

## 基于边缘智能计算的城市交通感知数据自适应恢复

向朝参<sup>1,2</sup> 程文辉<sup>1,2</sup> 张 昭<sup>1,2</sup> 焦贤龙<sup>1,2</sup> 屈毓鏗<sup>3</sup> 陈 超<sup>1,2</sup> 戴海鹏<sup>4</sup>

<sup>1</sup>(重庆大学计算机学院 重庆 400044)

<sup>2</sup>(信息物理社会可信服务计算教育部重点实验室(重庆大学) 重庆 400044)

<sup>3</sup>(南京航空航天大学电子信息工程学院 南京 210016)

<sup>4</sup>(南京大学计算机科学与技术系 南京 210023)

([xiangchaocan@cqu.edu.cn](mailto:xiangchaocan@cqu.edu.cn))

## Intelligent Edge Computing-Empowered Adaptive Urban Traffic Sensing Data Recovery

Xiang Chaocan<sup>1,2</sup>, Cheng Wenhui<sup>1,2</sup>, Zhang Zhao<sup>1,2</sup>, Jiao Xianlong<sup>1,2</sup>, Qu Yuben<sup>3</sup>, Chen Chao<sup>1,2</sup>, and Dai Haipeng<sup>4</sup>

<sup>1</sup>(College of Computer Science, Chongqing University, Chongqing 400044)

<sup>2</sup>(Key Laboratory of Dependable Service Computing in Cyber Physical Society(Chongqing University), Ministry of Education, Chongqing 400044)

<sup>3</sup>(College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016)

<sup>4</sup>(Department of Computer Science and Technology, Nanjing University, Nanjing 210023)

**Abstract** Intelligent transportation systems (ITSs) have been widely used in smart cities with a widespread problem of missing sensing data. The limited storage computing capability of traffic stations also severely restricts the recovery of sensing data and greatly affects the normal use of ITSs. Although the powerful computing capacity of edge nodes can be used to alleviate this issue, the high complexity and dynamics of the temporal and spatial correlation of sensing data still pose a serious challenge to the recovery process, making the result of data recovery, based on static edge nodes deployment and distribution, unsatisfactory. In order to effectively solve this series of problems, we propose an adaptive urban traffic sensing data recovery system based on intelligent edge computing. The system mainly consists of two parts: Firstly, the submodular optimization theory is used to design a suboptimal deployment and allocation scheme for edge nodes with a theoretical performance lower bound. Secondly, we address a data recovery method based on the low-rank theory. At the same time, the recovery results are used to calculate the corresponding non-missing theoretical lower bound, feed back to the edge nodes, and then update the data distribution scheme to ensure an accurate recovery of subsequent sensing data. The experiments based on large-scale ITSs traces of Australia illustrate that our method can achieve 90% of the optimal performance for the edge node deployment, and improve the data recovery accuracy by 43.8% in comparison with three baselines. Furthermore, the adaptive data recovery scheme can improve the accuracy by 40.3% on average.

**Key words** edge intelligent computing; intelligent transportation system; edge node deployment; sensing data recovery; submodular theory; low-rank theory

收稿日期: 2021-09-23; 修回日期: 2022-06-10

基金项目: 国家自然科学基金项目(62172063, 61872447, 62072064); 安徽省高校自然科学研究项目(KJ2021ZD0116)

This work was supported by the National Natural Science Foundation of China (62172063, 61872447, 62072064) and the College Natural Science Research Project of Anhui Province (KJ2021ZD0116).

通信作者: 陈超([cschaochen@cqu.edu.cn](mailto:cschaochen@cqu.edu.cn))

**摘要** 智能交通系统 (intelligent transportation systems, ITSs) 被广泛用于智慧城市中, 却普遍存在感知数据缺失问题, 而交通感知站点有限的存储计算能力严重制约感知数据的恢复, 极大影响 ITSs 的正常使用。虽然可以利用边缘节点强大的存储计算能力解决这个困境, 但边缘节点部署的高复杂性和感知数据时空相关性的高动态性对数据精确恢复提出挑战。为了解决上述挑战, 提出基于边缘智能计算的城市交通感知数据自适应恢复系统。具体地, 首先利用子模优化理论, 提出具有理论下界的边缘节点次优部署分配算法。然后, 基于低秩理论恢复感知数据, 并基于恢复结果估计非缺失下限, 通过反馈自适应调整感知站点的数据上传比例, 从而保证数据精确恢复。最后, 基于澳大利亚 600 个交通站点 1 年的感知数据构建原型系统, 对所提算法进行评估。实验结果表明, 所提算法的边缘节点部署性能达到最优性能的 90% 以上, 缺失数据恢复精度比 3 种对比方法提高 43.8% 以上。同时, 自适应数据恢复能够平均提高精度 40.3%。

**关键词** 边缘智能计算; 智能交通系统; 边缘节点部署; 感知数据恢复; 子模理论; 低秩理论

中图法分类号 TP393

随着城市的不断快速发展, 城市交通问题日益严峻, 如交通拥堵等<sup>[1]</sup>。为了解决这些问题, 大量智能交通系统(intelligent transportation systems, ITSs)被广泛地部署, 用于城市交通感知和监控<sup>[2]</sup>。例如, 澳大利亚新南威尔士州构建交通流量监控系统(traffic volume viewer system, TVVS)<sup>[3]</sup>, 部署超过 600 个交通感知站点。然而, 由于设备故障、数据传输和系统供电等问题, 感知数据缺失问题在 ITSs 中普遍存在<sup>[4]</sup>。例如, 文献[5]统计分析发现, 澳大利亚 TVVS 系统中约 25% 的站点感知数据缺失率超过 60%, 严重影响该系统的正常使用。因此, 准确实时地恢复大规模 ITSs 缺失感知数据对实现智慧城市的智能交通至关重要。

本文提出基于边缘智能计算的大规模交通缺失感知数据恢复系统, 既利用大量不同交通站点感知数据之间的相关性, 又利用部署在感知站点附近的边缘节点强大的计算处理能力, 从而实现大规模交通感知数据的精确自适应恢复。但是, 要实现该系统, 需要解决 2 方面的挑战:

1) 高计算复杂性的边缘节点部署。边缘节点部署, 既需要考虑部署在不同交通感知站点的成本与收益各不相同, 还与交通感知数据分配策略紧密相关, 从而使边缘节点最优部署问题更加复杂, 在 2.2 节被证明是 NP-hard 问题。所以, 如何解决这个高计算复杂性问题以实现最优部署非常具有挑战性。

2) 高时空动态的感知数据相关性。交通感知数据的时空相关性具有动态性、时变性和复杂性, 经常受道路拓扑、感知站点位置、天气以及社会事件等多方面因素的影响<sup>[6]</sup>。因此, 如何基于该动态的、时变的、难建模的时空相关性实现感知数据的精确恢复非常具有挑战性。进一步地, 数据之间的相关性影响各个交通感知站点的数据分配策略, 如当感知数据

之间的相关性较差时, 需要增加边缘节点的分配数据量, 以保证数据恢复的鲁棒性。但是感知数据之间的高时空动态相关性使保证感知数据恢复精度的鲁棒性非常困难。

为了解决上述 2 个挑战, 本文提出基于边缘智能计算的城市交通感知数据自适应恢复系统, 主要包括 2 部分:

1) 具有理论下界的边缘节点次优部署分配。针对挑战 1, 首先, 通过问题等价转化, 解耦边缘节点部署和感知数据分配之间的复杂关系。然后, 通过理论证明该问题具有非负的、单调的和子模的目标函数, 以及  $p$ -独立系统的约束条件。最后, 基于该问题的性质分析, 提出基于子模理论的边缘节点次优部署算法, 能够在多项式时间复杂度内获得具有理论下界的近似最优解。

2) 基于低秩理论的感知数据自适应精确恢复。针对挑战 2, 首先, 基于实际交通感知数据集, 对感知数据进行时空维度的低秩分析; 然后, 基于该分析结果, 提出基于低秩理论的时空维度联合数据恢复算法; 最后, 针对感知数据相关性的高时空动态变化, 提出基于奇异值分解的感知数据非缺失下限估计。基于该下限估计反馈, 自适应地调整各交通感知站点的感知数据分配比例, 以保证感知数据的精确恢复, 从而提高系统数据恢复性能的鲁棒性。

综上所述, 本文主要具有 3 方面的创新和贡献:

1) 提出基于边缘智能计算的城市交通感知数据自适应恢复系统, 既利用基于子模优化理论的边缘计算, 又利用基于低秩理论的智能计算, 还基于非缺失下限估计反馈, 自适应地调整交通站点感知数据分配比例, 从而实现交通感知数据的“闭环”处理, 提高系统的鲁棒性。

2) 针对边缘节点部署 NP-hard 问题, 基于子模优化理论, 通过问题解耦转化和性质分析, 提出近似比为  $1/(2 + \lceil \max\{c_{js}\} / \min\{c_{js}\} \rceil)$  的边缘节点次优部署分配算法, 其中  $c_{js}$  表示任意一个边缘节点  $e_s$  部署在任意一个感知站点  $\lambda_j$  处的成本. 同时, 提出缺失交通感知数据的自适应恢复算法, 利用基于低秩理论的感知数据恢复, 以及基于奇异值分解的非缺失下限估计反馈调整, 实现缺失感知数据的自适应精确恢复.

3) 基于澳大利亚 600 个交通感知站点 1 年的实际感知数据构建原型系统, 并基于该系统进行全面且深入的实验评估. 结果表明, 所提算法的边缘节点部署性能达到最优性能的 90% 以上, 缺失数据恢复精度比 3 种方法至少提高 43.8%. 同时, 自适应数据恢复精度平均提高 40.3%.

## 1 相关工作

为保障智能交通系统的正常运行, 促进智慧城市进一步发展, 大量工作开始致力于研究城市交通感知数据的精确恢复. 因此本文提出了基于边缘智能计算的城市交通感知数据自适应恢复系统, 所以下面将主要介绍交通感知数据恢复和边缘节点部署方面的研究工作.

1) ITSs 交通感知数据恢复. 智能交通系统在城市化进程不断深入的当下起着越来越重要的作用, 然而交通感知数据的普遍缺失却极大地影响了 ITSs 的效用, 为此越来越多的研究工作开始致力于交通感知数据的精确恢复. 例如, 文献 [7] 利用真实数据和仿真数据, 同时结合生成式对抗网络对交通感知数据进行恢复. 文献 [8] 中以高速公路交通数据为例, 提出了一种数据融合方法, 通过挖掘感知数据内在特性以重构缺失数据. 文献 [9] 中通过将交通感知数据恢复问题建模为一个高维张量补全问题, 并采用奇异值分解来进一步了解感知数据的内在联系, 以达到缺失数据补全的目的. 文献 [4] 将城市交通感知区域建模成一个具有时空相关性的张量, 同时结合交通感知数据的城市特性完成对缺失数据的填补. 文献 [10] 中提出一种联合空间多核学习和非负矩阵分解的策略机制, 去补全缺失感知数据. 该机制充分考虑了多感知区域间多层次的相关性, 使得其能获得一个较好的数据恢复精度. 有别于文献 [4, 7–10] 的工作, 本文提出基于边缘智能计算的城市交通感知数据自适应恢复系统, 实现了交通感知数据的自适应动态精确恢复.

2) 边缘计算中的节点部署. 由于感知数据恢复存在规模庞大和计算密集等问题, 本文提出通过边缘计算来解决这一难题, 充分利用边缘服务器强大的计算能力来完成对感知数据及时精准的恢复. 目前, 也有很多其他关于边缘计算应用的研究. 例如, 文献 [11–12] 中提出利用云端服务器和边缘节点协同处理任务; 文献 [13] 中研究多个边缘节点协作优化任务卸载和服务缓存, 以达到降低整体任务时延的目的; 文献 [14] 中提出利用边缘计算来存储整理车辆的道路感知信息, 以此为无人驾驶汽车提供及时准确的道路信息, 确保驾驶的安全; 文献 [15] 中研究了云服务和端服务之间通过事件机制进行的服务适配, 并结合图论和排队论方法实现了服务的平均响应时间最小. 此外, 关于边缘节点的相关研究也有很多, 例如文献 [16–17] 中研究了关于节点部署成本最小的问题, 文献 [18–20] 中也研究了在站点容量约束和带宽约束下最小化边缘节点部署成本的问题. 本文提出的部署方案主要为处理缺失感知数据而提出, 对每个边缘节点都有一定的数据上传量下限要求. 因此, 前人的研究成果并不适用于本文的研究. 虽然文献 [5] 中提出了一种边缘节点部署和数据恢复算法, 但是没有考虑数据恢复结果对于边缘节点处的反馈. 与该工作不同, 本文提出一种基于边缘智能计算的缺失感知数据自适应恢复系统, 基于恢复结果的反馈, 自适应调整恢复过程, 从而保证数据恢复精度, 提高系统恢复的鲁棒性.

## 2 系统模型和问题建模

### 2.1 系统模型

为解决大规模 ITSs 系统的感知数据缺失问题, 本文提出基于边缘智能计算的大规模交通缺失感知数据动态自适应恢复系统模型架构, 主要包括交通感知站点、边缘节点和云服务器.

1) 交通感知站点. 在智能交通系统中, 交通感知站点被部署在城市的各条道路或其交叉路口上, 感知和监测道路实时的交通信息, 如车流量、车速等<sup>[21]</sup>. 假设有  $N$  个站点, 用  $\lambda_i$  表示第  $i$  个站点, 其中  $i \in \{1, 2, \dots, N\}$ . 此外, 用  $w_i$  表示  $\lambda_i$  的感知数据规模大小. 表 1 给出本文主要数学符号说明.

2) 边缘节点. 所有交通感知数据被传送至边缘节点进行缺失数据恢复处理. 考虑到部署的便捷性及部署成本问题, 边缘节点将被部署在交通感知站点处. 假设共有  $S$  个边缘节点, 用  $e_s$  表示第  $s$  个边缘节

**Table 1 Main Mathematics Notations Description****表 1 主要数学符号的说明**

符号	符号含义
$N, \lambda_i, w_i$	交通站数量、第 $i$ 个交通站点和其数据量
$S, e_s, d_s$	边缘节点数量、第 $s$ 个边缘节点和其容量上限
$x_{js}, c_{js}$	表示边缘节点 $e_s$ 是否被部署到交通站点 $\lambda_j$ 和其部署成本
$l_{ij}, y_{ij}$	$\lambda_i$ 是否在 $\lambda_j$ 覆盖范围内、分配数据量比例
$\mathbf{x}, \mathbf{y}$	$\mathbf{x} = (x_{js}), \mathbf{y} = (y_{ij})$
$\mathcal{G}, \mathcal{A}$	交通站点和边缘节点配对集、可行的配对集
$\mathbf{M}_{js}, m_j$	在交通站点 $\lambda_j$ 处边缘节点 $e_s$ 的缺失矩阵及其非缺失下限

点, 其中 $s \in \{1, 2, \dots, S\}$ . 用 $x_{js} = 1$ 或 0 表示边缘节点 $e_s$ 是否被部署到交通站点 $\lambda_j$ 处,  $x_{js} = 1$ 表示被部署, 否则 $x_{js} = 0$ . 每个节点的部署需耗费一定成本, 用 $c_{js}$ 表示边缘节点 $e_s$ 部署到交通站点 $\lambda_j$ 的成本. 不同的边缘节点有不同的计算存储容量, 用 $d_s$ 表示 $e_s$ 的计算存储容量. 此外, 用 $y_{ij}$ 表示交通站点 $\lambda_i$ 传输到部署在 $\lambda_j$ 处边缘节点的数据比例, 即 $y_{ij} \in [0, 1]$ . 为方便处理, 在进行缺失数据恢复时, 边缘节点首先将各个站点上传的感知数据整合成一个含缺失数据的矩阵, 即缺失矩阵, 用 $\mathbf{M}_{js}$ 表示 $\lambda_j$ 处边缘节点 $e_s$ 整合的缺失矩阵. 同时, 为保证恢复精度, 缺失矩阵的元素缺失程度不能太大, 即传送到边缘节点的非缺失感知数据量需满足一定下限, 用 $m_j$ 表示 $\lambda_j$ 处边缘节点整合矩阵的非缺失下限. 特别地, 矩阵非缺失下限值与矩阵的秩大小密切相关, 具体见 3.2.2 节. 此外, 由于外界因素, 如环境、天气、节假日等的影响, 感知数据矩阵的秩也将随之变化, 进一步地将影响到数据最后的恢复精度. 为保证在变动的外界因素下, 缺失数据的恢复精度依然能满足一定要求, 边缘节点将依据恢复结果计算相应非缺失下限值, 并基于非缺失下限值调整各交通站点的感知数据分配比例, 以保证感知数据的自适应精确恢复. 此外, 为提高恢复效率, 边缘节点应能同时处理尽可能多的感知数据.

3) 云服务器. 云服务器在整个系统中主要有 2 个功能: ①管理和控制边缘节点. 连接所有边缘节点, 并灵活高效地控制和管理各个交通感知站点和边缘节点之间的数据传输和调度<sup>[22]</sup>. 具体地, 首先, 所有边缘节点根据恢复结果, 计算非缺失下限值, 并将该值发送至云服务器; 然后, 云服务器根据各个边缘节点上传的非缺失下限值重新计算感知数据分配方案, 并反馈回各个边缘节点; 最后, 边缘节点通知各个交通感知站点新的感知数据分配比例. ②复杂的大规模数据处理<sup>[23-25]</sup>. 当边缘节点恢复缺失感知数据之后,

利用完整的交通感知数据进一步地进行复杂的大规模数据分析和挖掘, 如基于深度学习的交通感知大数据态势分析和挖掘等<sup>[26-27]</sup>.

## 2.2 问题建模

基于 2.1 节所述的系统模型, 本文研究的主要包括: 1) 如何在部署成本约束下设计边缘节点部署方案, 提高部署性能使其能更好地服务感知数据恢复, 即能同时处理尽可能多的感知数据. 2) 如何基于边缘节点部署实现感知数据的动态自适应恢复.

1) 子问题 P1——边缘节点的次优部署. 已知边缘节点的计算存储容量 $\{d_s | s \in \{1, 2, \dots, S\}\}$ 和交通感知站点的感知数据规模 $\{w_i | i \in \{1, 2, \dots, N\}\}$ , 如何设计 $S$ 个边缘节点部署到 $N$ 个交通感知站点的部署策略 $\mathbf{x} = (x_{js} | \forall j \in \{1, 2, \dots, N\}, \forall s \in \{1, 2, \dots, S\})$ , 和 $N$ 个交通感知站点传输到已部署边缘节点的感知数据量分配方案 $\mathbf{y} = (y_{ij} | \forall i, j \in \{1, 2, \dots, N\})$ , 以在总部署成本预算 $C_0$ 约束下, 最大化所有部署边缘节点处理总数据量, 即交通感知站点总上传数据量.

$$(P1) \max_{\mathbf{x}, \mathbf{y}} \sum_{i=1}^N \sum_{j=1}^N w_i y_{ij}, \quad (1)$$

$$\text{s.t. } \sum_{j=1}^N y_{ij} \leq 1, \forall i \in \{1, 2, \dots, N\}, \quad (2)$$

$$\sum_{j=1}^N x_{js} \leq 1, \forall s \in \{1, 2, \dots, S\}, \quad (3)$$

$$\sum_{i=1}^N w_i y_{ij} \leq \sum_{s=1}^S x_{js} d_s, \forall j \in \{1, 2, \dots, N\}, \quad (4)$$

$$\sum_{i=1}^N w_i y_{ij} \geq \sum_{s=1}^S x_{js} m_j, \forall j \in \{1, 2, \dots, N\}, \quad (5)$$

$$\sum_{s=1}^S \sum_{j=1}^N x_{js} c_{js} \leq C_0, \quad (6)$$

$$y_{ij} \leq l_{ij}, \forall i, j \in \{1, 2, \dots, N\}. \quad (7)$$

其中: 式(1)中 $\sum_{i=1}^N \sum_{j=1}^N w_i y_{ij}$ 表示所有部署边缘节点同时处理的总数据量; 式(2)表示 $\lambda_j$ 将部分或全部感知数据上传到边缘节点进行处理; 式(3)表示 1 个边缘节点最多被部署到 1 个交通站点处; 式(4)表示 $\lambda_j$ 处边缘节点接收到的总处理数据量不超过其计算存储容量上限; 式(5)表示 $\lambda_j$ 处边缘节点整合的感知数据矩阵其缺失数据量要满足一定的下限约束, 以避免因缺失程度太大而导致无法精确地恢复; 式(6)表示所有边缘节点的总部署成本需满足成本预算限制, 特别地, 由于通信成本远小于部署成本<sup>[28]</sup>, 所以本文不

考虑数据传输带来的成本; 式(7)表示交通感知站点之间距离较远时, 其数据之间的空间相关性比较弱, 对缺失数据恢复作用非常小<sup>[29]</sup>, 为此, 只考虑作用范围内的感知站点, 用  $l_{ij} = 1(0)$  表示  $\lambda_i$  是否在  $\lambda_j$  的作用范围内, 如果是, 则  $l_{ij} = 1$ , 否则  $l_{ij} = 0$ .

2) 子问题 P2——感知数据的自适应恢复. 如何基于边缘节点整合的缺失矩阵, 通过矩阵中完整元素恢复缺失元素. 同时基于恢复结果, 估计当前感知数据相关性程度, 通过该相关性程度大小计算保证精确恢复所需的非缺失感知数据下限值, 从而基于下限值调节感知站点上传至各边缘节点感知数据比例, 达到动态自适应精确恢复的目的. 用  $T$  表示矩阵  $\mathbf{M}_{js}$  的元素个数, 用  $v_t$  和  $\hat{v}_t$  分别表示该矩阵第  $t$  个元素的实际值和估计值,  $t \in \{1, 2, \dots, T\}$ . 因此, 感知数据恢复问题可描述为: 如何设计矩阵恢复函数  $\Phi_t(\cdot)$ , 使得所有缺失感知数据的估计值与真实值之间的误差最小, 形式化表示为:

$$(P2) \min_{\Phi_t(\cdot)} \frac{1}{T} \sum_{t=1}^T |v_t - \hat{v}_t|, \quad (8)$$

$$\text{s.t. } \hat{v}_t = \Phi_t(\mathbf{M}_{js}), \forall t \in \{1, 2, \dots, T\}. \quad (9)$$

其中: 式(8)表示感知数据的估计误差, 即平均绝对值误差<sup>[30]</sup>; 式(9)表示感知数据的估计恢复,  $\Phi_t(\mathbf{M}_{js})$  表示利用矩阵恢复函数, 基于  $\mathbf{M}_{js}$  中所有非缺失数据估计其第  $t$  个元素的值. 同时, 为达到动态自适应精确恢复的目的, 还需依据 P2 恢复结果, 更新非缺失下限值, 以保证感知数据的恢复精度.

**定理 1.** 边缘节点的最优部署分配问题 P1 是 NP-hard 问题.

证明. 当每个交通站点的感知数据量规模  $\{w_i\}$  足够大, 远大于所有边缘节点的存储容量上限  $\{d_s\}$  时, 式(2)(5)可被松弛. 原问题简化为

$$\max_x \sum_{s=1}^S \sum_{j=1}^N x_{js} d_s, \quad (10)$$

s.t. 式(3)(6).

将  $S$  个边缘节点与  $N$  个交通站点的所有配对看成  $S \times N$  个物品  $\{(j, s) | \forall s \in \{1, 2, \dots, S\}, \forall j \in \{1, 2, \dots, N\}\}$ , 并将每个节点的配对集合分为一组, 共  $S$  组. 因此, 原问题可规约为一个经典的 0-1 背包问题: 已知每个物品  $(j, s)$  的价值和成本分别为  $d_s$  和  $c_{js}$ , 如何从每组物品中至多选择 1 个物品, 使在总成本  $C_0$  约束下最大化物品价值  $\sum_{s=1}^S \sum_{j=1}^N x_{js} d_s$ . 因为该 0-1 背包问题是 NP-hard 问题<sup>[31]</sup>, 所以, 问题 P1 也是 NP-hard 问题, 证毕.

### 3 算法系统设计

为了解决 2.2 节所述的子问题 P1 和 P2, 本文提出基于边缘智能计算的城市交通感知数据动态自适应恢复系统. 如图 1 所示, 该系统基于交通感知站点中存在缺失的交通感知数据和交通感知站点的交通拓扑网络图, 利用边缘智能计算进行感知数据的动态自适应恢复, 从而得到精确完整的交通感知数据. 具体地, 该系统主要包括 2 个模块:

1) 具有理论下界的边缘节点次优部署(3.1 节).

首先, 通过问题等价转化, 在 3.1.1 节分析 P1 问题的目标函数和约束条件性质; 然后, 基于理论分析结果,

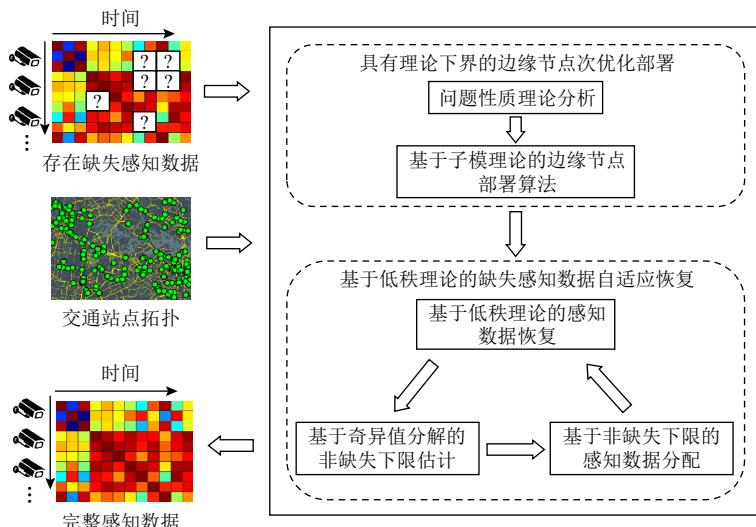


Fig. 1 Intelligent edge computing-empowered adaptive urban traffic sensing data recovery system

图 1 基于边缘智能计算的城市交通感知数据自适应恢复系统

在3.1.2节提出基于子模理论的边缘节点次优部署算法,在多项式时间内能获得该NP-hard问题的具有理论下界的近似最优解。

2) 基于低秩理论的感知数据自适应恢复(3.2节)。首先,分析交通感知数据的时空维度近似低秩特性,在3.2.1节提出基于低秩理论的交通感知数据恢复算法;然后,基于恢复结果,在3.2.2节利用奇异值分解和矩阵自由度理论估计交通感知数据的非缺失下限,依据此下限值更新感知数据分配比例,保证感知数据的精确恢复。

### 3.1 具有理论下界的边缘节点次优部署

#### 3.1.1 问题性质理论分析

1) 问题等价转化。问题P1包含2类决策变量,即边缘节点部署决策变量 $\mathbf{x}$ 和感知数据分配变量 $\mathbf{y}$ 。用 $\mathbf{y}^*(\mathbf{x})$ 表示给定边缘节点部署规划 $\mathbf{x}$ 的最优感知数据分配方案。通过理论分析,可得引理1。

**引理1。**对于问题P1,如果给定边缘节点部署方案,能够在多项式时间内获得最优的感知数据分配方案 $\mathbf{y}$ 。

证明。给定任意的边缘节点部署策略 $\mathbf{x}^0 = (x_{js}^0)$ ,问题P1转化为只关于 $\mathbf{y}$ 的感知数据分配问题P1'。

$$(P1') \max_{\mathbf{y}} \sum_{i=1}^N \sum_{j=1}^N w_i y_{ij}, \quad (11)$$

s.t. 式(2)(7),

$$\sum_{i=1}^N w_i y_{ij} \leq \sum_{s=1}^S x_{js}^0 d_s, \forall j \in \{1, 2, \dots, N\}, \quad (12)$$

$$\sum_{i=1}^N w_i y_{ij} \geq \sum_{s=1}^S x_{js}^0 m_j, \forall j \in \{1, 2, \dots, N\}. \quad (13)$$

P1'是一个简单的线性规划问题,可利用多种经典的线性规划方法<sup>[32]</sup>在多项式时间内求得最优解,

证毕。

用 $H(\mathbf{x}, \mathbf{y})$ 表示问题P1的目标函数。因此,根据引理1,可以将 $H(\mathbf{x}, \mathbf{y})$ 转化为只关于决策变量 $\mathbf{x}$ 的目标函数 $H(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))$ 。进一步地,定义集合函数 $\mathcal{G} := \{(j, s) \mid \forall j \in \{1, 2, \dots, N\}, s \in \{1, 2, \dots, S\}\}$ ,  $\mathcal{A} := \{(j, s) \mid x_{js} = 1, \forall j \in \{1, 2, \dots, N\}, s \in \{1, 2, \dots, S\}\}$ ,  $F(\mathcal{A}) := H(\mathbf{x}, \mathbf{y}^*(\mathbf{x}))$ 。因此,可将问题P1等价转化为集合函数优化问题P1'':

$$(P1'') \max_{\mathcal{A} \subseteq \mathcal{G}} F(\mathcal{A}), \quad (14)$$

$$\text{s.t. } \sum_{j:(j,s) \in \mathcal{A}} \mathbb{1}_{(j,s) \in \mathcal{A}} \leq 1, \forall s \in \{1, 2, \dots, S\}, \quad (15)$$

$$\sum_{(j,s) \in \mathcal{A}} c_{js} \leq C_0, \quad (16)$$

其中 $\mathbb{1}(\cdot)$ 表示指示函数。

2) 问题性质分析。下面基于定义1和定义2分别分析问题P1''的目标函数性质(如引理2)和约束条件性质(如引理3)。

**定义1.**非负性、单调性和子模性<sup>[33]</sup>。对于一个集合函数 $F(\cdot) : 2^{\mathcal{R}} \rightarrow \mathbb{R}$ ( $\mathcal{R}$ 是一个有限集),如果 $F(\emptyset) = 0$ 并且 $F(\mathcal{A}) \geq 0$  ( $\forall \mathcal{A} \subseteq \mathcal{R}$ ),则集合函数 $F(\cdot)$ 是非负的;如果 $\forall \mathcal{A}_1 \subseteq \mathcal{A}_2 \subseteq \mathcal{R}$ ,  $F(\mathcal{A}_1) \leq F(\mathcal{A}_2)$ ,则 $F(\cdot)$ 是单调的;当且仅当 $\forall \mathcal{A}_1 \subseteq \mathcal{A}_2 \subseteq \mathcal{R}$ ,  $\forall e \in \mathcal{R} \setminus \mathcal{A}_2$ ,  $F(\mathcal{A}_1 \cup \{e\}) - F(\mathcal{A}_1) \geq F(\mathcal{A}_2 \cup \{e\}) - F(\mathcal{A}_2)$ ,  $F(\cdot)$ 是子模的。

**引理2.** P1''的目标函数 $F(\mathcal{A})$ ( $\mathcal{A} \subseteq \mathcal{G}$ )是非负的、单调的和子模的。

证明。首先,由于 $F(\mathcal{A})$ 的物理含义是感知数据量,所以易证其是非负的和单调的。

其次,证明 $F(\mathcal{A})$ 是子模的。根据定义1,要证其是子模的,需证明 $\forall \mathcal{A}_1 \subseteq \mathcal{A}_2 \subseteq \mathcal{G}$ ,  $\forall (j_1, s_1) \in \mathcal{G} \setminus \mathcal{A}_2$ , 下列不等式恒成立:

$$\begin{aligned} F(\mathcal{A}_1 \cup \{(j_1, s_1)\}) - F(\mathcal{A}_1) &\geq \\ F(\mathcal{A}_2 \cup \{(j_1, s_1)\}) - F(\mathcal{A}_2). \end{aligned} \quad (17)$$

用 $\mathbf{y}^{(1)} = (y_{ij}^{(1)})$ ,  $\hat{\mathbf{y}}^{(1)} = (\hat{y}_{ij}^{(1)})$ ,  $\mathbf{y}^{(2)} = (y_{ij}^{(2)})$ ,  $\hat{\mathbf{y}}^{(2)} = (\hat{y}_{ij}^{(2)})$ 分别表示基于部署方案 $\mathcal{A}_1$ ,  $\mathcal{A}_1 \cup \{(j_1, s_1)\}$ ,  $\mathcal{A}_2$ ,  $\mathcal{A}_2 \cup \{(j_1, s_1)\}$ ,求解问题P1'所得到的最优感知数据分配解。为便于后续表述,用 $\mathbf{z}^{(1)} = (z_{ijs}^{(1)})$ ,  $\hat{\mathbf{z}}^{(1)} = (\hat{z}_{ijs}^{(1)})$ ,  $\mathbf{z}^{(2)} = (z_{ijs}^{(2)})$ ,  $\hat{\mathbf{z}}^{(2)} = (\hat{z}_{ijs}^{(2)})$ ,等价表示 $\mathbf{y}^{(1)} = (y_{ij}^{(1)})$ ,  $\hat{\mathbf{y}}^{(1)} = (\hat{y}_{ij}^{(1)})$ ,  $\mathbf{y}^{(2)} = (y_{ij}^{(2)})$ ,  $\hat{\mathbf{y}}^{(2)} = (\hat{y}_{ij}^{(2)})$ 。其中 $z_{ijs}$ 表示交通感知站点 $\lambda_i$ 上传至部署在 $\lambda_j$ 处边缘节点 $e$ 的感知数据比例。因此可得

$$F(\mathcal{A}_1 \cup \{(j_1, s_1)\}) - F(\mathcal{A}_1) = \sum_{(j,s) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)}) + \sum_{i=1}^N w_i \hat{z}_{i(j_1, s_1)}^{(1)}.$$

同理可得

$$F(\mathcal{A}_2 \cup \{(j_1, s_1)\}) - F(\mathcal{A}_2) = \sum_{(j,s) \in \mathcal{A}_2} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(2)} - z_{ijs}^{(2)}) + \sum_{i=1}^N w_i \hat{z}_{i(j_1, s_1)}^{(2)}.$$

特别地,由于问题P1'是线性规划问题,它的最优感知数据分配解可能不唯一。因此,为了分析方便,定义 $\hat{\mathbf{z}}^{(1)} = (\hat{z}_{ijs}^{(1)})$ 和 $\hat{\mathbf{z}}^{(2)} = (\hat{z}_{ijs}^{(2)})$ 为所有最优分配解中节点 $(j_1, s_1)$ 上感知数据最小的解。为证明不等式(17)成立,下面证明2个结论。

1) 证明

$$\sum_{(j,s) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)}) = \sum_{(j,s) \in \mathcal{A}_2} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(2)} - z_{ijs}^{(2)}).$$

$\sum_{(j,s) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)})$ 和 $\sum_{(j,s) \in \mathcal{A}_2} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(2)} - z_{ijs}^{(2)})$ 分别表示加入部署新节点 $(j_1, s_1)$ 后原部署方案 $\mathcal{A}_1$ 和 $\mathcal{A}_2$ 中边缘节点变化的感知数据量。

首先,证明

$$\sum_{(j, s) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)}) = 0.$$

因为  $(z_{ijs}^{(1)})$  是部署方案  $\mathcal{A}_1$  所对应的最优感知数据分配解, 所以

$$\sum_{(j_1, s_1) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)}) \leq 0.$$

然后利用反证法可证明

$$\sum_{(j_1, s_1) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)}) \geq 0.$$

具体证明: 假设该结论不成立, 即

$$\sum_{(j_1, s_1) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)}) < 0.$$

因此, 该不等式表明加入节点  $(j_1, s_1)$  后,  $\mathcal{A}_1$  中边缘节点的感知数据量减少. 然而, 根据目标函数的单调非减性可知,  $\mathcal{A}_1 \cup \{(j_1, s_1)\}$  中节点的感知数据量不会减少. 因此, 节点  $(j_1, s_1)$  的感知数据量增加, 所以可调整节点  $(j_1, s_1)$  的感知数据到  $\mathcal{A}_1$  中的边缘节点. 所以  $\mathcal{A}_1 \cup \{(j_1, s_1)\}$  对应的感知数据分配解不是所有最优解中  $(j_1, s_1)$  数据量最小的解. 与结论的前提条件矛盾, 即  $\hat{z}^{(1)} = (\hat{z}_{ijs}^{(1)})$  是所有最优解中  $(j_1, s_1)$  数据量最小的解. 所以假设不成立, 原结论成立, 即

$$\sum_{(j_1, s_1) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)}) \geq 0.$$

进一步地, 因为

$$\sum_{(j_1, s_1) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)}) \leq 0.$$

所以,

$$\sum_{(j_1, s_1) \in \mathcal{A}_1} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(1)} - z_{ijs}^{(1)}) = 0.$$

同理可证

$$\sum_{(j, s) \in \mathcal{A}_2} \sum_{i=1}^N w_i (\hat{z}_{ijs}^{(2)} - z_{ijs}^{(2)}) = 0.$$

综上, 结论 1) 成立.

2) 证明

$$\sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(1)} \geq \sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(2)}.$$

$\sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(1)}$  和  $\sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(2)}$  分别表示在部署方案  $\mathcal{A}_1$  和  $\mathcal{A}_2$  下新节点  $(j_1, s_1)$  分配的感知数据量. 因为  $\mathcal{A}_1 \subseteq \mathcal{A}_2$ , 所以  $\mathcal{A}_1 \cup \{(j_1, s_1)\} \cup (\mathcal{A}_2 \setminus \mathcal{A}_1) = \mathcal{A}_2 \cup \{(j_1, s_1)\}$ . 由结论 1) 可得, 在  $\mathcal{A}_1 \cup \{(j_1, s_1)\}$  中加入  $\mathcal{A}_2 \setminus \mathcal{A}_1$  后,  $\mathcal{A}_1 \cup \{(j_1, s_1)\}$

感知数据量不变. 因此, 利用反证法可证明

$$\sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(1)} \geq \sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(2)}.$$

具体证明: 假设该结论不成立, 即

$$\sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(1)} < \sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(2)}.$$

因此, 该不等式表明  $\mathcal{A}_1 \cup \{(j_1, s_1)\}$  加入  $\mathcal{A}_2 \setminus \mathcal{A}_1$  后, 节点  $(j_1, s_1)$  的感知数据量增加. 由于  $\mathcal{A}_1 \cup \{(j_1, s_1)\}$  中边缘节点的感知数据量在  $\mathcal{A}_2 \setminus \mathcal{A}_1$  加入前后保持不变, 所以可调节  $(j_1, s_1)$  上的感知数据到  $\mathcal{A}_1$  方案中的边缘节点. 所以,  $\mathcal{A}_2 \cup \{(j_1, s_1)\}$  对应的感知数据分配解不是所有最优解中  $(j_1, s_1)$  数据量最小的解. 所以假设不成立, 原结论成立, 即

$$\sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(1)} \geq \sum_{i=1}^N w_i \hat{z}_{ijs_1}^{(2)}.$$

结论 2) 成立.

基于上述 2 个结论, 可得不等式(17)成立. 因此,  $F(\mathcal{A})$  是子模的. 证毕.

**定义 2.**  $p$ -独立系统<sup>[34]</sup>. 给定一个任意有限集合  $\mathcal{G}$ , 用  $\mathcal{I}$  表示  $\mathcal{G}$  的子集构成的集合族, 即  $\mathcal{I} \subseteq 2^{\mathcal{G}}$ . 则集合对  $(\mathcal{G}, \mathcal{I})$  被称为独立系统, 当且仅当满足 2 个条件: ①  $\emptyset \in \mathcal{I}$ ; ② 如果  $\mathcal{A} \subseteq \mathcal{B} \in \mathcal{I}$ , 则  $\mathcal{A} \in \mathcal{I}$ .

说明:  $\mathcal{I}$  中的元素被称为独立集. 同时,  $\forall \mathcal{A} \subseteq \mathcal{G}$ , 如果  $\mathcal{A}$  中的某个独立集不是  $\mathcal{A}$  中的任何一个独立集的子集, 那么该独立集被称为  $\mathcal{A}$  的基. 用  $r(\mathcal{A})$  和  $l(\mathcal{A})$  分别表示  $\mathcal{A}$  的最大基和最小基元素的个数. 对于一个任意独立系统  $(\mathcal{G}, \mathcal{I})$  和任意集合  $\mathcal{A} \subseteq \mathcal{G}$ , 如果  $\max_{\mathcal{A} \subseteq \mathcal{G}} (r(\mathcal{A})/l(\mathcal{A})) \leq p$ , 则独立系统  $(\mathcal{G}, \mathcal{I})$  被称为  $p$ -独立系统.

**引理 3.** 式(15)(16)构成  $p$ -独立系统, 且  $p = 1 + \lceil \max\{c_{js}\} / \min\{c_{js}\} \rceil$ , 其中  $c_{js}$  表示任意一个边缘节点  $e_s$  部署在任意一个交通站点  $\lambda_j$  的成本,  $\forall j \in \{1, 2, \dots, N\}$ ,  $\forall s \in \{1, 2, \dots, S\}$ .

证明. 由定义 2 可知原证明等价于证明  $(\mathcal{G}, \mathcal{I}_p)$  构成  $p$ -独立系统, 其中  $\mathcal{G}$  表示边缘节点和交通站点所有可能的配对集合,  $\mathcal{I}_p$  表示所有可行部署方案的集合. 首先, 显然有  $\emptyset \in \mathcal{I}_p$ . 此外, 对于任意 2 个部署方案  $X, Y$ , 若其满足  $X \subseteq Y \in \mathcal{I}_p$ , 则由于满足问题约束条件的可行解的子集也必定满足约束条件, 所以任意可行解的子集也必为可行解. 所以必有  $X \in \mathcal{I}_p$ . 所以, 根据定义 2,  $(\mathcal{G}, \mathcal{I}_p)$  是一个独立系统.

考虑任意集合  $\mathcal{A} \subseteq \mathcal{G}$ , 因为  $\mathcal{I}_p$  是可行解集合. 所以  $\mathcal{A}$  中独立集都是可行解. 用  $\mathcal{A}_1$  和  $\mathcal{A}_2$  分别表示  $\mathcal{A}$  中

任意 2 个最大可行解(即满足全部约束已无法再加入元素), 现如果向  $\mathcal{A}_2$  中增加元素  $(j, s) \in \mathcal{A}_1 \setminus \mathcal{A}_2$ , 就需要在  $\mathcal{A}_2$  中拿出一些元素来满足约束. 为满足式(15), 需取出至多 1 个元素. 为了满足式(16), 需取出至多  $\lceil \max\{c_{js}\} / \min\{c_{js}\} \rceil$  个元素. 如此循环直到  $\mathcal{A}_2 = \mathcal{A}_1$ . 当最差情况发生时, 换入  $|\mathcal{A}_1|$  个元素需至多换出  $p|\mathcal{A}_1|$  个元素, 即  $\mathcal{A}_2$  中元素个数至多为  $\mathcal{A}_1$  中元素个数的  $p$  倍, 所以, 根据定义 2 可得式(15)(16)构成  $p$ -独立系统. 证毕.

### 3.1.2 基于子模理论的边缘节点次优部署算法

根据引理 2 和引理 3 可得, 子问题 P1 具有非负的、单调的、子模的目标函数, 以及  $p$ -独立系统的约束条件. 因此, 基于文献 [35] 的启发, 为求解子问题 P1, 本文提出基于子模理论的边缘节点次优部署算法. 如算法 1 所示.

#### 算法 1. 具有理论下界的边缘节点次优部署算法.

输入: 边缘节点容量上限  $\{d_s | s \in \{1, 2, \dots, S\}\}$ , 交通站点数据规模  $\{w_i | i \in \{1, 2, \dots, N\}\}$ , 边缘节点部署成本集合  $\{c_{js}\}$ , 总成本预算  $C_0$ ;

输出: 边缘节点部署方案  $\mathbf{x} = (x_{js})$ , 感知数据分配方案  $\mathbf{y} = (y_{ij})$ .

- ① 初始化  $\mathcal{G} = \{x_{js} = 0 | \forall j, s\}$ ;
- ② 初始化  $\mathcal{A} = \emptyset, \mathcal{B} = \emptyset$ ;
- ③ while  $\mathcal{B} \neq \mathcal{G}$
- ④  $v = \arg \max_{e \in \mathcal{G} \setminus \mathcal{B} \text{ satisfying (15)(16)}} F(\mathcal{A} \cup \{e\})$ ;
- ⑤  $\mathcal{A} = \mathcal{A} \cup \{v\}, \mathcal{B} = \mathcal{B} \cup \{v\}$ ;
- ⑥ for  $\forall e \in \mathcal{G} \setminus \mathcal{B}$
- ⑦ if  $\mathcal{A} \cup \{e\}$  不满足式(15)或式(16)
- ⑧  $\mathcal{B} = \mathcal{B} \cup \{e\}$ ;
- ⑨ end if
- ⑩ end for
- ⑪ end while
- ⑫ 令  $x_{js} = 1, \forall (j, s) \in \mathcal{A}$ ;
- ⑬ 基于  $\mathbf{x} = (x_{js})$ , 利用线性规划方法计算  $\mathbf{y}$ ;
- ⑭ 返回  $\mathbf{x}, \mathbf{y}$ .

算法 1 的核心思想是在每次迭代中, 贪心地选择当前满足约束的、边际效益值最大的边缘节点部署方案. 具体主要包含 2 步:

1) 添加新元素(行④⑤). 计算  $\mathcal{G} \setminus \mathcal{B}$  中所有元素边际效益值, 将其中边际效益值最大的元素加入解集  $\mathcal{A}$  中. 其中,  $\mathcal{B}$  表示当前加入  $\mathcal{A}$  中不满足约束的元素集合.

2) 剔除冲突元素(行⑥~⑩). 在每次向解集  $\mathcal{A}$  中添加新元素后, 遍历  $\mathcal{G} \setminus \mathcal{B}$ , 将当前无法再被添加至  $\mathcal{A}$

中的元素全添加至  $\mathcal{B}$  中. 当  $\mathcal{B} = \mathcal{G}$  时, 表明当前已无可添加至解集  $\mathcal{A}$  的元素. 此时得到的  $\mathcal{A}$  即为最终解, 而后基于  $\mathcal{A}$ , 进一步计算边缘部署方案  $\mathbf{x}$  和感知数据分配方案  $\mathbf{y}$ .

分析算法 1 的时间复杂度及解最优性, 可得定理 2.

**定理 2.** 算法 1 能够在多项式时间内获得近似比为  $1/(2 + \lceil \max\{c_{js}\} / \min\{c_{js}\} \rceil)$  的近似最优解.

证明. 基于引理 2 和 3, 子问题 P1 是具有非负、单调和子模的目标函数及  $p$ -独立系统约束的优化问题. 由文献 [35] 可得, 基于算法 1, 可求得近似比为  $1/(1+p)$  的近似最优解. 根据引理 3 得  $p = 1 + \lceil \max\{c_{js}\} / \min\{c_{js}\} \rceil$ , 因此, 算法 1 可获得近似比为  $1/(2 + \lceil \max\{c_{js}\} / \min\{c_{js}\} \rceil)$  的近似最优解. 此外, 算法 1 最多迭代  $S$  次, 每次迭代的时间复杂度为  $O(NS\Gamma)$ , 其中,  $N$  和  $S$  分别表示交通感知站点和边缘节点的数量,  $\Gamma$  表示求解线性规划问题 P1' 的多项式时间耗费<sup>[32]</sup>. 因此, 算法 1 具有多项式时间复杂度, 即  $O(NS^2\Gamma)$ , 证毕.

### 3.2 基于低秩理论的感知数据自适应恢复

#### 3.2.1 基于低秩理论的感知数据恢复

1) 基于低秩分析的问题转化. 利用实际智能交通感知数据集(详见 4.1 节), 分别基于时间维度和空间维度, 分析不同时间和不同站点交通感知数据的低秩性. 首先, 分析在时间维度上的低秩特性. 从数据集中随机选取 4 个交通站点 52h 的感知数据组成矩阵, 并对该矩阵进行奇异值分解. 如图 2(a) 所示, 矩阵前 10 个最大奇异值占全部奇异值之和的 85% 以上, 即奇异值累积贡献率为 85%. 类似地, 分析交通感知数据在空间维度的低秩特性. 对 25 个交通站点感知数据构成的矩阵进行奇异值分解, 如图 2(b) 所示, 前 5 个最大奇异值占全部奇异值之和的 90% 以上. 根据低秩理论<sup>[36]</sup>: 如果矩阵中存在少量奇异值之和占所有奇异值总和的比值较大, 则该矩阵可以近似认为低秩矩阵. 因此分析可得, 交通感知数据矩阵在时间维度和空间维度上都是近似低秩的.

基于上述结论可得,  $\mathbf{M}_{js}$  在时空维度是近似低秩的. 由文献 [37] 可知, 如果矩阵是近似低秩的, 则通过采样其元素可以找到另一个低秩矩阵  $\mathbf{X}$  来近似表示该矩阵. 因此子问题 P2 可等价转化为:

$$(P2') \min \text{rank}(\mathbf{X}), \quad (18)$$

$$\text{s.t. } P_{\Omega}(\mathbf{X}) = P_{\Omega}(\mathbf{M}_{js}), \quad (19)$$

其中  $\text{rank}(\cdot)$  表示矩阵的秩函数,  $P_{\Omega}(\cdot)$  表示投影采样算子. 因为该秩函数是非连续和非凸的, 求解非常困难, 且最小化矩阵秩即最小化矩阵非零奇异值个

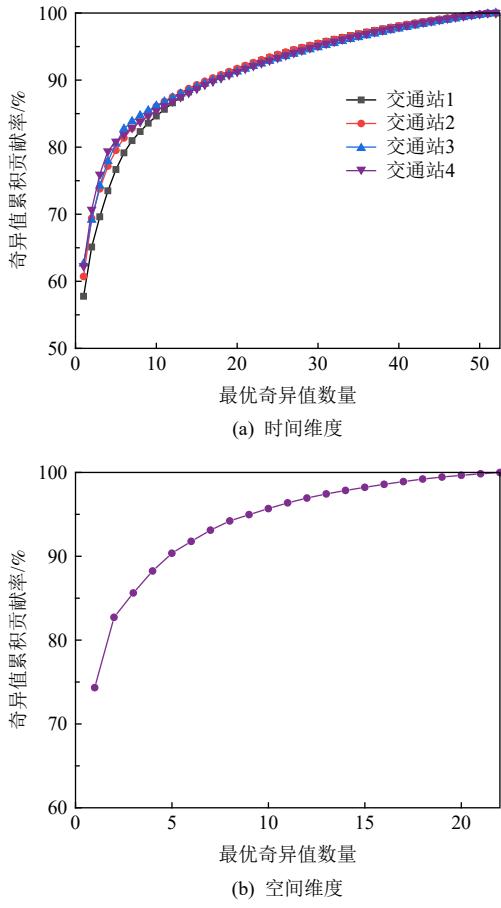


Fig. 2 Low-rank analysis of traffic sensing data matrix based SVD in terms of temporal and spatial dimensions

图 2 在时空维度下基于奇异值分解的交通感知数据低秩分析

数, 可近似看作最小化矩阵奇异值总和(即核范数最小)<sup>[37]</sup>. 因此, 为便于求解, 可用核范数 $\|X\|^*$ 函数近似代替目标秩函数 $\text{rank}(X)$ .

2) 感知数据精确恢复. 受文献[38]启发, 本文提出基于奇异值阈值截断(singular value thresholding, SVT)的缺失数据迭代恢复算法. 该算法的核心思想是在每次迭代中对矩阵进行奇异值分解, 然后将较小奇异值置为0, 构造新矩阵进行下一轮奇异值分解. 如算法2所示.

**算法2.** 基于低秩理论的感知数据自适应恢复算法.

输入: 边缘节点整合的含缺失数据的矩阵 $\mathbf{M}_{js}$ ;

输出: 完整交通感知数据矩阵 $\mathbf{X}^{\text{opt}}$ .

- ① 根据矩阵 $\mathbf{M}_{js}$ 及其缺失元素生成投影采样算子 $P_{\Omega}(\mathbf{M}_{js})$ ;
- ② 初始化 $\mathbf{Y}^0 = k_0 \delta P_{\Omega}(\mathbf{M}_{js})$ ;
- ③ for  $k = 1$  to  $k_{\max}$
- ④ 基于矩阵 $\mathbf{Y}^{k-1}$ 的奇异值分解, 得到 $\mathbf{U}^{k-1}, \Sigma^{k-1},$

- $\mathbf{V}^{k-1}$ ;
  - ⑤ 基于 $\mathbf{U}^{k-1}, \Sigma^{k-1}, \mathbf{V}^{k-1}$ , 根据式(21)(22)计算新矩阵 $\mathbf{X}^k$ ;
  - ⑥ if  $\|P_{\Omega}(\mathbf{X}^k - \mathbf{M}_{js})\|_F / \|P_{\Omega}(\mathbf{M}_{js})\|_F \leq \varepsilon$
  - ⑦  $\mathbf{X}^{\text{opt}} = \mathbf{X}^k$ ;
  - ⑧ break;
  - ⑨ else
  - ⑩ 根据式(20)计算新矩阵 $\mathbf{Y}^k$ ;
  - ⑪ end if
  - ⑫ end for
  - ⑬ 基于 $\mathbf{X}^{\text{opt}}$ , 根据式(26)估计恢复矩阵秩 $r_{js}$ ;
  - ⑭ 根据矩阵的秩估计 $r_{js}$ , 计算非缺失下限 $m_j$ , 反馈调整数据分配, 更新矩阵 $\mathbf{M}_{js}$ .
- 算法2主要包括2步:
- 1) 奇异值分解(行④). 对 $\mathbf{Y}^{k-1}$ 进行奇异值分解为 $\mathbf{U}^{k-1}, \Sigma^{k-1}, \mathbf{V}^{k-1}$ .
  - 2) 新矩阵构造(行⑤~⑩). 首先, 用 $\delta_b^{k-1}$ 表示 $\Sigma^{k-1}$ 中第 $b$ 个对角元素即奇异值(总个数为 $B$ ); 然后, 根据式(20), 基于 $\mathbf{U}^{k-1}, \Sigma^{k-1}, \mathbf{V}^{k-1}$ 构造用于下次迭代的新矩阵 $\mathbf{Y}^k$ , 其中 $\Omega$ 为采样元素索引集合.
- $$\mathbf{Y}_{a,b}^k = \begin{cases} 0, & \forall (a,b) \notin \Omega, \\ \mathbf{Y}_{a,b}^{k-1} + \delta((\mathbf{M}_{js})_{a,b} - \mathbf{X}_{a,b}^k), & \forall (a,b) \in \Omega, \end{cases} \quad (20)$$
- $$\mathbf{X}^k = \sum_{b=1}^B (\delta_b^{k-1} - \tau) \mathbf{U}_b^{k-1} \mathbf{V}_b^{k-1}, \quad (21)$$
- $$\delta_b^{*k-1} = \begin{cases} 0, & \forall \delta_b^{k-1} \leq \tau, \\ \delta_b^{k-1}, & \forall \delta_b^{k-1} > \tau. \end{cases} \quad (22)$$

以上2步相互迭代, 直至满足以下任意一个迭代停止条件: ①迭代达到最大次数 $k_{\max}$ (即行③); ②恢复得到的矩阵 $\mathbf{X}^k$ 与原矩阵 $\mathbf{M}_{js}$ 误差不大于 $\varepsilon$ (即行⑥).

### 3.2.2 基于奇异值分解的非缺失下限估计

当 $\mathbf{M}_{js}$ 中缺失元素过多, 即边缘节点 $e_s$ 处感知数据量较少, 而感知数据间的相关性程度又较小时, 很难精确地进行缺失数据恢复. 因此, 为保证数据恢复精度, 每个边缘节点分配到的感知数据量需满足一定下限, 即非缺失下限. 根据矩阵自由度理论<sup>[39]</sup>, 精确恢复一个含有缺失元素的矩阵需要的最少采样元素个数为

$$N \geq r(n_1 + n_2 - r), \quad (23)$$

其中 $r$ 表示该矩阵的秩即元素间相关性程度,  $n_1$ 和 $n_2$ 分别表示该矩阵的行列数. 由式(23)可知, 矩阵元素间相关性程度越小, 精确恢复缺失数据所需的最少采样元素个数越多.

根据式(23), 当计算基于算法2恢复得到的完整矩阵 $\mathbf{X}^{\text{opt}}$ 的非缺失下限 $m_j$ 时,  $n_1$ 和 $n_2$ 很容易基于其行

列数得到. 而矩阵的秩  $r_{js}$  需进行估计, 下面基于奇异值分解估计矩阵  $\mathbf{X}^{\text{opt}}$  的秩  $r_{js}$ :

首先对矩阵  $\mathbf{X}^{\text{opt}}$  进行奇异值分解为

$$\mathbf{X}^{\text{opt}} = \mathbf{U}_{m \times m} \boldsymbol{\Sigma}_{m \times n} \mathbf{V}_{n \times n}^T, \quad (24)$$

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{pmatrix}, \quad (25)$$

其中  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0$ .

由于矩阵的低秩性, 因此, 可用占全部奇异值总和比例为  $\eta$  ( $\eta \in [0, 1]$ ) 的最少奇异值个数作为矩阵秩的估计值:

$$r_{js} = \min \left\{ r \left| \begin{array}{l} \sum_{b=1}^r \sigma_b \\ \sum_{b=1}^B \sigma_b \geq \eta \end{array} \right. \right\}, \quad (26)$$

其中  $B$  为矩阵非零奇异值总个数.

最后, 根据矩阵秩的估计值  $r_{js}$ , 基于式(23)计算非缺失下限, 并将该结果反馈至云服务器重新调整感知数据分配方案(子问题 P1'), 以保证缺失感知数据的精确恢复.

## 4 基于大规模交通感知数据集的实验评估

### 4.1 数据集和实验方法介绍

1) 大规模智能交通感知数据集介绍. 本文利用澳大利亚新南威尔士州的交通流量监控系统 TVVS 的感知数据集进行实验评估. 该感知数据集由部署在道路旁的大约 600 个交通感知站点的交通监控感知系统产生, 基本覆盖整个新南威尔士州. 本文主要采用每个交通站点的交通流量感知数据, 采样间隔为 1h, 采样时间长度为 12 个月, 即 2018-01-01—2018-12-31.

2) 实验方法介绍. 为评估边缘节点部署性能, 本文参照实际场景, 设定交通站点数据量规模和边缘节点容量等其他参数, 同时通过变化交通站点数据量规模、边缘节点容量及部署成本预算, 分别评估 3 种因素对边缘节点部署性能, 即所有节点能同时处理的总感知数据量的影响. 在实验评估中, 同时考虑了 2 种不同规模的网络, 即大规模网络 ( $N = 100$ ,  $S = 20$ ) 和小规模网络 ( $N = 10$ ,  $S = 5$ ).

为了充分地评估交通感知数据恢复的性能, 随机地选择 24 个交通感知站点, 将它们中无缺失的感知数据组成实验数据矩阵. 进一步地, 在评估过程中, 首先变化缺失数据长度(即矩阵缺失元素个数); 然

后固定缺失数据长度, 变化交通感知站点的数量, 以评估站点数量对缺失感知数据恢复性能的影响.

与当前多数边缘节点部署工作<sup>[19, 28, 40]</sup>类似, 为了全面地评估边缘节点的部署性能, 本文采用 3 种性能不同的典型对比算法, 即最优的、近似最优的、一般的对比算法. 1) OPT. 即最优的边缘节点部署方案, 基于暴力搜索对所有可能的部署方案进行穷举以获得最优解. 由于本文的边缘节点部署问题是 NP-hard 问题, 因此, OPT 具有指数级计算复杂度, 只能适用于小规模网络. 2) HEU<sup>[41]</sup>. 即基于遗传算法的近似最优启发式算法, 将部署方案类比成生物种群, 初始化若干方案, 并通过模拟自然界种群进化的过程(即遗传、交叉和变异), 不断迭代更新以得到近似最优解. 3) RAN. 即随机部署算法, 将边缘节点随机地部署在任意交通站点处.

为了充分评估缺失感知数据恢复性能, 采用 3 种常用的数据恢复算法作为对比算法: 1) MR. 即基于回归拟合的缺失数据恢复算法, 利用非缺失数据拟合多项式, 从而基于该多项式补全缺失数据. 2) KNN<sup>[42]</sup>. 基于  $K$  近邻的缺失数据恢复算法, 利用感知数据的时空相关性, 通过距离缺失元素最近的  $k$  个数据的平均值来估计该缺失值. 3) RF<sup>[43]</sup>. 基于随机森林回归的缺失感知数据恢复算法, 即基于无缺失数据训练随机森林模型, 用该模型预测缺失数据.

在实验评估中, 本文采用 3 种评估指标: 1) 边缘节点处理总数据量, 即所有部署边缘节点接收到的总感知数据量, 如式(1); 2) 平均绝对误差 (mean absolute error, MAE), 即补全数据矩阵元素的估计值与真值之间误差的绝对平均值, 如式(27); 3) 平均绝对百分比误差 (mean absolute percentage error, MAPE), 即补全数据矩阵元素的估计误差与真值的百分比平均值, 如式(28).

$$MAE = \frac{1}{T} \sum_{t=1}^T |v_t - \hat{v}_t|, \quad (27)$$

$$MAPE = \frac{1}{T} \sum_{t=1}^T \left| \frac{v_t - \hat{v}_t}{v_t} \right|. \quad (28)$$

### 4.2 实验结果

#### 1) 边缘节点部署性能评估

① 评估不同交通感知站点的感知数据规模对边缘节点部署性能的影响. 如图 3 所示, 变化交通站点数据规模为实际规模的 1.0~2.0 倍. 实验结果表明, 在小规模网络场景下本文算法比 RAN 算法性能平均提升 25%. 同时, 本文算法与 HEU 算法都能达到 OPT

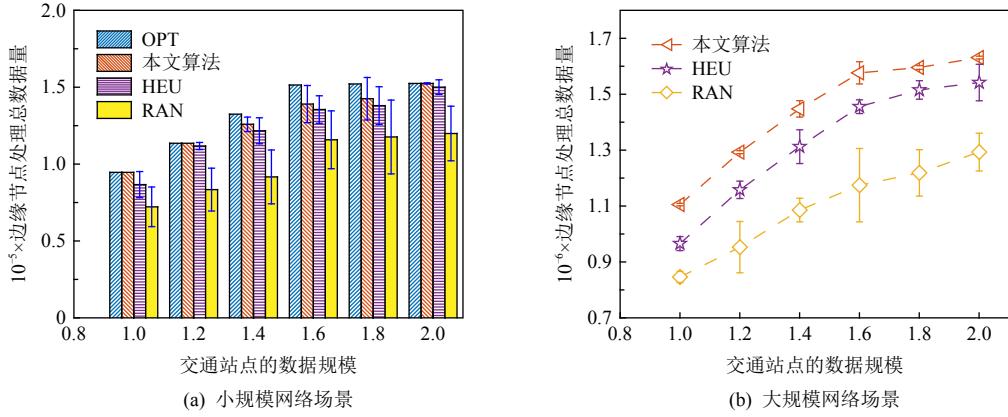


Fig. 3 Impact of different sensing data scales of traffic stations on the performance of edge node deployment

图3 不同交通站点感知数据规模对边缘节点部署性能的影响

算法的 95% 以上. 主要原因是小规模网络场景中, 满足约束的可行部署方案较少, 使得 HEU 算法有较大概率获得性能较优的解, 从而性能与本文算法以及 OPT 算法接近. 但是, 在大规模网络场景中, 本文算法比 HEU 算法平均提升性能 8.8%, 比 RAN 算法平均提升性能 31.6%.

②评估不同边缘节点容量对边缘节点部署性能的影响. 如图 4 所示, 本文算法比 RAN 算法, 在小规模和大规模网络场景中性能分别平均提升 25.9% 和 28.1%. 虽然本文算法和 HEU 算法在小规模网络场景中较为接近, 都达到了 OPT 性能的 95% 以上, 但是在大规模场景下本文算法比 HEU 性能平均提升 10.3%.

③评估不同部署成本预算对边缘节点部署性能的影响. 如图 5 所示, 本文算法比 RAN 算法, 在小规模和大规模网络场景中性能分别平均提升 45.2% 和 32.5%. 同时, 在小规模场景中, 本文算法和 HEU 算法都能达到 OPT 性能的 92%. 但是, 在大规模网络场景下, 本文算法的性能比 HEU 算法平均提高 11.9%.

## 2) 交通感知数据恢复精度评估

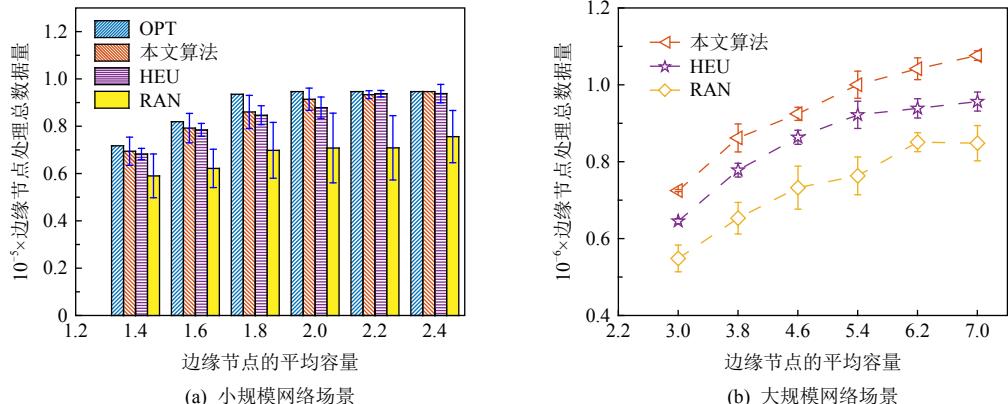


Fig. 4 Impact of different edge node capacities on the performance of edge node deployment

图4 不同边缘节点容量对边缘节点部署性能的影响

为了评估本文所提缺失感知数据恢复算法的性能, 分别与 3 种常见缺失数据恢复算法在变化缺失长度和交通站点数量情况下进行比较.

①评估不同感知数据缺失长度对缺失数据恢复精度的影响. 如图 6 所示, 实验结果表明, 随着感知数据缺失长度的增大, 缺失数据恢复的精度也不断降低. 该结果符合常理, 即缺失的感知数据量越大, 则数据所包含信息量就越小, 因此, 数据恢复的精度也就越差. 同时, 与 RF 算法、MR 算法和 KNN 算法比较, 本文算法的 MAE 指标分别平均降低 48.0%, 65.1%, 74.8%, MAPE 指标分别平均降低 47.8%, 62.4%, 78.9%.

此外, 当缺失数据长度较大时, MR 和 KNN 算法恢复的结果不够理想, 这是因为基于均值恢复和多项式拟合恢复数据的算法对数据的非缺失程度要求较高, 所以当数据缺失程度较大时, 选择 SVT 算法能够保证较好的补全精度.

②评估不同交通站点数量对缺失数据恢复精度的影响. 如图 7 所示, 实验结果表明, 随着交通站点

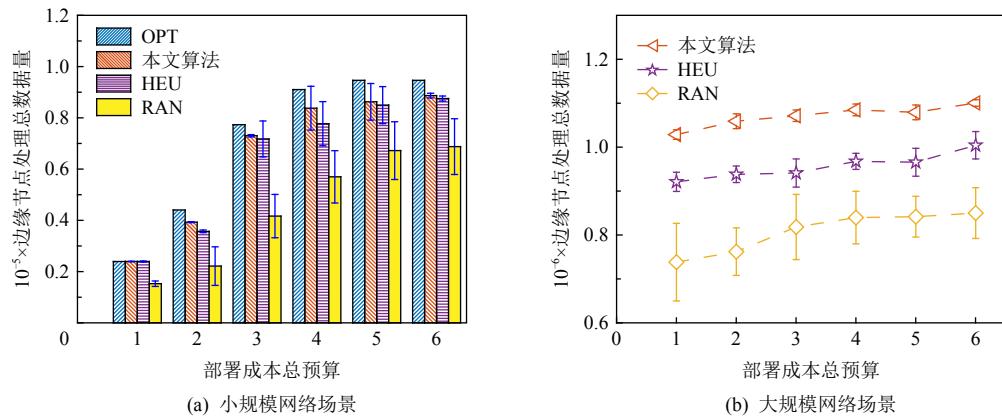


Fig. 5 Impact of different total consumption budgets on the performance of edge node deployment

图 5 不同总耗费预算对边缘节点部署性能的影响

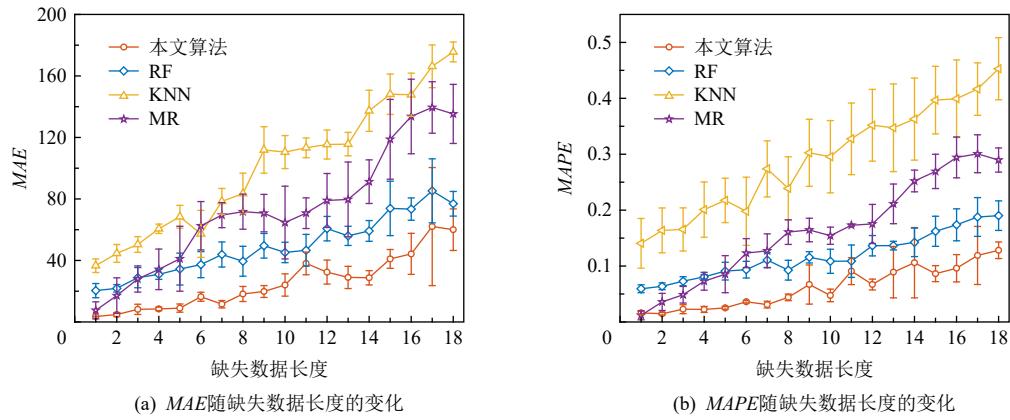


Fig. 6 Impact of different missing data lengths on the accuracy of missing sensing data recovery

图 6 不同缺失数据长度对缺失感知数据恢复精度的影响

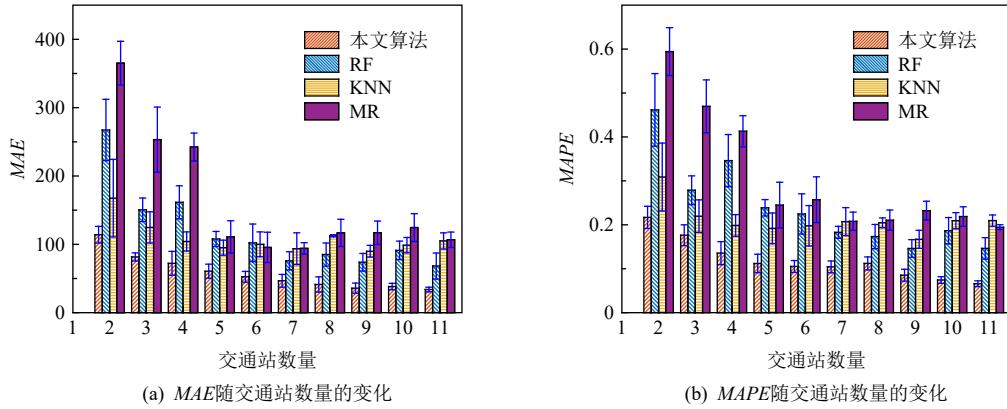


Fig. 7 Impact of different number of traffic stations on the accuracy of missing sensing data recovery

图 7 不同交通站点数量对缺失感知数据恢复精度的影响

数量的增多,缺失数据的恢复精度也不断提高.这是因为随着交通站点数量的增加,矩阵中非缺失元素的比例增加,从而使得缺失数据的恢复精度也不断提高.此外,与 RF 算法、MR 算法和 KNN 算法比较,本文算法的 MAPE 指标分别平均降低 50.2%、60.9%、43.8%,MAE 指标分别平均降低 51.2%、64.5%、47.1%.

同理,由于 KNN 和 MR 算法对矩阵中元素非缺失程度的依赖性,使得其在交通站点数量较少时表现出来的性能不够理想.

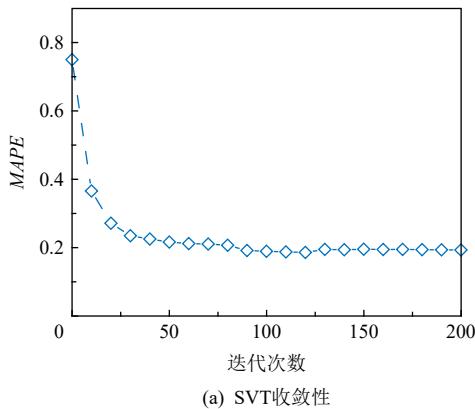
此外,本文还对 SVT 算法的收敛性进行了验证.首先,随机选择 24 个交通站点 24 天的感知数据组成实验数据矩阵;然后,利用 SVT 算法,设定不同迭代

次数对矩阵进行恢复, 得到的结果如图 8(a)所示. 可以发现当迭代次数达到 90 次后, 感知数据的恢复精度基本保持不变.

③评估缺失感知数据自适应恢复的性能. 具体地, 随机选择 16 个交通站点 14 天内的感知数据组成组成实验数据矩阵.

首先, 在实验前, 先探究了该矩阵秩随时间变化的情况, 结果如图 8(b)所示. 可以发现, 矩阵秩大小会随时间而产生波动, 如第 3 天(4 月 25 日, 澳大利亚退伍老兵日)、第 6 天和第 7 天(周末)的感知数据矩阵秩产生了较明显的下降.

然后, 分别采用本文提出的自适应缺失数据恢复算法, 即根据恢复结果, 计算非缺失下限并调整采样率和非自适应恢复算法, 即采样率不随恢复结果自适应变化, 对该矩阵进行数据恢复. 实验结果如图 9 所示, 自适应恢复算法比非自适应恢复算法的  $MAE$  指标和  $MAPE$  指标分别平均降低 40.3% 和 40.4%. 主要原因是感知数据矩阵的秩在随时间波动, 即感知数据的相关性程度随时间发生了变化. 而非



(a) SVT收敛性

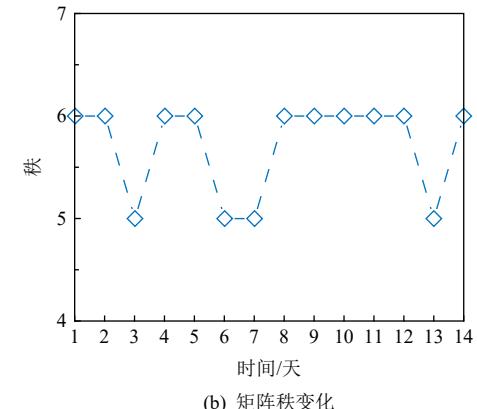
自适应恢复算法并没有根据相关性程度变化及时调整元素采样率, 如当相关性程度减小时, 提高采样率, 增加非缺失元素的比例. 而本文所提的自适应恢复算法充分考虑了这一因素, 因此保证了数据的恢复精度, 提高了系统恢复性能的鲁棒性.

此外, 本文基于澳大利亚 TVVS 系统中 600 个交通站点的感知数据, 构建基于边缘智能计算的大规模交通感知数据恢复原型系统, 如图 10 所示.

## 5 总结和展望

### 5.1 工作总结

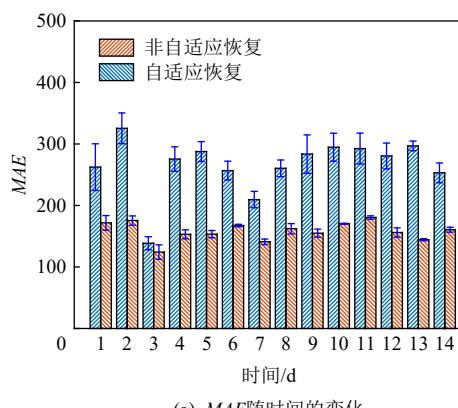
本文提出一种基于边缘智能计算的城市交通感知数据自适应恢复系统, 以解决高计算复杂度的边缘节点部署和高时空动态性的交通感知数据恢复问题. 首先, 提出具有理论下界的边缘节点次优部署算法, 以解决边缘节点部署这个 NP-hard 问题. 然后, 提出基于低秩理论的缺失感知数据自适应恢复算法, 利用低秩理论和奇异值分解, 基于恢复结果估计感



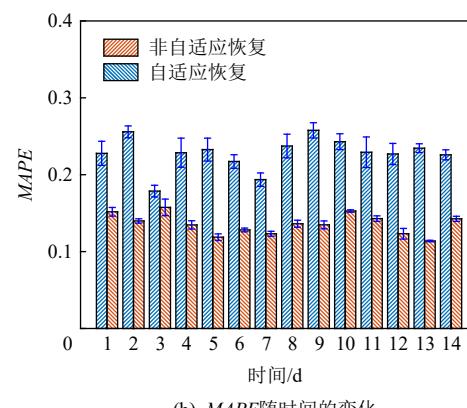
(b) 矩阵秩变化

Fig. 8 Convergence analysis of SVT and the trend of rank for traffic sensing data matrix

图 8 SVT 收敛性分析和交通感知数据矩阵秩的变化趋势



(a) MAE随时间的变化



(b) MAPE随时间的变化

Fig. 9 Evaluation of adaptive recovery performance of missing traffic sensing data

图 9 缺失交通感知数据自适应恢复性能评估



Fig. 10 Large-scale traffic sensing data recovery prototype system

图 10 大规模交通感知数据恢复原型系统

知数据矩阵的非缺失下限,并根据非缺失下限值的反馈,自适应调整交通站点感知数据分配,从而提高数据恢复的精确性和鲁棒性.最后,基于实际大规模交通感知数据集构建原型系统,并进行实验评估,实验结果验证所提算法的有效性和鲁棒性.

## 5.2 研究展望

在将来的工作中,进一步改进和完善本文存在的缺点和不足,主要包括3个方面:1)交通站点的拓扑关系与它们感知数据的相关性有关,因此,如何利用交通站点道路拓扑图来进一步提高缺失感知数据精度是下一步的研究工作.2)本文只是以交通流量这种典型的交通感知数据作为例子研究交通感知数据恢复且只考虑了来自静态交通感知站点的感知数据.下一步将延伸和扩展到更多不同来源种类的感知数据,如来自无人机感知<sup>[44]</sup>和人群感知<sup>[45]</sup>的城市感知数据等.3)本文在恢复交通感知数据时,主要利用感知数据之间的时空相关性,但这些感知数据之间还存在其他多维度的联系,如数据类型等.下一步将结合其他多维相关性进一步提高恢复精度.

致 谢 感谢重庆大学计算机学院张乃凡、程梁华、李耀宇对本文所做的贡献!

**作者贡献声明:**向朝参和程文辉提出了论文创新点,设计了论文实验方案,并撰写了论文;张昭实现了论文算法;焦贤龙、屈毓璘、陈超和戴海鹏指导了论文写作.

## 参 考 文 献

- [1] Meng Chuishi, Yi Xiuwen, Su Lu, et al. City-wide traffic volume inference with loop detector data and taxi trajectories [C/OL] //Proc of the 25th ACM SIGSPATIAL Int Conf on Advances in Geographic Information Systems. New York: ACM, 2017 [2022-04-08].<https://doi.org/10.1145/3139958.3139984>
- [2] Xu Yanyan, Kong Qingjie, Klette R, et al. Accurate and interpretable Bayesian MARS for traffic flow prediction [J]. *IEEE Transactions on Intelligent Transportation Systems*, 2014, 15(6): 2457–2469
- [3] New South Wales Government. System of traffic volume [EB/OL]. (2019-09-27)[2022-04-08].<https://www.rms.nsw.gov.au/about/corporate-publications/statistics/traffic-volumes/aadt-map/index.html>
- [4] Said A B, Erradi A. Spatiotemporal tensor completion for improved urban traffic imputation [J/OL]. *IEEE Transactions on Intelligent Transportation Systems*, 2021 [2022-04-08].<https://ieeexplore.ieee.org/abstract/document/9376705>
- [5] Xiang Chaocan, Zhang Zhao, Qu Yuben, et al. Edge computing-empowered large-scale traffic data recovery leveraging low-rank theory [J]. *IEEE Transactions on Network Science and Engineering*, 2020, 7(4): 2205–2218
- [6] Fan Xiaochen, Xiang Chaocan, Chen Chao, et al. BuildSenSys: Reusing building sensing data for traffic prediction with cross-domain learning [J]. *IEEE Transactions on Mobile Computing*, 2021, 20(6): 2154–2171
- [7] Chen Yuanyuan, Lv Yisheng, Wang Feiyue. Traffic flow imputation using parallel data and generative adversarial networks [J]. *IEEE Transactions on Intelligent Transportation Systems*, 2020, 21(4): 1624–1630
- [8] Cui Yanling, Jin Beihong, Zhang Fusang. Highway traffic condition detection with data fusion [J]. *Chinese Journal of Computers*, 2017, 40(8): 1798–1812 (in Chinese)  
(崔艳玲,金蓓弘,张扶桑. 基于数据融合的高速公路交通状况检测 [J]. *计算机学报*, 2017, 40(8): 1798–1812)
- [9] Chen Xinyu, He Zhaocheng, Wang Jiawei. Spatial-temporal traffic speed patterns discovery and incomplete data recovery via SVD-combined tensor decomposition [J]. *Transportation Research Part C: Emerging Technologies*, 2018, 86: 59–77
- [10] Gong Yongshun, Li Zhibin, Zhang Jian, et al. Missing value imputation for multi-view urban statistical data via spatial correlation learning [J/OL]. *IEEE Transactions on Knowledge and Data Engineering*, 2021 [2022-04-08].<https://ieeexplore.ieee.org/abstract/document/9403954>
- [11] Liang Bing, Ji Wen. Multiuser computation offloading for edge-cloud collaboration using submodular optimization [J]. *Journal on Communications*, 2020, 41(10): 25–36 (in Chinese)  
(梁冰,纪雯. 基于次模优化的边云协同多用户计算任务迁移方法 [J]. *通信学报*, 2020, 41(10): 25–36)
- [12] Yu Bowen, Pu Lingjun, Xie Yuting, et al. Joint task offloading and base station association in mobile edge computing [J]. *Journal of Computer Research and Development*, 2018, 55(3): 537–550 (in Chinese)  
(于博文,蒲凌君,谢玉婷,等. 移动边缘计算任务卸载和基站关联协同决策问题研究 [J]. *计算机研究与发展*, 2018, 55(3): 537–550)
- [13] Zhang Qiuping, Sun Sheng, Liu Min, et al. Online joint optimization mechanism of task offloading and service caching for multi-edge device collaboration [J]. *Journal of Computer Research and Development*, 2021, 58(6): 1318–1339 (in Chinese)

- (张秋平, 孙胜, 刘敏, 等. 面向多边缘设备协作的任务卸载和服务缓存在线联合优化机制[J]. *计算机研究与发展*, 2021, 58(6): 1318–1339)
- [14] Liu Qiang, Tao Han, Xie J L, et al. LiveMap: Real-time dynamic map in automotive edge computing [C/OL] //Proc of the 40th IEEE Int Conf on Computer Communications. Piscataway, NJ: IEEE, 2021 [2022-04-08]. <https://ieeexplore.ieee.org/abstract/document/9488872>
- [15] Zhang Shouli, Liu Chen, Han Yanbo, et al. Dance: A service adaption approach for the dynamic integration of cloud and edge[J]. *Chinese Journal of Computers*, 2020, 43(3): 423–439 (in Chinese)  
(张守利, 刘晨, 韩燕波, 等. Dance: 一种面向云端动态集成的服务适配方法[J]. *计算机学报*, 2020, 43(3): 423–439)
- [16] Ceselli A, Premoli M, Secci S. Mobile edge cloud network design optimization[J]. *IEEE/ACM Transactions on Networking*, 2017, 25(3): 1818–1831
- [17] Ceselli A, Premoli M, Secci S. Cloudlet network design optimization [C/OL] //Proc of the 14th IFIP Networking Conf. Piscataway, NJ: IEEE, 2015 [2022-04-08]. <https://ieeexplore.ieee.org/abstract/document/7145315>
- [18] Mondal S, Das G, Wong E. CCOMPASSION: A hybrid cloudlet placement framework over passive optical access networks [C] //Proc of the 38th IEEE Int Conf on Computer Communications. Piscataway, NJ: IEEE, 2018: 216–224
- [19] Feng Zeng, Ren Yongzheng, Deng Xiaoheng, et al. Cost-effective edge server placement in wireless metropolitan area networks[J/OL]. *Sensors*, 2019, 19(1). [2022-04-08]. <https://www.mdpi.com/1424-8220/19/1/32>
- [20] Mondal S, Das G, Wong E. Cost-optimal cloudlet placement frameworks over fiber-wireless access networks for low-latency applications[J]. *Journal of Network and Computer Applications*, 2019, 138: 27–38
- [21] Nellore K, Hancke G P. A survey on urban traffic management system using wireless sensor networks[J/OL]. *Sensors*, 2016, 16(2). [2022-04-08]. <https://www.mdpi.com/1424-8220/16/2/157>
- [22] Mao Yuyi, You Changsheng, Zhang Jun, et al. A survey on mobile edge computing: The communication perspective[J]. *IEEE Communications Surveys & Tutorials*, 2017, 19(4): 2322–2358
- [23] Yang Li, Lin Hao, Li Zhenhua, et al. A nationwide study on cellular reliability: Measurement, analysis, and enhancements [C] //Proc of the 35th ACM Int Conf on Applications, Technologies, Architectures, and Protocols for Computer Communication. New York: ACM, 2021: 597–609
- [24] Chen Jian, Zhao Minghao, Li Zhenhua, et al. Lock-free collaboration support for cloud storage services with operation inference and transformation [C] //Proc of the 18th Conf on File and Storage Technologies. Berkeley, CA: USENIX Association, 2020: 13–27
- [25] Li Zhenhua, Zhang Yongfeng, Liu Yunhao, et al. A quantitative and comparative study of network-level efficiency for cloud storage services[J]. *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, 2019, 4(1): 1–32
- [26] Mach P, Becvar Z. Mobile edge computing: A survey on architecture and computation offloading[J]. *IEEE Communications Surveys & Tutorials*, 2017, 19(3): 1628–1656
- [27] Abbas N, Zhang Yan, Taherkordi A, et al. Mobile edge computing: A survey[J]. *IEEE Internet of Things Journal*, 2018, 5(1): 450–465
- [28] Lu Dongyu, Qu Yuben, Wu Fan, et al. Robust server placement for edge computing [C] //Proc of the 34th IEEE Int Parallel and Distributed Processing Symp. Piscataway, NJ: IEEE, 2020: 285–294
- [29] Sarker A, Shen Haiying, Stankovic J A. MOPR: Data-driven multi-objective route planning and optimization for electric vehicles[J]. *ACM Interactive Mobile Wearable and Ubiquitous Technologies*, 2018, 1(4): 1–35
- [30] He Suining, Shin K G. Spatio-temporal adaptive pricing for balancing mobility-on-demand networks[J]. *ACM Transactions on Intelligent Systems and Technology*, 2019, 10(4): 1–28
- [31] Martello S. Knapsack Problems: Algorithms and Computer Implementations[M]. Hoboken, NJ: John Wiley & Sons, 1990
- [32] Murty K G. Linear Programming[M]. Berlin: Springer, 1983
- [33] Nemhauser G L, Wolsey L A, Fisher M L. An analysis of approximations for maximizing submodular set functions—I[J]. *Mathematical Programming*, 1978, 14(1): 265–294
- [34] Gupta A, Roth A, Schoenebeck G, et al. Constrained non-monotone submodular maximization: Offline and secretary algorithms [C] //Proc of the 6th Int Workshop on Internet and Network Economics. Berlin: Springer, 2010: 246–257
- [35] Fisher M L, Nemhauser G L, Wolsey L A. An analysis of approximations for maximizing submodular set functions—II[J/OL]. *Polyhedral Combinatorics*, 1978 [2022-04-08]. <https://link.springer.com/chapter/10.1007/BFb0121195>
- [36] Eckart C, Young G. The approximation of one matrix by another of lower rank[J]. *Psychometrika*, 1936, 1(3): 211–218
- [37] Candès E L, Recht B. Exact matrix completion via convex optimization[J]. *Foundations of Computational Mathematics*, 2009, 9(6): 717–772
- [38] Cai Jianfeng, Candès E J, Shen Zuowei. A singular value thresholding algorithm for matrix completion[J]. *SIAM Journal on Optimization*, 2010, 20(4): 1956–1982
- [39] Candès E J, Tao T. The power of convex relaxation: Near-optimal matrix completion[J]. *IEEE Transactions on Information Theory*, 2010, 56(5): 2053–2080
- [40] Li Yuanzhe, Zhou Ao, Ma Xiao, et al. Profit-aware edge server placement[J]. *IEEE Internet of Things Journal*, 2021, 9(1): 55–67
- [41] Chen Zhuo, Feng Gang, Liu Yijing, et al. Virtual network function deployment strategy based on improved genetic simulated annealing algorithm in MEC[J]. *Journal on Communications*, 2020, 41(4): 70–80 (in Chinese)  
(陈卓, 冯钢, 刘怡静, 等. MEC 中基于改进遗传模拟退火算法的虚拟网络功能部署策略[J]. *通信学报*, 2020, 41(4): 70–80)
- [42] Tak S, Woo S, Yeo H. Data-driven imputation method for traffic data in sectional units of road links[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2016, 17(6): 1762–1771
- [43] Li Huiying, Li Meng, Lin Xi, et al. A spatiotemporal approach for traffic data imputation with complicated missing patterns[J/OL]. *Transportation Research Part C: Emerging Technologies*, 2020 [2022-04-08]. <https://www.sciencedirect.com/science/article/abs/pii/S0968090X20306458>

- [44] Xiang Chaocan, Zhou Yanlin, Dai Haipeng, et al. Reusing delivery drones for urban crowdsensing[J/OL]. IEEE Transactions on Mobile Computing, 2021 [2022-04-08]. <https://ieeexplore.ieee.org/abstract/document/9612021>
- [45] Zhang Jinyi, Zhang Xinglin. Multi-task allocation in mobile crowd sensing with mobility prediction[J/OL]. IEEE Transactions on Mobile Computing, 2021 [2022-04-08]. <https://ieeexplore.ieee.org/abstract/document/9451627>



**Xiang Chaocan**, born in 1987. PhD, associate professor. Member of CCF. His main research interests include edge computing, mobile crowdsensing networks, urban computing, and AI.  
向朝参, 1987 年生. 博士, 副教授. CCF 会员.  
主要研究方向为边缘计算、移动群智感知网络、城市计算和人工智能.



**Cheng Wenhui**, born in 1998. Master candidate. His main research interests include urban computing and edge computing.  
程文辉, 1998 年生. 硕士研究生. 主要研究方向为城市计算和边缘计算.



**Zhang Zhao**, born in 1994. Master candidate. His main research interest is edge computing.  
张昭, 1994 年生. 硕士研究生. 主要研究方向为边缘计算.



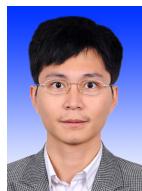
**Jiao Xianlong**, born in 1982. PhD, associate professor. Member of CCF. His main research interests include edge computing and wireless sensor network.

焦贤龙, 1982 年生. 博士, 副研究员. CCF 会员.  
主要研究方向为边缘计算和无线传感器网络.



**Qu Yuben**, born in 1987. PhD, professor. Member of CCF. His main research interests include edge computing and wireless sensor network.

屈毓铸, 1987 年生. 博士, 教授. CCF 会员.  
主要研究方向为边缘计算和无线传感器网络.



**Chen Chao**, born in 1985. PhD, professor. Member of CCF. His main research interests include urban edge computing and AI.

陈超, 1985 年生. 博士, 教授. CCF 会员.  
主要研究方向为城市边缘计算和人工智能.



**Dai Haipeng**, born in 1985. PhD, associate professor. Member of CCF. His main research interests include wireless sensor network, wireless charging, and edge computing.

戴海鹏, 1985 年生. 博士, 副教授. CCF 会员.  
主要研究方向为无线传感器网络、无线充电和边缘计算.