

面向工业生产的算力孪生网络数据处理优化方法

吴双¹, 谢人超^{1,2}, 唐琴琴¹, 谢高畅¹, 黄韬^{1,2}

(1.北京邮电大学网络与交换技术国家重点实验室, 北京 100876; 2.紫金山实验室, 江苏 南京 211111)

摘要: 针对工业场景的算力孪生网络中对高通量、异构工业数据的实时感知、同步和处理分析的需求, 提出一种面向工业生产的算力孪生网络数据处理优化方法, 以提高工业算网孪生的数据同步质量。首先, 考虑工业生产需求和算网孪生服务需求, 提出一种面向工业生产的算网孪生架构。其次, 针对数字孪生对低同步时延、高数据准确性、高效算力资源利用等方面的需求, 提出一个综合考虑上述因素的数字孪生数据处理模型放置问题。为解决该问题, 设计了一种基于和声搜索算法的优化算法, 能够高效求解模型放置问题。仿真实验结果表明, 相较于基准算法, 所提算法可以给出更优的模型放置决策且具备卓越的全局最优解搜索能力。

关键词: 工业物联网; 数字孪生; 算力网络; 数据处理

中图分类号: TP393.0

文献标志码: A

DOI: 10.11959/j.issn.1000-436x.2025025

Optimization method for data processing in digital twin-enabled computing power network for industrial production

WU Shuang¹, XIE Renchao^{1,2}, TANG Qinqin¹, XIE Gaochang¹, HUANG Tao^{1,2}

1. State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

2. Purple Mountain Laboratories, Nanjing 211111, China

Abstract: To meet the demand for real-time sensing, synchronization, and analysis of high-throughput, heterogeneous industrial data in computing power twin networks for industrial scenarios, an optimization method for data processing in industrial computing twin networks was proposed to enhance data synchronization quality. Firstly, an industrial computing twin architecture was proposed, considering industrial production requirements and computing twin services. Secondly, in response to the digital twin's requirements for low synchronization latency, high data accuracy, and efficient computing resource utilization, a data processing model placement problem was formulated by comprehensively incorporating these factors. To solve this problem, an optimization algorithm based on the harmony search algorithm was developed to efficiently address model placement. Simulation results show that, compared with benchmark algorithms, the proposed algorithm provides better model placement decisions and demonstrates superior global optimum search capabilities.

Keywords: industrial Internet of things, digital twin, computing power network, data processing

0 引言

工业物联网 (IIoT, industrial Internet of things) 作为新一代信息技术与制造业深度融合的产物, 通过将工业系统中的各类设备、传感器连接至互联网,

构建了一个全要素互联的综合体系^[1]。该体系为制造业的数字化、网络化、智能化转型升级及高质量发展提供了切实可行的路径, 能够提升工业生产效率和降低生产成本^[2-3]。随着 IIoT 的不断发展, 数字孪

收稿日期: 2024-10-09; 修回日期: 2025-02-06

通信作者: 谢人超, renchao_xie@bupt.edu.cn

基金项目: 国家自然科学基金资助项目(No.92367104)

Foundation Item: The National Natural Science Foundation of China (No.92367104)

生技术作为其数据闭环优化的核心使能技术,通过建立精确的孪生模型结合实时同步的数据,能够对工业设备和生产流程进行更加全面的监测与仿真模拟,减少由故障和决策失误带来的停机时间和成本,进一步提升工业生产的效率与智能化水平^[4-5]。然而,随着 IIoT 规模的不断扩大,数字孪生技术的应用需要消耗越来越多的算力资源,以支持高频同步数据的处理、分析,以及智能模型的训练与推理^[6]。

算力网络为 IIoT 数字孪生所面临的算力资源短缺问题提供了一种可行的解决方案^[7]。通过在网络协议中携带计算负载和网络成本等服务状态信息,算力网络控制器能够动态评估并协同调度各服务器的计算状态,实现对分布式算力资源的高效利用^[8]。将算力网络与工业数字孪生网络结合,不仅能够有效缓解局部算力不足,还能将数字孪生服务模块部署于分布式节点,从而减轻集中式数字孪生任务对本地网络资源需求的压力,并提升系统整体的可扩展性与可靠性^[9]。然而,不恰当的部署策略可能导致数字孪生数据出现误差或产生不可容忍的同步时延,因此有必要对数字孪生服务模块的部署策略展开深入研究。

目前,已有若干工作围绕数字孪生部署这一问题进行了研究。文献[10]提出了一种面向无线数字孪生网络的可持续部署方法,通过限制边缘服务器的过载概率来最小化系统整体成本。文献[11]则综合考虑无线数字孪生网络的动态、用户移动性、计算与通信资源限制等因素,利用深度强化学习对数字孪生部署进行优化,从而实现低时延、高效率的网络关联策略。文献[12]进一步结合网络动态性、资源异构性和用户移动性,利用合作博弈论中的 Shapley 值量化计算节点的协作贡献,提出了一种数字孪生放置与迁移方案,并通过双拍卖机制实现对资源分配和时延性能的优化。文献[13]面向最小化应用请求时延和满足数据时效性目标,提出了一种数字孪生放置策略,能够在保证响应时间的同时兼顾数据时效性,为物联网和工业场景中的数字孪生应用提供了行之有效的部署方案。文献[14]综合考虑时延和能耗等因素,提出了一种基于多臂赌博算法的数字孪生放置决策算法,以最小化无线数字孪生网络的总成本。文献[15]则基于信息时效性提出了一种用户服务满意度增强方法,通过引入新的用户满意度量化指标,分别定义静态和动态数字孪

生部署问题,并给出整数线性规划解决方案,为数字孪生应用的高效部署提供了新的思路。

现有研究主要围绕用户移动性、网络动态性、资源约束与数据时效性等维度展开,却忽略了数字孪生对超高数据同步准确率的特殊需求。实时更新且准确的数据是数字孪生服务的根基,在工业数据实时传输过程中若不考虑通信可能造成的数据误差,将难以保障同步数据的准确性,数字孪生也无法有效提供分析预测、仿真试错和资源调度等关键功能。然而,若仅注重数据同步质量,可能导致在算力网络中过度部署数据处理模型,占用大量计算和存储资源,进一步加剧算力资源的紧缺问题。

针对这一挑战,本文结合工业生产需求和算网孪生服务需求,提出一种面向工业生产的算网孪生架构,以实现 IIoT 数字孪生与算力网络的有机协同。基于该架构,本文根据算力网络下工业互联网数字孪生服务流程,建立了数据处理、模型部署率、数据同步时延和数据准确性的模型,并提出在最小化数据处理、模型部署率和平均数据同步时延的同时,最大化平均数据同步准确率的优化目标,以弥补现有研究在数字孪生对超高数据同步准确率需求方面的不足。针对该优化问题,本文设计了一种基于和声搜索(HS, harmony search)算法的数据处理模型放置优化算法,并引入参数自适应调整机制,以适应动态的工业算网孪生场景。仿真结果表明,所提方案和算法在不同场景、不同网络规模下能够显著提升数据处理效率和同步准确性,同时降低算力资源消耗,表现出较强的鲁棒性与适用性。

本文的主要工作如下。

1) 提出了一种面向工业生产的算网孪生架构,基于该架构建立了面向工业生产的算网孪生数据处理、模型部署率、数据同步时延、数据准确率模型,并提出了在最小化数据处理、模型部署率和平均数据同步时延的同时,最大化平均数据同步准确率的优化目标,以实现对工业场景下算网资源与数字孪生精度的平衡优化。

2) 为高效解决所提问题,提出了一种基于 HS 的数据处理模型放置优化算法,通过参数自适应调整机制实现算法的快速收敛,以适应动态的工业算网孪生场景。

3) 仿真结果表明,该算法能够在降低数据处理模型所需算力资源的同时,提高算网孪生数据同

步的实时性和准确性。此外,在不同方案对比中,所提方案相较于现有方案在优化算网资源使用量和数据同步准确率等方面表现出优越性。

1 系统模型

1.1 面向工业生产的算网孪生架构

在现代工业生产中,全周期的生产一般由多个复杂工业生产环节组成,各生产环节间往往存在耦合关系。以汽车制造业为例,现代汽车工艺流程包括冲压、焊接、涂装、总装等典型环节^[16]。当前,各生产环节通常由各工业现场独立的控制系统进行管理,在工业转型过程中,各控制系统正逐步融合特定的智能能力,以提高生产效率。然而,从整个生产周期的角度来看,这种“分而治之”的生产管控模式存在各生产环节割裂、智能能力难以互通,且缺乏全局统筹分析利用数据等问题。同时,各工业现场计算设备部署不均衡可能导致同时发生算力资源局部紧缺和局部过剩的情况,算网孪生在工业场景中的应用可以有效解决这些问题。具体来说,算力网络通过统一管理泛在异构算力资源,实现有限算力资源的协同编排,提高算力资源利用率,为工业智能化生产建立算力资源基础^[17-18]。数字孪生通过建立精准孪生模型、实时采集更新工业数据和网络状态数据,实现从物理世界到数字世界的准确映射,从而形成工业生产全周期的数字化,实现全链条工业生产情况的分析,并在此基础上实现对生产决策的仿真验证,从而提高全周期工业生产效率,降低工业生产试错成本^[19]。

本文提出的面向工业生产的算网孪生架构如图1所示,该架构由物理层和数字孪生层构成。物理层包括云中心节点、网络节点、边缘节点和工业终端组成。云中心节点、网络节点和边缘节点共同构成工业算力网络,促进分布式计算资源的协同使用。数字孪生层跟踪各物理实体的空间状态、资源状态和行为,建立相对应的数字孪生模型,形成全域、全链条工业生产的数字孪生,并使用智能模型进行状态分析、资源管理和模拟,提供工业现场故障检测、工业生产全流程优化等功能。

工业生产现场数据一般为现场图片、视频和各设备参数等多模态数据,在收集到最新数据后,终端将这些数据上传至最近的基站,由基站接入算力网络。每个基站配备了边缘服务器,被视作一个边

缘节点。边缘节点在算力网络边缘侧实时响应工业生产终端的时延敏感型任务,如故障检测。此外,边缘节点会将终端数据上传至云中心节点,支撑算网孪生系统中数据的实时同步更新。云中心节点基于实时数据响应生产环节协同优化等请求。如果将数据全部上传至云中心节点进行处理,可能导致云中心节点过载,数据处理和整合响应时间长,从而无法保证算网孪生系统中数据的实时同步更新,且原始状态数据中存在较多冗余信息,将其上传至云中心节点将造成带宽资源的浪费。因此,本文提出的算网孪生架构在网络节点上部署数据处理模型,对数据进行预处理后再上传至云中心节点,支撑算网孪生的建立和运维。相较于边缘节点,网络节点距离终端较远而距离云中心节点更近,可以为数据预处理提供更丰富的计算资源和存储资源。数据预处理主要包括数据查重、异常值检测、数据标准化,处理异构数据为数字孪生所需的标准化数据,使后续更新、处理更高效。

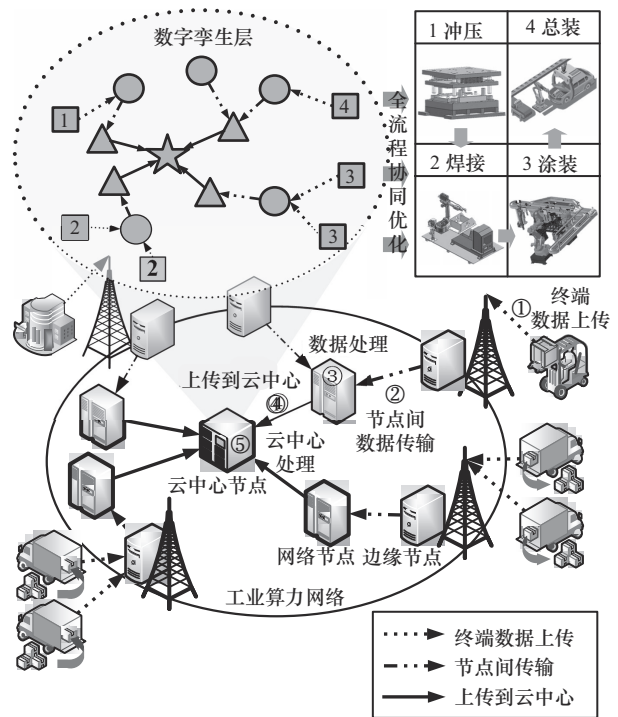


图1 面向工业生产的算网孪生架构

工业算力网络中共有一个云中心节点、 M 个网络节点 $S = \{s_1, s_2, \dots, s_M\}$ 和 I 个边缘节点 $V = \{v_1, v_2, \dots, v_I\}$ 。每个边缘节点 s_i 需要定期更新至数字孪生层的数据量为 d_i 。每个网络节点 s_m 所具备的计

算资源大小为 f_m ，所具备的存储资源大小为 c_m 。部署一个数据处理模型所需的计算资源大小为 f_0 ，所需的存储资源大小为 c_0 。系统中有 K 个工业终端 $U = \{u_1, u_2, \dots, u_K\}$ ，每个工业终端 u_k 需定期更新的数据量为 d_k 。数字孪生层将整个网络架构抽象为一个无向连通图 $\mathcal{G} = (\mathcal{V}, \mathcal{L})$ ， \mathcal{V} 是边缘节点、网络节点和云中心节点的集合， $\mathcal{L} = [L] \subseteq \mathcal{V} \times \mathcal{V}$ 表示网络链路集合， $l \in \mathcal{L}$ 表示节点间有直接网络链路，每条链路有非负权重。权重矩阵 $\mathbf{D} = (w_l) \in \mathbb{R}^{\mathcal{V} \times \mathcal{V}}$ 表示节点间传输的物理距离，若 $l \in \mathcal{L}$ 则 w_l 为有限正实数，否则 $w_l = \infty$ 。矩阵 $\mathbf{B} = (b_m) \in \mathbb{R}^M$ 表示网络节点 s_m 与云中心节点传输的可用带宽。

1.2 数据处理模型

本文提出的工业算网孪生架构中，分布式数据处理指对高通量多模态的工业生产现场状态数据进行数据清洗。数据清洗是对数据进行去重、处理异常值和转换数据格式等操作，提高数据的准确性和可靠性，进一步提高数据分析和孪生建模精度，同时可以节约网络带宽资源^[20]。

数据去重：去除数据集中的重复记录。未来对频繁采集的传感器数据进行高效去重，本文采用一种基于滑动窗口缓存的去重机制。每条数据到达后，首先提取数据的唯一性标识，标识由设备ID、采集时间戳和测量值的哈希值共同构成，以确保能够唯一标记每条数据。通过维护一个滑动时间窗口的缓存，存储最近一段时间内的标识信息，当新数据到达时，会检查标识是否已存在于缓存中。如果标识已存在，则判定为重复数据予以丢弃；如果标识不存在，则将其视为新数据，存入缓存并加入待上传的缓冲区。此外，缓存会定期清理超出时间窗口的过期标识，以确保内存占用在可控范围内。通过该去重机制减少冗余数据，为云中心进行数据分析和仿真验证提供高质量的数据输入，减少对通信资源的浪费。

异常值处理：检测和数据处理集中的异常值。异常值可以被删除或替换为可接受的值。传统的异常检测算法主要有基于统计的异常检测算法、基于距离的异常检测算法和基于密度的局部异常检测算法3类^[21]。在算网孪生中海量异构数据一般以高维形式存在，且要求实时处理，采用传统的异常值检测算法处理这类数据可能出现检测效果差、时间复

杂度高等问题。因此，本文采用基于角度的异常值检测（ABOD，angle-based outlier detection）算法来检测算网孪生中的高维异常数据。ABOD算法原理如图2所示，对每个点 o ，ABOD算法计算任意一对点 $x, y (x \neq o, y \neq o)$ 对其形成的 $\angle xoy$ 的大小。对于在簇中心的点，以点 a 为例，该点与其他点形成的角度变化很大；对于靠近簇边界的点，以点 b 为例，该点与其他点形成的角度变化略小；对于离群点，以点 c 为例，该点与其他点形成的角度变化很小。给定点集 D ，对于点 $o (o \in D)$ ，ABOD算法定义的基于角度的离群因子为

$$\text{ABOD}(o) = \frac{\text{VAR}_{x,y \in D, x \neq o, y \neq o} \langle ox, oy \rangle}{\|ox\|^2 \cdot \|oy\|^2} \quad (1)$$

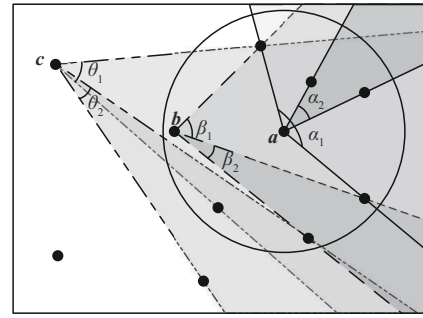


图2 ABOD算法原理

数据标准化：在工业算网孪生系统中，针对异构设备生成的数据，标准化是实现数据一致性和兼容性的关键步骤。通过建立统一的数据模型将不同设备的多样化数据映射为统一格式。标准化过程包括字段对齐和单位转换2个核心环节。字段对齐是将设备原始数据的字段名映射为统一的标准字段，例如将不同设备中的“temp_value”“temperature_reading”等映射为统一的“temperature”，并确保所有数据以标准字段名呈现。单位转换则通过预定义规则，将设备数据的度量单位统一，例如将温度从华氏度转换为摄氏度，从而使数据具备统一的物理含义。标准化后的数据采用通用的数据格式（如JSON或Protobuf）存储，为云中心的后续分析和仿真提供一致性强、易于解析的高质量输入。

1.3 模型部署率模型

本文用向量 $\mathbf{X} = [x_m], x_m \in \{0, 1\}$ 来表示在网络节点 s_m 上部署数据处理模型的决策， $x_m = 1$ 表示在网络节点 s_m 上部署数据处理模型， $x_m = 0$ 表示在节点 s_m 上未部署数据处理模型。根据放置决策 \mathbf{X} ，可

以计算得到数据处理模型部署率为

$$F(\mathbf{X}) = \sum_{m=1}^M \frac{x_m}{M} \quad (2)$$

因为部署数据处理模型所需的计算资源量和存储资源量都与部署个数线性相关,所以本文采用模型部署率来衡量该模型占用的资源量。

1.4 数据同步时延模型

终端数据上传时延:本文考虑终端通过当前蜂窝网络中广泛采用的正交频分多址方案与边缘节点通信,终端 u_k 上传数据至距离最近的边缘节点 v_i 的传输速率为

$$r_{k,i} = \omega_{k,i} \text{lb} \left(1 + \frac{p_k h_{k,i}^2}{n_0 \omega_{k,i}} \right) \quad (3)$$

其中, $\omega_{k,i}$ 是终端 u_k 的上行链路无线信道的带宽, p_k 是终端 u_k 的发射功率, $h_{k,i}$ 是信道系数, n_0 是噪声功率谱密度。终端 u_k 的上行链路传输时延 $T_{k,i}^{\text{wtr}}$ 为

$$T_{k,i}^{\text{wtr}} = \frac{d_k}{r_{k,i}} = \frac{d_k}{\omega_{k,i} \text{lb} \left(1 + \frac{p_k h_{k,i}^2}{n_0 \omega_{k,i}} \right)} \quad (4)$$

节点间数据传输时延:在算网孪生网络图 $\mathcal{G} = (\mathcal{V}, \mathcal{L})$ 中边缘节点 v_i 到网络节点 s_m 的最短传输路径用 $\text{tr}(v_i, s_m) = \langle v_i, e_1, \dots, e_n, s_m \rangle$ 表示,其中, v_i 是起点, s_m 是终点, $e_1, \dots, e_n \in \mathcal{I} \cup \mathcal{S}$ 是 n 个转发节点。序列中相邻2个节点间的链路 $l \in \mathcal{L}$ 。用 $D(i, j)$ 表示节点 i 和节点 j 之间的传输距离,则边缘节点 v_i 到网络节点 s_m 的最短传输距离 $D(v_i, s_m)$ 可由传输路径 $\text{tr}(v_i, s_m)$ 计算得到

$$D(v_i, s_m) = D(v_i, e_1) + \sum_{k=1}^{n-1} D(e_k, e_{k+1}) + D(e_n, s_m) \quad (5)$$

根据数据处理模型放置决策 \mathbf{X} ,边缘节点 v_i 将数据上传至距离最近的网络节点 s_{m^*} 为

$$s_{m^*} = \underset{s_m \in \mathcal{S} | x_m = 1}{\text{argmin}} D(v_i, s_m) \quad (6)$$

边缘节点 v_i 将数据上传至网络节点 s_{m^*} 的节点间传输时延 T_{i,m^*}^{nr} 为

$$T_{i,m^*}^{\text{nr}} = \phi \left(d_i + \sum_{u_k \in \mathcal{U}_i} d_k \right) D(v_i, s_{m^*}) \quad (7)$$

其中, \mathcal{U}_i 是将数据上传至边缘节点 v_i 的终端的集合, $d_i + \sum_{u_k \in \mathcal{U}_i} d_k$ 是边缘节点 v_i 需上传的数据量, ϕ 为每单位距离传输单位数据所需的时间^[11-12]。

数据处理时延:本文将每个数据处理模型提供服务视为一个 $M/M/1$ 队列,对网络节点 s_m ,数据处理请求的到达满足参数为 λ_m 的泊松分布,数据处理模型的平均服务率为 μ_m 。网络节点 s_m 处理由边缘节点 v_i 上传的数据所需的数据处理时延 $T_{i,m}^{\text{proc}}$ 为

$$T_{i,m}^{\text{proc}} = \frac{\lambda_m}{\mu_m(\mu_m - \lambda_m)} + \frac{1}{f_0} \left(d_i + \sum_{u_k \in \mathcal{U}_i} d_k \right) \quad (8)$$

其中, $\frac{\lambda_m}{\mu_m(\mu_m - \lambda_m)}$ 是排队时延,多个边缘节点同时向网络节点 s_m 传输数据,边缘节点 v_i 上传数据需排队等待处理, $\frac{1}{f_0} \left(d_i + \sum_{u_k \in \mathcal{U}_i} d_k \right)$ 是数据处理时延。

网络节点 s_m 处理后需上传至云中心节点的数据量为 $d_m = \lambda \left(d_i + \sum_{u_k \in \mathcal{U}_i} d_k \right)$ 。

网络节点数据上传时延:网络节点 m 上传数据至云中心节点的数据上传时延主要由可用传输带宽决定。根据带宽矩阵 $\mathbf{B} = (b_m) \in \mathbb{R}^M$,网络节点数据上传时延 T_m^{br} 为

$$T_m^{\text{br}} = \frac{d_m}{b_m} \quad (9)$$

对于终端 u_k 的算网孪生数据同步时延 T_k^{dt} 为

$$T_k^{\text{dt}} = T_{k,i}^{\text{wtr}} + T_{i,m^*}^{\text{nr}} + T_{i,m^*}^{\text{proc}} + T_{m^*}^{\text{br}} \quad (10)$$

1.5 数据同步准确率模型

为了衡量算网孪生系统中数据同步的准确性,本文引入数据同步准确率 P^{acc} 用于衡量终端设备采集的数据与上传至网络节点处理的数据之间的误差。信息量误差是衡量算网孪生数据同步准确率的重要指标,信息量误差越小,数据同步准确率越高^[12]。具体计算方式如下。

数据采集阶段:工业终端 u_k 通过传感器等感知设备获取工业生产过程及设备本身最新数据 d_k^{ori} ,在算网孪生系统中 d_k^{ori} 一般为文字、图像或视频等多模态数据,定义数据 d_k^{ori} 的原始比特信息量为 h_k^{ori} 。采集到原始数据后,终端会对原始数据进行压缩,数据压缩编码的失真率为 q_k ,则压缩后的信息量为

$$h_k^{\text{co}} = h_k^{\text{ori}} (1 - q_k^{\text{enc}}) \quad (11)$$

数据传输阶段:无线传输速率 $r_{k,i}$ 如式(3)所示,基于正交调制解调的传输方法,则算网孪生系统中无线传输的误码率为

$$p_{k,i}^{\text{werr}} = \frac{2 \left(1 - \frac{1}{\sqrt{N}} \right)}{\text{lb} \sqrt{N}} Q \left(\sqrt{\frac{3 \text{lb} \sqrt{N}}{N-1} 2E \left(\frac{p_k h_{k,i}^2}{n_0 \omega_{k,i}} \right)} \right) \quad (12)$$

其中, N 为调制阶数, $E(\cdot)$ 表示求期望, $Q(\cdot)$ 为 Q 函数, 表示为

$$Q(x) = \int_x^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t^2} dt \quad (13)$$

边缘节点与网络节点之间的通信可能会出现丢包问题, 从而导致算网孪生数据同步误差。设单跳传输丢包率为 ε , 由 $\text{tr}(v_i, s_m)$ 可得最短路径跳数为 $n+1$, 则节点间传输数据的准确率为

$$p_{i,m^*}^{\text{acc}} = (1 - \varepsilon)^{n+1} \quad (14)$$

数据处理阶段: 网络节点接收到数据后, 会进行清洗, 然后通过深度学习等算法提取数据中有用的信息。数据在采集和传输过程中的准确率越高, 算网孪生系统中的数据同步准确率也就越高。对于终端 u_k , 其数据同步准确率 P_k^{acc} 可表示为

$$P_k^{\text{acc}} = (1 - q_k^{\text{enc}}) (1 - p_{k,i}^{\text{werr}}) (1 - \varepsilon)^{n+1} \quad (15)$$

1.6 优化目标

本文的优化目标是找到最优的算网孪生数据处理模型的放置决策 \mathbf{X} , 在最小化数据处理、模型部署率和平均数据同步时延的同时, 最大化平均数据同步准确率, 其数学表达为

$$\begin{aligned} \min & \left(\omega_1 F(\mathbf{X}) + \omega_2 \sum_{k=1}^K \frac{T_k^{\text{dt}}(\mathbf{X})}{K} + \omega_3 \left(1 - \sum_{k=1}^K \frac{P_k^{\text{acc}}(\mathbf{X})}{K} \right) \right) \\ \text{s.t. } & C_1: \max \{ T_k^{\text{dt}} \} \leq T_{\max}^{\text{dt}} \\ & C_2: \min \{ P_k^{\text{acc}} \} \geq P_{\min}^{\text{acc}} \\ & C_3: x_m f_0 \leq f_m \\ & C_4: x_m c_0 \leq c_m \end{aligned} \quad (16)$$

其中, ω_1 、 ω_2 、 ω_3 为权重, T_{\max}^{dt} 和 P_{\min}^{acc} 分别为算网孪生系统可容忍的最大数据同步时延和最低数据同步准确率, f_0 和 c_0 分别为部署一个数据处理模型所需的计算资源和存储资源, f_m 和 c_m 分别为网络节点 s_m 所具备的计算资源和存储资源。

2 基于和声搜索的数据处理优化策略

本文所提数据处理模型放置优化问题的解空间大小为 2^M , 当网络规模较大时, 若采用贪心法或者枚举法等精确算法, 运行时间代价较高, 很难在面向工业生成的算网孪生中进行实际应用。因此, 本文采用相应的近似算法对其进行求解。HS、遗

传算法 (GA, genetic algorithm) 等启发式算法已在求解放置类型问题中得到了广泛有效的应用^[22-24]。与 GA 等其他启发式优化算法相比, HS 算法在优化过程中具有数学运算量更小、所需可调参数更少的优点^[25]。此外, HS 算法中解向量的各个分量可以独立调整, 使得搜索方式能够更好地针对具体问题进行设计。因此, 本文设计了一种基于 HS 的数据处理模型放置优化算法。

2.1 和声搜索

HS 算法是 Geem 等^[26]受乐队即兴演奏过程启发提出的一种启发式算法, 模拟音乐演奏的原理以寻求由目标函数评价确定的最佳状态。一个由 n 人组成的乐队, 乐队中每个人演奏一种乐器, 其和声可以表示为 $X = \{x_1, x_2, \dots, x_n\}$, 乐队在排练过程中会不断调整配合以实现最好的和声效果, 和声效果的好坏用函数 $f(X)$ 来表示。 $f(X)$ 相当于乐队中的乐师, 不断调整每个乐器的旋律直至达到完美和声。HS 算法中的主要参数如下。

和声记忆库大小 (HMS, harmony memory size): 和声记忆库 (HM, harmony memory) 中存储的候选解数量, 决定算法搜索范围的宽度。

和声记忆库取值概率 (HMCR, harmony memory considering rate): 从 HM 中选取已有解进行调整的概率, 表示利用历史信息的程度。

音调微调概率 (PAR, pitch adjusting rate): 对已选解各维度进行微调的概率, 增加解的多样性。

微调带宽 (BW, band width): 定义微调时的调整幅度, 影响解在搜索空间中的变化范围。

创作次数: 即算法在整个优化过程中需要重复的次数, 用于控制算法运行的时间和精度。

2.2 基于 HS 的数据处理模型放置优化算法

本文基于 HS 的算网孪生数据处理优化算法, 用于解决工业算网数字孪生数据处理优化问题。为了提高算法的适应性和灵活性, 本文设计的优化机制为: 1) 传统 HS 算法设计用于解决连续优化问题, 而数字孪生算力网络中的算网孪生数据处理模型放置策略是二值离散问题, 本文修改 HS 算法为适配二值离散化的方法; 2) 传统 HS 的初始 HM 是随机产生的, 随机生成的 HM 可能出现在解空间上分布不均匀的状况, 难以满足约束条件, 针对本文的优化问题, 改进后的算法初始解向量采用区间分割法生成, 从而保证随机产生的各个解向量分布均

匀; 3)传统HS算法存在易陷于局部最优解的问题, 本文针对这一问题, 设计了自适应参数调整机制, 参数HMCR和PAR根据迭代次数和结果自适应调整。基于这3点改进思路, 本文提出的基于HS的数据处理模型放置优化算法步骤如下。

步骤1 初始化HM。根据问题随机生成HM矩阵, 并设定算法参数 $HMSS_{H^*}$ 、 $HMCRp_H$ 、 $PARp_{PAR}$ 、微调带宽 B_H 和创作次数 T_{max} 。HM矩阵为

$$HM = \begin{bmatrix} X_1 & X_2 & \cdots & X_{S_H} \end{bmatrix}^T = \begin{bmatrix} x_{11} & \cdots & x_{1M} & x_{1(M+1)} & \cdots & x_{1(IM)} \\ x_{21} & \cdots & x_{2M} & x_{2(M+1)} & \cdots & x_{2(IM)} \\ \vdots & & \vdots & \vdots & & \vdots \\ x_{S_H1} & \cdots & x_{S_HM} & x_{S_H(M+1)} & \cdots & x_{S_H(IM)} \end{bmatrix} \quad (17)$$

每个和声, 即对应一个放置决策 X 。本文将每个和声分割为 I 个长度为 M 的区间, 每个区间内的元素取值按顺序生成

$$x_{iM+m} = \left[\text{rand} + 0.5 \left(1 - \frac{1}{M} \sum_{n=1}^{m-1} X_{iM+n} \right) \right], \quad \forall i \in \mathcal{I}, m \in \mathcal{M} \quad (18)$$

其中, rand 是随机生成的, 且 $\text{rand} \in [0, 1)$ 。该区间分割法是针对算网孪生数据处理模型放置问题所设计的一种初始HM生成法, 使其能生成分布更为均匀的解, 更贴合工业算网中需要均匀部署数据处理模型的要求。

步骤2 生成新和声。新和声向量由随机选择HM中向量微调、解空间随机生成产生。随机生成 γ_1 ($\gamma_1 \in [0, 1]$), 若 $\gamma_1 < p_H$, 则从HM抽取一个和声, 否则从解空间生成一个新的和声。本文自适应调整参数 p_H 取值, 随着迭代次数 t 增加, $p_H(t)$ 取值为

$$p_H(t) = \begin{cases} p_H^{\min}, & t = 1 \\ p_H(t-1) + \frac{p_H^{\max} - p_H^{\min}}{T_{\max}}, & 1 < t \leq T_{\max} \text{ 且 } n < 50 \\ p_H^{\min}, & 1 < t \leq T_{\max} \text{ 且 } n \geq 50 \end{cases} \quad (19)$$

其中, n 指目标函数值在连续的 n 次演奏创作保持不变。当目标函数 $f(X)$ 值连续保持不变, 即算法趋于收敛, 为了避免算法陷入局部最优解, 调整 $p_H(t)$ 值为可取最小值 p_H^{\min} , 以在更大范围内探索解空间中新生成的和声。

步骤3 微调新解。以 p_{PAR} 的概率对选出的解进

行微调, 调整幅度由 B_H 控制。 p_{PAR} 的调整主要与当前HM中的各和声目标函数值有关, 当目前和声库中目标函数值趋于稳定, 希望进行全局搜索避免陷入局部最优解, 随着迭代次数 t 增加, $p_{PAR}(t)$ 取值为

$$p_{PAR}(t) = p_{PAR}^{\max} - \frac{f(X_{t-1})_{\max} - f(X_{t-1})_{\min}}{f(X_{t-1})_{\max}} (p_{PAR}^{\max} - p_{PAR}^{\min}) \quad (20)$$

步骤4 更新HM。评估新解的质量, 若新解优于和声库中最差的解, 则替换该解, 保持和声库中的解始终优质。

步骤5 重复上述过程, 完成创作次数 T_{max} 。

本文提出的基于HS的数据处理模型放置优化算法的伪代码如算法1所示。

算法1 基于HS的数据处理模型放置优化算法

输入 I 、 S_H 、 p_H^{\min} 、 p_H^{\max} 、 p_{PAR}^{\min} 、 p_{PAR}^{\max} 、 B_H 和 T_{max}

输出 放置决策 X

- 1) 初始化HM、 γ_1 、 γ_2 、 $p_H(0)$ 和 $p_{PAR}(0)$
- 2) while $t < T_{max}$ do
- 3) $t = t + 1$
- 4) 根据式(19)更新 $p_H(t)$, 根据式(20)更新 $p_{PAR}(t)$, 生成 γ_1 、 γ_2
- 5) for $k = 1:1:S_H$
- 6) for $i = 1:1:I$
- 7) if $\gamma_1 < p_H(t)$ then
- 8) $X_{new}^k = X^k$
- 9) if $\gamma_2 < p_{PAR}(t)$ then
- 10) $X_{new}^k = (X^k + \lfloor B_H \times \text{rand} \rfloor) \% 2$
- 11) end if
- 12) else
- 13) 随机生成 X_{new}^k
- 14) end if
- 15) if $f(X_{new}^k) > f_{\min}(X_{HM})$
- 16) $X_{worst} \leftarrow X_{new}^k$
- 17) end if
- 18) end for
- 19) end for
- 20) end while

3 仿真分析

为验证本文提出的基于HS的数据处理模型放置优化算法的性能, 采用编程语言Python 3.12.0进行仿真实验。仿真参数如表1所示^[27]。

表1 仿真参数

仿真参数	参数值
数据处理模型所需计算资源/TFLOPS	0.5
数据处理模型所需内存/MB	256
终端发射功率/dBm	23
上行链路带宽/MHz	20
$n_0/(dBm \cdot Hz^{-1})$	-174
同步数据任务大小/MB	(0.1, 10)
HMS/个	10
P_H^{\max}	0.98
P_H^{\min}	0.7
P_{PAR}^{\max}	0.98
P_{PAR}^{\min}	0.7
T_{\max}	500

首先, 为了验证本文算法的可行性, 设置了4组的网络节点数 M 、边缘节点数 I 和工业终端数 K , 分别为: 1) $M = 20, I = 40, K = 120$; 2) $M = 20, I = 80, K = 240$; 3) $M = 40, I = 80, K = 240$; 4) $M = 40, I = 160, K = 480$, 在本组实验中, 工业终端均匀分布在每个边缘节点下。不同网络规模下的目标函数值随迭代次数变化如图3所示。在不同网络规模下, 最优目标函数值和平均目标函数值随迭代次数的变化如图3(a)和图3(b)所示。由图3可以观察到, 本文算法均可实现算法收敛, 证明了所提算法的可行性与可扩展性。HM中的平均目标函数值随着迭代次数增加不断减小, 证明了本文算法中HM保留优质解的有效性。在网络规模较大时, 可以观察到本文算法具有较好的逃出局部最优

解的效果, 证明了本文设计的参数自适应机制的有效性。

其次, 本文提出的数据处理模型放置优化问题是模型部署率、平均数据同步时延和平均数据同步准确率协同优化的问题, 本文设置了4组权重取值, 如表2所示。在不同网络规模下, 权重取值对模型部署率、数据同步时延和数据同步准确率的影响如图4所示。从图4可以看到, 当权重取值为权重2时, 在不同网络规模下模型部署率都远小于其他3组权重取值, 对平均数据同步时延和平均数据同步准确率优化效果较差, ω_1 取值过高; 当权重取值为权重3时, 在不同网络规模下平均数据同步时延远小于其他3组权重取值, 对模型部署率和平均数据同步准确率优化效果较差, ω_2 取值过高; 当权重取值为权重4时, 在不同网络规模下平均数据同步准确率都高于其他3组权重取值, 对模型部署率和平均数据同步时延优化效果较差, ω_3 取值过高; 而当权重取值为权重1时, 对模型部署率、平均数据同步时延和平均数据同步准确率的优化综合表现较好, 因此本文后续仿真时延的权重取值都为权重1, 保证3项指标的协同优化。此外, 该组实验还可以证明本文优化方法可适应不同的网络环境和业务需求, 仅需调整权重取值即可满足不同的业务需求。

表2 权重取值

权重取值	ω_1	ω_2	ω_3
权重1	30	0.5	3 000
权重2	120	0.5	3 000
权重3	30	2	3 000
权重4	30	0.5	12 000

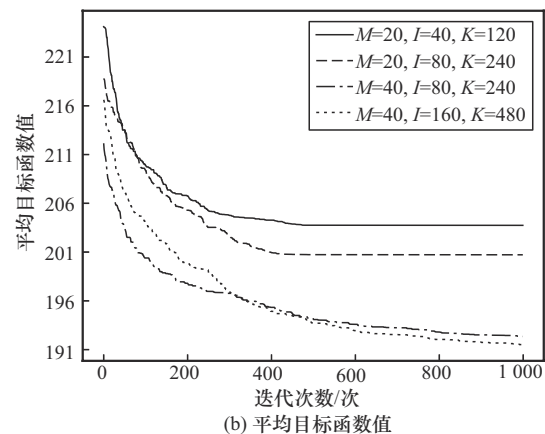
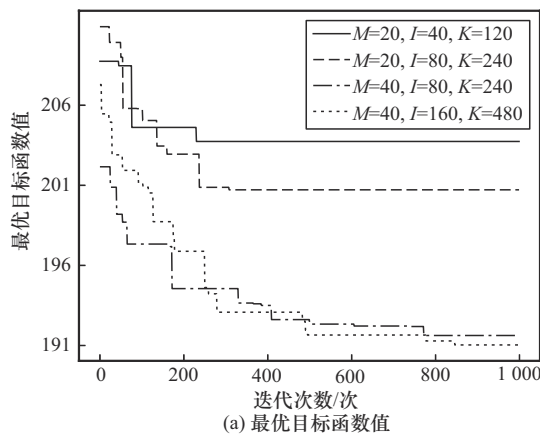


图3 不同网络规模下的目标函数值随迭代次数变化

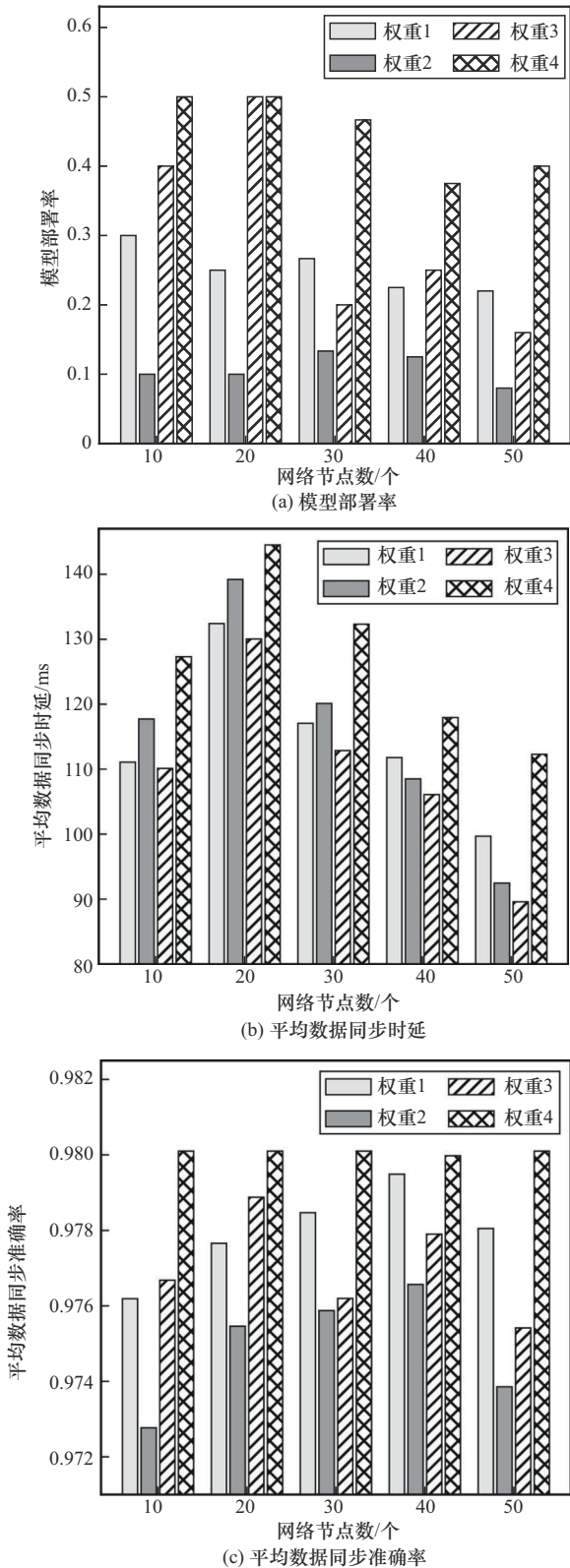


图4 不同权重取值对模型部署率、数据同步时延和数据同步准确率的影响

为了验证本文放置方案的性能,对比了以下2种放置方案。

1) 对比方案1: Lu等^[11]提出的一种聚焦优化数

据同步时延的放置方案,该方案中主要考虑降低数字孪生服务时延。

2) 对比方案2: 集中式处理方案,即目前面向工业生产的数字孪生服务架构中常见的数据由终端采集上传至云中心统一进行处理与分析的方案。

不同放置方案下对模型部署率、数据同步时延和数据同步准确率随网络规模变化如图5所示。由图5(a)可以观察到,本文方案在降低平均数据同步时延方面略逊于对比方案1,但其优化了模型部署率,模型部署率降低了0.5以上,同时有效提高了平均数据同步准确率,从而降低了因数据错误导致数字孪生服务不可靠的风险。对比方案2将数据集中收集至云中心统一处理,因此未考虑模型部署率。本文假设云中心的数据处理能力是网络节点上所部署的数据处理模型的10倍。由图5(b)可以观察到,随着网络规模的增大和数据处理需求的提升,对比方案2的平均数据同步时延远高于本文方案 and 对比方案1,主要原因在于集中式处理导致了较高的排队时延,并且此排队时延会随网络规模和数据处理需求的增加而进一步加剧。由图5(c)可以观察到,对比方案2的平均数据同步准确率同样低于本文方案及对比方案1。其主要原因是云中心距离工业终端较远,传输路径更长,发生数据错误的概率相应增加,而本文方案则可在靠近用户端的位置进行数据去重和异常值检测等预处理操作,从而降低数字孪生接收到错误数据的可能性。

为了验证本文算法的有效性,将本文算法与Random及2种经典的启发式算法GA和粒子群优化(PSO, particle swarm optimization)进行比较。不同网络规模下的目标函数值随迭代次数变化如图6所示。本组实验中,工业终端均匀分布在边缘节点下,网络节点数、边缘节点数和各边缘节点对应的终端数的比例保持1:2:6不变。由图6可知,当网络节点数为10和15,本文算法与GA、PSO得到的最优目标函数值相同,说明在小规模网络中,各种启发式算法均能迅速收敛到最优解。随着网络规模不断扩大,本文算法始终优于对比算法,表明其在全局搜索与避免陷入局部最优方面更具优势,能够得到更优的部署决策。此外,随着网络节点规模的增大,3种算法的最优目标函数值均呈现下降趋势,其原因在于较大的网络规模提供了更多可行部署方案,从而有更大概率为

各工业终端找到合适的数据处理模型部署方式，取得更好的优化效果。

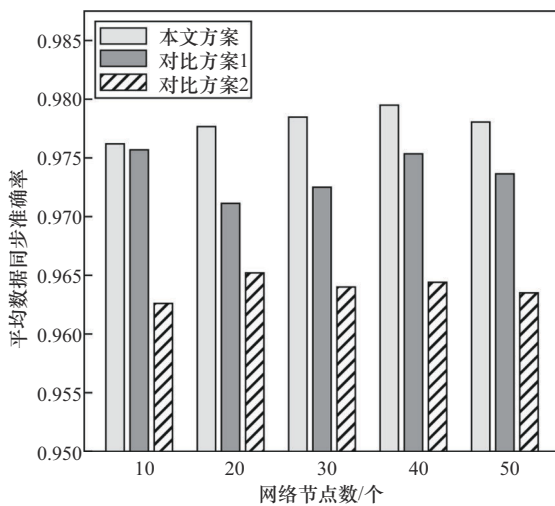
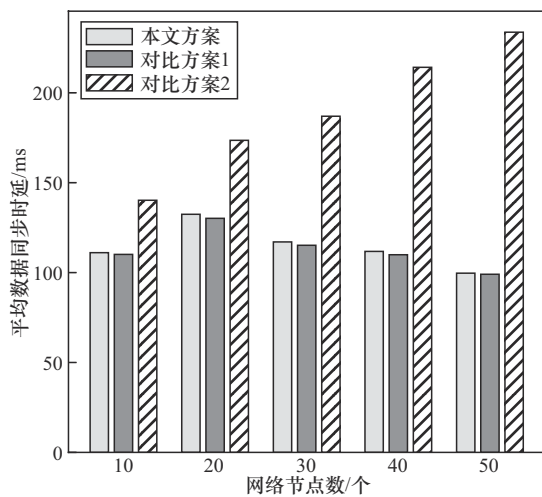
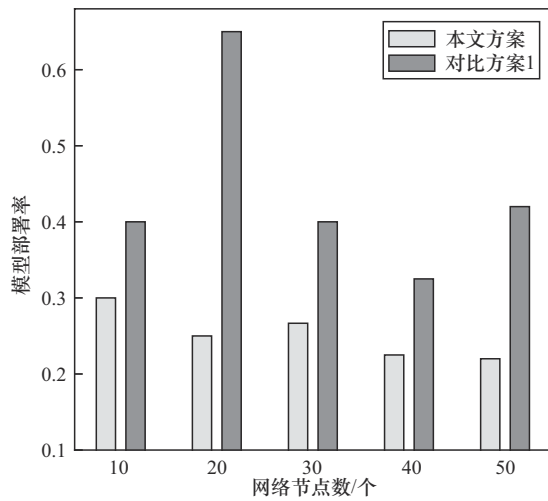


图5 不同放置方案下对模型部署率、数据同步时延和数据同步准确率随网络规模变化

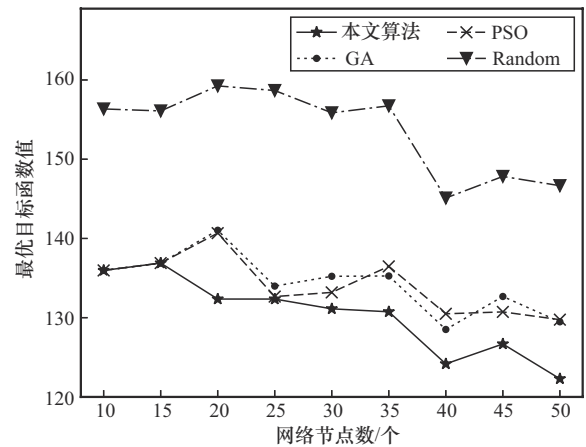


图6 不同网络规模下的目标函数值随迭代次数变化

终端随机分布下的目标函数值随终端数变化如图7所示。本组实验中，网络节点数设为30，边缘节点数为60，工业终端在网络中随机分布。由图7可知，在终端分布不均的网络环境中，本文算法获得了最优的优化效果，证明本文算法也可以适应非均匀分布的网络状态。随着网络终端数量的增加，4种算法的最优目标函数值均随之上升，原因在于终端数量越多，需要同步与处理的数据量也相应增长，导致数据传输时延、排队时延及处理时延的叠加效应更为明显，从而使整体的目标函数值呈现上升趋势。

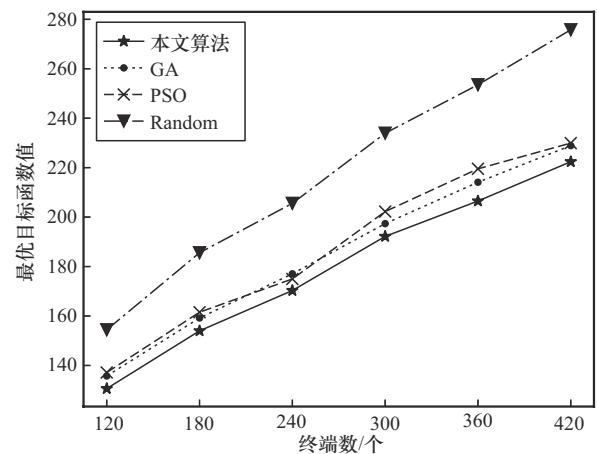


图7 终端随机分布下的目标函数值随终端数变化

4 结束语

本文面向IIoT场景，研究了算网孪生数据实时同步过程中的数据处理优化问题。首先，设计了一种云边协同的算网孪生数据处理架构模型，为工业生产环境下的数据实时同步与高效处理提供了可行

的系统框架。随后,以同时最小化数据处理、模型部署率、平均数据同步时延及数据同步错误率,并最大化数据同步准确率为目标,建立了工业算网孪生系统中的数据处理优化模型。针对该优化问题,设计了一种基于HS算法的数据处理模型放置优化方案。最后,通过仿真实验证明,本文方案和算法不仅能够有效降低平均数据同步时延和模型部署率,还能显著提高数据同步准确率,从而验证了本文算法在数据处理优化中的有效性与优越性。在未来的工作中,将聚焦IIoT多业务场景,进一步研究多数字孪生模型间的协同处理,从而提升数字孪生服务整体的智能化与自适应能力。

参考文献:

- [1] SISINNI E, SAIFULLAH A, HAN S, et al. Industrial Internet of Things: challenges, opportunities, and directions[J]. *IEEE Transactions on Industrial Informatics*, 2018, 14(11): 4724-4734.
- [2] NASRALLAH A, THYAGATURU A S, ALHARBI Z, et al. Ultra-low latency (ULL) networks: the IEEE TSN and IETF DetNet standards and related 5G ULL research[J]. *IEEE Communications Surveys & Tutorials*, 2019, 21(1): 88-145.
- [3] KHALIL R A, SAEED N, MASOOD M, et al. Deep learning in the industrial Internet of things: potentials, challenges, and emerging applications[J]. *IEEE Internet of Things Journal*, 2021, 8(14): 11016-11040.
- [4] HU Y, CAO N, LU H, et al. Multi-dimensional resource management with deep deterministic policy gradient for digital twin-enabled industrial Internet of things in 6 generation[J]. *Transactions on Emerging Telecommunications Technologies*, 2024, 35(4): e4962.
- [5] TAO F, ZHANG H, LIU A, et al. Digital twin in industry: state-of-the-art[J]. *IEEE Transactions on Industrial Informatics*, 2019, 15(4): 2405-2415.
- [6] SONG J, KANG Y, SONG Q Y, et al. Distributed resource optimization with blockchain security for immersive digital twin in IIoT[J]. *IEEE Transactions on Industrial Informatics*, 2023, 19(5): 7258-7267.
- [7] HU Y J, JIA Q M, YAO Y, et al. Industrial Internet of Things intelligence empowering smart manufacturing: a literature review[J]. *IEEE Internet of Things Journal*, 2024, 11(11): 19143-19167.
- [8] 贾庆民, 胡玉姣, 张华宇, 等. 确定性算力网络研究[J]. *通信学报*, 2022, 43(10): 55-64.
JIA Q M, HU Y J, ZHANG H Y, et al. Research on deterministic computing power network[J]. *Journal on Communications*, 2022, 43(10): 55-64.
- [9] COSTANTINI A, DI MODICA G, AHOUANGONOU J C, et al. IIoT-wins: toward implementation of distributed digital twins in industry 4.0 settings[J]. *Computers*, 2022, 11(5): 67.
- [10] ZHOU Y Z, FU Y R, SHI Z, et al. Sustainable placement with cost minimization in wireless digital twin networks[J]. *IEEE Transactions on Vehicular Technology*, 2025, 74(1): 1064-1077.
- [11] LU Y L, MAHARJAN S, ZHANG Y. Adaptive edge association for wireless digital twin networks in 6G[J]. *IEEE Internet of Things Journal*, 2021, 8(22): 16219-16230.
- [12] ZHANG Y D, ZHANG H B, LU Y L, et al. Adaptive digital twin placement and transfer in wireless computing power network[J]. *IEEE Internet of Things Journal*, 2024, 11(6): 10924-10936.
- [13] VAEZI M, NOROOZI K, TODD T D, et al. Digital twin placement for minimum application request delay with data age targets[J]. *IEEE Internet of Things Journal*, 2023, 10(13): 11547-11557.
- [14] GU J L, FU Y R, HUNG K. On intelligent placement decision-making algorithms for wireless digital twin networks via bandit learning[J]. *IEEE Transactions on Vehicular Technology*, 2024, 73(6): 8889-8902.
- [15] LI J, GUO S, LIANG W F, et al. AoI-aware user service satisfaction enhancement in digital twin-empowered edge computing[J]. *IEEE/ACM Transactions on Networking*, 2024, 32(2): 1677-1690.
- [16] 张文博, 李森, 杨哲建, 等. 基于工业算网的汽车制造业数字化转型研究与应用[J]. *新型工业化*, 2023, 13(6): 58-67.
ZHANG W B, LI S, YANG Z J, et al. Research and application of digital transformation of automobile manufacturing industry based on industrial computing network[J]. *The Journal of New Industrialization*, 2023, 13(6): 58-67.
- [17] TANG X Y, CAO C, WANG Y X, et al. Computing power network: The architecture of convergence of computing and networking towards 6G requirement[J]. *China Communications*, 2021, 18(2): 175-185.
- [18] XIE G C, XIE R C, TANG Q Q, et al. Joint task scheduling and intelligence optimization in CPN-enabled connected intelligence systems [C]// *Proceedings of the GLOBECOM 2023 - 2023 IEEE Global Communications Conference*. Piscataway: IEEE Press, 2023: 1101-1106.
- [19] GEHRMANN C, GUNNARSSON M. A digital twin based industrial automation and control system security architecture[J]. *IEEE Transactions on Industrial Informatics*, 2020, 16(1): 669-680.
- [20] MA L C, PEI Q Q, ZHOU L, et al. Federated data cleaning: collaborative and privacy-preserving data cleaning for edge intelligence[J]. *IEEE Internet of Things Journal*, 2021, 8(8): 6757-6770.
- [21] WANG T, KE H X, ZHENG X, et al. Big data cleaning based on mobile edge computing in industrial sensor-cloud[J]. *IEEE Transactions on Industrial Informatics*, 2020, 16(2): 1321-1329.
- [22] DING Z J, WANG S, JIANG C J. Kubernetes-oriented microservice placement with dynamic resource allocation[J]. *IEEE Transactions on Cloud Computing*, 2023, 11(2): 1777-1793.
- [23] HE Z L, LI K L, LI K Q. Cost-efficient server configuration and placement for mobile edge computing[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2022, 33(9): 2198-2212.
- [24] ZHU X K, YAO W B, HOU Y Y, et al. RDRM: real-time dynamic replica management with joint optimization for edge computing[J]. *IEEE Transactions on Services Computing*, 2024, 17(6): 4183-4196.
- [25] AL-OMOUSH A A, ALSEWARI A A, ALAMRI H S, et al. Comprehensive review of the development of the harmony search algorithm and its applications[J]. *IEEE Access*, 2019, 7: 14233-14245.

[26] DIAO R, SHEN Q. Feature selection with harmony search[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2012, 42(6): 1509-1523.

[27] XIE G C, XIONG Z H, ZHANG X Y, et al. GAI-IoV: bridging generative AI and vehicular networks for ubiquitous edge intelligence[J]. IEEE Transactions on Wireless Communications, 2024, 23(10): 12799-12814.



唐琴琴 (1994-), 女, 广西桂林人, 博士, 北京邮电大学副研究员, 主要研究方向为算力网络、网络人工智能、网络孪生等。

[作者简介]



吴双 (2000-), 女, 江苏盐城人, 北京邮电大学博士生, 主要研究方向为算力网络、数字孪生。



谢高畅 (1997-), 男, 山东泰安人, 北京邮电大学博士生, 主要研究方向为算力网络、边缘智能。



谢人超 (1984-), 男, 福建南平人, 博士, 北京邮电大学教授、博士生导师, 主要研究方向为信息中心网络、工业互联网、算力网络、边缘计算、数字孪生。



黄韬 (1980-), 男, 重庆人, 博士, 北京邮电大学教授、博士生导师, 主要研究方向为路由与交换、软件定义网络、内容分发网络、确定性网络、算力网络等。