}(\mathbf{T}^{(k,i)})^\top \\
&- 2\alpha \cdot (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)}\mathbf{T}^{(i,j)}\mathbf{T}^{(j,k)}\mathbf{T}^{(k,i)}(\mathbf{S}^{(i)})^\top (\mathbf{T}^{(k,i)})^\top - \gamma \cdot \mathbf{1}\mathbf{1}^\top.
\end{aligned}
$$

$$
\begin{aligned}
(3)\, &\frac{\partial \mathcal{L}\left(\mathbf{T}^{(i,j)}, \mathbf{T}^{(j,k)}, \mathbf{T}^{(k,i)}, \beta, \gamma, \theta\right)}{\partial \mathbf{T}^{(k,i)}} \\
&= 2 \cdot \mathbf{S}^{(k)}\mathbf{T}^{(k,i)}(\mathbf{T}^{(k,i)})^\top (\mathbf{S}^{(k)})^\top \mathbf{T}^{(k,i)} \\
&+ 2 \cdot (\mathbf{S}^{(k)})^\top \mathbf{T}^{(k,i)}(\mathbf{T}^{(k,i)})^\top \mathbf{S}^{(k)}\mathbf{T}^{(k,i)} \\
&+ 2\alpha \mathbf{T}^{(k,i)}(\mathbf{S}^{(i)})^\top (\mathbf{T}^{(k,i)})^\top \mathbf{T}^{(k,i)}\mathbf{S}^{(i)} \\
&+ 2\alpha \mathbf{T}^{(k,i)}\mathbf{S}^{(i)}(\mathbf{T}^{(k,i)})^\top \mathbf{T}^{(k,i)}(\mathbf{S}^{(i)})^\top \\
&- 2 \cdot \mathbf{S}^{(k)}\mathbf{T}^{(k,i)}(\mathbf{S}^{(i)})^\top - 2 \cdot (\mathbf{S}^{(k)})^\top \mathbf{T}^{(k,i)}\mathbf{S}^{(i)} \\
&- 2\alpha \cdot (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top (\mathbf{S}^{(i)})^\top \mathbf{T}^{(i,j)}\mathbf{T}^{(j,k)}\mathbf{T}^{(k,i)}\mathbf{S}^{(i)} \\
&- 2\alpha \cdot (\mathbf{T}^{(j,k)})^\top (\mathbf{T}^{(i,j)})^\top \mathbf{S}^{(i)}\mathbf{T}^{(i,j)}\mathbf{T}^{(j,k)}\mathbf{T}^{(k,i)}(\mathbf{S}^{(i)})^\top - \theta \cdot \mathbf{1}\mathbf{1}^\top.
\end{aligned}
$$

References

1. Avriel, M.: Nonlinear Programming: Analysis and Methods. Prentice-Hall, Englewood Cliffs (1976)
2. Backstrom, L., Dwork, C., Kleinberg, J.: Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In: WWW (2007)
3. Bayati, M., Gerritsen, M., Gleich, D., Saberi, A., Wang, Y.: Algorithms for large, sparse network alignment problems. In: ICDM (2009)
4. Bhattacharya, I., Getoor, L.: Collective entity resolution in relational data. TKDD (2007)
5. Conte, D., Foggia, P., Sansone, C., Vento, M.: Thirty years of graph matching in pattern recognition. IJPRAI (2004)
6. Deo, N.: Graph Theory with Applications to Engineering and Computer Science. Prentice Hall Series in Automatic Computation. Prentice-Hall Inc. (1974)
7. Doan, A., Madhavan, J., Domingos, P., Halevy, A.: Ontology matching: a machine learning approach. In: Handbook on Ontologies (2004)
8. Dubins, L., Freedman, D.: Machiavelli and the gale-shapley algorithm. Am. Math. Mon. (1981)
9. Flannick, J., Novak, A., Srinivasan, B., McAdams, H., Batzoglou, S.: Graemlin: general and robust alignment of multiple large interaction networks. Genome Res. (2006)
10. Jin, S., Zhang, J., Yu, P., Yang, S., Li, A.: Synergistic partitioning in multiple large scale social networks. In: IEEE BigData (2014)
11. Kalaev, M., Bafna, V., Sharan, R.: Fast and accurate alignment of multiple protein networks. In: RECOMB (2008)
12. Khan, A., Gleich, D., Pothen, A., Halappanavar, M.: A multithreaded algorithm for network alignment via approximate matching. In: SC (2012)
13. Kong, X., Zhang, J., Yu, P.: Inferring anchor links across multiple heterogeneous social networks. In: CIKM (2013)
14. Koutra, D., Tong, H., Lubensky, D.: Big-align: fast bipartite graph alignment. In: ICDM (2013)
15. Kunen, K.: Set Theory. Elsevier Science Publishers (1980)
16. Lee, J., Han, W., Kasperovics, R., Lee, J.: An in-depth comparison of subgraph isomorphism algorithms in graph databases. VLDB (2012)

17. Liao, C., Lu, K., Baym, M., Singh, R., Berger, B.: Isorankn: spectral methods for global alignment of multiple protein networks. Bioinformatics (2009)
18. Manne, F., Halappanavar, M.: New effective multithreaded matching algorithms. In: IPDP (2014)
19. Melnik, S., Garcia-Molina, H., Rahm, E.: Similarity flooding: a versatile graph matching algorithm and its application to schema matching. In: ICDE (2002)
20. Park, D., Singh, R., Baym, M., Liao, C., Berger, B.: Isobase: a database of functionally related proteins across ppi networks. Nucleic Acids Res. (2011)
21. Sharan, R., Suthram, S., Kelley, R., Kuhn, T., McCuine, S., Uetz, P., Sittler, T., Karp, R., Ideker, T.: Conserved patterns of protein interaction in multiple species (2005)
22. Shi, C., Li, Y., Zhang, J., Sun, Y., Yu, P.: A survey of heterogeneous information network analysis. CoRR (2015). arXiv:1511.04854
23. Shih, Y., Parthasarathy, S.: Scalable global alignment for multiple biological networks. Bioinformatics (2012)
24. Singh, R., Xu, J., Berger, B.: Pairwise global alignment of protein interaction networks by matching neighborhood topology. In: RECOMB (2007)
25. Singh, R., Xu, J., Berger, B.: Global alignment of multiple protein interaction networks with application to functional orthology detection. In: Proceedings of the National Academy of Sciences (2008)
26. Smalter, A., Huan, J., Lushington, G.: Gpm: a graph pattern matching kernel with diffusion for chemical compound classification. In: IEEE BIBE (2008)
27. Tsikerdekis, M., Zeadally, S.: Multiple account identity deception detection in social media using nonverbal behavior. IEEE TIFS (2014)
28. Umeyama, S.: An eigendecomposition approach to weighted graph matching problems. IEEE TPAMI (1988)
29. Wipf, D., Rao, B.: L0-norm minimization for basis selection. In: NIPS (2005)
30. Zafarani, R., Liu, H.: Connecting users across social media sites: a behavioral-modeling approach. In: KDD (2013)
31. Zhan, Q., Zhang, J., Wang, S., Yu, P., Xie, J.: Influence maximization across partially aligned heterogenous social networks. In: PAKDD (2015)
32. Zhang, J., Shao, W., Wang, S., Kong, X., Yu, P.: Partial network alignment with anchor meta path and truncated generalized stable matching. In: IRI (2015)
33. Zhang, J., Yu, P.: Integrated anchor and social link predictions across social networks. In: IJCAI (2015)
34. Zhang, J., Yu, P.: Mcd: mutual clustering across multiple heterogeneous networks. In: IEEE BigData Congress (2015)
35. Zhang, J., Yu, P., Multiple anonymized social networks alignment. In: ICDM (2015)
36. Zhang, J., Yu, P., Pct: partial co-alignment of social networks. In: WWW (2016)
37. Zhang, J., Yu, P., Zhou, Z.: Meta-path based multi-network collective link prediction. In: KDD (2014)