控制与决策
Control and Decision

基于多智能体深度强化学习的轨道车辆组装分布式异构柔性作业调度

孟祥恒, 郭鹏, 李嘉雯, 史海超, 张志瑶, 马永敬, 孙轶杰

引用本文:
孟祥恒, 郭鹏, 李嘉雯, 等. 基于多智能体深度强化学习的轨道车辆组装分布式异构柔性作业调度[J]. 控制与决策, 2026, 41(5): 1219-1228.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2025.0857

基于多智能体深度强化学习的轨道车辆组装分布式异构柔性作业调度

孟祥恒[1]，郭 鹏[1,2†]，李嘉雯[1]，史海超[1]，张志瑶[1]，马永敬[3]，孙轶杰[3]

(1. 西南交通大学 机械工程学院，成都 610031；
2. 轨道交通运维技术与装备四川省重点实验室，成都 610031；
3. 中车青岛四方机车车辆股份有限公司，山东 青岛 266111)

摘 要: 针对轨道车辆组装作业中多车型混线生产、工序复杂、工艺路线差异显著及制造资源高度异构带来的分布式异构柔性作业车间调度挑战, 提出一种两阶段多智能体深度强化学习方法. 将调度流程建模为多阶段马尔可夫决策过程, 决策涵盖工件分配、工序排序和机器选择, 通过奖励设计引导智能体最小化全局最大完工时间. 上层智能体基于分层异构图注意力网络提取产线全局状态, 实现工件在不同组装线或工区间的合理分配与负载均衡; 下层智能体采用双智能体协作策略, 利用基于图神经网络的编码器-解码器结构捕捉工序间前后约束及资源占用等依赖关系, 实现局部优化. 基于实际作业场景数据, 通过计算验证该方法在缩短制造周期方面的有效性, 展现出良好的泛化能力.

关键词: 深度强化学习；异构资源；多智能体；分布式调度；柔性作业车间；图神经网络

中图分类号: TP18 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2025.0857

引用格式: 孟祥恒, 郭鹏, 李嘉雯, 等. 基于多智能体深度强化学习的轨道车辆组装分布式异构柔性作业调度 [J]. 控制与决策, 2026, 41(5): 1219-1228.

Distributed heterogeneous flexible job shop scheduling for railway vehicle assembly using multi-agent deep reinforcement learning

MENG Xiang-heng[1]， *GUO Peng*[1,2†]， *LI Jia-wen*[1]， *SHI Hai-chao*[1]， *ZHANG Zhi-yao*[1]， *MA Yong-jing*[3]， *SUN Yi-jie*[3]

(1. School of Mechanical Engineering， Southwest Jiaotong University， Chengdu 610031, China；2. Technology and Equipment of Rail Transit Operation and Maintenance Key Laboratory of Sichuan Province， Chengdu 610031， China；3. CRRC Qingdao Sifang CO.， LTD.， Qingdao 266111， China)

Abstract: A two-stage multi-agent deep reinforcement learning method is proposed to address the scheduling challenge of the distributed heterogeneous flexible job shop, posed by multi-model mixed-flow production, complex processes, significant process route variations, and highly heterogeneous manufacturing resources in rail vehicle assembly operations. The scheduling process is modeled as a multi-stage Markov decision process, where decisions encompass job allocation, operation sequencing, and machine selection, and agents are guided by reward design to minimize the global makespan. The upper-level agent, based on a hierarchical heterogeneous graph attention network, extracts the global state of the production line to achieve reasonable job allocation and load balancing across different assembly lines or work zones. The lower-level agent utilizes a dual-agent collaboration strategy and an encoder-decoder structure based on a graph neural network to capture dependencies such as precedence constraints between operations and resource occupancy, enabling local optimization. Based on data from actual operational scenarios, the effectiveness of the proposed method in shortening the manufacturing cycle is validated computationally, and it exhibits good generalization capability.

Keywords: deep reinforcement learning；heterogeneous resources；multi-agent；distributed scheduling；flexible job shop；graph neural network

收稿日期: 2025-08-20；**录用日期:** 2025-12-30.
基金项目: 国家自然科学基金项目 (52405220)；四川省自然科学基金项目 (2024ZHCG0028).
责任编委: 李新宇.
†通信作者. E-mail: pengguo318@swjtu.edu.cn.

0 引 言

为满足下一代高速列车的高精度制造需求, 并解决其与既有系列列车混线生产带来的一致性差、效率低等难题, 亟需探索新的混线生产模式. 该模式需在不显著增加成本的前提下, 有效适配多系列列车的差异化工艺要求[1], 这对生产调度系统提出了更高挑战. 轨道交通装备制造业正加速向分布式制造模式转型[2]. 分布式制造系统通过跨地域或跨单元的资源协同, 能够显著提升多车型混线生产的灵活性以及关键设备的资源利用率[3].

面向轨道车辆多车型混线生产的实际需求, 车体组装产线通常采用分布式空间布局, 并将加工设备与辅助资源等集成为若干相对独立的柔性制造单元 (FMU). 整车组装依赖这些 FMU 之间的协同配合完成. 不同 FMU 在工装与工具配置、人员编组及节拍安排等方面存在差异, 这些条件会影响装配操作的可执行时机与处理方式. 例如, 若某工位具备特定工装, 则相关操作可提前在该工位完成; 否则需在具备条件的其他工位或在后续阶段处理. 同样, 人员配置的差异也会改变操作的合并或拆分方式. 上述条件的不同使得相同车体在不同 FMU 中呈现出不完全一致的工序组合与顺序, 形成工艺路线的异构, 而 FMU 内部设备能力的差别又带来了资源选择的柔性. 基于这些跨单元工艺差异与内部资源可选性的共同作用, 本文将轨道车辆分布式组装过程抽象为分布式异构柔性作业车间调度问题 (DHFJSP).

针对分布式车间调度问题, 国内外学者提出了多种优化方法. Han 等[4] 提出了一种改进贪心迭代算法, 有效提升了求解分布式流水车间调度问题的质量和稳定性; Wang 等[5] 针对分布式作业车间调度问题, 构建工厂分配与工序顺序的联合解空间, 描述异构工厂间的工艺约束与机器选择多样性; Tian 等[6] 针对分布式装配作业车间调度问题, 利用三向量编码方案实现工序-工厂-装配线的决策解耦; Deng 等[7] 和 Wang 等[8] 分别提出知识驱动的模因算法和自适应模因算法, 解决分布式场景下工件的分配与工序的排序; Wei 等[9] 针对共享制造环境下的供应-需求匹配问题, 提出估计分布算法-禁忌搜索混合方法增强全局搜索能力; Zhao 等[10] 在 DHFJSP 中, 提出多目标适应度景观估计分布算法, 分离工厂工艺约束与机器选择逻辑.

近年来, 深度强化学习 (DRL) 凭借其在高维决策问题中的强大表达能力和自学习能力, 在车间调度领域得到广泛应用[11]. DRL 提供的解决方案质量与元启发式相当, 但计算时间更短[12], 为解决车间调度相关问题提供了新的解决方案. 孙爱红等[13] 通过智能体自主选择调度规则, 在作业车间中实现机器和 AGV 联合调度; Luo[14] 针对动态柔性车间调度问题, 采用双深度Q网络与软目标更新机制提升策略稳定性; Lei 等[15]针对动态分布式作业车间调度问题, 设计 11 种复合调度规则作为动作空间, 验证了近端策略优化 (PPO) 等 5 种 DRL 方法的有效性. 值得注意的是, 基于 DRL 的调度方法高度依赖状态表征精度, 图神经网络 (GNN) 凭借其对实体关系之间拓扑结构的建模能力, 逐渐成为 DRL 框架中状态表征的核心技术. Wang 等[16] 提出一种双注意网络实现高效的特征提取, 提升柔性车间调度效果; Huang 等[17] 提出一种基于 GNN 的多动作策略, 优化分布式车间调度.

综上, 现有分布式车间调度研究大多基于单元同构的假设, 在流水车间和作业车间等典型场景中取得了一定进展. 然而, 这一假设在轨道车辆组装等实际生产环境中面临挑战, 因其柔性制造单元功能异构性突出, 导致同一工件在不同单元中需遵循差异化的工序序列, 从而对调度系统的全局协调能力提出了更高要求. 另一方面, 现有基于 GNN 的 DRL 方法通常局限于单一车间环境或简化的分布式场景, 尚未构建能够同时捕捉单元间工艺异构性与资源适配性的全局状态表征, 难以支撑复杂异构环境下的协同优化. 同时, DHFJSP 需要处理工件分配、工序排序与机器选择多重决策, 现有优化框架在同时处理这些决策并兼顾工艺路线异构性方面仍存在局限性.

针对轨道车辆组装中存在的多单元协同、工艺路径差异以及资源能力异构等调度特征, 本文的主要工作包含以下 3 个方面: 1) 构建能够刻画跨单元工艺差异与内部资源柔性的分布式异构柔性作业车间调度数学模型, 并在此基础上提出两阶段多智能体深度强化学习框架 (2S-MADRL). 该框架通过形式化的多阶段马尔可夫决策过程, 实现了跨单元工件分配与单元内部调度的协同决策. 2) 为反映系统层面的结构特征, 设计用于表征工件在不同 FMU 差异化工艺序列的全局异构图, 为智能体获取跨单元信息提供结构化支撑. 3) 构建分层异构图注意力网络, 使模型能够在更细粒度上刻画工件在各单元中的工艺需求与资源能力之间的匹配关系, 从而提升调度决策的适应性. 最后, 基于实际产线数据验证所提方法在不同规模场景下的有效性与泛化性能.

1　问题描述与模型构建

1.1　问题描述

DHFJSP 核心特征在于工件的工艺路线在不同 FMU 中具有不同的设定. 该问题可以表述为: 给定由 f 个 FMU 构成的分布式制造系统 $\mathcal{U}=\{U^1,U^2,\ldots,U^l,\ldots,U^f\}$, 每个 FMU U^l 包含 m 台异构机器的集合 $\mathcal{M}^l=\{M_1^l,M_2^l,\ldots,M_k^l,\ldots,M_m^l\}$. 待加工的工件集合为 $\mathcal{J}=\{J_1,J_2,\ldots,J_i,\ldots,J_n\}$, 所有工件在零时刻静态存在且可调度. 每个工件 J_i 在不同 FMU 中具有异构的工艺路线: 若分配至 U^l, 则其加工过程需遵循该单元特定的工序序列 $\mathcal{O}_i^l=\{O_{i1}^l,O_{i2}^l,\ldots,O_{ij}^l,\ldots,O_{in_{il}}^l\}$, 其中 n_{il} 表示在 U^l 中加工 J_i 所需的工序总数. 对于任意工序 O_{ij}^l, 其可加工机器集合为 $\mathcal{M}_{ij}^l\subseteq\mathcal{M}^l$, 且在机器 $M_k^l\in\mathcal{M}_{ij}^l$ 上的加工时间为 p_{ijk}^l. 调度目标是将每个工件分配至合适的 FMU, 并确定各 FMU 内工序的加工顺序和加工机器, 以最小化全局最大完工时间 $C_{\max}=\max\limits_{l\in\{1,\ldots,f\}}C^l$, 其中 C^l 表示 U^l 中所有工件的最后一道工序的完工时间.

1.2　数学模型

针对上述问题, 以最小化最大完工时间为目标构建数学模型, 符号与描述见表 1.

表1　符号与描述

符号	描述
B	足够大的常数
x_{il}	工件 J_i 分配给 U^l 为1, 否则为0
y_{ijkl}	工序 O_{ij}^l 在 U^l 内分配到机器 M_k^l 为1, 否则为0
S_{ij}^l	工序 O_{ij}^l 的开始加工时间
C_{ij}^l	工序 O_{ij}^l 的完工时间
$z_{iji'j'kl}$	工序 O_{ij}^l 先于工序 $O_{i'j'}^l$ 在机器 M_k^l 加工为1, 否则为0

$$\min\ C_{\max}. \tag{1}$$

$$\text{s.t.}\ \sum_{l\in U}x_{il}=1,\ i\in\mathcal{J}; \tag{2}$$

$$\sum_{k\in\mathcal{M}_{ij}^l}y_{ijkl}=x_{il},\ i\in\mathcal{J},l\in\mathcal{U},j\in\mathcal{O}_i^l; \tag{3}$$

$$S_{i(j+1)}^l\geqslant C_{ij}^l-B(1-x_{il}),\ i\in\mathcal{J},l\in\mathcal{U},j\in\mathcal{O}_i^l; \tag{4}$$

$$\begin{aligned}&S_{ij}^l\geqslant C_{i'j'}^l-B(1-z_{iji'j'kl}),\\&i,i'\in\mathcal{J},l\in\mathcal{U},k\in\mathcal{M}^l,j\in\mathcal{O}_i^l,j'\in\mathcal{O}_{i'}^l;\end{aligned} \tag{5}$$

$$\begin{aligned}&S_{i'j'}^l\geqslant C_{ij}^l-Bz_{iji'j'kl},\\&i,i'\in\mathcal{J},l\in\mathcal{U},k\in\mathcal{M}^l,j\in\mathcal{O}_i^l,j'\in\mathcal{O}_{i'}^l;\end{aligned} \tag{6}$$

$$C_{ij}^l=S_{ij}^l+\sum_{k\in\mathcal{M}_{ij}^l}p_{ijk}^ly_{ijkl},\ i\in\mathcal{J},l\in\mathcal{U},j\in\mathcal{O}_i^l; \tag{7}$$

$$C^l\geqslant C_{in_{il}}^l,\ i\in\mathcal{J},l\in\mathcal{U}; \tag{8}$$

$$C_{\max}\geqslant C^l,\ l\in\mathcal{U}. \tag{9}$$

工件分配约束 (2) 保证每个工件 J_i 必须且仅可分配到一个 FMU. 工序配分约束 (3) 保证若工件 J_i 分配到 U^l, 其在该 FMU 中的每道工序 O_{ij}^l 必须从可加工机器集合 $\mathcal{M}_{ij}^l$ 中选择一台机器加工. 工序顺序约束 (4) 确保在 U^l 中, 工件 J_i 的工序必须按照其特定工艺路线 $\mathcal{O}_i^l$ 执行, 后一道工序的开始时间不得早于前一道工序的完工时间. 机器加工不重叠约束 (5) 和 (6) 保证在 U^l 的同一台机器 M_k^l 上, 不同工件的工序不能重叠加工. 约束 (7) 表示工序 O_{ij}^l 的完工时间等于开始时间加上在所分配的机器上的加工时间. 约束 (8) 确保 U^l 的完工时间 C^l 是该 FMU 内所有工件的最后一道工序完工时间的最大值. 约束 (9) 保证全局最大完工时间 $C_{\max}$ 是所有 FMU 完工时间的最大值.

2　两阶段多智能体强化学习框架

针对轨道车辆分布式组装场景下的 DHFJSP 中工艺路线可选性和加工资源高度异构带来的调度挑战, 提出一种两阶段多智能体深度强化学习框架如图 1 所示. 第 1 阶段为工件分配阶段, 上层智能体 agent_a 基于分层异构图注意力网络 (HHGAT) 聚合全局异构图中的节点特征, 量化不同 FMU 对各工件的适配度权重, 从而确定分配决策. 工件分配完毕后进入局部排产阶段, 下层智能体 agent_s 为工序选择智能体 agent_{s1} 和机器分配智能体 agent_{s2}, 两个智能体基于局部异构图的特征表示, 采用编码器-解码器架构的策略模型, 为 FMU 内的工件确定每个时间步的工序选择和机器分配, 从而生成各 FMU 的调度方案.

2.1　全局异构图

为了表征工件在不同 FMU 中的工艺路线安排及 FMU 加工能力, 提出了如图 1 所示的全局异构图 $\mathcal{G}=(\mathcal{J}\bigcup\mathcal{U},\mathcal{O}\bigcup\mathcal{M},\mathcal{C},\mathcal{E})$. 其中: 工序集合为 $\mathcal{O}=\{\mathcal{O}^1,\mathcal{O}^2,\ldots,\mathcal{O}^f\}$, 即同一工件对不同 FMU 具有独立的工序集合; $\mathcal{M}$ 表示所有 FMU 中的机器集合; 工序节点 $O_{ij}^l\in\mathcal{O}$ 和机器节点 $M_k^l\in\mathcal{M}$ 的原始特征向量分别为 μ_{ij}^l 和 ν_k^l; 工件节点集合 $\mathcal{J}$ 表示所有待分配的工件; $\mathcal{U}$ 表示所有 FMU 的集合, 节点 $U^l\in\mathcal{U}$ 提供了该 FMU 的原始特征向量 ξ^l; $\mathcal{C}$ 是工序间的连接弧,

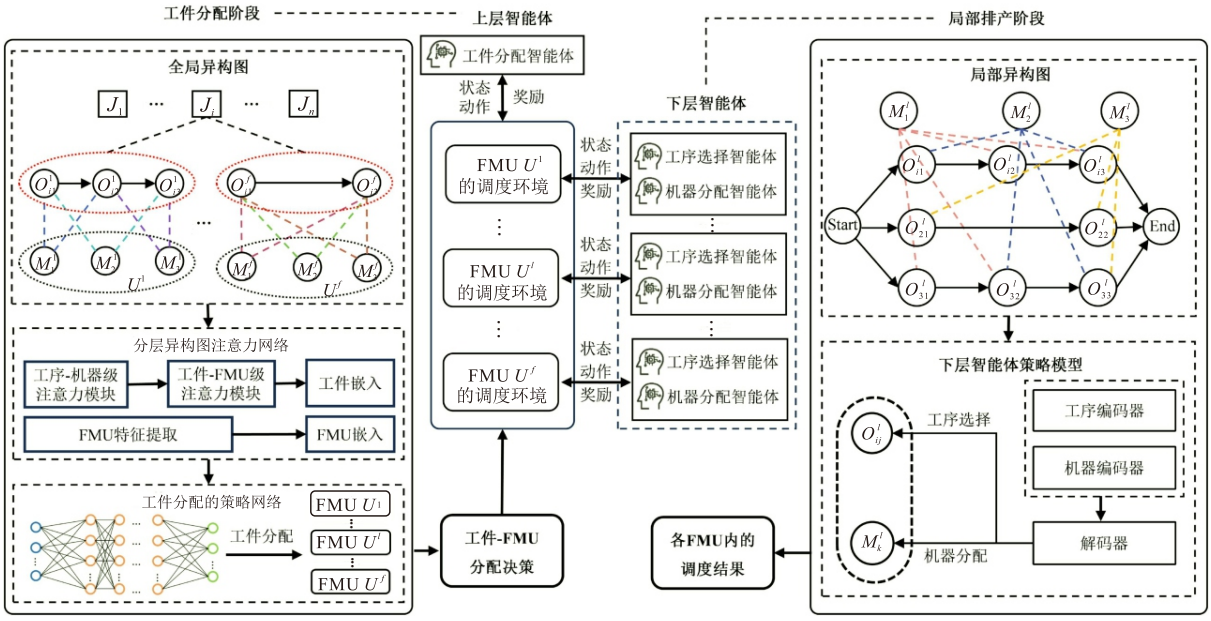


图1 针对 DHFJSP 的两阶段多智能体深度强化学习框架

表示工序顺序约束; $\mathcal{E}$是各工序与兼容机器间弧的集合; 每条工序-机器弧 (O-M弧) 的原始特征为λ_{ijk}^{l}.

2.2 马尔可夫决策过程

整个调度流程可以视为一个多阶段的决策过程, 工件分配阶段需要选择工件分配到合适的 FMU, 局部排产阶段确定 FMU 内的调度方案. 每个时间步, 智能体需要根据当前的系统状态决定最合适的工件分配或进行工序排序与机器选择. 以下是多阶段马尔可夫决策过程 (MMDP) 的具体描述.

2.2.1 状态表示

工件分配阶段, agent_a在时间步t的状态s_t^*通过全局异构图$\mathcal{G}_t=(\mathcal{J}_t\bigcup\mathcal{U},\mathcal{O}_t\bigcup\mathcal{M},\mathcal{C}_t,\mathcal{E}_t)$包含的信息进行表征. 其中: 工序的原始特征$\mu_{ij}^{l}$包含了工序$O_{ij}^{l}$的可加工机器数、平均加工时间、$J_i$在$U^l$的工序数、$J_i$在$U^l$加工预估完成时间等指标; 机器的原始特征$\nu_k^l$包含了机器$M_k^l$邻接工序数、可加工工序的加工时间等指标, 连接节点对(O_{ij}^{l},M_k^{l})的O-M弧的原始特征λ_{ijk}^{l}是机器M_k^l加工工序O_{ij}^{l}的加工时间.

已定义的工序、机器及连接弧特征能够聚合得到工件的特征嵌入, 为了让agent_a从全局角度综合考虑工件分配, 即使某工件在特定 FMU 中加工工艺最简化、加工时间最短, 但可能该 FMU 的负载已经远超其他 FMU, 此时选择该 FMU 是不明智的. 因此需要描述 FMU 的整体生产信息, 使得能够考虑 FMU 之间的生产均衡. 为此, 提取 FMU 的原始特征ξ^l包含U^l中潜在加工负载、待加工工序数、待加工工序平均加工时间等指标.

在局部排产阶段, agent_s在时间步t的状态$s_t^{\triangledown}$通过异构图$\mathcal{H}_t^l=(\mathcal{O}_t^l,\mathcal{M}_t^l,\mathcal{C}_t^l,\mathcal{E}_t^l)$表示. 其中: $\mathcal{O}_t^l$表示U^l包含的工序集合, $\mathcal{M}_t^l$表示U^l中的可用机器集合, $\mathcal{C}_t^l$表示$\mathcal{O}_t^l$间的连接弧集合, $\mathcal{E}_t^l$表示U^l中O-M弧集合. 分配至U^l后, 工序节点O_{ij}^l的原始特征$\mu_{ij}^{l\triangledown}$除了含有$\mu_{ij}^l$中的特征外, 还包含调度状态 (0 为未调度, 1 为已调度)、最早开始时间两个特征; 机器M_k^l的原始特征$\nu_k^{l\triangledown}$由可加工工序数、加工负载等指标组成, O-M弧的原始特征$\lambda_{ijk}^{l\triangledown}$继承了$\lambda_{ijk}^l$的表示.

2.2.2 动作空间

上层智能体agent_a的主要任务是选择工件并分配给某个合适的 FMU. 为了简化该任务, 两个决策 (即选择待分配的工件、选择合适的 FMU) 被整合为一个单一的动作. 具体地, 定义工件和 FMU 的配对 (J-U) 作为动作空间中的元素$a_t^*\in A_{\text{up}}(t)$, 动作空间定义为

$$A_{\text{up}}(t)=\{(J_i^l)\in\mathcal{J}\times\mathcal{U}\},\tag{10}$$

其中J_i^l表示将工件J_i分配给U^l.

下层智能体agent_s的决策任务涉及在 FMU 内部进行工序选择和机器分配. 其中: agent_{s1}的动作a_t^o表示从当前待调度的工件中选择合适的工序, 其动作空间为 FMU 内尚未完成工件的可选工序集; 而agent_{s2}的动作a_t^m是在 FMU 内为所选工序分配合适的机器进行加工, 其动作空间包含 FMU 内的所有可加工所选工序的机器.

2.2.3 奖励函数

在 DHFJSP 中, 全局目标 (最小化最大完工时间) 受 FMU 间资源协调和 FMU 内工序调度的双重影响, 若采用单一的全局奖励, 则多智能体训练往往

会出现收敛慢、稳定性不足等问题, 从而降低学习效率[18], 尤其是在分配阶段并未开始下层的调度, 上层智能体难以通过最大完工时间的差值评估策略的优劣. 为此, 本文分别为上层和下层智能体设计差异化奖励, 以避免因联合奖励产生的策略振荡.

为此, 定义上层智能体的奖励r_t^*, 它基于每次决策前后最大的 FMU 负载变化进行计算, 有

$$r_t^* = \mathcal{L}_{\max}(t) - \mathcal{L}_{\max}(t+1), \tag{11}$$

其中$\mathcal{L}_{\max}(t)$表示在决策步t, 所有 FMU 的负载的最大值, 反映了整个系统中最大负载. 奖励值为负值表示负载增加, 因此agent_a通过最大化该奖励实现负载均衡. 这一奖励设计鼓励智能体在工件分配时优化负载均衡, 从而达到缩短全局完工时间的目标[19]. 下层智能体的目标是最小化 FMU 内部的最大完工时间, 奖励函数为$r_t^\nabla = C(s_t^\nabla) - C(s_{t+1}^\nabla)$, 其中$C(s_t^\nabla)$为时间步$t$当前 FMU 的最大完工时间.

2.2.4　状态转移

agent_a在状态s_t^*选定工件分配到 U^l 后, 从缓冲池中删除该工件的状态信息, 随后读取该工件在 U^l 内的工艺信息, 提取其状态特征并纳入 U^l 的调度流程, U^l 的状态信息也同步更新. 而双智能体协同决策的触发与上层工件分配结果直接关联, 上层完成所有工件分配后, 仅当 FMU 同时存在待安排工序与可用机器时, 下层双智能体于决策点t启动动作. 具体而言, agent_{s1}先依据当前状态s_t^∇执行工序选择动作, 从待安排工序中确定目标工序O_{ij}^l; agent_{s2}实时接收该工序信息后, 同步开展机器分配动作, 将该工序调度至适配机器M_k^l, 并结合该机器的空闲时间与工序工艺约束, 计算工序的最早开工时间, 实现两者决策的有序协同.

2.3　参数化策略

2.3.1　工件分配策略

针对 DHFJSP 中工件工艺路线异构导致工序节点状态独立的特征, 本文设计分层异构图注意力网络, 以精准表征多维异构特征与跨层级关联信息. 如图 2 所示, 通过两类核心元路径实现跨类型节点信息传递: 工序-机器元路径建立工序与加工机器的关联, HHGAT 的工序-机器级注意力模块依托该元路径, 捕捉工序加工要求与机器功能、性能的匹配关系; 工件-FMU 元路径指工件与其在不同 FMU 内对应工序的连接链路, 对应的工件-FMU 级注意力模块侧重学习工件在不同 FMU 的潜在适配性, 同时关联 FMU 工艺能力与负载状态. 两模块协同作用, 实现全局与局部异构特征的有效融合.

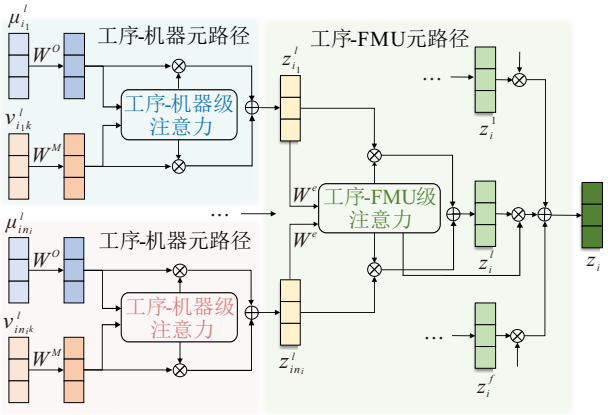


图2　HHGAT 特征提取流程

1) 首先采用工序-机器级注意力模块聚合工序的信息以保留独立的状态特征. 在每个 FMU 中, 工序 O_{ij}^l 对其邻接机器节点的注意力得分定义为

$$e_{ijk}^l = \text{LeakyReLU}(a^{\mathrm{T}}(W^O\mu_{ij}^l \| W^M\nu_{ijk}^l)), \tag{12}$$

$$e_{ij}^l = \text{LeakyReLU}(a^{\mathrm{T}}(W^O\mu_{ij}^l \| W^O\mu_{ij}^l)). \tag{13}$$

其中: a、W^O和W^M为可学习参数, T表示转置操作, $\|$ 表示向量拼接, LeakyReLU 为采用的激活函数, $\nu_{ijk}^l = [\nu_k^l \| \lambda_{ijk}^l]$为将$O$-$M$连接信息拼接到机器特征中. 随后, 将邻接机器与工序自身的得分通过 softmax 函数归一化, 得到工序与机器的注意力系数α_{ijk}^l和α_{ij}^l, 从而获得工序O_{ij}^l 针对U^l 的特征嵌入

$$z_{ij}^l = \sigma\Big(\sum_{k\in\mathcal{N}(O_{ij}^l)} \alpha_{ijk}^l W^M\nu_{ijk}^l + \alpha_{ij}^l W^O\mu_{ij}^l\Big). \tag{14}$$

其中: σ为激活函数, $\mathcal{N}(O_{ij}^l)$ 为工序 O_{ij}^l 的邻接机器节点集. z_{ij}^l保留了工序O_{ij}^l在U^l中的工艺路线信息. 为了增强特征提取能力, 堆叠H层工序-机器级注意力模块, 以获得最终工序嵌入$\mathcal{Z}_{ij}^l$.

2) 随后采用工件-FMU 级注意力模块学习工件 J_i对不同 FMU 的特征嵌入$\mathcal{Z}_i^l$. 首先通过多层感知机 (MLP) 来变换$\mathcal{Z}_{ij}^l$, 然后使用变换后的特征嵌入与语义注意力向量q的相似性量化ω_{ij}^l, 如果变换后的嵌入与注意力向量q高度相似, 则说明该工序所在元路径在智能体决策过程中的贡献较大. 通过ω_{ij}^l归一化聚合$\mathcal{Z}_{ij}^l$可以获得$\mathcal{Z}_i^l$, 即

$$\omega_{ij}^l = q^{\mathrm{T}} \cdot \tanh(W^e \times \mathcal{Z}_{ij}^l + b), \tag{15}$$

$$\mathcal{Z}_i^l = \sum_{j=1}^{n_{il}} \Big(\frac{\exp(\omega_{ij}^l)}{\sum_{j=1}^{n_{il}} \exp(\omega_{ij}^l)} \times \mathcal{Z}_{ij}^l\Big). \tag{16}$$

其中: W^e为可学习参数, b为偏差向量, tanh为激活函数, q为语义注意力向量. 随后利用工件-FMU 元路径权重求和, 可以有效表征工件-FMU 元路径的重

要性. 因此, 工件J_i的嵌入可以通过如下公式得到:

$$\mathcal{Z}_i=\sum_{l\in\mathcal{U}}\Big(\frac{\exp\Big(\frac{1}{n_{il}}\sum\limits_{j=1}^{n_{il}}\omega_{ij}^{l}\Big)}{\sum\limits_{l\in\mathcal{U}}\exp\Big(\frac{1}{n_{il}}\sum\limits_{j=1}^{n_{il}}\omega_{ij}^{l}\Big)}\times\mathcal{Z}_i^{l}\Big).\tag{17}$$

3) 在进行工件分配时, 还需要考虑 FMU 的潜在负载信息, 因此堆叠H层 MLP 对 FMU 的原始状态特征ξ^l进行编码得到其嵌入$\mathcal{F}_l$. 将得到的工件嵌入、FMU 嵌入与它们的池化向量输入到由 MLP 组成的策略网络, 得到工件-FMU 动作的选择得分

$$\rho(a_t^*,s_t^*)=\\ \mathrm{MLP}\Big(\mathcal{Z}_i\|\mathcal{F}_l\|\frac{1}{|\mathcal{J}|}\sum_{i\in\mathcal{J}}\mathcal{Z}_i\|\frac{1}{|\mathcal{U}|}\sum_{l\in\mathcal{U}}\mathcal{F}_l\Big).\tag{18}$$

通过 softmax 函数将动作得分归一化为概率分布.

2.3.2 局部排产策略

下层智能体的策略模型依赖于图神经网络, 采用编码器-解码器架构, 旨在学习和提取工序与机器之间的复杂依赖关系, 并通过注意力机制聚焦于重要特征. 基于 FMU 内部车间调度进行以下关于编码器-解码器架构的描述.

1) 工序编码器对于每个源节点 (工序O_{ij}^l), 提取兼容目标节点 (机器M_k^l) 及其O-M连接弧信息, 以生成每个工序节点的嵌入向量. 以工序为中心聚合邻接机器信息, 其中O-M连接弧信息拼接到机器特征中$\nu_{ijk}^{l\nabla}=[\nu_k^{l\nabla}\|\lambda_{ijk}^{l\nabla}]$. 通过两个线性变化$W^o$与$W^m$将工序原始特征$\mu_{ij}^{l\nabla}$与$\nu_{ijk}^{l\nabla}$分别变换到相同维度, 可以获得工序$O_{ij}^l$的注意力系数, 即机器$M_k^l$与其自身对$O_{ij}^l$的重要性, 公式如下:

$$e_{ijk}=\mathrm{LeakyReLU}(a^{\mathrm{T}}(W^o\mu_{ij}^{l\nabla}\|W^m\nu_{ijk}^{l\nabla})),\tag{19}$$

$$e_{ij}=\mathrm{LeakyReLU}(a^{\mathrm{T}}(W^o\mu_{ij}^{l\nabla}\|W^o\mu_{ij}^{l\nabla})).\tag{20}$$

将注意力系数e_{ijk}与e_{ij}通过 softmax 函数进行归一化后可以得到每个邻接节点的权重α_{ijk}与α_{ij}, 通过权重信息融合得到工序节点的特征嵌入

$$\mu_{ij}'=\sigma\Big(\alpha_{ij}W^o\mu_{ij}^{l\nabla}+\sum_{k\in\mathcal{N}(O_{ij}^l)}\alpha_{ijk}W^m\nu_{ijk}^{l\nabla}\Big).\tag{21}$$

2) 机器编码器由H层 MLP 组成, 用于提取选定工序的兼容机器节点及连接弧信息. 将机器特征编码并映射到d维空间作为机器M_k^l的嵌入.

3) 为了增强特征提取能力, 分别堆叠具有相同结构但独立可训练参数的L层工序编码器与机器编码器, 以获得最终特征嵌入$\mu_{ij}^{'(L)}$与$\nu_k^{'(L)}$. 通过自注意力机制一步增强工序间的依赖关系, 将获得的μ_{ij}^{attn}通过平均池化操作获得池化向量h_{ij}, 机器特征嵌入的池化向量表示为u_k, 随后通过 MLP 生成工序得分s_{ope}和机器得分s_{ma}, 即

$$\mu_{ij}^{\mathrm{attn}}=\mathrm{SelfAttn}(W^Q\mu_{ij}^{'(L)},W^K\mu_{ij}^{'(L)},W^V\mu_{ij}^{'(L)}),\tag{22}$$

$$s_{\mathrm{ope}}(a_t^o,s_t^{\nabla})=\mathrm{MLP}_o(\mu_{ij}^{\mathrm{attn}}\|h_{ij}),\tag{23}$$

$$s_{\mathrm{ma}}(a_t^m,s_t^{\nabla})=\mathrm{MLP}_m(\nu_k^{'(L)}\|u_k),\tag{24}$$

其中W^Q,W^K,W^V为可学习参数. 最后, 用 softmax 函数将动作得分转换为概率分布.

2.4 智能体训练

为了避免多智能体训练不稳定, 本文采用分阶段的训练方式[20]: 下层智能体首先在固定大小的实例上进行训练, 上层智能体随后使用下层智能体得出的策略进行训练, 着重全局工件分配. 在工件分配阶段, 上层智能体的训练过程采用 PPO 算法[21], 通过最大化累积奖励优化工件分配策略, 并结合 HHGAT 学习环境有效信息以最小化系统负载波动.

在局部排产阶段, 对于下层的双智能体, 采用多智能体近端策略优化 (MAPPO), 两个智能体具有独立的策略网络和价值网络, 能够更加精确地针对双智能体各自的任务进行优化. 在训练过程中利用采样策略进行探索, 在测试中采用贪婪策略选择动作.

3 实验结果与分析

本文引入大量计算分析以评估所提出框架的性能. 测试环境为配备 Intel Xeon Gold 5218R CPU、英伟达 GeForce RTX 3090 GPU 和 Ubuntu 20.04 系统的工作站.

3.1 算例生成

基于中车多车型 (动车组、地铁、机车) 混线生产模式设计验证算例. 如表 2 所示, 采用均匀分布生成实例, 考虑不同规模 (车型数n-制造单元数f-关键设备数m) 对求解轨道车辆 DHFJSP 的影响, 验证 2S-MADRL 性能. 按照规模n-f-m各生成 10 个算例.

表2 生成算例的参数

参数	值
FMU数(f)	{2, 3, 4}
每个FMU中机器数(m)	10
工件数(n)	{20, 30, 50, 70}
每个工序的兼容机器数	Unif [1, m]
工件在各FMU的工序数	Unif [4, 8]
工序加工时间	Unif [5, 20]

3.2 训练参数配置

为验证关键参数配置对算法性能的影响并确定最优组合, 本文针对所提出方法的 4 个核心参数开展正交实验. 各参数的水平设置如下: 学习率$l_r\in\{1\times$

$10^{-4}, 2\times10^{-4}, 3\times10^{-4}\}$, 奖励折扣因子$\gamma\in\{0.9, 0.95, 1\}$, HHGAT 迭代次数$H\in\{1,2,3\}$, 神经网络隐藏层维度$d\in\{64,128,256\}$. 选择正交组数$L_9(3^4)$, 包含 9 组参数水平组合. 训练总迭代次数统一设定为 2 000 次, 每 20 次迭代更换训练算例, 每 10 次迭代在独立验证集上评估模型性能. 以最大完工时间的均值最小化为评估指标, 对不同参数组合训练的模型进行测试, 通过绘制图 3 关键参数主效应图, 最终确定最佳参数组合为 $l_r=2\times10^{-4}$, $\gamma=1$, $H=2$, $d=128$.

图 4 展示了 20-2-10 规模下模型的训练曲线及完工时间收敛曲线, 记录了智能体每回合奖励及每 10 回合奖励的平均值. 由图 4 可见, 随着训练推进, 上层与下层智能体的奖励平稳上升并逐渐收敛, 表明智能体能够捕捉生产环境特征, 掌握合理的工件分配与调度排产策略以最大化奖励, 从而实现最小

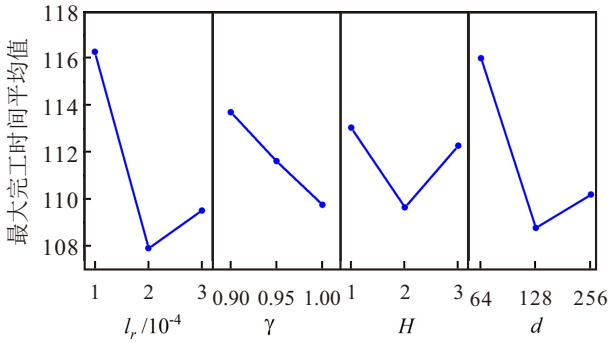
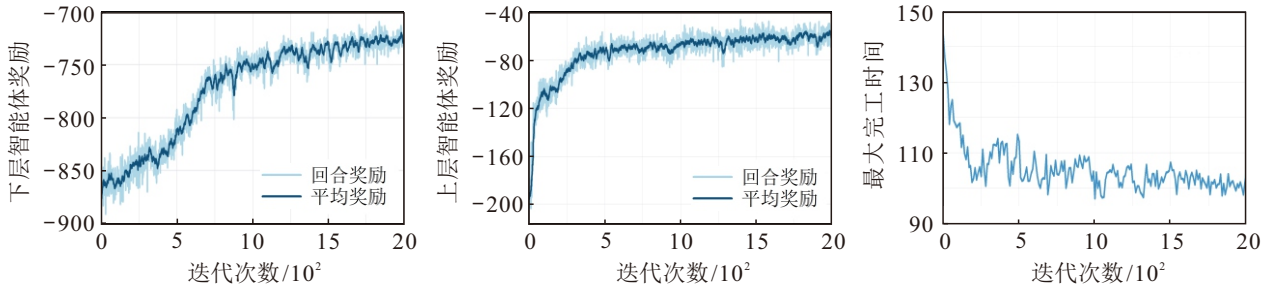


图3　关键参数主效应图

化最大完工时间的优化目标. 图 5 展示了不同规模下的训练时长, 随着问题规模增大, 训练阶段的计算成本相应提升, 这主要源于多智能体在学习跨单元与单元内部协同策略时需要大量交互样本. 需要强调的是, 本文方法采用离线训练方式, 训练过程可在非生产时段独立完成, 不影响实际调度的实时性, 模型训练完成后, 在线求解时间与调度规则相当.



(a) 下层智能体收敛曲线　(b) 上层智能体收敛曲线　(c) 完工时间收敛曲线

图4　20-2-10 规模下智能体的训练曲线

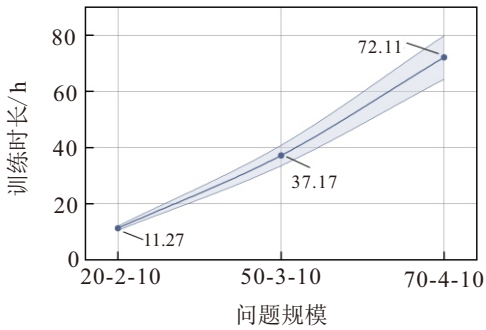


图5　不同问题规模下模型的训练时长

3.3　基线方法

为了展示所提出方法的优越性, 与实际生产普遍采用的调度规则方法作对比. 其中, 工件分配采用以下 3 种工件分配规则 (JAR):

JAR1: 工件J_i在U^l中的预估加工时间为$\mathrm{TPT}_i^l=\sum_{j=1}^{n_{il}}\frac{1}{|M_{ij}^l|}\sum_{k\in M_{ij}^l}p_{ijk}^l$. 选择$\mathrm{TPT}_i^l$最小的$J$-$U$组合.

JAR2: 计算每个工件的平均总加工时间$\overline{\mathrm{TPT}}_i=\sum_{l=1}^{f}\frac{\mathrm{TPT}_i^l}{f}$, 选择$\overline{\mathrm{TPT}}_i$最大的工件分配至当前剩余工件数量最少的 FMU.

JAR3: 定义工件的工艺复杂度指标为$\mathrm{MON}_i=\min\limits_{l\in[1,f]}n_{il}$, 选择$\mathrm{MON}_i$最大的工件分配至对应$U^l$.

工序选择采用实际生产调度中表现较好的 4 种调度规则, 分别为最多剩余作业 (MWKR)、先进先出 (FIFO)、最多剩余工序数 (MOR) 以及最少剩余作业 (LWKR). 机器分配调度规则采用最短处理时间 (SPT) 和最早结束时间 (EET). 将 3 类调度规则组合可得到共 24 种复合调度规则作为基准方法对比.

为更进一步评估 2S-MADRL 方法, 本文还将其与元启发式方法 HGTSA[22] 进行对比. 该方法原用于求解分布式柔性作业车间调度问题, 本文对其进行适应性修改以适配 DHFJSP 场景. 为降低实验随机性带来的偏差, 以上基线方法在每个算例上均独立运行 5 次, 以平均值作为实验结果. 此外, 本文以 Gurobi 求解器对第 1.2 节混合整数规划模型的求解结果为参考, 求解时间限定为 3 600 s.

3.4　实验结果分析

3.4.1　消融实验

为了验证上层智能体和下层智能体的有效性,

将其与表现较优的调度规则 (JAR1+SPT+MWKR) 进行组合作为对比. 记录各方法在不同规模的 10 个算例上的平均目标值 (Obj), 并以最优方法的目标值作为计算 Gap 值的基准.

随着规模增加, 各对比方法与 2S-MADRL 的 Obj 与 Gap 值差距变化趋势如图 6 所示. 在 20-2-10 规模下, JAR1 配合 agent_s 显著优于 agent_a + SPT + MWKR 组合, 然而规模扩展至 50-3-10 与 70-4-10 时, agent_a 全局分配的优化效果逐渐凸显, 其与 SPT + MWKR 组合的 Gap 值呈降低趋势, 而 JAR1 + agent_s 的 Gap 值显著提升, 这表明了上层智能体在跨工厂协同中的有效性. 此外, 下层智能体在大规模场景中展现出独特的局部优化价值, 如在 70-4-10 规模中, 2S-MADRL 的目标值小于agent_a + SPT + MWKR 组合, 表明其能有效补偿复杂负载波动.

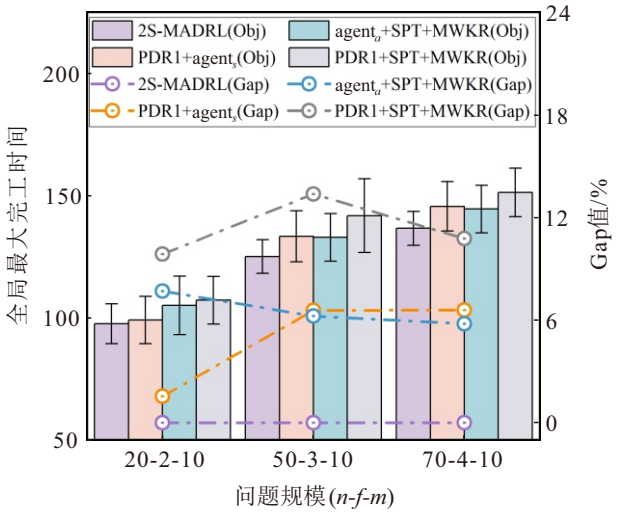


图6 消融实验目标值与 Gap 值变化趋势

3.4.2 对比分析

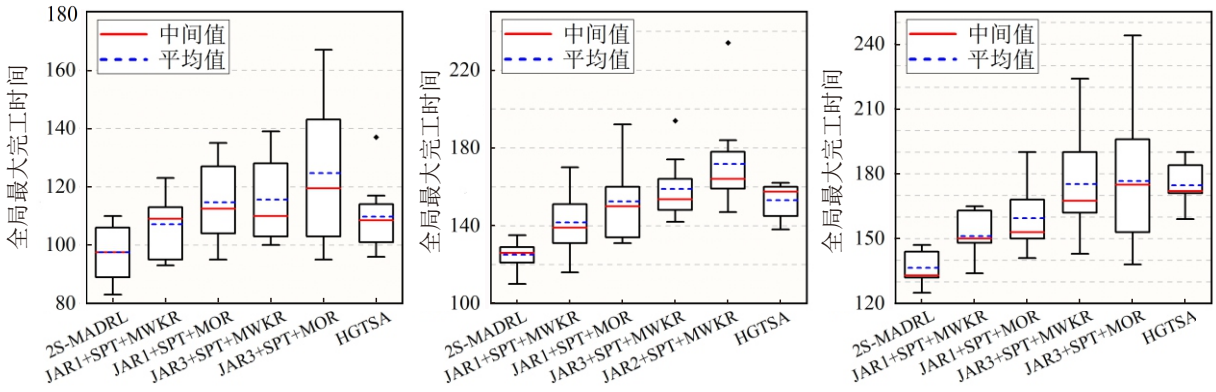
本节对各基线方法进行了广泛的对比实验. 表 3 展示了各方法在每个规模的 10 个随机算例上的平均目标值 (Obj), Gap 值和平均运行时间, 其中仅给出 8 种表现最好的调度规则组合的结果. 表 3 结果显示, 所提出方法在所有测试规模下均优于复合调度规则和 HGTSA, 在小规模下与 Gurobi 求解结果差距相对较小. 其中最佳复合调度规则组合 (JAR1 + SPT + MWKR) 在所有规模下均优于其他规则组合, 但其在 70 个工件规模下, 求解质量随着 FMU 数增长而有所下降, 揭示其难以适应 FMU 规模扩展带来的决策复杂度提升.

在限定时间内, Gurobi 求解质量随着问题规模扩大而下降, 而所提出方法依然适用, 且取得了更突出的表现. 在计算效率方面, 所提出方法计算时间显著小于 HGTSA, 即使随规模增长而有所增加, 但增加幅度在合理范围内, 与调度规则保持同一数量级.

为更直观对比数据的分布情况, 绘制了如图 7 所示箱线图. 可以看出, 2S-MADRL 的解集分布相比其他方法更为紧凑, 箱体高度显著低于各对比方法,

表3 2S-MADRL 与基线方法对比结果

规模		2S-MADRL	JAR1			EET	JAR2		JAR3		HGTSA	Gurobi
			SPT				SPT		SPT			
			MWKR	LWKR	MOR	MOR	MWKR	MOR	MWKR	MOR		
20-2-10	Obj	**97.5**	107.1	287.6	114.7	282.5	125.1	134.2	115.6	124.7	107.7	87.2
	Gap/%	**11.81**	22.82	229.82	31.54	223.97	43.46	53.90	32.57	43.00	23.51	0.00
	Time/s	1.71	1.79	1.76	1.71	1.72	2.09	1.97	1.77	1.70	632	3600
30-2-10	Obj	**132.0**	143.1	403.5	152.6	430.2	157.2	157.3	151.4	158.0	137.7	118.7
	Gap/%	**11.20**	20.56	239.93	28.56	262.43	32.43	32.52	27.55	33.11	16.01	0.00
	Time/s	3.55	3.79	3.52	3.57	3.58	4.36	4.13	3.78	3.59	993	3600
50-2-10	Obj	**163.9**	177.3	586.6	207.3	586.8	216.9	221.2	206.7	222.6	203.6	186.8
	Gap/%	**0.00**	8.18	257.90	26.48	258.02	32.34	34.96	26.11	35.81	24.22	13.94
	Time/s	9.51	9.66	8.96	9.08	9.04	11.17	10.34	9.66	9.04	1292	3600
50-3-10	Obj	**125.0**	141.7	436.1	152.4	460.7	171.7	179.8	158.8	175.4	153.1	128.6
	Gap/%	**0.00**	13.36	248.88	21.92	268.56	37.36	43.84	27.04	40.32	22.48	2.88
	Time/s	8.53	9.01	8.58	8.45	8.45	11.34	10.78	9.08	8.52	2167	3600
70-2-10	Obj	**216.8**	234.0	816.2	264.5	880.4	268.9	284.1	266.8	282.5	278.3	344.0
	Gap/%	**0.00**	7.93	276.48	22.00	306.09	24.03	31.04	23.06	30.30	28.37	58.67
	Time/s	20.44	19.56	19.19	20.56	20.56	19.19	21.63	19.76	20.21	2563	3600
70-3-10	Obj	**161.5**	176.5	539.0	194.3	611.6	224.4	231.5	201.3	210.9	206.5	181.0
	Gap/%	**0.00**	9.29	233.75	20.31	278.70	38.95	43.34	24.64	30.59	27.86	12.07
	Time/s	17.53	17.40	16.53	16.18	16.15	21.89	20.02	17.36	16.13	2530	3600
70-4-10	Obj	**136.5**	151.2	395.1	159.4	423.2	181.9	192.3	175.3	176.6	174.7	156.0
	Gap/%	**0.00**	10.77	189.45	16.78	210.04	33.26	40.88	28.42	29.38	27.99	14.29
	Time/s	16.53	16.46	12.06	15.33	15.30	21.73	20.60	16.34	15.20	2300	3600



(a) 20-2-10 规模　　(b) 50-3-10 规模　　(c) 70-4-10 规模

图7　对比方法在不同规模下的箱线图

表4　大规模实例上泛化结果

规模		2S-MADRL	JAR1				JAR2		JAR3	
			SPT			EET	SPT		SPT	
			MWKR	LWKR	MOR	MOR	MWKR	MOR	MWKR	MOR
80-3-10	Obj	**195.4**	205.3	610.0	219.1	632.2	246.0	239.2	221.2	223.7
	Gap/%	**0.00**	5.07	212.18	12.13	223.54	25.90	22.42	13.20	14.48
	Time/s	19.52	22.22	20.03	20.72	20.73	28.08	26.02	22.26	20.63
160-4-10	Obj	**259.1**	265.0	820.0	278.1	901.3	320.9	341.5	317.6	327.8
	Gap/%	**0.00**	2.28	216.48	7.33	247.86	23.85	31.80	22.58	26.51
	Time/s	84.12	86.06	79.08	78.27	80.37	113.07	101.16	85.02	77.58

且各规模下均无异常值, 进一步验证了所提出算法在性能和稳定性方面的优越性.

3.4.3　大规模实例上的泛化性能

进一步探讨所提出策略的模型在泛化到从未遇到过的大规模实例时的性能. 为此, 利用在小规模 (20-2-10) 实例上训练的模型直接应用于更大规模的实例 (80-3-10 和 160-4-10), 以评估其在扩展场景下的调度性能. 实验结果如表 4 所示, 在扩展到比训练规模大 8 倍的实例时, 最优调度规则组合与 2S-MADRL 的 Gap 值有所减小, 但该模型策略的调度结果依然优于所有对比的复合调度规则. 这表明, 2S-MADRL 在未见过的大规模问题上仍然能够保持一定的优化能力, 具有较强的泛化性.

4　结　论

本文针对轨道车辆分布式组装场景中工艺路线多样性与资源高度异构的复杂性, 提出了两阶段多智能体强化学习框架. 该方法采用协同决策机制应对车型工艺差异与关键设备异构的优化挑战: 在全局层级, 分层异构图注意力网络融合跨制造单元多维特征, 实现了车体等大型部件分配与 FMU 负载均衡; 在局部层级, 双智能体协作策略解析工序与关键设备的依赖关系, 生成了排产方案. 实验结果表明, 所提出方法在静态多车型混线环境下显著优于传统调度规则, 且在产线规模扩展时具有良好的稳定性与泛化性能. 未来将探索动态事件 (如设备故障、临时增补车型) 干扰下的实时重调度机制, 同时拓展多目标协同优化能力, 集成能耗与成本约束以提升绿色工厂场景的适用性.

参考文献 (References)

[1] 王忠凯, 史天运, 张惟皎, 等. 城际铁路动车组修造合一车间调度优化方法与应用系统[J]. 计算机集成制造系统, 2023, 29(8): 2773-2791.
(Wang Z K, Shi T Y, Zhang W J, et al. Optimization method of integration job shop scheduling of overhaul and manufacture and computer manufacturing system of intercity railway rolling stock[J]. Computer Integrated Manufacturing Systems, 2023, 29(8): 2773-2791.)

[2] 刘泰, 吴士林, 崔玉龙, 等. 区块链技术在轨道交通装备制造业应用方案研究[J]. 科技创新导报, 2021, 18(9): 99-103.
(Liu T, Wu S L, Cui Y L, et al. The application of blockchain technology in rail transit equipment manufacturing industry[J]. Science and Technology Innovation Herald, 2021, 18(9): 99-103.)

[3] 王凌, 邓瑾, 王圣尧. 分布式车间调度优化算法研究综述[J]. 控制与决策, 2016, 31(1): 1-11.
(Wang L, Deng J, Wang S Y. Survey on optimization algorithms for distributed shop scheduling[J]. Control and Decision, 2016, 31(1): 1-11.)

[4] Han X, Han Y Y, Chen Q D, et al. Distributed flow shop scheduling with sequence-dependent setup times using an improved iterated greedy algorithm[J]. Complex System Modeling and Simulation, 2021, 1(3): 198-217.

[5] Wang S H, Li X Y, Gao L, et al. A multi-disjunctive-graph model-based memetic algorithm for the distributed job shop scheduling problem[J]. Advanced Engineering Informatics, 2024, 60: 102401.

[6] Tian S C, Zhang C J, Fan J X, et al. A genetic algorithm with critical path-based variable neighborhood search for distributed assembly job shop scheduling problem[J]. Swarm and Evolutionary Computation, 2024, 85: 101485.

[7] Deng L B, Qiu Y X, Di Y Z, et al. A knowledge-driven memetic algorithm for distributed green flexible job shop scheduling considering the endurance of machines[J]. Applied Soft Computing, 2025, 170: 112697.

[8] Wang G C, Wang P, Zhang H G. A self-adaptive memetic algorithm for distributed job shop scheduling problem[J]. Mathematics, 2024, 12(5): 683.

[9] Wei G Y, Ye C M, Xu J N. Shared manufacturing-based distributed flexible job shop scheduling with supply-demand matching[J]. Computers & Industrial Engineering, 2024, 189: 109950.

[10] Zhao F Q, Li M J, Zhu N N, et al. Multi-objective fitness landscape-based estimation of distribution algorithm for distributed heterogeneous flexible job shop scheduling problem[J]. Applied Soft Computing, 2025, 171: 112780.

[11] Guo P, Shi H C, Wang Y, et al. Multi-objective scheduling of cloud-edge cooperation in distributed manufacturing via multi-agent deep reinforcement learning[J]. International Journal of Production Research, 2024: 1-25.

[12] 王艳红, 付威通, 张俊, 等. 基于改进近端策略优化算法的柔性作业车间调度[J]. 控制与决策, 2025, 40(6): 1883-1891.
(Wang Y H, Fu W T, Zhang J, et al. Flexible job-shop scheduling based on improved proximal policy optimization algorithm[J]. Control and Decision, 2025, 40(6): 1883-1891.)

[13] 孙爱红, 雷琦, 宋豫川, 等. 基于深度强化学习求解作业车间机器与 AGV 联合调度问题[J]. 控制与决策, 2024, 39(1): 253-262.
(Sun A H, Lei Q, Song Y C, et al. Deep reinforcement learning for solving the joint scheduling problem of machines and AGVs in job shop[J]. Control and Decision, 2024, 39(1): 253-262.)

[14] Luo S. Dynamic scheduling for flexible job shop with new job insertions by deep reinforcement learning[J]. Applied Soft Computing, 2020, 91: 106208.

[15] Lei Y, Deng Q W, Liao M Q, et al. Deep reinforcement learning for dynamic distributed job shop scheduling problem with transfers[J]. Expert Systems with Applications, 2024, 251: 123970.

[16] Wang R Q, Wang G, Sun J, et al. Flexible job shop scheduling via dual attention network-based reinforcement learning[J]. IEEE Transactions on Neural Networks and Learning Systems, 2024, 35(3): 3091-3102.

[17] Huang J P, Gao L, Li X Y. A hierarchical multi-action deep reinforcement learning method for dynamic distributed job-shop scheduling problem with job arrivals[J]. IEEE Transactions on Automation Science and Engineering, 2025, 22: 2501-2513.

[18] Liu R K, Piplani R, Toro C. Deep reinforcement learning for dynamic scheduling of a flexible job shop[J]. International Journal of Production Research, 2022, 60(13): 4049-4069.

[19] Huang J P, Gao L, Li X Y, et al. A cooperative hierarchical deep reinforcement learning based multi-agent method for distributed job shop scheduling problem with random job arrivals[J]. Computers & Industrial Engineering, 2023, 185: 109650.

[20] Gupta J K, Egorov M, Kochenderfer M. Cooperative multi-agent control using deep reinforcement learning[C]. Autonomous Agents and Multiagent Systems. Cham: Springer, 2017: 66-83.

[21] Schulman J, Wolski F, Dhariwal P, et al. Proximal policy optimization algorithms[J/OL]. 2017, arXiv: 1707.06347.

[22] Xie J, Li X Y, Gao L, et al. A hybrid genetic tabu search algorithm for distributed flexible job shop scheduling problems[J]. Journal of Manufacturing Systems, 2023, 71: 82-94.

作者简介

孟祥恒 (2001−), 男, 硕士生, 主要研究方向为智能决策优化、深度强化学习, E-mail: 2024200252@my.swjtu.edu.cn;

郭鹏 (1988−), 男, 副教授, 博士, 主要研究方向为智能装备运维、设备状态检测, E-mail: pengguo318@swjtu.edu.cn;

李嘉雯 (2002−), 女, 硕士生, 主要研究方向为运筹优化、生产调度, E-mail: lijiawen1016@my.swjtu.edu.cn;

史海超 (2000−), 男, 硕士生, 主要研究方向为智能决策优化、深度强化学习, E-mail: haichaoshi01@163.com;

张志瑶 (1993−), 女, 讲师, 博士, 主要研究方向为智能装备运维、设备状态监测, E-mail: zhiyaozhang@swjtu.edu.cn;

马永敬 (1987−), 男, 高级工程师, 硕士, 主要研究方向为工业大数据、智能制造, E-mail: mayongjing@cqsf.com;

孙轶杰 (1990−), 男, 工程师, 硕士, 主要研究方向为生产计划与控制, E-mail: sunyijie.sf@crrcgc.cc.