Coupled Graphs and Tensor Factorization for Recommender Systems and Community Detection

Vassilis N. Ioannidis, *Student Member, IEEE*, Ahmed S. Zamzam, *Student Member, IEEE*, Georgios B. Giannakis, *Fellow, IEEE*, and Nicholas D. Sidiropoulos, *Fellow, IEEE*

Abstract—Joint analysis of data from multiple information repositories facilitates uncovering the underlying structure in heterogeneous datasets. Single and coupled matrix-tensor factorization (CMTF) has been widely used in this context for imputation-based recommendation from ratings, social network, and other user-item data. When this side information is in the form of item-item correlation matrices or graphs, existing CMTF algorithms may fall short. Alleviating current limitations, we introduce a novel model coined coupled graph-tensor factorization (CGTF) that judiciously accounts for graph-related side information. The CGTF model has the potential to overcome practical challenges, such as missing slabs from the tensor and/or missing rows/columns from the correlation matrices. A novel alternating direction method of multipliers (ADMM) is also developed that recovers the nonnegative factors of CGTF. Our algorithm enjoys closed-form updates that result in reduced computational complexity and allow for convergence claims. A novel direction is further explored by employing the interpretable factors to detect graph communities having the tensor as side information. The resulting community detection approach is successful even when some links in the graphs are missing. Results with real data sets corroborate the merits of the proposed methods relative to state-of-the-art competing factorization techniques in providing recommendations and detecting communities.

Index Terms—Tensor-matrix factorization, tensor-graph imputation, graph data, recommender systems, community detection

1 Introduction

Multi-relational data emerge in applications as diverse as social networks, recommender systems, biomedical imaging, computer vision and communication networks, and are typically modeled using high-order tensors [3]. However, in many real settings only a subset of the data is observed due to application-specific restrictions. For example, in recommender systems ratings of new users are missing; in social applications individuals may be reluctant to share personal information due to privacy concerns; and brain data may contain misses due to inadequate spatial resolution. In this context, a task of paramount importance is to infer unavailable entries given the available data.

Inference of unavailable tensor data can certainly benefit from side information that can be available in the form of correlations, social interactions, or, biological relations, all of which can be captured by a graph [4]. In recommender systems for instance, one may benefit from available user-user interactions over a social network to impute the missing ratings, and also extrapolate (that is predict) profitable recommendations to new costumers.

In addition to graph-aided inference of tensor data, benefits can be effected in the opposite direction through tensor data employed to improve graph inference tasks, such as community detection (CD). CD amounts to finding clusters of vertices densely connected within each cluster and scarcely connected across clusters [5]. A major challenges emerges here when some links in the graph are missing due to privacy or observation constraints. In a social network for example, not all users will provide their social network connections. Additional data organized in a tensor can be utilized to improve CD performance and cope with the missing lnks of the graph.

The present paper develops a novel approach to inference with incomplete data by jointly leveraging tensor factorization and associated graphs.

1.1 Related Work

Matrix factorization (MF) techniques have been employed for matrix completion with documented success in user-item recommender systems [6]. MF-based techniques assume that the ratings matrix is of low rank, and hence can be modeled by a reduced number of factors. Although the two-relation recommendation model has wide applicability, multi-relation data motivate the use of high-dimensional tensor models. Scalable algorithms for nonnegative tensor factorization

- *V. N. Ioannidis, A. S. Zamzam, and G. B. Giannakis are with the ECE Department and Digital Tech. Center, University of Minnesota, Minneapolis, MN 55455 USA. E-mail: {ioann006, ahmedz, georgios}@umn.edu.*
- *N. D. Sidiropoulos is with the ECE Department, University of Virginia, Charlottesville, VA 22903 USA. E-mail: nikos@virginia.edu.*

Manuscript received 21 Sept. 2018; revised 1 June 2019; accepted 13 Aug. 2019. Date of publication 16 Sept. 2019; date of current version 3 Feb. 2021. (Corresponding author: Vassilis N. Ioannidis.)
Recommended for acceptance by J. Caverlee.
Digital Object Identifier no. 10.1109/TKDE.2019.2941716

(TF) have been pursued [7], but do not consider further structure on tensor modes or any other form of side information.

Side information in the form of matrices sharing factors with a data tensor has been investigated in the so-termed coupled matrix-tensor factorization (CMTF) [8], [9], [10]. Typically, CMTF adopts a low-rank model for the tensor to recover the missing entries. Misses in both the side information and the tensor were handled in [8], [9], but not with the use of graph adjacency matrices. Using a Bayesian approach, inference relying on tensor factorization with low-rank covariance regularization, was reported in [11]. Albeit interesting, this approach assumes that the similarity matrices are fully observable, which is not the case in several applications e.g., social networks.

1.2 Our Contributions

Alleviating the limitations of existing approaches, this paper introduces a novel factorization model coined coupled graph and tensor factorization (CGTF) to account for the graph structure of side information. The CGTF factors are estimated via a novel algorithm based on the alternating method of multipliers (ADMM) to infer missing entries in both the matrices and the tensor. The CGTF is subsequently explored to detect communities in the partially observed coupled graphs. Specifically, the contribution of this paper is fourfold.

- C1. A novel model is introduced to link multiple repositories of information bearing data and their correlations in the form of high-order tensors and graphs. The proposed approach can overcome practical challenges, such as missing slabs from the tensor and/or missing rows/columns from the correlation matrices (graph links), known as the *cold start* problem.
- C2. A novel ADMM algorithm is developed that features convergence guarantees and low computational complexity by using closed-form updates. Our accelerated ADMM solver leverages data sparsity [9] and can easily incorporate other types of constraints on the latent factors.
- C3. The proposed approach is applied to recommender systems and markedly improves the rating prediction performance. The results in two real datasets corroborate that the novel method is successful in providing accurate recommendations as well as recovering missing links in graphs.
- C4. Finally, the proposed coupled factorization approach enables detection of communities on graphs by using the recovered factors. Experiments testify to the ability of CGTF to exploit the tensor data for CD even when graph links are missing; e.g., cold start problem.

The novel contribution of this work concerning CD is in the coupling between tensor and graph data. Nodes in the recovered communities have similar graph connections and tensor data. Different than traditional CD methods [12], [13], [14], [15], [16], [17], [18] that find communities given only the graph, our CGTF finds communities from the *coupled* tensor and graph data and hence can be even applied when graph links are missing.

The rest of this paper is organized as follows. Section 2 describes the model and the problem formulation. Section 3 introduces the novel algorithm, and Section 4 deals with the application of CGTF to community detection. Section 5 demonstrates the effectiveness of the proposed approach in real and synthetic data. Finally, Section 6 summarizes some closing remarks.

Throughout, lower and upper boldface letters are used to denote vectors and matrices, respectively. The tensors are denoted by underlined upper case boldface symbols. For any general matrix $\mathbf{X}$, $\mathbf{X}^T$, $\mathbf{X}^{-1}$, $\mathrm{Tr}(\mathbf{X})$, and $\mathrm{diag}(\mathbf{X})$ denote respectively the transpose, inverse, trace, and diagonal of $\mathbf{X}$. The Khatri-Rao and Hadamard products of two matrices $\mathbf{X}$ and $\mathbf{Y}$ are denoted by $\mathbf{X} \odot \mathbf{Y}$ and $\mathbf{X} * \mathbf{Y}$, respectively. The operator $\mathrm{vec}(\cdot)$ denotes the vectorization of $(\cdot)$.

2 Coupled Factorization Model

Consider a tensor $\underline{\mathbf{X}}$ of order N and size $I_1 \times I_2 \times \cdots \times I_N$. An entry of $\underline{\mathbf{X}}$ is denoted by $[\underline{\mathbf{X}}]_{(i_1,i_2,\ldots,i_N)}$, where index i_n refers to the nth mode of the tensor. The focus of this paper is on tensors with positive entries that appear in diverse applications such as recommender systems, finance, or biology. The mode-k matricization of $\underline{\mathbf{X}}$ is denoted by the matrix $\mathbf{X}_k$, which arranges the mode-k one-dimensional fibers as columns of the resulting matrix; see [3] for details.

Without loss of generality, consider 3-way tensors $\underline{\mathbf{X}} \in \mathbb{R}_+^{I_1 \times I_2 \times I_3}$. In many real settings, tensors have low rank and hence can be expressed via the well-known parallel factor (PARAFAC) decomposition [3] that models a rank-R tensor as

$$[\underline{\mathbf{X}}]_{(i_1,i_2,i_3)} = \sum_{r=1}^{R} [\mathbf{A}_1]_{(i_1,r)} [\mathbf{A}_2]_{(i_2,r)} [\mathbf{A}_3]_{(i_3,r)} + [\underline{\mathbf{E}}]_{(i_1,i_2,i_3)},$$

where $\{\mathbf{A}_n \in \mathbb{R}_+^{I_n \times R}\}_{n=1}^3$ represent the low-rank factor matrices corresponding to the three modes of the tensor, and $\underline{\mathbf{E}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ captures model mismatch. The PARAFAC model is written in tensor-matrix form as

$$\underline{\mathbf{X}} = \big[[\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3]\big] + \underline{\mathbf{E}}, \tag{1}$$

where $\big[[\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3]\big]$ is the outer product of these matrices resulting in a tensor. Oftentimes, only a subset of entries of $\underline{\mathbf{X}}$ is observable due to application-specific constraints such as privacy in social network applications; experimental error in the data collection process; or missing ratings in recommender systems. Hence, we write $\underline{\mathbf{X}} = \underline{\mathbf{X}}^A + \underline{\mathbf{X}}^M$, where $\underline{\mathbf{X}}^A$ contains the available tensor entries and otherwise is zero and $\underline{\mathbf{X}}^M$ holds the missing values and zeros elsewhere.

The tensor entries are also related through a set of per-mode graph adjacency or similarity matrices $\{\mathbf{G}_n \in \mathbb{R}_+^{I_n \times I_n}\}_{n=1}^3$. The (i,i')th entry of $\mathbf{G}_n$ reflects the similarity between the ith and i'th data items of the nth tensor mode and thus, $\mathbf{G}_n$ captures the connectivity of the corresponding mode-n graph. This prior information for the tensor entries is well-motivated since network data are available across numerous disciplines including sociology, biology, neuroscience and engineering. In these domains, subsets of entries (here graph nodes) form communities in the sense that they exhibit dense intra-connections and sparse inter-connections, which are captured by $\mathbf{G}_n$. Such connections are common in e.g., social networks [19], where friends tend to form dense clusters. We will model this graph-induced side information on tensor data using a symmetric nonnegative matrix

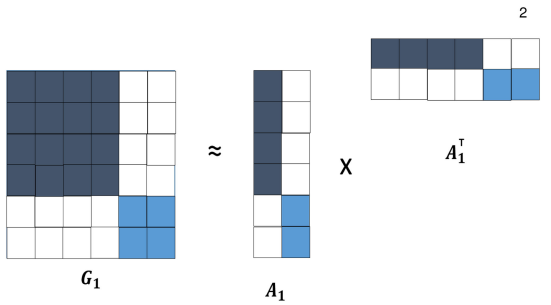


Fig. 1. Illustration of the SNMF model (2) for $\mathbf{G}_1$ with $\mathbf{d}_1 = \mathbf{1}$. White cells correspond to small-value entries. The rows and columns of $\mathbf{G}_1$ have been reorganized to place nodes in the same community one after the other.

factorization (SNMF) model [16], which can efficiently provide identifiable factors and recover graph clusters. Specifically, we advocate the following diagonally-scaled SNMF model

$$\mathbf{G}_n = \mathbf{A}_n \operatorname{diag}(\mathbf{d}_n)\mathbf{A}_n^\top + \mathbf{V}_n, \ n = 1, 2, 3, \tag{2}$$

where $\{\mathbf{V}_n \in \mathbb{R}^{I_n \times I_n}\}_n$ capture modeling errors; $\{\mathbf{d}_n \in \mathbb{R}_+^{R\times 1}\}_n$ weight the factor matrices; and $\{\mathbf{A}_n \in \mathbb{R}^{I_n \times R}\}_n$ denote factor matrices of rank $R < I_n$ that readily reveal communities in the graphs corresponding to $\{\mathbf{G}_n\}_n$ [16], [20]. Recovering the community of the ith node in the nth graph is straightforward, by selecting the largest entry in the ith row of $\mathbf{A}_n$ [16], [20]; see Fig. 1. Unfortunately, the topologies of $\{\mathbf{G}_n\}$ may contain missing entries, which can be attributed to privacy concerns in social networks, or down-sampling massive networks. Hence, the graph matrices are modeled as $\mathbf{G}_n = \mathbf{G}_n^A + \mathbf{G}_n^M$, where $\mathbf{G}_n^A$ contains the available links and $\mathbf{G}_n^M$ holds the unavailable ones.

The factors $\{\mathbf{A}_n\}_n$ are shared among the tensor and the graph of each corresponding item, which justifies the name of the proposed model as coupled graph tensor factorization. Whereas classical CMTF approaches model the side information as $\mathbf{A}_n\mathbf{B}_n^\top$, the novel CGTF captures the graph structure by employing $\mathbf{A}_n \operatorname{diag}(\mathbf{d}_n)\mathbf{A}_n^\top$. Adding the diagonal loading matrices endows the model with the ability to adjust the relative weight between the tensor and the side information matrices. The novel CGTF model is depicted in Fig. 2.

Problem Statement. Given $\underline{\mathbf{X}}^A$ and $\{\mathbf{G}_n^A\}_{n=1}^3$, our goal is to estimate $\underline{\mathbf{X}}^M$ and $\{\mathbf{A}_n, \mathbf{d}_n, \mathbf{G}_n^M\}_{n=1}^3$ by employing the CGTF model in (1) and (2). As a byproduct, the recovered $\{\mathbf{A}_n, \mathbf{d}_n\}_{n=1}^3$ will be utilized to detect communities.

3 Coupled Graph Tensor Factorization

Given (1) and (2), this section develops a novel algorithm to infer the latent factor matrices and hence estimate $\underline{\mathbf{X}}^M$ and $\{\mathbf{G}_n^M\}_{n=1}^3$. To this end, consider the optimization task

$$\begin{aligned} \underset{\underline{\mathbf{X}}^M, \{\mathbf{A}_n, \mathbf{d}_n, \mathbf{G}_n^M\}_{n=1}^3}{\text{minimize}} \quad & \|\underline{\mathbf{X}} - \left[\!\left[\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3\right]\!\right]\|_F^2 \\ & + \mu \sum_{n=1}^3 \|\mathbf{G}_n - \mathbf{A}_n \operatorname{diag}(\mathbf{d}_n)\mathbf{A}_n^\top\|_F^2 \\ \text{s. t.} \quad & \mathbf{A}_n \geq \mathbf{0}, \ \mathbf{d}_n \geq \mathbf{0}, \\ & \underline{\mathbf{X}} = \underline{\mathbf{X}}^A + \underline{\mathbf{X}}^M, \ \mathbf{G}_n = \mathbf{G}_n^A + \mathbf{G}_n^M, \\ & \mathcal{P}_{\underline{\Omega}}(\underline{\mathbf{X}}^M) = \underline{\mathbf{0}}, \ \mathcal{P}_{\Omega_n}(\mathbf{G}_n^M) = \mathbf{0}, \ n = 1, 2, 3, \end{aligned} \tag{3}$$

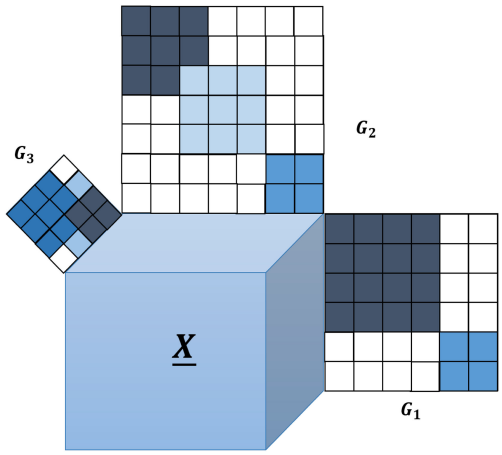


Fig. 2. Illustration of the tensor and graphs that partake in the CGTF model. The heat maps suggest that $\{\mathbf{G}_n\}_{n=1}^3$ exhibit community structure.

where $\mu > 0$ tunes the relative importance of the fit between the tensor and the graph-induced side information. The first term accounts for the LS fitting error of the PARAFAC model (1), and the second sum of LS costs accounts for the SNMF model (2). The positivity constraints stem from prior knowledge related to the factor and diagonal matrices. The equality conditions constrain $\underline{\mathbf{X}}$ and $\{\mathbf{G}_n\}_{n=1}^3$ to be equal to $\underline{\mathbf{X}}^A$ and $\{\mathbf{G}_n^A\}_{n=1}^3$ at the observed entries and to the optimization variables $\underline{\mathbf{X}}^M$ and $\{\mathbf{G}_n^M\}_{n=1}^3$ otherwise. The operators $\mathcal{P}_{\underline{\Omega}}$ and $\mathcal{P}_{\Omega_n}$ force the optimization variables to be zero at the observed entries.

The optimization problem in (3) is non-convex due to the trilinear terms $\left[\!\left[\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3\right]\!\right]$ and $\mathbf{A}_n \operatorname{diag}(\mathbf{d}_n)\mathbf{A}_n^\top$. The next section develops an efficient solver for (3) based on the ADMM [7].

Remark 1. In some applications, a graph $\mathbf{G}_n$ may not be available for one or more modes n of the tensor. Hence, before solving (3) one may remove the corresponding fitting term $\|\mathbf{G}_n - \mathbf{A}_n \operatorname{diag}(\mathbf{d}_n)\mathbf{A}_n^\top\|_F^2$ and related graph constraints. As a byproduct, our novel framework may utilize the recovered factor $\mathbf{A}_n$ and obtain a similarity matrix $\mathbf{G}_n$ (2).

3.1 ADMM for CGTF

First notice that the optimization problem (3) is even non-convex for each $\mathbf{A}_n$ separately due to the product of factor matrices in the SMNF model. This poses an additional challenge to any ADMM algorithm that iteratively pursues per block minimizers of the augmented Lagrangian. Hence, we introduce $\{\bar{\mathbf{A}}_n\}_n$ auxiliary variables and rewrite the SMNF cost as

$$\|\mathbf{G}_n - \mathbf{A}_n \operatorname{diag}(\mathbf{d}_n)\bar{\mathbf{A}}_n^\top\|_F^2. \tag{4}$$

Furthermore, to handle the positivity constraints we introduce

$$g(\mathbf{M}) = \begin{cases} 0, & \text{if } \mathbf{M} \geq \mathbf{0} \\ \infty, & \text{otherwise} \end{cases}, \tag{5}$$

and the auxiliary variables $\{\tilde{\mathbf{A}}_n, \tilde{\mathbf{d}}_n\}_n$. Next, we rewrite (3) to an equivalent form as

$$L\big(\underline{\mathbf{X}}^M,\mathbf{\Phi},\{\mathbf{G}_n^M,\mathbf{Y}_{\bar{\mathbf{A}}_n},\mathbf{Y}_{\tilde{\mathbf{A}}_n},\mathbf{y}_{\tilde{\mathbf{d}}_n}\}_{n=1}^3\big) := f\big(\underline{\mathbf{X}}^M,\mathbf{\Phi},\{\mathbf{G}_n^M\}_{n=1}^3\big) + \sum_{f=1}^{3}\big\{\mathrm{Tr}(\mathbf{Y}_{\bar{\mathbf{A}}_n}^\top(\mathbf{A}_n-\bar{\mathbf{A}}_n)+\frac{\rho_{\bar{\mathbf{A}}_n}}{2}\|\mathbf{A}_n-\bar{\mathbf{A}}_n\|_F^2 + \mathrm{Tr}(\mathbf{Y}_{\tilde{\mathbf{A}}_n}^\top(\mathbf{A}_n-\tilde{\mathbf{A}}_n))+\frac{\rho_{\tilde{\mathbf{A}}_n}}{2}\|\mathbf{A}_n-\tilde{\mathbf{A}}_n\|_F^2+\mathbf{y}_{\tilde{\mathbf{d}}_n}^\top(\mathbf{d}_n-\tilde{\mathbf{d}}_n)+\frac{\rho_{\tilde{\mathbf{d}}_n}}{2}\|\mathbf{d}_n-\tilde{\mathbf{d}}_n\|_F^2\big\}. \tag{7}$$

$$\begin{aligned}
&\underset{\substack{\underline{\mathbf{X}}^M,\{\mathbf{A}_n,\bar{\mathbf{A}}_n,\tilde{\mathbf{A}}_n,\\ \mathbf{d}_n,\tilde{\mathbf{d}}_n,\mathbf{G}_n^M\}_{n=1}^3}}{\text{minimize}} \ \|\underline{\mathbf{X}}-\big[[\mathbf{A}_1,\mathbf{A}_2,\mathbf{A}_3]\big]\|_F^2+\sum_{f=1}^{3}g(\tilde{\mathbf{A}}_n)\\
&+\mu\sum_{n=1}^{3}\|\mathbf{G}_n-\mathbf{A}_n\,diag\,(\mathbf{d}_n)\bar{\mathbf{A}}_n^\top\|_F^2+\sum_{f=1}^{3}g(\tilde{\mathbf{d}}_n)\\
&\text{s. t.}\qquad \mathbf{A}_n=\bar{\mathbf{A}}_n,\ \mathbf{A}_n=\tilde{\mathbf{A}}_n,\ \mathbf{d}_n=\tilde{\mathbf{d}}_n,\\
&\qquad\quad \underline{\mathbf{X}}=\underline{\mathbf{X}}^A+\underline{\mathbf{X}}^M,\ \mathbf{G}_n=\mathbf{G}_n^A+\mathbf{G}_n^M,\\
&\qquad\quad \mathcal{P}_{\underline{\Omega}}(\underline{\mathbf{X}}^M)=\underline{\mathbf{0}},\ \mathcal{P}_{\Omega_n}(\mathbf{G}_n^M)=\mathbf{0},\ n=1,2,3.
\end{aligned} \tag{6}$$

Even though (6) is still non-convex in all the variables, it is convex with respect to each block variable separately. Towards deriving an ADMM solver, we introduce the dual variables $\{\mathbf{Y}_{\bar{\mathbf{A}}_n}\in\mathbb{R}^{I_n\times R},\mathbf{Y}_{\tilde{\mathbf{A}}_n}\in\mathbb{R}^{I_n\times R},\mathbf{y}_{\tilde{\mathbf{d}}_n}\in\mathbb{R}^{R\times 1}\}_n$ and the penalty parameters $\{\rho_{\bar{\mathbf{A}}_n}>0,\rho_{\tilde{\mathbf{A}}_n}>0,\rho_{\tilde{\mathbf{d}}_n}>0\}_n$.

The augmented Lagrangian is given in (7), at the bottom of the next page, where $f(\cdot)$ represents the cost function in (6) and we collect all factor variables in $\mathbf{\Phi}:=(\{\mathbf{A}_n,\bar{\mathbf{A}}_n,\tilde{\mathbf{A}}_n,\mathbf{d}_n,\tilde{\mathbf{d}}_n\}_{n=1}^3)$. For ease of notation no ADMM superscripts will be used in the following equations. For brevity, only the ADMM updates for $n=1$ will be presented.

The update for $\mathbf{A}_1$ can be obtained by taking the derivative of L in (7) with respect to (w.r.t.) $\mathbf{A}_1$ and equating it to zero that yields

$$\begin{aligned}
&\hat{\mathbf{A}}_1(\mathbf{M}_1^\top\mathbf{M}_1+\mu\mathbf{D}_1\bar{\mathbf{A}}_1^\top\bar{\mathbf{A}}_1\mathbf{D}_1+(\rho_{\tilde{\mathbf{A}}_1}+\rho_{\bar{\mathbf{A}}_1})\mathbf{I}_R)\\
&=\mathbf{X}_1^\top\mathbf{M}_1+\mu\mathbf{G}_1\bar{\mathbf{A}}_1\mathbf{D}_1+\rho_{\bar{\mathbf{A}}_1}\bar{\mathbf{A}}_1+\rho_{\tilde{\mathbf{A}}_1}\tilde{\mathbf{A}}_1-\mathbf{Y}_{\tilde{\mathbf{A}}_1}-\mathbf{Y}_{\bar{\mathbf{A}}_1},
\end{aligned} \tag{8a}$$

where $\mathbf{M}_1:=\mathbf{A}_3\odot\mathbf{A}_2$, and $\mathbf{D}_1:=diag\,(\mathbf{d}_1)$. The update for $\mathbf{d}_1$ can be obtained likewise as

$$\begin{aligned}
&((\bar{\mathbf{A}}_1\odot\mathbf{A}_1)^\top(\mu\bar{\mathbf{A}}_1\odot\mathbf{A}_1)+\rho_{\tilde{\mathbf{d}}_1}\mathbf{I}_R)\hat{\mathbf{d}}_1\\
&=\mu(\bar{\mathbf{A}}_1\odot\mathbf{A}_1)^\top\mathbf{g}_1+\rho_{\tilde{\mathbf{d}}_1}\tilde{\mathbf{d}}_1-\mathbf{y}_{\tilde{\mathbf{d}}_1}.
\end{aligned} \tag{8b}$$

where $\mathbf{g}_n:=\mathrm{vec}(\mathbf{G}_n)$. Accordingly, the update for the $\bar{\mathbf{A}}_1$ is given by

$$\begin{aligned}
&\hat{\bar{\mathbf{A}}}_1(\mu\mathbf{D}_1\mathbf{A}_1^\top\mathbf{A}_1\mathbf{D}_1+\rho_{\bar{\mathbf{A}}_1}\mathbf{I}_R)\\
&=\mu\mathbf{G}_1^\top\mathbf{A}_1\mathbf{D}_1+\rho_{\bar{\mathbf{A}}_1}\mathbf{A}_1+\mathbf{Y}_{\bar{\mathbf{A}}_1}.
\end{aligned} \tag{8c}$$

The auxiliary variables $\tilde{\mathbf{A}}_1,\tilde{\mathbf{d}}_1$ are updated by projecting to the nonnegative orthant as follows

$$\begin{aligned}
\hat{\tilde{\mathbf{A}}}_1&=\left(\mathbf{A}_1+\frac{1}{\rho_{\tilde{\mathbf{A}}_1}}\mathbf{Y}_{\tilde{\mathbf{A}}_1}\right)_+,\\
\hat{\tilde{\mathbf{d}}}_1&=\left(\mathbf{d}_1+\frac{1}{\rho_{\tilde{\mathbf{d}}_1}}\mathbf{y}_{\tilde{\mathbf{d}}_1}\right)_+.
\end{aligned} \tag{8d}$$

Using the estimated factors $\{\hat{\mathbf{A}}_n\}_n$ the updates for the missing tensor elements are given by

$$\hat{\underline{\mathbf{X}}}^M=\mathcal{P}_{\underline{\Omega}}(\big[[\hat{\mathbf{A}}_1,\hat{\mathbf{A}}_2,\hat{\mathbf{A}}_3]\big]). \tag{8e}$$

Similarly, the missing entries in $\mathbf{G}_1$ can be obtained by

$$\hat{\mathbf{G}}_1^M=\mathcal{P}_{\Omega_1}(\hat{\mathbf{A}}_1\,diag\,(\hat{\mathbf{d}}_1)\hat{\bar{\mathbf{A}}}_1^\top). \tag{8f}$$

Finally, the updates for the Lagrange multipliers are

$$\begin{aligned}
\mathbf{Y}_{\bar{\mathbf{A}}_1}&=\mathbf{Y}_{\bar{\mathbf{A}}_1}+\rho_{\bar{\mathbf{A}}_1}(\mathbf{A}_1-\bar{\mathbf{A}}_1)\\
\mathbf{Y}_{\tilde{\mathbf{A}}_1}&=\mathbf{Y}_{\tilde{\mathbf{A}}_1}+\rho_{\tilde{\mathbf{A}}_1}(\mathbf{A}_1-\tilde{\mathbf{A}}_1)\\
\mathbf{y}_{\tilde{\mathbf{d}}_1}&=\mathbf{y}_{\tilde{\mathbf{d}}_1}+\rho_{\tilde{\mathbf{d}}_1}(\mathbf{d}_1-\tilde{\mathbf{d}}_1).
\end{aligned} \tag{8g}$$

The steps of our CGTF algorithm are listed in Algorithm 1. Since (6) is a non-convex problem, a judicious initialization of $\{\mathbf{A}_n\}_n$ is required. Towards that end, we adopt an efficient algorithm for SNMF, see [21], to initialize the factor matrices using only the available elements in the corresponding graphs $\{\mathbf{G}_n^A\}$, while $\{\mathbf{d}_n\}$ are initialized as all-ones vectors. Since SNMF is unique under certain conditions, the initialization is likely to be a good one [21]. The ADMM algorithm stops when the primal residuals and the dual feasibility residuals are sufficiently small. Even though $\{\tilde{\mathbf{A}}_n,\tilde{\mathbf{d}}_n\}_n$ are by construction non-negative, $\{\mathbf{A}_n,\bar{\mathbf{A}}_n,\mathbf{d}_n\}_n$ are not necessarily non-negative, but they become so upon convergence.

The advantage of introducing the auxiliary variables is threefold. First, by employing $\bar{\mathbf{A}}_n$, we bypass solving the non-convex SNMF that would require a costly iterative algorithm per factor update. Second, by introducing $\{\tilde{\mathbf{A}}_n,\tilde{\mathbf{d}}_n\}$, we avoid the solution to a constrained optimization problem, resulting in a more computationally affordable update compared to constrained least-squares based algorithms. In a nutshell, our novel reformulation allows for closed-form updates per step of the ADMM solver. Lastly, the closed-form updates allow us to make convergence claims to a stationary point of (6) in Section 3.2.

Remark 2. The era of data science brings opportunities for adversaries that aim to corrupt the data, e.g., recommendation data may be corrupted by malicious users that provide fake ratings, or social networks may contain spamming users. The CGTF model can be extended to account for anomalies in the graph links and the tensor data. Specifically, consider the robust CGTF (R-CGTF) as $\underline{\mathbf{X}}=\big[[\mathbf{A}_1,\mathbf{A}_2,\mathbf{A}_3]\big]+\underline{\mathbf{O}}+\underline{\mathbf{E}}$ and $\mathbf{G}_n=\mathbf{A}_n\,diag\,(\mathbf{d}_n)\mathbf{A}_n^\top+\mathbf{O}_n+\mathbf{V}_n$ for the tensor and the graph matrices respectively. The variables $\underline{\mathbf{O}}\in\mathbb{R}^{I_1\times I_2\times I_3}$ and $\{\mathbf{O}_n\in\mathbb{R}^{I_n\times R}\}_n$ model the anomalies in the tensor and graphs that should occur infrequently, and hence most entries of $\underline{\mathbf{O}}$ and $\{\mathbf{O}_n\}_n$ are zero. Hence, the optimization (3) and the ADMM solver can be readily modified to obtain sparse estimates of $\underline{\mathbf{O}}$ and $\{\mathbf{O}_n\}_n$ as well; see e.g., [22] and [20].

3.2 Convergence

Here, convergence of Algorithm 1 is examined when all the measurements are available $\{\mathbf{G}_n^A = \mathbf{G}_n\}_n$, and $\underline{\mathbf{X}}^A = \underline{\mathbf{X}}$, the extension for the case with misses is straightforward [23].

A point $\mathbf{\Phi} := (\{\mathbf{A}_n, \bar{\mathbf{A}}_n, \tilde{\mathbf{A}}_n, \mathbf{d}_n, \tilde{\mathbf{d}}_n\}_{n=1}^3)$ satisfies the Karush-Kuhn-Tucker (KKT) conditions for problem (6) if there exist dual variables $\mathbf{\Psi} := (\{\mathbf{Y}_{\tilde{\mathbf{A}}_n}, \mathbf{Y}_{\tilde{\mathbf{A}}_n}, \mathbf{y}_{\tilde{\mathbf{d}}_n}\}_{n=1}^3)$ such that

$$\begin{aligned}
(\mathbf{X}_n - \mathbf{A}_n\mathbf{M}_n^\top)\mathbf{M}_n + \mu(\mathbf{G}_n - \mathbf{A}_n\mathbf{D}_n\bar{\mathbf{A}}_n^\top)\bar{\mathbf{A}}_n\mathbf{D}_n \\
-\mathbf{Y}_{\tilde{\mathbf{A}}_n} - \mathbf{Y}_{\bar{\mathbf{A}}_n} = \mathbf{0} \\
\mu(\bar{\mathbf{A}}_n \odot \mathbf{A}_n)^\top(\mathbf{g}_n - \bar{\mathbf{A}}_n \odot \mathbf{A}_n\mathbf{d}_n) - \mathbf{y}_{\tilde{\mathbf{d}}_n} = \mathbf{0} \\
\mu(\mathbf{G}_n - \bar{\mathbf{A}}_n\mathbf{D}_n\mathbf{A}_n^\top)\mathbf{A}_n\mathbf{D}_n - \mathbf{Y}_{\tilde{\mathbf{A}}_n} = \mathbf{0} \qquad (9) \\
\mathbf{A}_n - \bar{\mathbf{A}}_n = \mathbf{0}, \quad \mathbf{A}_n - \tilde{\mathbf{A}}_n = \mathbf{0}, \quad \mathbf{d}_n - \tilde{\mathbf{d}}_n = \mathbf{0} \\
\mathbf{Y}_{\tilde{\mathbf{A}}_n} \leq \mathbf{0} \geq \tilde{\mathbf{A}}_n, \quad \mathbf{y}_{\tilde{\mathbf{d}}_n} \leq \mathbf{0} \geq \tilde{\mathbf{d}}_n \\
\mathbf{Y}_{\tilde{\mathbf{A}}_n} * \tilde{\mathbf{A}}_n = \mathbf{0}, \quad \mathbf{y}_{\tilde{\mathbf{d}}_n} * \tilde{\mathbf{d}}_n = \mathbf{0}, \quad n = 1, 2, 3.
\end{aligned}$$

Proposition 1. *Let $\{\mathbf{\Phi}^l, \mathbf{\Psi}^l\}_l$ be a sequence generated by Algorithm 1. If the sequence of dual variables $\{\mathbf{\Psi}^l\}_l$ is bounded and satisfies*

$$\sum_{l=0}^{\infty} \|\mathbf{\Psi}^{l+1} - \mathbf{\Psi}^l\|_F^2 < \infty, \qquad (10)$$

then any accumulation point of $\{\mathbf{\Phi}^l\}_l$ satisfies the KKT conditions of (6). Hence, any accumulation point of $\{\{\mathbf{A}_n^l, \mathbf{d}_n^l\}_{n=1}^3\}_l$ satisfies the KKT conditions for problem (3).

Proof. See Section 7. □

Proposition 1 suggests that upon convergence of the dual variables $\{\mathbf{\Psi}^l\}_l$, the sequence $\{\mathbf{\Phi}^l\}_l$ reaches a KKT point. Note that the closed-form updates of Algorithm 1 are instrumental in establishing the convergence claim. Empirical convergence with numerical tests is provided in Section 5.

Algorithm 1. ADMM for CGTF

Input: $\underline{\mathbf{X}}^A$ and $\{\mathbf{G}_n^A\}_{n=1}^3$
1: *Initialization:* SNMF for $\{\mathbf{A}_n\}_n$ using [21].
2: **while** iterates not converge **do**
3: Update $\hat{\mathbf{A}}_n$ using (8a).
4: Update $\hat{\mathbf{d}}_n$ using (8b).
5: Update $\hat{\bar{\mathbf{A}}}_n$ using (8c).
6: Update $\{\hat{\tilde{\mathbf{A}}}_n, \hat{\tilde{\mathbf{d}}}_n\}$ using (8d).
7: Update $\hat{\underline{\mathbf{X}}}^M$ using (8e).
8: Update $\hat{\mathbf{G}}_n^M$ using (8f).
9: Update Lagrange multipliers using (8g).
10: **end while**
Output: $\hat{\underline{\mathbf{X}}}^M, \{\hat{\mathbf{A}}_n, \hat{\mathbf{d}}_n, \hat{\mathbf{G}}_n^M\}_n$

4 Community Detection via CGTF

A task of major practical importance in network science is the identification of groups of vertices or communities that are more densely connected to each other than to the rest of the nodes in the network. Community detection unveils the structure of the network and facilitates a number of applications. For example, clustering web clients improves the performance of web services, identifying communities among customers leads to accurate recommendations, or grouping proteins based on their dependencies enables the development of targeted drugs [5]. This section exemplifies how the novel CGTF can recover the communities in graphs even when some links are missing; what is known as the cold start problem.

Community detection methods aim to learn for each node $i \in \{1, \ldots, I_n\}$ of $\mathbf{G}_n$ a mapping to a cluster assignment $\alpha_{n,i} \in \{1, \ldots, C_n\}$, where C_n is the number of communities in the nth graph. Collecting all the nodal assignments, one seeks an $I_n \times 1$ vector $\boldsymbol{\alpha}_n := [\alpha_{n,1}, \ldots, \alpha_{n,I_n}]^\top$.

If C_n is not known a priori, the recovered factor $\mathbf{A}_n$ can be directly utilized to provide a community assignment. First, we scale $\mathbf{A}_n$ to account for the weighting vector $\mathbf{C}_n := \mathbf{A}_n\, diag(\sqrt{\mathbf{d}_n})$. The largest entry in each row of $\mathbf{C}_n$ indicates clustering assignments [16]. Specifically, we estimate the community assignment of a node i as

$$\hat{\alpha}_{n,i} = \arg\max_{r=1,\ldots,R} [\mathbf{C}_n]_{(i,r)}, \qquad (11)$$

and $\hat{\boldsymbol{\alpha}}_n := [\hat{\alpha}_{n,1}, \ldots, \hat{\alpha}_{n,I_n}]^\top$ is the estimated assignment vector. Hence, in lieu of prior information about C_n we implicitly assume that $C_n = R$ for $n = 1, 2, 3$.

Oftentimes, in CD problems C_n is available. If $C_n \neq R$ one cannot apply directly (11) to recover the communities. In such a case, we regard $\mathbf{C}_n$ as a representation of $\mathbf{G}_n$ in a latent space of lower dimension. Hence, we apply the celebrated k-means algorithm [24] obtain

$$\hat{\boldsymbol{\alpha}}_n = k\text{-means}(\mathbf{C}_n, C_n). \qquad (12)$$

The community assignment procedure is summarized in Algorithm 2. Note that the discussed method amounts to a hard community assignment in the sense that each node is assigned to exactly one community. Nonetheless, the factors can be utilized to perform soft community assignment, where one node may belong to more than one communities. If the rows of $\mathbf{C}_n$ are normalized to sum to 1, $[\mathbf{C}_n]_{(i,r)}$ can be interpreted as the probability of the ith node belonging to the rth community.

Algorithm 2. Community Detection via CGTF

Input: $\underline{\mathbf{X}}^A$, $\{\mathbf{G}_n^A\}_{n=1}^3$, and $\{C_n\}_{n=1}^3$
1: *Initialization:* Algorithm 1 for $\{\mathbf{A}_n, \mathbf{d}_n\}_{n=1}^3$.
2: **for** $n = 1, 2, 3$
3: $\mathbf{C}_n := \mathbf{A}_n\, diag\,(\sqrt{\mathbf{d}_n})$
4: **if** $C_n = R$ **do**
5: Compute $\hat{\boldsymbol{\alpha}}_n$ using (11)
6: **else**
7: Compute $\hat{\boldsymbol{\alpha}}_n$ using (12)
Output: $\{\hat{\boldsymbol{\alpha}}_n\}_n$

4.1 Community Detection Evaluation

For a graph of I nodes and graph adjacency matrix $\mathbf{G}$, we define the cover set $\mathcal{S} := \{\mathcal{C}_c\}_{c=1}^C$ where $\mathcal{C}_c$ contains all the nodes that belong to community c as captured by the assignment vector $\boldsymbol{\alpha}$, i.e., $\mathcal{C}_c := \{i | \text{s.t. } \alpha_i = c\}$. The estimated cover set is defined as $\hat{\mathcal{S}}$ that uses $\hat{\boldsymbol{\alpha}}$ from Algorithm 2.

For networks with ground truth communities, we employ the normalized mutual information (NMI) metric

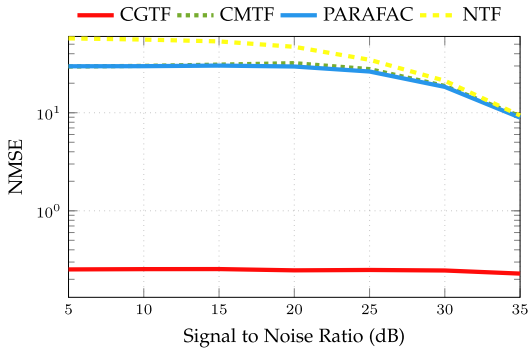


Fig. 3. Tensor imputation performance based on NMSE.

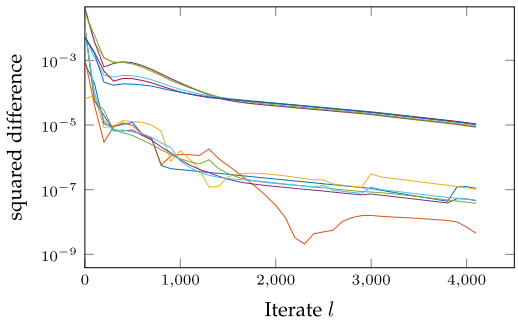


Fig. 4. Convergence of ADMM iterates $\{\|\mathbf{\Phi}^l - \mathbf{\Phi}^{l-1}\|_F^2, \|\mathbf{\Psi}^l - \mathbf{\Psi}^{l-1}\|_F^2\}_l$, and $\|\{\mathbf{A}_n^l - \tilde{\mathbf{A}}_n^l\|_F^2, \|\mathbf{A}_n^l - \bar{\mathbf{A}}_n^l\|_F^2, \|\mathbf{d}_n^l - \tilde{\mathbf{d}}_n^l\|_F^2\}_l$.

[25] to evaluate the recovered communities, $\hat{\boldsymbol{\alpha}}$. The NMI takes values between 0 and 1 and is defined as

$$\text{NMI} := \frac{2\text{I}(\mathcal{S}, \hat{\mathcal{S}})}{\text{H}(\mathcal{S}) + \text{H}(\hat{\mathcal{S}})}, \tag{13}$$

where H denotes the entropy ($|\mathcal{C}|$ is the cardinality of $\mathcal{C}$)

$$\text{H}(\mathcal{S}) := -\sum_{c=1}^{C} \frac{|\mathcal{C}_c|}{I} \log \frac{|\mathcal{C}_c|}{I}, \tag{14}$$

and $\text{I}(\mathcal{S}, \hat{\mathcal{S}})$ stands for the mutual information (MI) between $\mathcal{S}$ and $\hat{\mathcal{S}}$ defined as

$$\text{I}(\mathcal{S}, \hat{\mathcal{S}}) := \sum_{c=1}^{C} \sum_{c'=1}^{\hat{C}} \frac{|\hat{\mathcal{C}}_{c'} \cap \mathcal{C}_c|}{I} \log \frac{|\hat{\mathcal{C}}_{c'} \cap \mathcal{C}_c| I}{|\hat{\mathcal{C}}_{c'}||\mathcal{C}_c|}. \tag{15}$$

Whereas MI encodes how similar two community cover sets are, the entropy measures the level of uncertainty in each cover set individually; see e.g., [5]. For successful clustering algorithms, the resulting NMI is close to 1, and otherwise 0.

On the other hand, to evaluate the quality of a recovered community $\hat{\mathcal{C}}$ even without ground-truth community labels, the conductance $\pi(\hat{\mathcal{C}})$ is traditionally employed [26]

$$\pi(\hat{\mathcal{C}}) := \frac{\sum_{i \in \hat{\mathcal{C}}} \sum_{i' \notin \hat{\mathcal{C}}} [\mathbf{G}]_{(i,i')}}{\min(\text{vol}(\hat{\mathcal{C}}), \text{vol}(\hat{\mathcal{C}}^c))} \tag{16}$$

where

$$\text{vol}(\hat{\mathcal{C}}) := \sum_{i \in \hat{\mathcal{C}}} \sum_{i'=1}^{I} [\mathbf{G}]_{(i,i')}, \tag{17}$$

and the set $\hat{\mathcal{C}}^c$ contains all nodes in the graph not in $\hat{\mathcal{C}}$. For successful CD, the connections among nodes in $\hat{\mathcal{C}}$ are dense and otherwise sparse that leads to small scores of $\pi(\hat{\mathcal{C}})$.

A metric that summarizes the conductance across communities $\{\hat{\mathcal{C}}_c\}_c \in \hat{\mathcal{S}}$ is the so-termed coverage

$$\chi(\hat{\mathcal{S}}, \alpha) := \frac{1}{I} \Big| \bigcup_{\pi(\hat{\mathcal{C}}_c) < \alpha} \hat{\mathcal{C}}_c \Big|, \quad \{\hat{\mathcal{C}}_c\}_c \in \hat{\mathcal{S}} \tag{18}$$

where $\alpha \in [0, 1]$ is a suitable threshold. The coverage gives the portion of nodes that belong to communities with conductance less than α and since low conductance scores correspond to more cohesive communities, large values of coverage for small thresholds are desirable.

5 Experimental Evaluation

This section evaluates the performance of the proposed CGTF on synthetic and real data. The approaches compared include the CANDECOMP/PARAFAC Weighted OPTimization (PARAFAC) algorithm [10]; the nonnegative tensor factorization (NTF) implemented as in [27]; and the CMTF [8]. The algorithms were initialized using the proposed SNMF scheme, which enhances the performance of all methods. Unless stated otherwise, the following parameters were selected for CGTF: $\{\rho_{\tilde{\mathbf{A}}_n} = 100, \rho_{\bar{\mathbf{A}}_n} = 100, \rho_{\tilde{\mathbf{d}}_n} = 100\}_n, \mu = 1$.[1]

5.1 Tensor Imputation

Synthetic tensor data $\underline{\mathbf{X}} \in \mathbb{R}^{350 \times 350 \times 30}$ with $R = 4$ was generated according to the PARAFAC model (1), where the true factors $\{\mathbf{A}_n\}_{n=1}^3$ are drawn from a uniform distribution. Matrices $\{\mathbf{G}_n\}_{n=1}^3$ were generated using the SMNF (2).

To evaluate the performance of the various factorization algorithms, the entries of $\underline{\mathbf{X}}$ were corrupted with i.i.d. Gaussian noise. Fig. 3 depicts the normalized mean squared error $\text{NMSE} := \sum_{i_3=1}^{I_3} \|\hat{\underline{\mathbf{X}}}(:,:,i_3) - \underline{\mathbf{X}}(:,:,i_3)\|_F^2 / \sum_{i_3=1}^{I_3} \|\underline{\mathbf{X}}(:,:,i_3)\|_F^2$ against the signal-to-noise ratio (SNR) of the tensor data. The novel CGTF exploits the graph adjacency matrices and achieves superior performance relative to the competing methods.

Furthermore, the convergence of the proposed approach is evaluated. Fig. 4 testifies to the theoretical convergence results established in Prop. 1.

5.2 Community Detection

To evaluate the performance of the CGTF in detecting communities, we employed the Lancicchinetti-Fortunato-Radicci (LFR) benchmark [28] that generates graphs with ground truth communities. LFR graphs capture properties of real-world networks such as heterogeneity in the distributions of node degrees and also in the community sizes.

First, we generated 3 LFR networks $\{\mathbf{G}_n\}_{n=1}^3$ with $I_1 = 100$, $I_2 = 300$, and $I_3 = 500$ nodes, correspondingly comprising $C_1 = 5$, $C_2 = 3$, $C_3 = 4$ communities; see Fig. 5. We recover the factors $\{\mathbf{A}_n\}_n$ of $\{\mathbf{G}_n\}_n$ using SNMF, and construct $\underline{\mathbf{X}}$ using (1). Next, we observe noisy versions of the tensor data and the corresponding graph adjacency matrices; for $\mathbf{G}_1$ we observe only 10 percent of its entries and $R = 5$.

1. The ADMM implementation of the proposed CGTF method can be found in https://github.com/bioannidis/Coupled_tensors_graphs

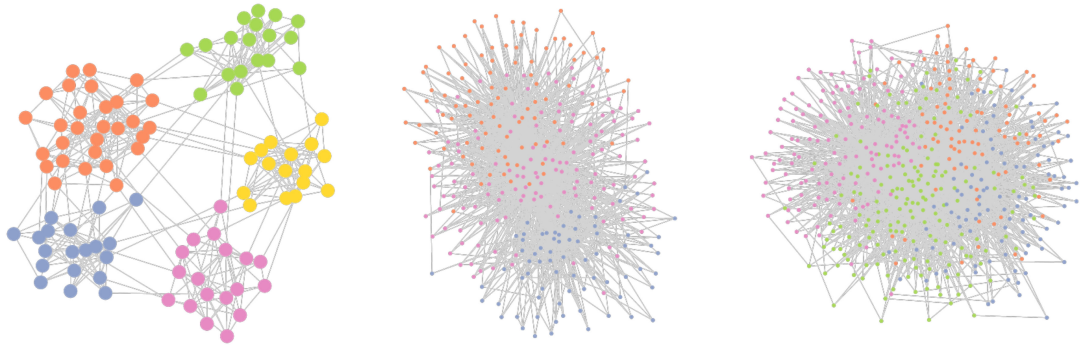
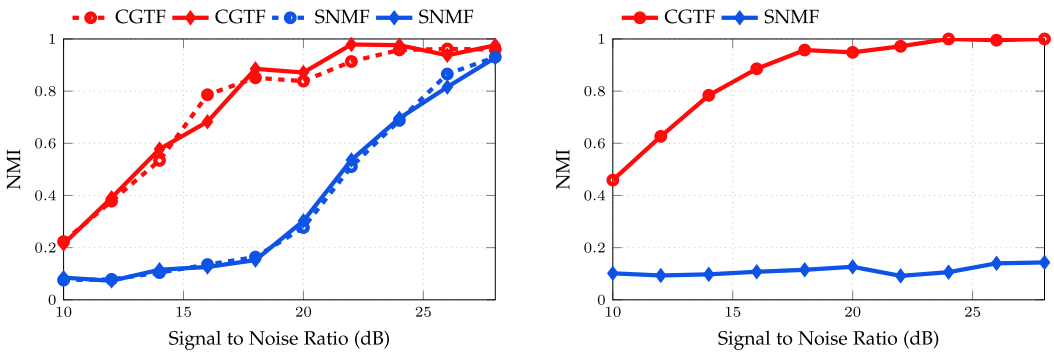


Fig. 5. LFR clustered graphs; $\mathbf{G}_1$ left, $\mathbf{G}_2$ middle, and $\mathbf{G}_3$ right.



(a) $\mathbf{G}_2$ (dashed) and $\mathbf{G}_3$ (solid) (b) $\mathbf{G}_1$

Fig. 6. Community detection performance based on NMI.

Fig. 6a shows the NMI performance of CGTF and SNMF [16], as we increase the SNR for $\mathbf{G}_2$ and $\mathbf{G}_3$. The proposed approach recovers successfully robust community assignments.

Furthermore, Fig. 6b depicts the NMI performance of the algorithms with 90 percent entries of $\mathbf{G}_1$ missing. As expected, SNMF cannot recover the community assignments of the nodes in this partially observed $\mathbf{G}_1$. On the other hand, the novel CGTF exploits the tensor data, copes with missing links, and provides reliable estimates of $\boldsymbol{\alpha}_1$.

5.3 Activities of Users at Different Locations

To assess the potential of our approach in providing accurate recommendations, we further tested a real recommendation dataset that comprises a three-way tensor indicating the frequency of a user performing an activity at a certain location [29]. It contains information about 164 users, 168 locations and 5 activities. A binary tensor $\underline{\mathbf{X}}$ is constructed to represent the links between users, their locations and corresponding activities. In other words, $\underline{\mathbf{X}}(i_1, i_2, i_3)$ equals 1 if user i_1 visited location i_2 and performed activity i_3; otherwise, it is 0. Additionally, similarity matrices between the users and the activities are provided. The similarity value between two locations is defined by the inner product of the corresponding feature vectors. The dataset is missing social network information for 28 users, and feature vectors for 32 locations. The parameters of CGTF were $\{\rho_{\tilde{\mathbf{A}}_n} = 100, \rho_{\tilde{\mathbf{A}}_n} = 100, \rho_{\tilde{\mathbf{d}}_n} = 100\}_n, \mu = 10^{-4}$, and for all approaches $R = 5$.

TABLE 1
NMSE for Different Ratios of Missing Data

Missing	NTF	PARAFAC	CMTF	CGTF
40%	0.995	1.016	0.98	**0.46**
50%	0.99	0.96	0.99	**0.68**

Table 1 lists the NMSE for variable percentages of missing tensor data. The CGTF model exploits judiciously the structure of the available graph information, which enables our efficient ADMM solver to outperform competing alternatives, and lead to improved recommendations.

In order to assess the recommendation quality of the proposed approach, we changed the thereshold for detecting an activity (edge) on the tensor (graphs). Per threshold value, we then obtained the probabilities of detection and false alarm.

Fig. 8 depicts the receiver operating characteristic (ROC) for the tensor entries, and as expected the novel CGTF outperforms the alternative. Moreover, Fig. 7 shows the ROC for discovering concealed links in the user-graph with only 10 percent of observed graph entries when the factors are initialized either using the SNMF or randomly. In both cases, CGTF performs successful edge identification and yields accurate link predictions. The performance gap among CGTF and CMTF, becomes more pronounced when the factors are initialized randomly, which suggests that initialization is crucial in achieving a good stationary point.

5.3.1 *Community Detection*

Furthermore, CD is pursued for the user and location graphs with 70 percent entries missing in the tensor and no misses in the graphs. We compare our CD performance against the

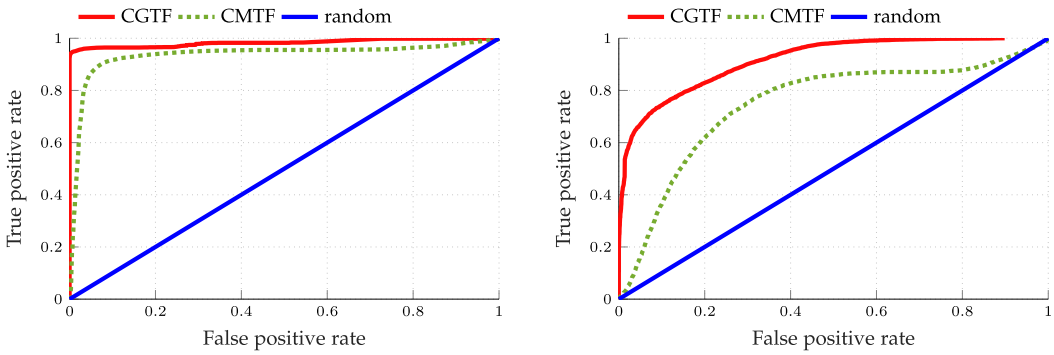


Fig. 7. ROC curve for $\mathbf{G}_1$ using the SNMF for initialization of $\mathbf{A}_1$(left), random initialization (right) with 40 percent misses in $\underline{\mathbf{X}}$ and 90 percent misses in $\mathbf{G}_1$ and $\mathbf{G}_2$.

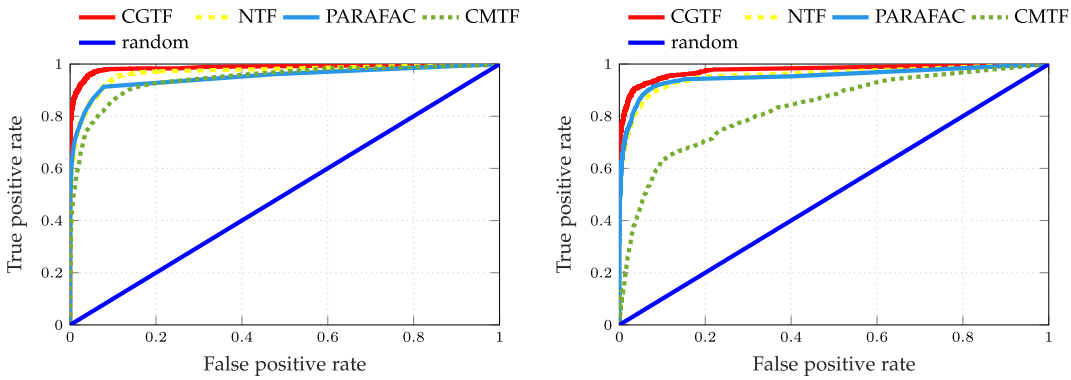


Fig. 8. ROC for 40 percent (left); and 50 percent (right) tensor missing entries.

following baselines: Potts [13], NewmanF [12], SP [14], AFG [15] and SNMF [16].[2] In lieu of ground-truth communities, we evaluate the CD performance by the maximum conductance-coverage curve. This curve is plotted by varying α from 0 to 1 (cf. (18)) and reporting the corresponding coverage value on the x-axis and maximum conductunce the y-axis. Low values of conductance for large values of coverage correspond to more cohesive communities. Hence, a smaller area under curve (AUC) implies better performance; see Section 4.1. Fig. 9 reports the coverage scores relative to the maximum conductance (α) for the users graph (left) and the locations graph (right). The proposed CGTF achieves higher coverage scores for smaller conductance and outperforms competing approaches. CGTF achieves the smallest AUC value in the user graph and one of the smallest in the location graph. Hence, the factors obtained by our coupled approach indeed improve CD performance.

5.4 Posts of Users in a Social Network

We also tested the performance of CGTF on the Digg dataset. Digg is a social network that allows users to submit, Digg, and comment on news stories. In [30], the data was collected from a large subset of users and stories. The dataset includes stories, and users along with their time-stamped actions with respect to stories, as well as the social network of users. In addition, a set of keywords is assigned to each story.

After discretizing the time into 20 time intervals over 3 days, we construct a tensor comprising the number of comments that user i wrote on story j during the kth time interval stored in the (i, j, k) item. Also, a story-story graph is constructed where any two stories are connected only if they share more than two keywords. The original tensor containing all users and stories includes a large number of inactive users and unpopular stories. In order to assess performance of the proposed method, the data is subsampled so that the 175 most active users and the 800 most popular stories are kept. Hence, the size of the tensor in this experiment is $I_1 = 175$ users, $I_2 = 800$ stories and $I_2 = 20$ time intervals. In addition, the side information comprises two graphs that represent the users' social network and the similarities of the stories.

The tensor and the two graphs are fused jointly as in (3) with $R = 10$. Then, the proposed ADMM-based algorithm is employed to obtain the latent factors of the CGTF model. As there is no graph on the third mode (time intervals), the term $\|\mathbf{G}_3 - \mathbf{A}_3\, diag\,(\mathbf{d}_3)\mathbf{A}_3^\top\|_F^2$ is not included in (3). We assume that 40 percent of the tensor entries, as well as 30 percent of the links in the user-user and story-story graphs are missing.

In Figs. 10 and 11, the ROC is presented for the tensor and the graphs. The proposed approach outperforms competing approaches in completing the missing tensor entries as well as predicting the missing links in the graph, and leads to accurate recommendations for previously unseen data.

5.4.1 *Community Detection under Missing Links*

In this experiment we assume that 40 percent of the tensor entries and 50 percent of the graph links are missing. The

2. We use the Matlab implementations provided by the authors.

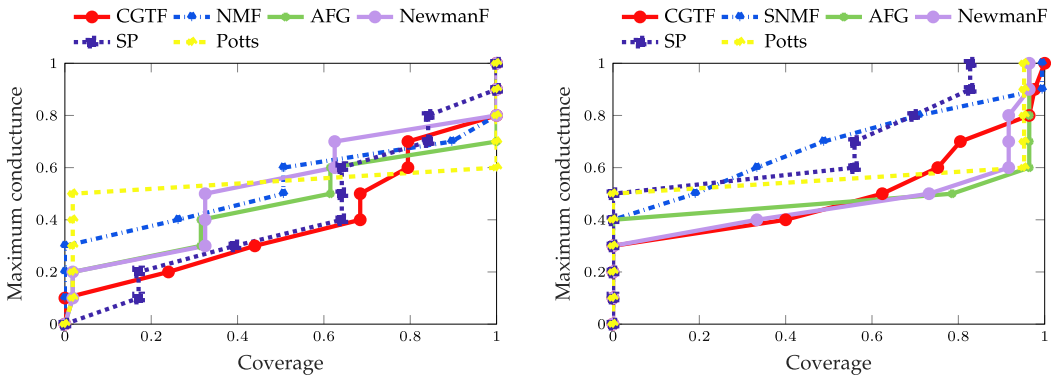


Fig. 9. Community detection performance based on coverage for the user graph (left); and the location graph (right).

goal here is to examine whether CGTF recovers the communities in the graphs even with hidden graph links. Fig. 12 reports the coverage scores relative to the maximum conductance for the users graph (left) and the stories graph (right).[3] Competing approaches that only utilize the partially observed graphs can not recover crisp graph communities. On the other hand, our novel CGTF utilizes judiciously the partially observed graphs and tensors and reports superior performance. The advantage of the proposed framework in community detection is more evident in this experiment (compare Figs. 9 and 12).

5.5 Runtime Comparisons

The scalabilty of CGTF is reflected on the relative runtime comparisons listed in Fig. 13, for recovering the tensor entries for the Activities and Digg datasets in Figs. 8 and 10 respectively. All experiments were run on a machine with i7-4790 @3.60 Ghz CPU, and 32 GB of RAM. We used the Matlab implementations provided by the authors of the compared algorithms. The bars in Fig. 13 indicate the runtime of the algorithms relative to CGTF's runtime. Evidently, our efficient yet effective CGTF implementation is almost as fast as the PARAFAC, while achieving superior tensor imputation performance (see Figs. 8 and 10).

6 Conclusions and Future Work

This paper investigates the inference of unavailable entries in tensors and graphs based on a novel CGTF model. An efficient algorithm is developed to identify the factor matrices and recover the missing entries. The ADMM solver features closed-form updates and is amenable to parallel and accelerated implementation. In addition, the proposed method can overcome the so-called *cold-start* problem, where the tensor has missing slabs or the similarity matrices are not complete. The novel algorithm makes accurate prediction of the missing values and can be used in many real world settings, especially in recommender systems. A novel direction is further explored by employing the interpretable factors of CGTF to detect communities of nodes in the graphs having the tensor as side information. Through numerical tests with synthetic as well as real-data, the novel algorithm was observed to perform markedly better than existing alternatives and further yield accurate recommendations, as well as effective identification of communities.

Our future research agenda will focus in two direction. Todays era of data deluge has grown the interest for robust methods that can handle anomalies in collections of high-dimensional data. Towards this end, we aim at a robust CGTF to handle anomalies in the tensor and graph data. Furthermore, in many scenarios prior information on the tensor and graph data can be accounted for to improve imputation performance. CGTF may incorporate such knowledge by introducing a probabilistic prior for certain graphs e.g., stochastic block models [31].

7 Proof of Proposition 1

In what follows, we omit the terms $\underline{\mathbf{X}}^M, \{\mathbf{G}_n^M\}_{n=1}^3$, although the proof can be easily modified to accommodate misses in the graphs and in the tensor. First, we claim

$$\begin{aligned} \mathbf{\Phi}^{l+1} - \mathbf{\Phi}^{l} &\to \mathbf{0}, \\ \mathbf{\Psi}^{l+1} - \mathbf{\Psi}^{l} &\to \mathbf{0}. \end{aligned} \tag{19}$$

Observe that the Lagragian $L(\mathbf{\Phi}, \mathbf{\Psi})$ is bounded from below which follows because

$$L(\mathbf{\Phi}, \mathbf{\Psi}) := f(\mathbf{\Phi}) + \sum_{f=1}^{3} \left\{ \frac{\rho_{\bar{\mathbf{A}}_n}}{2} \|\mathbf{A}_n - \bar{\mathbf{A}}_n + \frac{\mathbf{Y}_{\bar{\mathbf{A}}_n}}{\rho_{\bar{\mathbf{A}}_n}}\|_F^2 \right.$$

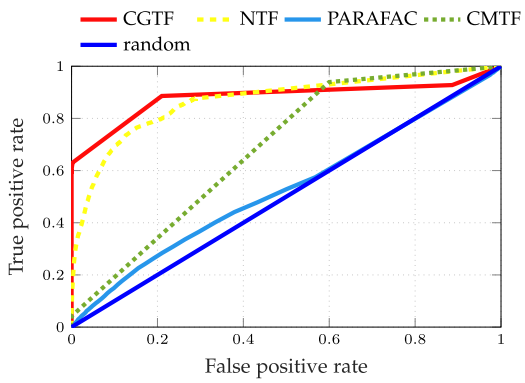


Fig. 10. ROC for 40 percent tensor missing entries.

3. AFG did not provide meaningful results and was not included.

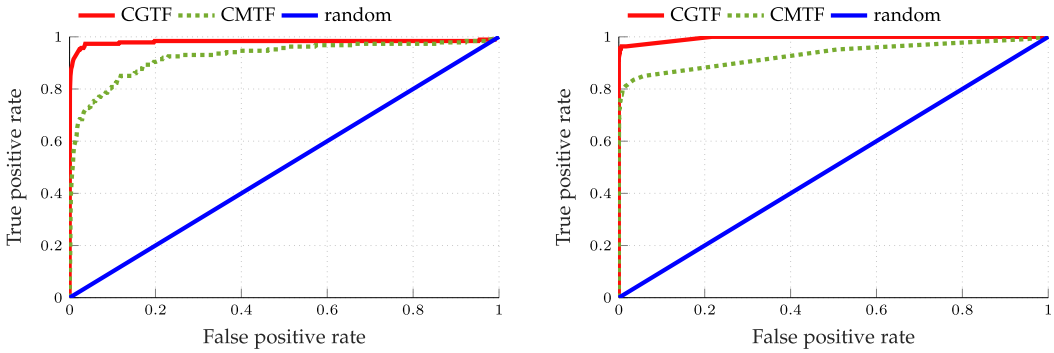


Fig. 11. ROC for the prediction in the users' social network $\mathbf{G}_1$ (left); and the story graph adjacency $\mathbf{G}_2$ (right).

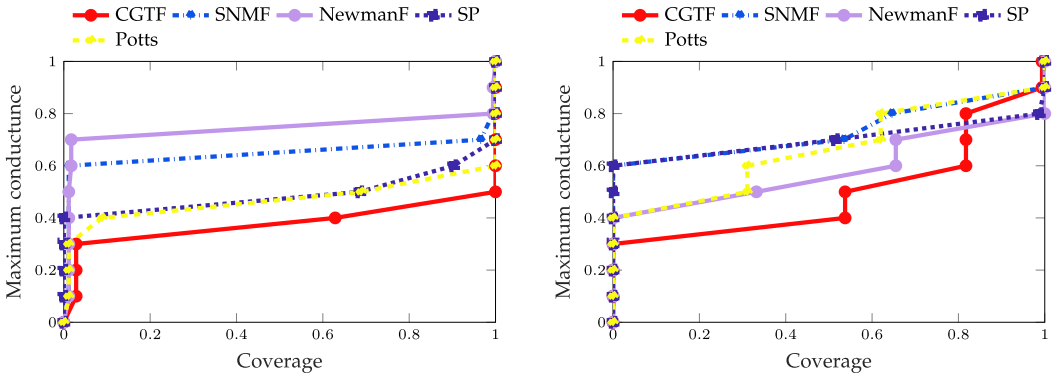


Fig. 12. Community detection performance based on coverage for the user graph (left); and the story graph (right).

$$-\frac{1}{2\rho_{\tilde{\mathbf{A}}_n}}\|\mathbf{Y}_{\tilde{\mathbf{A}}_n}\|_F^2+\frac{\rho_{\tilde{\mathbf{A}}_n}}{2}\|\mathbf{A}_n-\tilde{\mathbf{A}}_n+\frac{\mathbf{Y}_{\tilde{\mathbf{A}}_n}}{\rho_{\tilde{\mathbf{A}}_n}}\|_F^2$$
$$-\frac{1}{2\rho_{\tilde{\mathbf{d}}_n}}\|\mathbf{Y}_{\tilde{\mathbf{d}}_n}\|_F^2+\frac{\rho_{\tilde{\mathbf{d}}_n}}{2}\|\mathbf{d}_n-\tilde{\mathbf{d}}_n+\frac{\mathbf{y}_{\tilde{\mathbf{d}}_n}}{\rho_{\tilde{\mathbf{d}}_n}}\|_2^2-\frac{1}{2\rho_{\tilde{\mathbf{d}}_n}}\|\mathbf{y}_{\tilde{\mathbf{d}}_n}\|_2^2\Big\},$$

and $\mathbf{\Psi}$ is bounded. Owing to the appropriate reformulation (6), $L(\cdot)$ is strongly convex w.r.t. each matrix variable $\mathbf{V}\in\{\mathbf{A}_n,\bar{\mathbf{A}}_n,\tilde{\mathbf{A}}_n,\mathbf{d}_n,\tilde{\mathbf{d}}_n\}_{n=1}^3$ separately. As a result, it holds for $\mathbf{V}$ that

$$L(\mathbf{V}+\delta\mathbf{V})-L(\mathbf{V})\geq\partial_{\mathbf{V}}L(\mathbf{V})^\top\delta\mathbf{V}+\rho\|\delta\mathbf{V}\|_F^2, \tag{20}$$

where ρ is a properly selected parameter, while the variables except $\mathbf{V}$ are kept the same. Moreover, if $\mathbf{V}^*:=\arg\min_{\mathbf{V}}L(\mathbf{V})$ it follows that $\partial_{\mathbf{V}}L(\mathbf{V}^*)^\top\delta\mathbf{V}\geq 0$. Hence, for $\delta\mathbf{V}=\mathbf{V}^l-\mathbf{V}^{l+1}$ and since $\mathbf{V}^{l+1}:=\arg\max_{\mathbf{V}}L(\mathbf{V})$ at the lth iteration, it follows from (20) that

$$L(\mathbf{V}^l)-L(\mathbf{V}^{l+1})\geq\rho\|\mathbf{V}^l-\mathbf{V}^{l+1}\|_F^2. \tag{21}$$

Specifying (21) to each variable in $\mathbf{\Phi}$, yields for $n=1,2,3$

$$L(\mathbf{A}_n^l)-L(\mathbf{A}_n^{l+1})\geq\frac{\rho_{\tilde{\mathbf{A}}_n}+\rho_{\tilde{\mathbf{A}}_n}}{2}\|\mathbf{A}_n^l-\mathbf{A}_n^{l+1}\|_F^2 \tag{22a}$$

$$L(\bar{\mathbf{A}}_n^l)-L(\bar{\mathbf{A}}_n^{l+1})\geq\frac{\rho_{\bar{\mathbf{A}}_n}}{2}\|\bar{\mathbf{A}}_n^l-\bar{\mathbf{A}}_n^{l+1}\|_F^2 \tag{22b}$$

$$L(\tilde{\mathbf{A}}_n^l)-L(\tilde{\mathbf{A}}_n^{l+1})\geq\frac{\rho_{\tilde{\mathbf{A}}_n}}{2}\|\tilde{\mathbf{A}}_n^l-\tilde{\mathbf{A}}_n^{l+1}\|_F^2 \tag{22c}$$

$$L(\mathbf{d}_n^l)-L(\mathbf{d}_n^{l+1})\geq\frac{\rho_{\tilde{\mathbf{d}}_n}}{2}\|\mathbf{d}_n^l-\mathbf{d}_n^{l+1}\|_F^2 \tag{22d}$$

$$L(\tilde{\mathbf{d}}_n^l)-L(\tilde{\mathbf{d}}_n^{l+1})\geq\frac{\rho_{\tilde{\mathbf{d}}_n}}{2}\|\tilde{\mathbf{d}}_n^l-\tilde{\mathbf{d}}_n^{l+1}\|_F^2. \tag{22e}$$

It follows then for $R:=\min\{\rho_{\tilde{\mathbf{A}}_n},\rho_{\bar{\mathbf{A}}_n},\rho_{\tilde{\mathbf{d}}_n}\}_n$ that

$$L(\mathbf{\Phi}^l,\mathbf{\Psi}^l)-L(\mathbf{\Phi}^{l+1},\mathbf{\Psi}^l)\geq R\|\mathbf{\Phi}^l-\mathbf{\Phi}^{l+1}\|_F^2. \tag{23}$$

On the other hand, it holds for the dual variables that

$$\begin{aligned}L\Big(\mathbf{Y}_{\tilde{\mathbf{A}}_n}^l\Big)-L\Big(\mathbf{Y}_{\tilde{\mathbf{A}}_n}^{l+1}\Big)&=\mathrm{Tr}\Big(\mathbf{Y}_{\tilde{\mathbf{A}}_n}^l-\mathbf{Y}_{\tilde{\mathbf{A}}_n}^{l+1}\Big)^\top\Big(\bar{\mathbf{A}}_n^l-\bar{\mathbf{A}}_n^{l+1}\Big)\\&=-\frac{1}{\rho_{\tilde{\mathbf{A}}_n}}\|\mathbf{Y}_{\tilde{\mathbf{A}}_n}^l-\mathbf{Y}_{\tilde{\mathbf{A}}_n}^{l+1}\|_F^2,\end{aligned} \tag{24a}$$

where the last equality follows from (8g), and similarly

$$L\Big(\mathbf{Y}_{\tilde{\mathbf{A}}_n}^l\Big)-L\Big(\mathbf{Y}_{\tilde{\mathbf{A}}_n}^{l+1}\Big)=-\frac{1}{\rho_{\tilde{\mathbf{A}}_n}}\|\mathbf{Y}_{\tilde{\mathbf{A}}_n}^l-\mathbf{Y}_{\tilde{\mathbf{A}}_n}^{l+1}\|_F^2 \tag{24b}$$

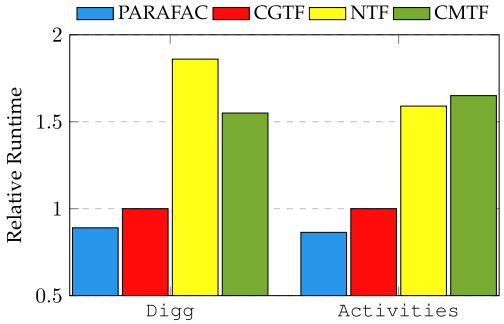


Fig. 13. Runtime comparisons relative to CGTF.

$$L\left(\mathbf{y}^{l}_{\tilde{\mathbf{d}}_n}\right) - L\left(\mathbf{y}^{l+1}_{\tilde{\mathbf{d}}_n}\right) = -\frac{1}{\rho_{\tilde{\mathbf{d}}_n}}\|\mathbf{y}^{l}_{\tilde{\mathbf{d}}_n} - \mathbf{y}^{l+1}_{\tilde{\mathbf{d}}_n}\|_F^2. \tag{24c}$$

Hence, we find that

$$L(\mathbf{\Phi}^{l+1}, \mathbf{\Psi}^{l}) - L(\mathbf{\Phi}^{l+1}, \mathbf{\Psi}^{l+1}) \geq -\frac{1}{R}\|\mathbf{\Psi}^{l} - \mathbf{\Psi}^{l+1}\|_F^2, \tag{25}$$

and upon combining (25) and (23), we arrive at

$$\begin{aligned} &L(\mathbf{\Phi}^{l}, \mathbf{\Psi}^{l}) - L(\mathbf{\Phi}^{l+1}, \mathbf{\Psi}^{l+1}) \geq \\ &\qquad R\|\mathbf{\Phi}^{l} - \mathbf{\Phi}^{l+1}\|_F^2 - \frac{1}{R}\|\mathbf{\Psi}^{l} - \mathbf{\Psi}^{l+1}\|_F^2. \end{aligned} \tag{26}$$

Since $L(\cdot)$ is bounded, we have

$$\sum_{l=0}^{\infty} R\|\mathbf{\Phi}^{l} - \mathbf{\Phi}^{l+1}\|_F^2 - \frac{1}{R}\|\mathbf{\Psi}^{l} - \mathbf{\Psi}^{l+1}\|_F^2 < \infty, \tag{27}$$

and after applying (10) we establish that $\mathbf{\Phi}^{l+1} - \mathbf{\Phi}^{l} \to \mathbf{0}$ and $\mathbf{\Psi}^{l+1} - \mathbf{\Psi}^{l} \to \mathbf{0}$.

Next, we rewrite the ADMM updates in (8) as

$$\begin{aligned} &[\mathbf{A}_n^{l+1} - \mathbf{A}_n^{l})(\mathbf{M}_n^{\top}\mathbf{M}_n + \mu \mathbf{D}_n^{l}\bar{\mathbf{A}}_n^{l\,\top}\bar{\mathbf{A}}_n^{l}\mathbf{D}_n^{l} + (\rho_{\tilde{\mathbf{A}}_n} + \rho_{\bar{\mathbf{A}}_n})\mathbf{I}] \\ &= (\mathbf{X}_n - \mathbf{A}_n^{l}\mathbf{M}_n^{\top})\mathbf{M}_n + \mu(\mathbf{G}_n - \mathbf{A}_n^{l}\mathbf{D}_n^{l}\bar{\mathbf{A}}_n^{l\,\top})\bar{\mathbf{A}}_n^{l}\mathbf{D}_n^{l} \\ &\quad + \rho_{\tilde{\mathbf{A}}_n}(\mathbf{A}_n^{l} - \tilde{\mathbf{A}}_n^{l}) + \rho_{\bar{\mathbf{A}}_n}(\mathbf{A}_n^{l} - \bar{\mathbf{A}}_n^{l}) - \mathbf{Y}^{l}_{\tilde{\mathbf{A}}_n} - \mathbf{Y}^{l}_{\bar{\mathbf{A}}_n} \end{aligned} \tag{28a}$$

$$\begin{aligned} &(\mathbf{d}_n^{l+1} - \mathbf{d}_n^{l})((\bar{\mathbf{A}}_n^{l} \odot \mathbf{A}_n^{l})^{\top}(\mu\bar{\mathbf{A}}_n^{l} \odot \mathbf{A}_n^{l}) + \rho_{\tilde{\mathbf{d}}_n}\mathbf{I}) \\ &= \mu(\bar{\mathbf{A}}_n^{l} \odot \mathbf{A}_n^{l})^{\top}(\mathbf{g}_n - \bar{\mathbf{A}}_n^{l} \odot \mathbf{A}_n^{l}\tilde{\mathbf{d}}_n^{l}) + \rho_{\mathbf{y}_{\tilde{\mathbf{d}}_n}}(\mathbf{d}_n^{l} - \tilde{\mathbf{d}}_n^{l}) - \mathbf{y}^{l}_{\tilde{\mathbf{d}}_n} \end{aligned} \tag{28b}$$

$$\begin{aligned} &(\bar{\mathbf{A}}_n^{l+1} - \bar{\mathbf{A}}_n^{l})(\mu\mathbf{D}_n^{l}\mathbf{A}_n^{l\,\top}\mathbf{A}_n^{l}\mathbf{D}_n^{l} + \rho_{\bar{\mathbf{A}}_n}\mathbf{I}) \\ &= \mu(\mathbf{G}_n - \bar{\mathbf{A}}_n^{l}\mathbf{D}_n^{l}\mathbf{A}_n^{l\,\top})\mathbf{A}_n^{l}\mathbf{D}_n^{l} + \rho_{\bar{\mathbf{A}}_n}(\mathbf{A}_n^{l} - \bar{\mathbf{A}}_n^{l}) - \mathbf{Y}^{l}_{\bar{\mathbf{A}}_n} \end{aligned} \tag{28c}$$

$$\tilde{\mathbf{A}}_n^{l+1} - \tilde{\mathbf{A}}_n^{l} = \left(\mathbf{A}_n^{l} + \frac{1}{\rho_{\tilde{\mathbf{A}}_n}}\mathbf{Y}^{l}_{\tilde{\mathbf{A}}_n}\right)_{+} - \tilde{\mathbf{A}}_n^{l} \tag{28d}$$

$$\tilde{\mathbf{d}}_n^{l+1} - \tilde{\mathbf{d}}_n^{l} = \left(\mathbf{d}_n^{l} + \frac{1}{\rho_{\tilde{\mathbf{d}}_n}}\mathbf{y}^{l}_{\tilde{\mathbf{d}}_n}\right)_{+} - \tilde{\mathbf{d}}_n^{l}, \tag{28e}$$

and for the dual updates

$$\begin{aligned} \mathbf{Y}^{l+1}_{\bar{\mathbf{A}}_n} - \mathbf{Y}^{l}_{\bar{\mathbf{A}}_n} &= \rho_{\bar{\mathbf{A}}_n}(\mathbf{A}_n^{l} - \bar{\mathbf{A}}_n^{l}) \\ \mathbf{Y}^{l+1}_{\tilde{\mathbf{A}}_n} - \mathbf{Y}^{l}_{\tilde{\mathbf{A}}_n} &= \rho_{\tilde{\mathbf{A}}_n}(\mathbf{A}_n^{l} - \tilde{\mathbf{A}}_n^{l}) \\ \mathbf{y}^{l+1}_{\tilde{\mathbf{d}}_n} - \mathbf{y}^{l}_{\tilde{\mathbf{d}}_n} &= \rho_{\tilde{\mathbf{d}}_n}(\mathbf{d}_n^{l} - \tilde{\mathbf{d}}_n^{l}). \end{aligned} \tag{28f}$$

Next, we leverage (19) and establish that the left hand side of the equations in (28) is equal to $\mathbf{0}$. Hence, from (28f) we deduce that $\mathbf{A}_n^{l} - \bar{\mathbf{A}}_n^{l} \to \mathbf{0}$, $\mathbf{A}_n^{l} - \tilde{\mathbf{A}}_n^{l} \to \mathbf{0}$, and $\mathbf{A}_n^{l} - \bar{\mathbf{A}}_n^{l} \to \mathbf{0}$. So far we have proved that the KKT conditions (9) relating to the primal variables $\mathbf{\Phi}$, are satisfied. The variables $\tilde{\mathbf{A}}_n$ and $\tilde{\mathbf{d}}_n$ are nonnegative by construction. For the dual variables, notice from (28f) that if $[\mathbf{A}_n^{l}]_{(i_n,r)} = [\tilde{\mathbf{A}}_n^{l}]_{(i_n,r)} = 0$ then $\left([\mathbf{Y}^{l}_{\tilde{\mathbf{A}}_n}]_{(i_n,r)}\right)_{+} = 0$, which implies that $[\mathbf{Y}^{l}_{\tilde{\mathbf{A}}_n}]_{(i_n,r)} \leq 0$, else if $[\mathbf{A}_n^{l}]_{(i_n,r)} = [\tilde{\mathbf{A}}_n^{l}]_{(i_n,r)} \geq 0$ then $[\mathbf{Y}^{l}_{\tilde{\mathbf{A}}_n}]_{(i_n,r)} = 0$. The same argument applies for $\mathbf{y}^{l}_{\tilde{\mathbf{d}}_n}$ and hence we have established satisfaction of the last KKT conditions concerning $\mathbf{Y}^{l}_{\tilde{\mathbf{A}}_n}$ and $\mathbf{y}^{l}_{\tilde{\mathbf{d}}_n}$.

ACKNOWLEDGMENTS

The work of V.N. Ioannidis and G.B. Giannakis was supported by NSF grants 1442686, 1514056, and 1711471. The work of A.S. Zamzam and N.D. Sidiropoulos was partially supported by NSF grant CIF-1525194. Preliminary results of this work were presented in [1], [2]. A summary of differences is included in the supplementary material.

REFERENCES

[1] V. N. Ioannidis, A. S. Zamzam, G. B. Giannakis, and N. D. Sidiropoulos, "Imputation of coupled tensors and graphs," in *Proc. Global Conf. Signal Inf. Process.*, Nov. 2018, pp. 1331–1335.

[2] A. S. Zamzam, V. N. Ioannidis, and N. D. Sidiropoulos, "Coupled graph tensor factorization," in *Proc. Asilomar Conf. Signal Syst. Comput.*, Nov. 2016, pp. 1755–1759.

[3] N. D. Sidiropoulos, L. D. Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, Jul. 2017.

[4] V. N. Ioannidis, M. Ma, A. Nikolakopoulos, G. B. Giannakis, and D. Romero, "Kernel-based inference of functions on graphs," in *Adaptive Learning Methods for Nonlinear System Modeling*, D. Comminiello and J. Principe, Eds. Amsterdam, The Netherlands: Elsevier, 2018.

[5] S. Fortunato, "Community detection in graphs," *Phys. Reports*, vol. 486, no. 3–5, pp. 75–174, 2010.

[6] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Comput.*, no. 8, pp. 30–37, Aug. 2009.

[7] A. P. Liavas and N. D. Sidiropoulos, "Parallel algorithms for constrained tensor factorization via alternating direction method of multipliers," *IEEE Trans. Signal Process.*, vol. 63, no. 20, pp. 5450–5463, Oct. 2015.

[8] B. Ermiş, E. Acar, and A. T. Cemgil, "Link prediction in heterogeneous data via generalized coupled tensor factorization," *Data Mining Knowl. Discovery*, vol. 29, no. 1, pp. 203–236, 2015.

[9] E. E. Papalexakis, C. Faloutsos, T. M. Mitchell, P. P. Talukdar, N. D. Sidiropoulos, and B. Murphy, "Turbo-SMT: Accelerating coupled sparse matrix-tensor factorizations by 200x," in *SDM*. Philadelphia, PA, USA: SIAM, 2014, pp. 118–126.

[10] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, "Scalable tensor factorizations with missing data," in *SDM*. Philadelphia, PA, USA: SIAM, 2010, pp. 701–712.

[11] J. A. Bazerque, G. Mateos, and G. B. Giannakis, "Rank regularization and bayesian inference for tensor completion and extrapolation," *IEEE Trans. Signal Process.*, vol. 61, no. 22, pp. 5689–5703, Nov. 2013.

[12] L. Danon, A. Díaz-Guilera, and A. Arenas, "The effect of size heterogeneity on community identification in complex networks," *J. Statistical Mech.: Theory Experiment*, vol. 2006, no. 11, pp. P11010–P11022, 2006.

[13] P. Ronhovde and Z. Nussinov, "Local resolution-limit-free potts model for community detection," *Phys. Rev. E*, vol. 81, no. 4, 2010, Art. no. 046114.

[14] J. P. Hespanha, "An efficient matlab algorithm for graph partitioning," 2004.

[15] A. Arenas, A. Fernandez, and S. Gomez, "Analysis of the structure of complex networks at different resolution levels," vol. 10, no. 5, pp. P053039–P053061, 2008.

[16] D. Kuang, H. Park, and C. H. Ding, "Symmetric nonnegative matrix factorization for graph clustering," in *SDM*, vol. 12. Philadelphia, PA, USA: SIAM, 2012, pp. 106–117.

[17] F. Huang, U. Niranjan, M. U. Hakeem, and A. Anandkumar, "Online tensor methods for learning latent variable models," *J. Mach. Learn. Res.*, vol. 16, no. 1, pp. 2797–2835, 2015.

[18] A. Anandkumar, R. Ge, D. Hsu, and S. M. Kakade, "A tensor approach to learning mixed membership community models," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 2239–2312, 2014.

[19] F. Sheikholeslami and G. B. Giannakis, "Overlapping community detection via constrained parafac: A divide and conquer approach," in *Proc. IEEE Int. Conf. Data Mining*, Nov. 2017, pp. 127–136.

[20] B. Baingana and G. B. Giannakis, "Joint community and anomaly tracking in dynamic networks," *IEEE Trans. Signal Process.*, vol. 64, no. 8, pp. 2013–2025, Sep. 2016.

[21] K. Huang and N. Sidiropoulos, "Putting nonnegative matrix factorization to the test: A tutorial derivation of pertinent Cramer-Rao bounds and performance benchmarking," *IEEE Signal Process. Mag.*, vol. 31, no. 3, pp. 76–86, Apr. 2014.
[22] M. Mardani, G. Mateos, and G. B. Giannakis, "Dynamic anomalography: Tracking network anomalies via sparsity and low rank," *IEEE J. Select. Topics Signal Process.*, vol. 7, no. 1, pp. 50–66, Feb. 2013.
[23] Y. Xu, W. Yin, Z. Wen, and Y. Zhang, "An alternating direction algorithm for matrix completion with nonnegative factors," *Frontiers Math. China*, vol. 7, no. 2, pp. 365–384, 2012.
[24] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *J. Roy. Statistical Soc.. Series C (Appl. Statist.)*, vol. 28, no. 1, pp. 100–108, 1979.
[25] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *J. Stat. Mech.: Theory Experiment*, vol. 2005, no. 09, Sep. 2005, Art. no. P09008.
[26] B. Bollobás, *Modern Graph Theory*, vol. 184, Berlin, Germany: Springer, 2013.
[27] C. A. Andersson and R. Bro, "The N-way toolbox for MATLAB," *Chemometrics Intell. Laboratory Syst.*, vol. 52, no. 1, pp. 1–4, 2000.
[28] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Phys. Rev. E*, vol. 78, Oct. 2008, Art. no. 046110.
[29] V. W. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang, "Collaborative filtering meets mobile recommendation: A user-centered approach," in *Proc. 24th AAAI Conf. Artif. Intell.*, 2010, vol. 10, pp. 236–241.
[30] Y.-R. Lin, J. Sun, P. Castro, R. Konuru, H. Sundaram, and A. Kelliher, "MetaFac: Community discovery via relational hypergraph factorization," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2009, pp. 527–536.
[31] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing, "Mixed membership stochastic blockmodels," *J. Mach. Learn. Res.*, vol. 9, no. Sep, pp. 1981–2014, 2008.

Vassilis N. Ioannidis (S'16) received the diploma degree in electrical and computer engineering from the National Technical University of Athens, Greece, in 2015, and the MSc degree in electrical engineering from the University of Minnesota (UMN), Twin Cities, Minneapolis, MN, in 2017. Currently, he is working towards the PhD degree in the Department of Electrical and Computer Engineering, University of Minnesota, Twin Cities. He received the Doctoral Dissertation Fellowship (2019) from the University of Minnesota. He also received Student Travel Awards from the IEEE Signal Processing Society (2017 and 2018) and from the IEEE (2018). From 2014 to 2015, he worked as a middleware consultant for Oracle in Athens, Greece, and received a Performance Excellence Award. His research interests include machine learning, big data analytics, and network science. He is a student member of the IEEE.

Ahmed S. Zamzam (S'14) received the BSc degree from Cairo University, in 2013, and the MSc degree from Nile University, in 2015. He is working toward the PhD degree at the Department of Electrical and Computer Engineering , University of Minnesota, where he is also affiliated with the Signal and Tensor Analytics Research (STAR) group under the supervision of Professor N. D. Sidiropoulos. He received the Louis John Schnell Fellowship (2015), and the Doctoral Dissertation Fellowship (2018) from the University of Minnesota. He also received Student Travel Awards from the IEEE Signal Processing Society (2017), the IEEE Power and Energy Society (2018), and the Council of Graduate Students at the University of Minnesota (2016 and 2018). His research interests include control and optimization of smart grids, large-scale complex energy systems, grid data analytics, and machine learning. He is a student member of the IEEE.

Georgios B. Giannakis (F'97) received the diploma degree in electrical engineering from the National Technical University of Athens, Greece, 1981, the MSc degree in electrical engineering, the MSc degree in mathematics, and the PhD degree in electrical engineering from the University of Southern California, in 1983, 1986, and 1986, respectively. He was a faculty member with the University of Virginia from 1987 to 1998, and since 1999 he has been a professor with the University of Minnesota, where he holds an ADC Endowed Chair, a University of Minnesota McKnight Presidential Chair in ECE, and serves as director of the Digital Technology Center. His general interests span the areas of statistical learning, communications, and networking—subjects on which he has published more than 450 journal papers, 750 conference papers, 25 book chapters, two edited books, and two research monographs (h-index 140). Current research focuses on data science, and network science with applications to the Internet of Things, social, brain, and power networks with renewables. He is the (co-) inventor of 32 patents issued, and the (co-) recipient of nine best journal paper awards from the IEEE Signal Processing (SP) and Communications Societies, including the G. Marconi Prize Paper Award in Wireless Communications. He also received Technical Achievement Awards from the SP Society (2000), from EURASIP (2005), a Young Faculty Teaching Award, the G. W. Taylor Award for distinguished research from the University of Minnesota, and the IEEE Fourier Technical Field Award (inaugural recipient in 2015). He is a Fellow of EURASIP, and has served the IEEE in a number of posts, including that of a distinguished lecturer for the IEEE-SPS. He is a fellow of the IEEE.

Nicholas D. Sidiropoulos (F'09) received the diploma degree in electrical engineering from the Aristotle University of Thessaloniki, Greece, in 1988, and the MS and PhD degrees in electrical engineering from the University of Maryland at College Park, in 1990 and 1992, respectively. He has served on the faculty of the University of Virginia, University of Minnesota, and the Technical University of Crete, Greece, prior to his current appointment as Louis T. Rader professor and chair of ECE at UVA. From 2015 to 2017, he was an ADC chair professor with the University of Minnesota. His research interests are in signal processing, communications, optimization, tensor decomposition, and factor analysis, with applications in machine learning and communications. He received the NSF/CAREER award in 1998, the IEEE Signal Processing Society (SPS) Best Paper Award in 2001, 2007, and 2011, respectively, served as the IEEE SPS Distinguished Lecturer (2008-2009), and currently serves as vice president - membership of IEEE SPS. He received the 2010 IEEE Signal Processing Society Meritorious Service Award, and the 2013 Distinguished Alumni Award from the University of Maryland, Department of ECE. He is a Fellow of the IEEE (2009) and a Fellow of EURASIP (2014).