Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 17, No. 3, pp. 487–505, 2021

# TENSOR FACTORIZATION WITH TOTAL VARIATION FOR INTERNET TRAFFIC DATA IMPUTATION

Gaohang Yu*, Liqin Wang, Shaochun Wan, Liqun Qi and Yanwei Xu

**Abstract:** Recovering network traffic data from incomplete observed data becomes increasingly critical in network engineering and management. To fully exploit the spatial-temporal features of the Internet traffic data, this paper presents a new tensor completion model which combines the T-product-based tensor factorization with total variation (TV) regularization. To tackle the proposed model, an effective Proximal Alternating Minimization (PAM) algorithm with guaranteed convergence is designed. Extensive experiments on the real-word traffic datasets show that the proposed method has superiority over the existing state-of-the-art methods.

**Key words:** *tensor completion, tensor factorization, total variation, internet traffic data, T-product*

**Mathematics Subject Classification:** *15A69, 65K10*

## 1 Introduction

Estimating the traffic data between a set of originating and destination (OD) pairs in the network is an important part of network engineering tasks, including capacity planning[4], anomaly detection[12] and traffic engineering[8], etc. Unfortunately, in traditional IP network systems, no matter what kind of traffic measurement system is used, data missing is unavoidable. Since many network engineering tasks require complete flow data volume information or are highly sensitive to the missing data, how to accurately recover missing values from partial traffic data is critical.

For the inference of missing data in network traffic data, people have done a lot of research work. Reference[16] first proposed a low-rank matrix approach with a spatio-temporal regularization, and then some other matrix recovery models and algorithms were proposed[7, 19, 23]. Although these methods show good performance when the data missing rate is low, their practical performance will be greatly affected when the missing rate is large. In particular, those matrix-based methods do not work on some extreme case, such as the traffic data lost in continuous time intervals.

In order to improve the recovery performance of the above matrix-based methods, some tensor completion methods [2, 24] have been proposed to recover the missing traffic data, which are based on CANDECOMP/PARAFAC(CP) decomposition [11], or Tucker decomposition [17] etc. Zhou et al. [24] modeled the traffic data as a third-order tensor $\mathcal{G} \in \mathbb{R}^{N\times N\times T}$,

*Prof. G. Yu was supported in part by National Natural Science Foundation of China (Nos. 12071104 and 11661007) and Natural Science Foundation of Zhejiang Province (No. LD19A010002).

where $N$ corresponds to the set of origins and the set of destinations, and $T$ denotes the total number of the time intervals. Then, with the help of the CP decomposition, they developed a traffic data recovery model with temporal and spatial constraints. However, the third-order tensor modeled by Zhou et al. [24] cannot fully utilize the traffic periodicity in the traffic data, so the recovery accuracy is not very high. Moreover, taking the Abilene data [1] as an example, the size of the tensor modeled in [24] is $11 \times 11 \times 1008$, which does not satisfy the balance of the three dimensions, which will also greatly affect the recovery performance. In order to make full use of the traffic features of the temporal periodicity, Xie et al. [20] modeled the considered traffic data as a third-order tensor $\mathcal{Z} \in \mathbb{R}^{o \times t \times d}$, where $o$ corresponds to the set of $N \times N$ OD pairs, $t$ denotes the total number of time intervals in each day and $d$ corresponds to the total number of days to consider. And then they proposed two tensor completion algorithms to recover Internet traffic data. The tensor in $\mathbb{R}^{144 \times 288 \times 168}$ modeled in [20] has better recovery performance than the one in[24] due to the more balanced size of its three dimensions.

In order to accurately recover internet traffic data, it is necessary to consider temporal and spatial prior knowledge, that is, the temporal stability and periodicity features of the original traffic data. Inspired for the successful application of total variation (TV) in spatio-temporal video recovery [13], this paper applies TV regularization to the network traffic data recovery problem.

The structure of this paper is as follows. In the next section, we introduce some notations and preliminary knowledge of tensors. In Section 3, we propose a new internet traffic tensor recovery model based on tensor decomposition and TV regularization. The PAM algorithm is proposed to solve the model and the convergence of the algorithm is analyzed in Section 4. In Section 5, we conduct extensive numerical experiments on Abilene and GÉANT datasets to evaluate the performance of the proposed algorithm. Conclusions and future work are discussed in Section 6.

## 2 Notations and Preliminaries

In this section, we briefly introduce some symbols, basic definitions and lemmas used in this paper.

### 2.1 Notations

In this paper, matrices are denoted by capital letters $(A, B, \ldots)$, tensors by Euler script letters $(\mathcal{A}, \mathcal{B}, \ldots)$, and $\mathbb{R}$ represents a real number space, $\mathbb{C}$ represents the complex number space. For a third-order tensor $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$, its $(i,j,k)$-th element $\mathcal{A}(i,j,k)$ can be represented by $a_{ijk}$, and we use the Matlab notations $\mathcal{A}(i,:,:)$, $\mathcal{A}(:,i,:)$ and $\mathcal{A}(:,:,i)$ to represent the $i$-th horizontal, lateral and frontal slice of the $\mathcal{A}$ respectively[11]. The frontal slice $\mathcal{A}(:,:,i)$ is represented by $A_i$. We also use the Matlab notation $\mathcal{A}(:)$ and define $\|\mathcal{A}\|_{l_1} := \|\mathcal{A}(:)\|_1 = \Sigma_{ijk}|a_{ijk}|$, $\|\mathcal{A}\|_F := \|\mathcal{A}\|_{l_2} = \|\mathcal{A}(:)\|_2 = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle} = \sqrt{\Sigma_{ijk}|a_{ijk}|^2}$. $\mathcal{A}^*$ and $\mathcal{A}^\dagger$ respectively represent the conjugate transpose and pseudo-inverse of $\mathcal{A}$.

Discrete Fourier Transformation (DFT) plays a core role in the tensor-tensor product introduced later. For the tensor $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$, $\bar{\mathcal{A}} \in \mathbb{C}^{m_1 \times m_2 \times m_3}$ represents the result of DFT on $\mathcal{A}$ along the third dimension. In fact, we can use the Matlab command $\bar{\mathcal{A}} = \mathrm{fft}(\mathcal{A}, [\,], 3)$ to directly calculate $\bar{\mathcal{A}}$, and can use the inverse DFT to calculate $\mathcal{A}$ from $\bar{\mathcal{A}}$, that is, $\mathcal{A} = \mathrm{ifft}(\bar{\mathcal{A}}, [\,], 3)$.

The projection operator on $\Omega$ is denoted by $\mathcal{P}_\Omega$, which is defined as:

$$(\mathcal{P}_\Omega(\mathcal{A}))_{i_1,\cdots,i_N} = \begin{cases} x_{i_1,\cdots,i_N}, & \Omega_{i_1,\cdots,i_N} = 1, \\ 0, & \text{others.} \end{cases}$$

## $\boxed{2.2}$ Preliminaries

**Definition 2.1.** (block-diagonal matrix)[25]: Given the tensor $\bar{\mathcal{A}} \in \mathbb{C}^{m_1 \times m_2 \times m_3}$, then

$$\bar{A} = \text{bdiag}(\bar{\mathcal{A}}) = \begin{bmatrix} \bar{A}_1 & & & \\ & \bar{A}_2 & & \\ & & \ddots & \\ & & & \bar{A}_{m_3} \end{bmatrix}$$

is a block diagonal matrix of size $m_1 m_3 \times m_2 m_3$.

**Definition 2.2.** (block circulant matrix)[25]: For a given tensor with $m_3$ frontal slices $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$, then

$$\text{bcirc}(\mathcal{A}) = \begin{bmatrix} A_1 & A_{m_3} & A_{m_3-1} & \dots & A_3 & A_2 \\ A_2 & A_1 & A_{m_3} & \dots & A_4 & A_3 \\ A_3 & A_2 & A_1 & \dots & A_5 & A_4 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ A_{m_3-1} & A_{m_3-2} & A_{m_3-3} & \dots & A_1 & A_{m_3} \\ A_{m_3} & A_{m_3-1} & A_{m_3-2} & \dots & A_2 & A_1 \end{bmatrix}$$

is a block circulant matrix of size $m_1 m_3 \times m_2 m_3$.

**Definition 2.3.** (Identity tensor)[21]: The identity tensor $\mathcal{I} \in \mathbb{R}^{m_1 \times m_1 \times m_3}$ is defined as the tensor whose first frontal slice $\mathcal{I}(:,:,1)$ is the identity matrix of size $m_1 \times m_1$, and all other frontal slices are zeros.

**Definition 2.4.** ($f$-diagonal tensor)[10]: A tensor $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ is called f-diagonal tensor if each of its frontal slices is a diagonal matrix.

**Definition 2.5.** (tensor-tensor product)[21]: $\mathcal{A} \in \mathbb{R}^{m_1 \times r \times m_3}$ and $\mathcal{B} \in \mathbb{R}^{r \times m_2 \times m_3}$ are two real tensors, then the tensor product $\mathcal{A} * \mathcal{B}$ is a real tensor of size $m_1 \times m_2 \times m_3$ defined by the following formula

$$\mathcal{A} * \mathcal{B} = \text{fold}(\text{bcirc}(\mathcal{A}) \cdot \text{unfold}(\mathcal{B}))$$

where $\text{unfold}(\mathcal{A}) = [A_1; A_2; \cdots; A_{m_3}] \in \mathbb{R}^{m_1 m_3 \times r}$ and its inverse operator fold is defined as $\text{fold}(\text{unfold}(\mathcal{A})) = \mathcal{A}$, "$\cdot$" represents the standard matrix product.

**Definition 2.6.** (Conjugate transpose)[21]: The conjugate transpose of tensor $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ is a tensor of size $\mathbb{R}^{m_2 \times m_1 \times m_3}$, denoted by $\mathcal{A}^*$. $\mathcal{A}^*$ is obtained by conjugate transposing each of the frontal slices and then reversing the order of transposed frontal slices 2 through $m_3$.

**Definition 2.7.** (Orthogonal tensor)[21]: Tensor $\mathcal{A} \in \mathbb{R}^{m_1 \times m_1 \times m_3}$ is an orthogonal tensor, if it satisfies

$$\mathcal{A} * \mathcal{A}^* = \mathcal{A}^* * \mathcal{A} = \mathcal{I}$$

**Definition 2.8.** (t-SVD)[10]: Tensor $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ can be decomposed into

$$\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^*$$

Where $\mathcal{U} \in \mathbb{R}^{m_1 \times m_1 \times m_3}$ and $\mathcal{V} \in \mathbb{R}^{m_2 \times m_2 \times m_3}$ are orthogonal tensors, $\mathcal{S} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ is $f$-diagonal tensor.

**Definition 2.9.** (Tensor tubal rank)[9]: The tubal rank $\mathrm{rank_t}(\mathcal{A})$ of tensor $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ is defined as the number of non-zero singular tubes of $\mathcal{S}$.i.e., $\mathrm{rank_t}(\mathcal{A}) = \sharp\{i|\mathcal{S}(i,i,:) \neq 0\}$, where $\mathcal{S}$ is from $\mathcal{A} = \mathcal{U} * \mathcal{S} * \mathcal{V}^*$.

**Lemma 2.10.** [25] *If* $\mathrm{rank_t}(\mathcal{F}) = r$, *then* $\mathcal{F}$ *can be written into the form of tensor product* $\mathcal{F} = \mathcal{G} * \mathcal{H}$, *where* $\mathcal{G} \in \mathbb{R}^{m_1 \times r \times m_3}$ *and* $\mathcal{H} \in \mathbb{R}^{r \times m_2 \times m_3}$ *are two tensors of smaller sizes and they meet* $\mathrm{rank_t}(\mathcal{G}) = \mathrm{rank_t}(\mathcal{H}) = r$.

**Lemma 2.11.** [10] *Suppose that* $\mathcal{A} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ *and* $\mathcal{B} \in \mathbb{R}^{m_2 \times m_4 \times m_3}$ *are two arbitrary tensors, let* $\mathcal{F} = \mathcal{A} * \mathcal{B}$, *then the following properties hold:*
$(1)\|\mathcal{A}\|_F^2 = \frac{1}{m_3}\|(\bar{A})\|_F^2$
$(2)\mathcal{F} = \mathcal{A} * \mathcal{B}$ *is equivalent to* $\bar{F} = \bar{A}\bar{B}$.

## $\boxed{3}$ Tensor Completion Model for Internet Traffic Data Imputation

Let $\mathcal{G} = (g_{i_1 i_2 i_3}) \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ be a given incomplete tensor, we use the tensor $\mathcal{W} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$ to model the internet traffic data. Based on the low rank of traffic data, the problem of traffic data recovery can be formulated as the following tensor rank minimization problem:

$$\begin{aligned} &\min_{\mathcal{W}} \mathrm{rank}(\mathcal{W}) \\ &s.t. P_\Omega(\mathcal{W}) = P_\Omega(\mathcal{G}), \end{aligned} \tag{3.1}$$

where $\mathrm{rank}(\mathcal{W})$ represents the rank of tensor $\mathcal{W}$, $\Omega$ is the set of the positions of the observed data, $P_\Omega$ is a linear operator, which extracts the known elements in the subset $\Omega$, and the elements outside $\Omega$ are filled with 0.

Since solving the problem (3.1) directly is NP-hard, nuclear norm minimization methods [5, 14] were proposed to approximate the rank by using the convex relaxation substitute. However, the nuclear norm minimization methods require the calculation of Singular Value Decomposition (SVD), which is very expensive in computation. To avoid calculating SVD, Zhou et al.[25] proposed a tensor decomposition method based on tensor-tensor product (t-product):

$$\begin{aligned} &\min_{\mathcal{X},\mathcal{Y},\mathcal{W}} \frac{1}{2}\|\mathcal{X} * \mathcal{Y} - \mathcal{W}\|_F^2 \\ &s.t. P_\Omega(\mathcal{W}) = P_\Omega(\mathcal{G}) \\ &\mathcal{X} \in R^{m_1 \times r \times m_3}, \mathcal{Y} \in R^{r \times m_2 \times m_3}, \mathcal{W} \in R^{m_1 \times m_2 \times m_3}. \end{aligned} \tag{3.2}$$

In real-world internet, traffic data usually changes slowly over time, which exhibit temporal stability feature; users often have similar network access behaviors in the same time period on different dates, which exhibit the periodicity of traffic data. The tensors $\mathcal{D}_1 \in \mathbb{R}^{m_1 \times m_1 \times m_3}$ and $\mathcal{D}_2 \in \mathbb{R}^{m_2 \times m_2 \times m_3}$ are used to characterize the traffic temporal properties, where $\mathcal{D}_1$ is used to capture the stability of the traffic data at two adjacent times, $\mathcal{D}_2$ is used to capture the periodicity of traffic data. The proposed tensor completion model by spatio-temporal regularization tensor factorization with TV regularization (SRTFTV) can be described as follows.

$$\begin{aligned}
&\min_{\mathcal{W},\mathcal{X},\mathcal{Y}} \frac{1}{2}\|\mathcal{X} * \mathcal{Y} - \mathcal{W}\|_F^2 + \alpha_1\|\mathcal{D}_1 * \mathcal{W}\|_{l_1} + \alpha_2\|\mathcal{W} * \mathcal{D}_2\|_{l_1}\\
&s.t. P_\Omega(\mathcal{W}) = P_\Omega(\mathcal{G})\\
&\quad v_{\min} \leq \mathcal{W} \leq v_{\max}\\
&\quad \mathcal{X} \in R^{m_1\times r\times m_3}, \mathcal{Y} \in R^{r\times m_2\times m_3}, \mathcal{W} \in R^{m_1\times m_2\times m_3}.
\end{aligned} \tag{3.3}$$

Here

$$\mathcal{D}_1(:,:,i) = \begin{cases} L_{(m_1)}, & i = 1,\\ O_{(m_1)}, & i > 1.\end{cases} \quad \mathcal{D}_2(:,:,i) = \begin{cases} L^*_{(m_2)}, & i = 1,\\ O_{(m_2)}, & i > 1.\end{cases}$$

$$L_{(m)} = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 0\\ -1 & 1 & 0 & \dots & 0 & 0\\ 0 & -1 & 1 & \dots & 0 & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & 0 & 0 & \dots & 1 & 0\\ 0 & 0 & 0 & \dots & -1 & 1 \end{bmatrix} \in \mathbb{R}^{m\times m}$$

$\alpha_1$ and $\alpha_2$ are regularization parameters, $\|\mathcal{D}_1 * \mathcal{W}\|_{l_1} + \|\mathcal{W} * \mathcal{D}_2\|_{l_1}$ is the TV regularization, and the second constraint in (3.3) imposes all values of the output tensor to be included in a range $[v_{\min}, v_{\max}]$.

The optimization problem (3.3) can be rewritten as the following unconstrained optimization problem:

$$\min F(\mathcal{X},\mathcal{Y},\mathcal{W}) = \frac{1}{2}\|\mathcal{X} * \mathcal{Y} - \mathcal{W}\|_F^2 + \alpha_1\|\mathcal{D}_1 * \mathcal{W}\|_{l_1} + \alpha_2\|\mathcal{W} * \mathcal{D}_2\|_{l_1} + \delta_S(\mathcal{W}), \tag{3.4}$$

where

$$\begin{aligned}\mathcal{S} =& \{\mathcal{W} \in R^{m_1\times m_2\times m_3} | \mathcal{W}(i,j,k) = \mathcal{G}(i,j,k)\ for\ (i,j,k) \in \Omega,\\ & \mathcal{W}(i,j,k) \in [v_{\min}, v_{\max}]\ for\ (i,j,k) \notin \Omega\},\end{aligned}$$

$$\delta_S(\mathcal{W}) = \begin{cases} 0, & \mathcal{W} \in S,\\ +\infty, & \mathcal{W} \notin S.\end{cases}$$

## 4 Proximal alternating minimization algorithm for the problem (3.4)

### 4.1 Algorithm description

In this section, we will introduce the algorithm for the internet traffic data completion problem (3.4) in detail. The objective function in (3.4) is not a convex function of $(\mathcal{X},\mathcal{Y},\mathcal{W})$, but it is a convex function for each variable $\mathcal{X},\mathcal{Y},\mathcal{W}$. So, we can use the alternate minimization strategy to solve the problem. In order to improve the theoretical convergence and numerical stability of the alternate minimization algorithm, proximal items are suggested to add to the sub-problems generated by the AM algorithm, which is called the Proximal Alternate Minimization (PAM) algorithm.

Given the initial point $(\mathcal{X}^{(0)}, \mathcal{Y}^{(0)}, \mathcal{W}^{(0)})$ of the problem (3.4), the PAM iteration is defined as follows:

$$\mathcal{X}^{(k+1)} = \arg\min_{\mathcal{X}} F(\mathcal{X}, \mathcal{Y}^{(k)}, \mathcal{W}^{(k)}) + \frac{\rho_1}{2}\|\mathcal{X} - \mathcal{X}^{(k)}\|_F^2 \tag{4.1}$$

$$\mathcal{Y}^{(k+1)} = \arg\min_{\mathcal{Y}} F(\mathcal{X}^{(k+1)}, \mathcal{Y}, \mathcal{W}^{(k)}) + \frac{\rho_2}{2}\|\mathcal{Y} - \mathcal{Y}^{(k)}\|_F^2 \tag{4.2}$$

$$\mathcal{W}^{(k+1)} = \arg\min_{\mathcal{W}} F(\mathcal{X}^{(k+1)}, \mathcal{Y}^{(k+1)}, \mathcal{W}) + \frac{\rho_3}{2}\|\mathcal{W} - \mathcal{W}^{(k)}\|_F^2 \tag{4.3}$$

where $\rho_1, \rho_2, \rho_3$ are the given parameters with $\rho_1, \rho_2, \rho_3 > 0$.

It can be seen that the sub-problems (4.1)-(4.3) are all strongly convex optimization problems, the existence and uniqueness of the solutions are guaranteed. The details are as follows:

**$\mathcal{X}$-Subproblem**

$$\mathcal{X}^{(k+1)} = (\rho_1 \mathcal{X}^{(k)} + \mathcal{W}^{(k)} * (\mathcal{Y}^{(k)})^*)((\mathcal{Y}^{(k)}) * (\mathcal{Y}^{(k)})^* + \rho_1 I)^\dagger \tag{4.4}$$

**$\mathcal{Y}$-Subproblem**

$$\mathcal{Y}^{(k+1)} = ((\mathcal{X}^{(k+1)})^* * \mathcal{X}^{(k+1)} + \rho_2 I)^\dagger ((\mathcal{X}^{(k+1)})^* * \mathcal{W}^{(k)} + \rho_2 \mathcal{Y}^{(k)}) \tag{4.5}$$

**$\mathcal{W}$-Subproblem**

Equation (4.3) is equivalent to the following equality constraint problem:

$$\begin{aligned} \mathcal{W} = \arg\min_{\mathcal{W}, \mathcal{Q}_1, \mathcal{Q}_2} \frac{1}{2}\|\mathcal{X}^{(k+1)} * \mathcal{Y}^{(k+1)} - \mathcal{W}\|_F^2 + \alpha_1 \|\mathcal{Q}_1\|_{l_1} \\ + \alpha_2 \|\mathcal{Q}_2\|_{l_1} + \frac{\rho_3}{2}\|\mathcal{W} - \mathcal{W}^{(k)}\|_F^2 + \delta_S(\mathcal{W}) \\ s.t. \mathcal{Q}_1 = \mathcal{D}_1 * \mathcal{W}, \mathcal{Q}_2 = \mathcal{W} * \mathcal{D}_2 \end{aligned} \tag{4.6}$$

Let $\mathcal{P}$ and $\mathcal{Z}$ be the Lagrange multiplier of (4.6). The iterative scheme for (4.6) can be described as follows:

$$\mathcal{Q}_1^{(k+1)} = \arg\min_{\mathcal{Q}_1} \alpha_1 \|\mathcal{Q}_1\|_{l_1} + \frac{\beta}{2}\|\mathcal{Q}_1 - \mathcal{D}_1 * \mathcal{W}^{(k)} + \frac{\mathcal{P}^{(k)}}{\beta}\|_F^2 \tag{4.7}$$

$$\mathcal{Q}_2^{(k+1)} = \arg\min_{\mathcal{Q}_2} \alpha_2 \|\mathcal{Q}_2\|_{l_1} + \frac{\lambda}{2}\|\mathcal{Q}_2 - \mathcal{W}^{(k)} * \mathcal{D}_2 + \frac{\mathcal{Z}^{(k)}}{\lambda}\|_F^2 \tag{4.8}$$

$$\begin{aligned} \mathcal{W}^{(k+1)} = \arg\min_{\mathcal{W}} \frac{1}{2}\|\mathcal{X}^{(k+1)} * \mathcal{Y}^{(k+1)} - \mathcal{W}\|_F^2 + \frac{\beta}{2}\|\mathcal{Q}_1^{(k+1)} - \mathcal{D}_1 * \mathcal{W} + \frac{\mathcal{P}^{(k)}}{\beta}\|_F^2 \\ + \frac{\lambda}{2}\|\mathcal{Q}_2^{(k+1)} - \mathcal{W} * \mathcal{D}_2 + \frac{\mathcal{Z}^{(k)}}{\lambda}\|_F^2 + \frac{\rho_3}{2}\|\mathcal{W} - \mathcal{W}^{(k)}\|_F^2 + \delta_S(\mathcal{W}), \end{aligned} \tag{4.9}$$

with

$$\mathcal{P}^{(k+1)} = \mathcal{P}^{(k)} + \beta(\mathcal{Q}_1^{(k+1)} - \mathcal{D}_1 * \mathcal{W}^{(k+1)}) \tag{4.10}$$

and

$$\mathcal{Z}^{(k+1)} = \mathcal{Z}^{(k)} + \lambda(\mathcal{Q}_2^{(k+1)} - \mathcal{W}^{(k+1)} * \mathcal{D}_2), \tag{4.11}$$

where $\beta > 0$ and $\lambda > 0$ are the penalty parameters. According to the soft threshold, (4.7) and (4.8) have the following unique solutions:

$$\mathcal{Q}_1^{(k+1)}(i_1,i_2,i_3) = T_{\frac{\alpha_1}{\beta}}((\mathcal{D}_1 * \mathcal{W}^{(k)} + \frac{\mathcal{P}^{(k)}}{\beta})(i_1,i_2,i_3)), \tag{4.12}$$

$$\mathcal{Q}_2^{(k+1)}(i_1,i_2,i_3) = T_{\frac{\alpha_2}{\lambda}}((\mathcal{W}^{(k)} * \mathcal{D}_2 + \frac{\mathcal{Z}^{(k)}}{\lambda})(i_1,i_2,i_3)), \tag{4.13}$$

here, for $y \in \mathbb{R}$,

$$T_\mu(y) := \begin{cases} (|y|-\mu)\text{sign}(y), & |y| > \mu, \\ 0, & |y| \leq \mu. \end{cases}$$

According to **Lemma 2.2**, (4.9) is equivalent to following problem

$$\begin{aligned} \bar{W}^{(k+1)} = \arg\min_{\bar{W}} & \frac{1}{2}\|\bar{W} - \bar{X}^{(k+1)}\bar{Y}^{(k+1)}\|_F^2 + \frac{\beta}{2}\|\bar{Q}_1^{(k+1)} - \bar{D}_1\bar{W} + \frac{\bar{P}^{(k)}}{\beta}\|_F^2 \\ & + \frac{\lambda}{2}\|\bar{Q}_2^{(k+1)} - \bar{W}\bar{D}_2 + \frac{\bar{Z}^{(k)}}{\lambda}\|_F^2 + \frac{\rho_3}{2}\|\bar{W} - \bar{W}^{(k)}\|_F^2 + \delta_S(\bar{W}) \end{aligned} \tag{4.14}$$

Treating real part and imaginary part of $\bar{W}$ as real variables of the objective function, then the unique solution of the above problem is actually the solution of the following matrix equation.

$$\bar{W} + \beta\bar{D}_1^*\bar{D}_1\bar{W} + \lambda\bar{W}\bar{D}_2\bar{D}_2^* + \rho_3\bar{W} = \Gamma, \tag{4.15}$$

where $\Gamma = \bar{X}^{(k+1)}\bar{Y}^{(k+1)} + \bar{D}_1^*(\beta\bar{Q}_1^{(k+1)} + \bar{P}^{(k)}) + (\lambda\bar{Q}_2^{(k+1)} + \bar{Z}^{(k)})\bar{D}_2^* + \rho_3\bar{W}^{(k)}$. Note that the matrices in (4.15) all have block diagonal structure. From the definition of $\mathcal{D}_1$ and $\mathcal{D}_2$, it can be seen that the diagonal blocks of $\bar{D}_1, \bar{D}_2$ are identical. Therefore, the formula (4.15) is equivalent to $m_3$ matrix equations of smaller size as follows:

$$\bar{W}_l + \beta H_{(m_1)}\bar{W}_l + \lambda\bar{W}_l H_{(m_2)} + \rho_3\bar{W}_l = \Gamma_l, \quad l = 1,2,\ldots,m_3, \tag{4.16}$$

where $\bar{W}_l$ and $\Gamma_l$ represent the lth diagonal block of $\bar{W}$ and $\Gamma$ respectively, and

$$H_{(m)} := L_{(m)}^T L_{(m)} = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{m\times m}, \quad H_{(m)}^0 := I_{(m)}.$$

It is easy to verify that $H_{(m)}$ has the following orthogonal diagonalization form

$$H_{(m)} = K_{(m)}\Lambda_{(m)}K_{(m)}^T,$$

$$K_{(m)} = \sqrt{\frac{2}{m}}\left[\sqrt{(1+\delta_{j,1})^{-1}}\cos(\frac{\pi(2\mathrm{i}-1)(\mathrm{j}-1)}{2\mathrm{m}})\right]_{i,j=1}^m,$$

$$\delta_{j,1} = \begin{cases} 1, & j = 1, \\ 0, & \text{otherwise}, \end{cases}$$

$$\Lambda_{(m)} = 4 \times \text{diag}\left(\sin^2\left(\frac{(\mathrm{i}-1)\pi}{2\mathrm{m}}\right)\right)_{i=1}^m.$$

That is, $\Lambda_{(m)}$ is a non-negative diagonal matrix of size $m \times m$, so (4.15) is further equivalent to

$$\bar{W}_l + \beta K_{(m_1)} \Lambda_{(m_1)} K_{(m_1)}^T \bar{W}_l + \lambda \bar{W}_l K_{(m_2)} \Lambda_{(m_2)} K_{(m_2)}^T + \rho_3 \bar{W}_l = \Gamma_l, \quad l = 1, 2, \ldots, m_3. \tag{4.17}$$

Multiplying $K_{(m_1)}^T$ from the left and $K_{(m_2)}$ from the right on both sides of (4.17), then we can obtain the following equation

$$(1+\rho_3)\hat{W}_l + \beta \Lambda_{(m_1)} \hat{W}_l + \hat{W}_l \lambda \Lambda_{(m_2)} = \hat{\Gamma}_l, \quad l = 1, 2, \ldots, m_3, \tag{4.18}$$

where $\hat{W}_l = K_{(m_1)}^T \bar{W}_l K_{(m_2)}$, $\hat{\Gamma}_l = K_{(m_1)}^T \Gamma_l K_{(m_2)}$. For a positive integer $n$, denote $\Xi(n) := \{1, 2, \cdots, n\}$, then

$$\hat{W}_l(m,n) = \frac{\hat{\Gamma}_l(m,n)}{((1+\rho_3)I + \beta \Lambda_{(m_1)})(m) + \lambda \Lambda_{(m_2)}(n)}, \quad (m,n) \in \Xi(m_1) \times \Xi(m_2). \tag{4.19}$$

Therefore, $\mathcal{W}$ in (4.6) can be solved by

$$\begin{aligned} &\mathcal{W}^{(k+1)}(i_1, i_2, i_3) \\ &= \begin{cases} \min\left\{\max\{\text{ifft}(\text{blockdiag}(K_{(m_1)} \hat{W}_l K_{(m_2)}^T)_{l=1}^{m_3}, [\,], 3), v_{\min}\}, v_{\max}\right\}, & (i_1, i_2, i_3) \notin \Omega, \\ \mathcal{G}(i_1, i_2, i_3), \ (i_1, i_2, i_3) \in \Omega. \end{cases} \end{aligned} \tag{4.20}$$

The detail pseudo code is described as follows.

**Algorithm** (SRTFTV, tensor completion by spatio-temporal regularization tensor factorization with TV regularization)

```
Input: The tensor data $\mathcal{G} \in \mathbb{R}^{m_1 \times m_2 \times m_3}$, the initialized rank $r^0 \in \mathbb{R}^{m_3}$, the observed set
$\Omega$, the regularization parameters $\alpha_1, \alpha_2, \beta, \lambda, \rho_1, \rho_2, \rho_3 > 0$, and $\epsilon = 1e-6$.
while not converge do
  (1) For every $k \in [m_3]$, fix $\mathcal{Y}^{(k)}$ and $\mathcal{W}^{(k)}$ to update $\mathcal{X}^{(k+1)}$ via (4.4).
  (2) For every $k \in [m_3]$, fix $\mathcal{X}^{(k+1)}$ and $\mathcal{W}^{(k)}$ to update $\mathcal{Y}^{(k+1)}$ via (4.5).
  (3) For every $k \in [m_3]$, fix $\mathcal{X}^{(k+1)}$ and $\mathcal{Y}^{(k+1)}$ to update $\mathcal{W}^{(k+1)}$ via (4.20).
  (4) Check the termination criterion: $\frac{\|\mathcal{W}^{(k+1)} - \mathcal{W}^{(k)}\|_F^2}{\|\mathcal{W}^{(k+1)}\|_F^2} \leq \epsilon$.
end while
Output : $\mathcal{X}^{(k+1)}, \mathcal{Y}^{(k+1)}, \mathcal{W}^{(k+1)}$
```

## 4.2 Convergence analysis

In this subsection, we will prove the global convergence of the proposed algorithm. To formalize the discussion, we express the tensor-form variables $\mathcal{X}, \mathcal{Y}$ and $\mathcal{W}$ as vectors in what follows.

For positive integers $l_1$,$l_2$,$l_3$, define bijections $\mathbb{V}_{[l_1,l_2,l_3]} : \mathcal{A} \in \mathbb{R}^{l_1 \times l_2 \times l_3} \mapsto \mathbb{V}_{[l_1,l_2,l_3]}(\mathcal{A}) \in \mathbb{R}^{l_1 l_2 l_3 \times 1}$ and $\mathbb{I}_{[l_1,l_2,l_3]} : (i_1, i_2, i_3) \in \Xi(l_1) \times \Xi(l_2) \times \Xi(l_3) \mapsto \mathbb{I}_{[l_1,l_2,l_3]}(i_1, i_2, i_3) \in \Xi(l_1 l_2 l_3)$ by:

$$\begin{aligned} &\mathbb{V}_{[l_1,l_2,l_3]}(\mathcal{A}) := \mathcal{A}(:), \\ &\tilde{\mathcal{A}}(i_1, i_2, i_3) = \left[\tilde{\mathcal{A}}(:)\right]\left(\mathbb{I}_{[l_1,l_2,l_3]}(i_1, i_2, i_3)\right) \text{ holding for each } \tilde{\mathcal{A}} \in \mathbb{R}^{l_1 \times l_2 \times l_3}. \end{aligned}$$

where $\Xi(l_1 l_2 l_3) = \{1, 2, \ldots, l_1 l_2 l_3\}$. Note that $\mathbb{I}_{[l_1,l_2,l_3]}$ maps an index $(i_1, i_2, i_3)$ of an entry in $\mathcal{A} \in \mathbb{R}^{l_1 \times l_2 \times l_3}$ into the index of the same entry in $\mathcal{A}(:)$ and $\mathbb{V}_{[l_1,l_2,l_3]}$ is a bi-continuous linear bijection.

Denote $\mathbb{V}_{[m_1,r,m_3]}$, $\mathbb{V}_{[r,m_2,m_3]}$, $\mathbb{V}_{[m_1,m_2,m_3]}$, $\mathbb{I}_{[m_1,r,m_3]}$, $\mathbb{I}_{[r,m_2,m_3]}$, $\mathbb{I}_{[m_1,m_2,m_3]}$ as $\mathbb{V}_1$,$\mathbb{V}_2$,$\mathbb{V}_3$,$\mathbb{I}_1$,$\mathbb{I}_2$,$\mathbb{I}_3$, respectively. Denote $s_1 = m_1 r m_3$, $s_2 = r m_2 m_3$,$s_3 = m_1 m_2 m_3$, $s = s_1 + s_2 + s_3$. Then, (3.4) can be equivalently rewritten as the following vector form

$$\arg\min_{v \in \mathbb{R}^n} \tilde{F}(v), \tag{4.21}$$

where $v = (v_1; v_2; v_3) \in \mathbb{R}^s$, $v_i \in \mathbb{R}^{s_i}$ $(i = 1, 2, 3)$,

$$\begin{aligned}
&\tilde{F}(v) := G(v) + \delta_{\tilde{S}}(v_3) + g(v_3), \\
&G(v) = \frac{1}{2}\|\mathbb{V}_1^{-1}(v_1) * \mathbb{V}_2^{-1}(v_2) - \mathbb{V}_3^{-1}(v_3)\|_{\ell_2}^2, \\
&g(v_3) := \alpha_1 \|D_1(v_3)\|_{\ell_1} + \alpha_2 \|D_2(v_3)\|_{\ell_1}, \\
&D_1(v_3) := \mathcal{D}_1 * \left[\mathbb{V}_3^{-1}(v_3)\right], \\
&D_2(v_3) := \left[\mathbb{V}_3^{-1}(v_3)\right] * \mathcal{D}_2, \\
&\tilde{S} := \{w \in \mathbb{R}^{s_3} | w(i) = [\mathbb{V}_3(\mathcal{M})](i) \text{ for } i \in \mathbb{I}_3(\Omega), w(i) \in [v_{\min}, v_{\max}] \text{ for } i \notin \mathbb{I}_3(\Omega)\},
\end{aligned}$$

and

$$\delta_{\tilde{S}}(v_3) := \begin{cases} 0, & v_3 \in \tilde{S}, \\ +\infty, & v_3 \notin \tilde{S}. \end{cases}$$

Due to the bilinearity of "$*$" operation, it is clear that $D_1 : \mathbb{R}^{m_1 m_2 m_3 \times 1} \to \mathbb{R}^{m_1 \times m_2 \times m_3}$ and $D_2 : \mathbb{R}^{m_1 m_2 m_3 \times 1} \to \mathbb{R}^{m_1 \times m_2 \times m_3}$ are both linear operators. Moreover, it is easy to see that $\tilde{S}$ is a non-empty closed set, which means $\delta_{\tilde{S}}(\cdot)$ is a proper lower semi-continuous (PLSC) function on $\mathbb{R}^{s_3}$ and $\tilde{F}(\cdot)$ is a PLSC function on $\mathbb{R}^s$.

(4.1)-(4.3) is equivalent to

$$v_1^{(k+1)} \in \arg\min_{v_1} \tilde{F}\left(v_1; v_2^{(k)}; v_3^{(k)}\right) + \frac{\rho_1}{2}\|v_1 - v_1^{(k)}\|_2^2, \tag{4.22}$$

$$v_2^{(k+1)} \in \arg\min_{v_2} \tilde{F}\left(v_1^{(k+1)}; v_2; v_3^{(k)}\right) + \frac{\rho_2}{2}\|v_2 - v_2^{(k)}\|_2^2, \tag{4.23}$$

$$v_3^{(k+1)} \in \arg\min_{v_3} \tilde{F}\left(v_1^{(k+1)}; v_2^{(k+1)}; v_3\right) + \frac{\rho_3}{2}\|v_3 - v_3^{(k)}\|_2^2, \tag{4.24}$$

where $(v_1^{(0)}; v_2^{(0)}; v_3^{(0)})$ is an initial guess for (4.21).

Before proceeding with our convergence analysis, let us briefly review the Kurdyka-Łojasiewicz(KŁ) property, which is the core of the convergence analysis.

**Definition 4.1.** (KŁ property[3])

(a) The function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is said to have the KŁ property at $\bar{x} \in dom(\partial f)$, if there exist $\eta \in (0, +\infty]$, a neighbourhood U of $\bar{x}$ and a continuous concave function $\phi : [0, \eta) \to [0, +\infty)$, such that:

   (i) $\phi(0) = 0$,

   (ii) $\phi$ is first-order continuous on $(0, \eta)$,

   (iii) $\phi'$ is positive on $(0, \eta)$,

(iv) for each $x \in U \cap [f(\bar{x}) < f < f(\bar{x}) + \eta]$, the KŁ inequality holds:

$$\phi'(f(x) - f(\bar{x}))\, dist\,(0, \partial f(x)) \geq 1.$$

(b) PLSC functions which satisfies the KŁ property at each point of $dom\,(\partial f)$ are called KŁ functions, where the norm involved in $dist\,(\cdot,\cdot)$ is $\|\cdot\|_2$ and the convention $dist\,(0,\emptyset) := +\infty$.

**Lemma 4.2.** [3] *Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a PLSC function. Let $\left\{x^{(k)}\right\}_{k\in\mathbb{N}} \subset \mathbb{R}^n$ be a sequence such that*

***H1*** (*Sufficient decrease condition*) *For each $k \in \mathbb{N}$, there exits $a \in (0,+\infty)$ such that $f\left(x^{(k+1)}\right) + a\|x^{(k+1)} - x^{(k)}\|_2^2 \leq f\left(x^{(k)}\right)$ hold,*

***H2*** (*Relative error condition*) *For each $k \in \mathbb{N}$, there exits $w^{(k+1)} \in \partial f\left(x^{(k+1)}\right)$ and a constant $b \in (0,+\infty)$ such that$\|w^{(k+1)}\|_2 \leq b\|x^{(k+1)} - x^{(k)}\|_2$ hold,*

***H3*** (*Continuity condition*) *There exists a subsequence $\left\{x^{(k_j)}\right\}_{j\in\mathbb{N}}$ and $\bar{x} \in \mathbb{R}^n$ such that*

$$x^{(k_j)} \to \bar{x} \text{ and } f\left(x^{(k_j)}\right) \to f(\bar{x}), j \to \infty.$$

*If $f$ has the KŁ property at $\bar{x}$, then*

(i) $x^{(k)} \to \bar{x}$

(ii) *$\bar{x}$ is a critical point of $f$, i.e., $0 \in \partial f(\bar{x})$;*

(iii) *the sequence $\left\{x^{(k)}\right\}_k \in \mathbb{N}$ has a finite length, i.e.,*

$$\sum_{k=0}^{+\infty} \|x^{(k+1)} - x^{(k)}\|_2 < +\infty.$$

Next, we show that the objective function $\tilde{F}$ in (4.21) and the iterative sequence $(v_1^{(k)}; v_2^{(k)}; v_3^{(k)})_{k\in\mathbb{N}}$ generated by iteration (4.22)-(4.24) satisfy the assumptions imposed in **Lemma 4.2**, by which we establish the convergence of the proposed algorithm.

Denote $\hat{S} = \mathbb{R}^{s_1} \times \mathbb{R}^{s_2} \times \tilde{S} \subset \mathbb{R}^s$. Firstly, we prove that $\tilde{F}$ satisfies the KŁ property at each $v \in \hat{S}$.

**Lemma 4.3.** *$\tilde{F}$ is semi-algebraic on $\hat{S}$. Hence, $\tilde{F}$ satisfies the KŁ property at each $v \in \hat{S}$.*

*Proof.* On $\hat{S}$, $\tilde{F}$ can be expressed as

$$\tilde{F} := G(v) + g(v_3),\ v \in \hat{S}.$$

It's clear that $\hat{S}$ is a semi-algebraic set([13] **Proposition 2**).

Note that $\mathbb{V}_i^{-1}$ $(i = 1, 2, 3)$ are linear mappings between finite-dimensional spaces. Hence, each element of $\mathbb{V}_i^{-1}(v_i)$ $(i = 1, 2, 3)$ is actually a linear polynomial of $v := (v_1; v_2; v_3)$. Additionally, by the definition of $\|\cdot\|_{\ell_2}$, we know that $G(v)$ is a polynomial of $v$. Thus, $G$ is semi-algebraic on $\hat{S}$([13] **Proposition 2**)

$D_1$ and $D_2$ are linear mappings between finite-dimensional spaces. Hence, each element of $D_1(v_3)$ and $D_2(v_3)$ can be regarded as a linear polynomial of $v = (v_1; v_2; v_3)$, therefore, $g(v_3)$ are simply finite sums of compositions of absolute value function and linear polynomial of $v$. So, $g(v_3)$ is semi-algebraic on $\hat{S}$([13] **Proposition 2,3**).

In summary, $\tilde{F}$ is semi-algebraic on $\hat{S}$([13]**Proposition 2**). The proof is complete. □

**Theorem 4.4.** *Assume that the sequence* $v^{(k)} = (v_1^{(k)}; v_2^{(k)}; v_3^{(k)})$ *generated by iteration* (4.22)-(4.24) is bounded, then it converges to a critical point of $\tilde{F}$.

*Proof.* It has been mentioned above that $\tilde{F}$ is a PLSC function on $\mathbb{R}^s$. From (4.22)-(4.24), we see that

$$\begin{aligned}
&\tilde{F}(v_1^{(k+1)}; v_2^{(k)}; v_3^{(k)}) + \frac{\rho_1}{2}\|v_1^{(k+1)} - v_1^{(k)}\|_2^2 \leq \tilde{F}(v_1^{(k)}; v_2^{(k)}; v_3^{(k)}), k \in \mathbb{N},\\
&\tilde{F}(v_1^{(k+1)}; v_2^{(k+1)}; v_3^{(k)}) + \frac{\rho_2}{2}\|v_2^{(k+1)} - v_2^{(k)}\|_2^2 \leq \tilde{F}(v_1^{(k+1)}; v_2^{(k)}; v_3^{(k)}), k \in \mathbb{N},\\
&\tilde{F}(v_1^{(k+1)}; v_2^{(k+1)}; v_3^{(k+1)}) + \frac{\rho_3}{2}\|v_3^{(k+1)} - v_3^{(k)}\|_2^2 \leq \tilde{F}(v_1^{(k+1)}; v_2^{(k+1)}; v_3^{(k)}), k \in \mathbb{N}.
\end{aligned}$$

Summing over the three inequalities above, we obtain

$$\tilde{F}(v^{(k+1)}) + \frac{\rho_{\min}}{2}\|v^{(k+1)} - v^{(k)}\|_2^2 \leq \tilde{F}(v^{(k)}), k \in \mathbb{N}. \tag{4.25}$$

Where $\rho_{\min} = \min\{\rho_1, \rho_2, \rho_3\}$. Hence, **H1**(sufficient decrease condition) is satisfied, where $a = \rho_{\min}/2$.

Let $\partial_{v_i}$ and $\nabla_{v_i}$ denote sub-differential and gradient with respect to variable $v_i$, respectively, for $i = 1, 2, 3$. $G$ is a polynomial function and thus is infinitely differentiable. (4.22) and **Proposition 1**[13] imply that

$$\begin{aligned}
0 &\in \partial_{v_1}\left[\tilde{F}(\cdot; v_2^{(k)}; v_3^{(k)}) + \frac{\rho_1}{2}\|\cdot - v_1^{(k)}\|_2^2\right]\Bigg|_{v_1 = v_1^{(k+1)}}\\
&= \partial_{v_1}\tilde{F}(v_1^{(k+1)}; v_2^{(k)}; v_3^{(k)}) + \rho_1\|v_1^{(k+1)} - v_1^{(k)}\|_2^2\\
&= \nabla_{v_1}G(v_1^{(k+1)}; v_2^{(k)}; v_3^{(k)}) + \rho_1(v_1^{(k+1)} - v_1^{(k)}), k \in \mathbb{N}
\end{aligned} \tag{4.26}$$

Similarly, (4.23)-(4.24) and Proposition 1[13] imply that

$$0 \in \nabla_{v_2}G(v_1^{(k+1)}; v_2^{(k+1)}; v_3^{(k)}) + \rho_2(v_2^{(k+1)} - v_2^{(k)}), k \in \mathbb{N}, \tag{4.27}$$

$$0 \in \nabla_{v_3}G(v_1^{(k+1)}; v_2^{(k+1)}; v_3^{(k+1)}) + \partial(\delta_{\tilde{S}} + g)(v_3^{(k+1)}) + \rho_3(v_3^{(k+1)} - v_3^{(k)}), k \in \mathbb{N}. \tag{4.28}$$

(4.26)-(4.28) imply that there exists $w_1^{(k+1)} \in \partial(\delta_{\tilde{S}} + g)(v_3^{(k+1)})$ such that

$$-\tilde{w}^{(k+1)} = u^{(k)} + (\rho_1(v_1^{(k+1)} - v_1^{(k)}); \rho_2(v_2^{(k+1)} - v_2^{(k)}); \rho_3(v_3^{(k+1)} - v_3^{(k)})), k \in \mathbb{N},$$

where

$$\begin{aligned}
&\tilde{w}^{(k+1)} = (0; 0; w_1^{(k+1)}),\\
&u^{(k)} = (\nabla_{v_1}G(v_1^{(k+1)}; v_2^{(k)}; v_3^{(k)}); \nabla_{v_2}G(v_1^{(k+1)}; v_2^{(k+1)}; v_3^{(k)}); \nabla_{v_3}G(v_1^{(k+1)}; v_2^{(k+1)}; v_3^{(k+1)})).
\end{aligned}$$

Denote $w^{(k+1)} = \tilde{w}^{(k+1)} + \nabla G(v^{(k+1)})$. It's clear that $w^{(k+1)} \in \partial\tilde{F}(v^{(k+1)})$ and

$$-w^{(k+1)} = u^{(k)} - \nabla G(v^{(k+1)}) + (\rho_1(v_1^{(k+1)} - v_1^{(k)}); \rho_2(v_2^{(k+1)} - v_2^{(k)}); \rho_3(v_3^{(k+1)} - v_3^{(k)})), k \in \mathbb{N}, \tag{4.29}$$

Denote $E = \{v^{(k)} | k \in \mathbb{N}\}$. For $v = (v_1; v_2; v_3) \in \mathbb{R}^s$, $v_i \in \mathbb{R}^{s_i} (i = 1, 2, 3)$, define the coordinate projections by

$$\Pi_i(v) \coloneqq v_i, i = 1, 2, 3.$$

Denote $\hat{E} = \Pi_1(v) \times \Pi_2(v) \times \Pi_3(v)$. Since $E$ is bounded, therefore, $\hat{E} \subset \mathbb{R}^s$ is also bounded. Since $G$ is a polynomial, it is easy to prove that $\nabla G$ is Lipschitz continuous on any bounded subset of $\mathbb{R}^s$. Hence, there exists a constant $c > 0$ such that

$$\|\nabla G(v) - \nabla G(w)\|_2^2 \leq c\|v - w\|_2^2, v, w \in \hat{E}.$$

Hence,

$$\begin{aligned}\|u^{(k)} - \nabla G(v^{(k+1)})\|_2 = &\Big[\|\nabla_{v_1} G(v_1^{(k+1)}; v_2^{(k)}; v_3^{(k)}) - \nabla_{v_1} G(v^{(k+1)})\|_2^2 \\ &+\|\nabla_{v_2} G(v_1^{(k+1)}; v_2^{(k+1)}; v_3^{(k)}) - \nabla_{v_2} G(v^{(k+1)})\|_2^2\Big]^{\frac{1}{2}} \\ \leq &\Big[c^2(\|v_2^{(k)} - v_2^{(k+1)}\|_2^2 + \|v_3^{(k)} - v_3^{(k+1)}\|_2^2) + c^2(\|v_3^{(k)} - v_3^{(k+1)}\|_2^2)\Big]^{\frac{1}{2}} \\ \leq &\sqrt{2}c\|v^{(k+1)} - v^{(k)}\|_2^2, k \in \mathbb{N},\end{aligned}$$

According to (4.29), there are

$$\begin{aligned}\|w^{(k+1)}\|_2 &\leq \|u^{(k)} - \nabla G(v^{(k+1)})\|_2 + \rho_{\max}\|v^{(k+1)} - v^{(k)}\|_2 \\ &\leq (\sqrt{2}c + \rho_{\max})\|v^{(k+1)} - v^{(k)}\|_2, k \in \mathbb{N}.\end{aligned}$$

where $\rho_{\max} = \max\{\rho_1, \rho_2, \rho_3\}$. Therefore, **H2**(Relative error condition) is satisfied with $b = \sqrt{2}c + \rho_{\max}$.

Moreover, since $\left\{v^{(k)}\right\}_{k\in\mathbb{N}} \subset \mathbb{R}^s$ is bounded and thus relative compact, there exists a subsequence $\left\{v^{(k_j)}\right\}_{j\in\mathbb{N}}$ and $\bar{v} \in \mathbb{R}^s$ such that $v^{(k_j)} \to \bar{v}$, $j \to +\infty$. Since $\left\{v_3^{(k)} | k \in \mathbb{N}\right\} \subset \tilde{S}$, so $\left\{v^{(k)} | k \in \mathbb{N}\right\} \subset \hat{S}$ holds. Since $\hat{S}$ is closed, $\bar{v} \in \hat{S}$, and $\tilde{F}$ is continuous on $\hat{S}$. Therefore, $\tilde{F}\left(v^{(k_j)}\right) \to \tilde{F}\left(\bar{v}\right)$, $j \to +\infty$. Hence, **H3**(Continuity condition) is satisfied.

By **Lemma 4.3**, $\tilde{F}$ satisfies the KŁ property at $\bar{v} \in \hat{S}$. According to **Lemma 4.2**, the sequence $\{v^{(k)}\}_{k\in\mathbb{N}}$ converges to the critical point of $\tilde{F}$. The proof is complete. □

## 5 Numerical experiments

In this section, we conduct numerical experiments to evaluate the performance of the SRTFTV method, and compare it with the following seven data completion methods, the first is the matrix completion via iterated soft thresholding algorithm(IST MC)[5], which does not consider the temporal and spatial structure of the internet traffic matrix, the second is the sparse regularized matrix factorization (SRMF) method [16], which is a low-rank matrix completion method with spatio-temporal regularization, the third method is the CP tensor completion method [2], and the fourth method is the CP tensor completion method with spatio-temporal regularization(STTC)[24], the fifth approach is tensor completion based on tensor-SVD(t-SVD)[22], the sixth method is a tensor completion method based on tensor decomposition(TCTF)[25], the last one is a tensor completion method that combines tensor decomposition, TV regularization and Tikhonov regularization (TCTFTVT)[13]. A series of loss scenarios are simulated, from low loss to high loss probability, from random loss to highly structured loss patterns. We use the normalized mean absolute error (NMAE) in the missing value as a measure of the recovered data. NMAE is defined as follows

$$\text{NMAE} = \frac{\Sigma_{(i,j)\notin\Omega}|X_{ij} - \hat{X}_{ij}|}{\Sigma_{(i,j)\notin\Omega}|X_{ij}|},$$

where $X$ and $\hat{X}$ are the original data and estimated data respectively.

The method parameters used for compared methods were the parameters given in the paper. The proposed method is implemented with parameters $\alpha_1 = \mu\frac{\|\mathcal{D}_1 * \mathcal{W}_{init}\|_{l_1}}{\|\mathcal{G}_\Omega\|_{l_1}}, \alpha_2 = \mu\frac{\|\mathcal{W}_{init} * \mathcal{D}_2\|_{l_1}}{\|\mathcal{G}_\Omega\|_{l_1}}, \mu = 10,\ \beta = 80, \lambda = 0.1, \rho_1 = 10, \rho_2 = \rho_3 = 5e-6$, we set the parameter $\mu = 0.01, \rho_1 = 5e-6$ for consecutive data missing scenario, where $\mathcal{W}_{init}$ corresponds to an initial estimate values. In all experiments, we set tolerance $\epsilon = 1e-6$. If the number of iterations reaches 200, the methods are stopped. The platform is Matlab R2017a under Windows 10 on a PC of a 1.19GHz CPU and 8GB memory.

## 5.1 Dataset

We conduct experiments on two real-world traffic datasets. The first is the widely used Abilene dataset [1], which has 11 routers and therefore 11 × 11 = 121 OD pairs. From December 8, 2003 to December 14, 2003, the network traffic data of each OD pair is recorded every 5 minutes. Therefore, each OD pair has 7 × 24 × 12 = 2016 numbers, so we obtain a network traffic matrix $X$ of size 121 × 2016. To make full use of the time stability and periodicity of the traffic data, the size of the tensor modeled in this paper is 288 × 7 × 121. The second real-world dataset is the GÉANT traffic dataset [18], which has 23 routers, so there are 529 OD pairs. For each OD pair, network traffic is recorded for every 15 minutes in a month from March 21, 2005 to April 15, 2005, so we obtain an internet traffic matrix $X$ of size 529 × 2496, which can be modeled into a tensor of size 96 × 26 × 529.

## 5.2 Performance under random loss

Random missing patterns referred to the elements of a given traffic matrix (TM) data are uniformly and randomly missing. We randomly drop the data independently with probability from 10% to 95% to evaluate the performance.

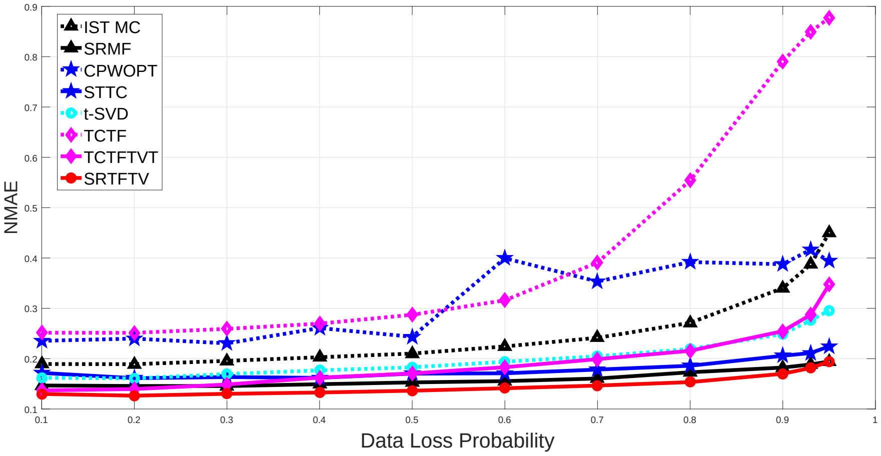

Figure 1: Abilene data, NMAE under random loss

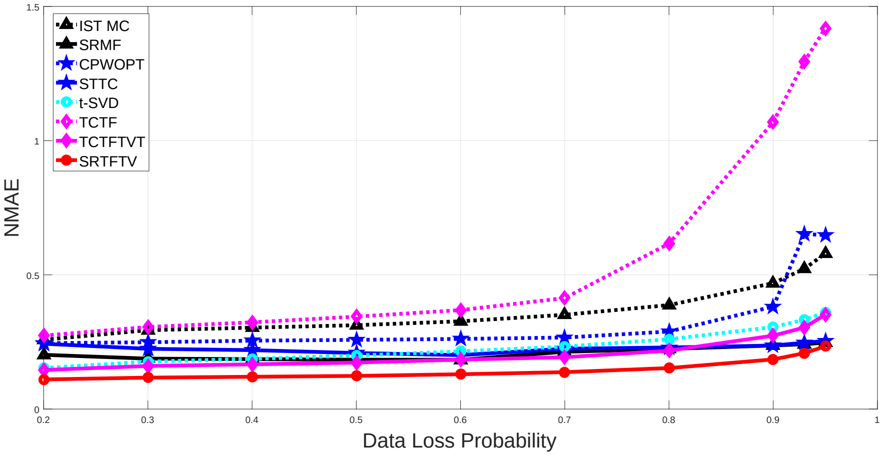

Figure 2: GÉANT data, NMAE under random loss

Figure 1 shows the experimental results applied to the Abilene dataset. The X axis represents the probability of data missing, and the Y axis represents the value of NMAE. It can be seen from Figure 1 that the proposed method SRTFTV is significantly better than the normal TCTF method. Similarly, the performance of the spatio-temporal regularized SRMF method is better than that of the non-regularized IST MC method, and the performance of the spatio-temporal regularized STTC method is better than that of the non-regularized CPWOPT method. This phenomenon shows that the spatio-temporal structure in the internet traffic data is very valuable and has been used to improve the recovery accuracy. Further, the performance of the method SRTFTV method is better than that of TCTFTVT, indicating that the method SRTFTV can capture the structural features of traffic data better than TCTFTVT. When the probability of data loss is less than 95%, the performance of the SRTFTV is better than all other methods. SRMF is just behind SRTFTV and achieves strong performance over the entire loss range. In addition, we apply these methods to the GÉANT traffic dataset, similar results could be observed from Figure 2.

### 5.3 Performance under structured loss

Not all data loss is random. In fact, network traffic losses are often highly structured due to software or hardware reasons[15]. In this section, we simulate two typical data structure loss patterns.

Time-mode Loss(TL): Usually, the overload of the monitoring equipment would cause some random proportion of data to be lost in a certain period of time. We simulate this loss by randomly selecting a certain percentage of time intervals and dropping data points with a certain probability.

Spatial-mode Loss(SL): Unreliable transport protocol (UDP) may cause random data loss of some nodes. We simulate this loss by randomly selecting a certain proportion of OD pairs and dropping data points with a certain probability.

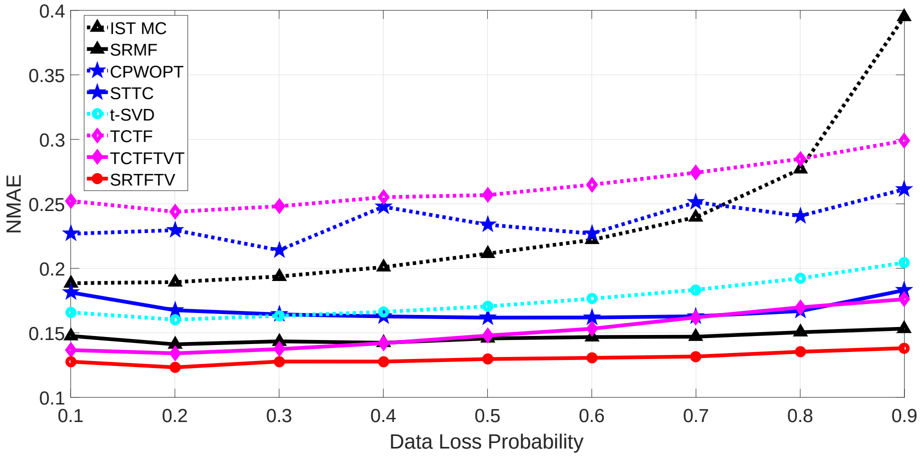

Figure 3: Abilene dataset, TL, 60% time intervals chosen

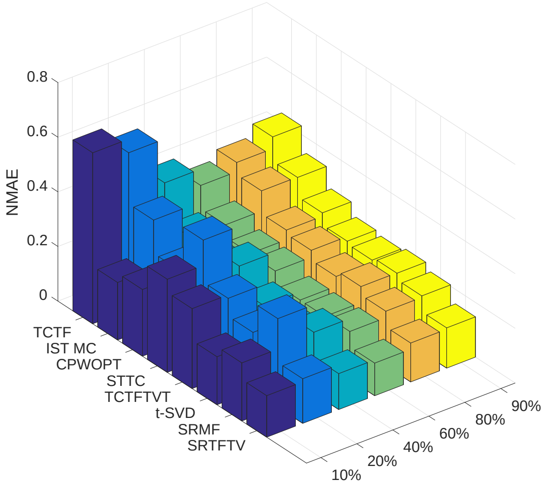

Figure 4: GÉANT dataset, SL, loss probability=0.6

Figure 3 shows the TL pattern, where 60% sampling time intervals are randomly chosen. In these time intervals, loss probability is from 0.1 to 0.9. In this case, SRTFTV achieves the best performance.

Figure 4 shows the SL pattern, where OD pairs are randomly selected from 10% to 90%. In these OD pairs, loss probability is fixed at 0.6. In this case, SRTFTV is superior to other algorithms. SRMF cannot make good use of the correlation between adjacent data to fill the missing data due to the density of data loss, while SRTFTV makes good use of the three-dimensional structure of the tensor, and performs better than SRMF.

Traffic data is usually obtained through continuous measurement. We conduct experiments on Abilene dataset, let the measurement of Wednesday and Thursday be lost, and then calculate the NMAE on the two day's data, as shown in Figure 5. Obviously, The consecutive data missing results in the consecutive column missing in the traffic matrix. From the related literature [6], we know that the normal matrix completion algorithm can recover data only when there is no row or the column is completely empty. If a row or column is missing, the matrix completion algorithm is invalid to fill these missing elements, so under the matrix completion algorithm IST MC, NMAE is 1. The tensor completion method utilizes information along three dimensions, while matrix completion only considers constraints along two specific dimensions, therefore, the tensor completion method SRTFTV performs better than matrix completion-based algorithm SRMF.

In order to observe the recovery accuracy more intuitively, we illustrate the recovered data for the NYCM-LOSA OD pair of Abilene data, as we can see in Figure 6, SRTFTV method can achieve better recovery performance than the compared methods.

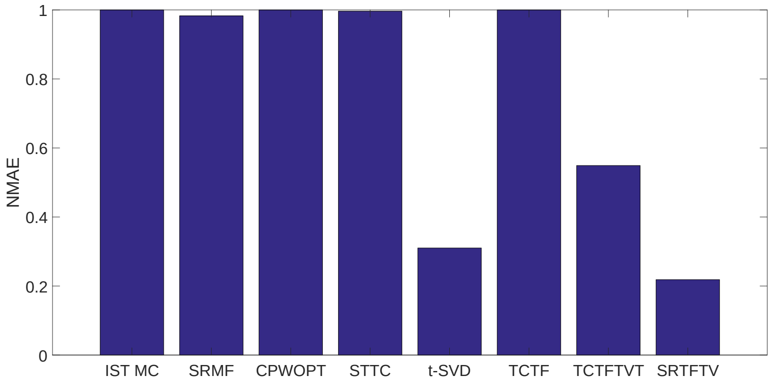

Figure 5: NMAE under consecutive data missing from Abilene Data

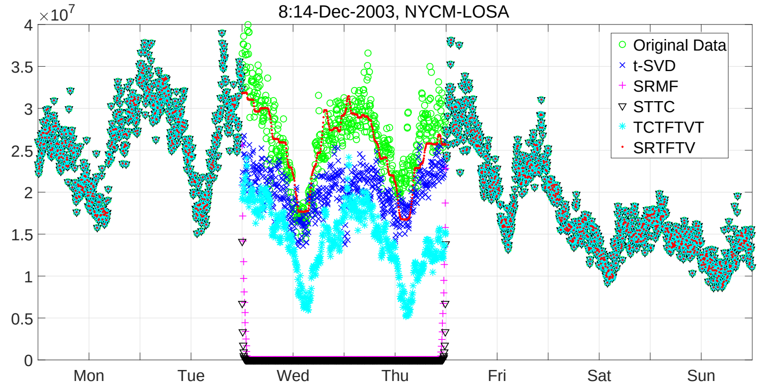

Figure 6: Visualization of the recovered data by t-SVD, SRMF, STTC, TCTFTVT and SRTFTV

## 6 Conclusion

Considering the global structure and local smoothness features of network traffic data, we propose a network traffic data imputation model based on low rank tensor completion and TV regularization. In addition, an easy-to-operate and relatively effective algorithm PAM is used to solve the problem, and the convergence of the algorithm is proved. Numerical experiments on widely used real-world datasets show that the method has excellent performance.

## Acknowledgment

The authors would like to thank two anonymous referees who have contributed to improve the quality of the paper.

## References

[1] The Abilene Observatory Data Collections, `http://abilene.internet2.edu/observatory/data-collections.html.`

[2] E. Acar, D.M. Dunlavy, T.G. Kolda and M. Mørup, Scalable tensor factorizations for incomplete data, *Chemometrics and Intelligent Laboratory Systems* 106 (2011) 41–56.

[3] H. Attouch, J. Bolte and B.F. Svaiter, Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss-Seidel methods, *Mathematical Programming* 137 (2013) 91–129.

[4] R.S. Cahn, *Wide Area Network Design Concepts and Tools for Optimization*, USA: Morgan Kaufmann,1998.

[5] J.F. Cai, E.J. Candès and Z. Shen, A singular value thresholding algorithm for matrix completion, *SIAM Journal on Optimization* 20 (2010) 1956–1982.

[6] E.J. Candès and B. Recht, Exact matrix completion via convex optimization, *Foundations of Computational mathematics* 9 (2009): 717.

[7] R. Du, C. Chen, B. Yang and X. Guan, Vanet based traffic estimation: A matrix completion approach, in: *2013 IEEE Global Communications Conference (GLOBECOM)*, IEEE, 2013, pp. 30–35.

[8] B. Fortz and M. Thorup, Optimizing OSPF/IS-IS weights in a changing world, *IEEE Journal on Selected Areas in Communications* 20 (2002) 756–767.

[9] M.E. Kilmer, K. Braman, N. Hao and R.C. Hoover, Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging, *SIAM Journal on Matrix Analysis and Applications* 34 (2013) 148–172.

[10] M.E. Kilmer and C.D. Martin, Factorization strategies for third-order tensors, *Linear Algebra and its Applications* 435 (2011) 641–658.

[11] T.G. Kolda and B.W. Bader, Tensor decompositions and applications, *SIAM review* 51 (2009) 455–500.

[12] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E.D. Kolaczyk and N. Taft, Structural analysis of network traffic flows,in: *Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems*, 2004, pp. 61-72.

[13] X. Lin, M.K. Ng and X. Zhao, Tensor factorization with total variation and Tikhonov regularization for low-rank tensor completion in imaging data, *Journal of Mathematical Imaging and Vision* 62 (2020) 900–918.

[14] B. Recht, M. Fazel and P.A. Parrilo, Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization, *SIAM review* 52 (2010) 471–501.

[15] M. Roughan, A case study of the accuracy of SNMP measurements, *Journal of Electrical and Computer Engineering*, 2010, Article ID 812979. https://doi.org/10.1155/2010/812979.

[16] M. Roughan, Y. Zhang, W. Willinger and L. Qiu, Spatiotemporal compressive sensing and Internet traffic matrices (extended version), *IEEE/ACM Transactions on Networking (ToN)* 20 (2012) 662–676.

[17] L.R. Tucker, Some mathematical notes on three-mode factor analysis, *Psychometrika* 31 (1966) 279–311.

[18] S. Uhlig, B. Quoitin, J. Lepropre and S. Balon, Providing public intradomain traffic matrices to the research community, *ACM SIGCOMM Computer Communication Review* 36 (2006) 83–86.

[19] K. Xie, L. Wang, X. Wang, G. Xie, G. Zhang, D. Xie and J. Wen, Sequential and adaptive sampling for matrix completion in network monitoring systems, in: *2015 IEEE Conference on Computer Communications (INFOCOM)*, IEEE, 2015, pp. 2443–2451.

[20] K. Xie, L. Wang, X. Wang, G. Xie, J. Wen, G. Zhang and D. Zhang, Accurate recovery of internet traffic data: A sequential tensor completion approach, *IEEE/ACM Transactions on Networking* 26 (2018) 793–806.

[21] Z. Zhang and S. Aeron, Exact tensor completion using t-SVD, *IEEE Transactions on Signal Processing* 65 (2016) 1511–1526.

[22] Z. Zhang, G. Ely, S. Aeron, N. Hao and M. Kilmer, Novel methods for multilinear data completion and de-noising based on tensor-SVD, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 3842–3849.

[23] H. Zhou, D. Zhang and K. Xie, Accurate traffic matrix completion based on multi-Gaussian models, *Computer Communications*102 (2017) 165–176.

[24] H. Zhou, D. Zhang, K. Xie and Y. Chen, Spatio-temporal tensor completion for imputing missing internet traffic data, in: *IEEE 34th International Performance Computing and Communications Conference (IPCCC)*, 2015, pp. 1–7.

[25] P. Zhou, C.Y. Lu, Z.C. Lin and C. Zhang, Tensor factorization for low-rank tensor completion, *IEEE Transactions on Image Processing* 27 (2017) 1152–1163.

*Manuscript received 3 April 2021*
*revised 31 May 2021*
*accepted for publication 1 June 2021*

Gaohang Yu
Department of Mathematics, Hangzhou Dianzi University
310018, China
E-mail address: maghyu@hdu.edu.cn

Liqin Wang
Department of Mathematics, Hangzhou Dianzi University
310018, China
E-mail address: 710342426@qq.com

Shaochun Wan
Department of Mathematics, Hangzhou Dianzi University
310018, China
E-mail address: shaochunw@hdu.edu.cn

Liqun Qi
Huawei Theory Research Lab, Hong Kong, China
Department of Mathematics, Hangzhou Dianzi University
310018, China
Department of Applied Mathematics
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
E-mail address: liqun.qi@polyu.edu.hk

Yanwei Xu
Huawei Theory Research Lab, Hong Kong, China
E-mail address: xuyanwei1@huawei.com