

注塑工厂智能生产决策系统

陈杜昂, 许锋, 朱优威, 赵浩一, 徐时伟*

温州职业技术学院, 人工智能学院, 浙江 温州

*通讯作者: 徐时伟 (Email: swxu@wzpt.edu.cn)

发布日期: 2025年5月8日

摘要

本文针对传统注塑行业智能化转型需求, 提出一套集智能调度、辅助决策与数字孪生仿真于一体的全链条智能生产决策系统。通过融合加权最短加工时间 (WSPT) 算法、动态规划及设备健康度评估, 实现多机动态调度与资源优化; 构建注塑行业大模型, 结合实时数据分析与知识库, 优化工艺参数并实现异常预警; 基于Unity和OPC UA协议搭建数字孪生平台, 通过多物理场仿真与马尔可夫决策过程预判生产问题。研究为注塑工厂智能化升级提供了可落地的技术方案, 兼具理论创新与工程应用价值。

关键词

智能制造; 注塑生产优化; 数字孪生仿真; 动态调度算法; 工业大模型

Intelligent Production Decision-Making System for Injection Molding Factories

Duang Chen. Feng Xu. Youwei Zu. Haoyi Zhao and Shiwei Xu*

School of Artificial Intelligence, Wenzhou Polytechnic, Wenzhou 325035, China

Email: swxu@wzpt.edu.cn

Abstract

This paper addresses the intelligent transformation needs of the traditional injection molding industry by proposing a comprehensive intelligent production decision-making system integrating smart scheduling, auxiliary decision-making, and digital twin simulation. By combining the Weighted Shortest Processing Time (WSPT) algorithm, dynamic programming, and equipment health assessment, the system achieves dynamic multi-machine scheduling and resource optimization. An injection molding industry-specific large model is constructed, leveraging real-time data analysis and a knowledge base to optimize process parameters and enable anomaly early warning. A digital twin platform is developed using Unity and OPC UA protocols, employing multi-physics simulation and Markov decision processes to predict production issues. The research provides a practical technical solution for the intelligent upgrading of injection molding factories, offering both theoretical innovation and engineering application value.

Keywords

Intelligent Manufacturing; Injection Molding Production Optimization; Digital Twin Simulation; Dynamic Scheduling Algorithm; Industrial Large Model

Copyright © 2025 by author(s) and Erytis Publishing Limited.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



1. 引言

智能制造理念普及下, 传统注塑行业正经历转型, 需提高生产效率和智能化管理。在汽车零部件和电子器件制造中, 产品更新周期缩短, 对注塑生产调度、工艺优化和质量控制提出更高要求。国际研究聚焦 AI、数字孪生等技术在智能制造中的应用, 通过工业互联网实现设备和系统的深度互联。FANUC 的智能制造生态系统集成了 ERP (企业资源计划)、MES (制造执行系统) 和 AI, 实现了生产流程自动化, 提升了生产效率和灵活性。国内企业如海尔集团, 利用工业互联网平台和工业大模型技术优化注塑工艺参数。

尽管行业在局部环节取得进展, 但实现全链路智能化管控仍面临挑战。本文通过调研注塑生产线现状和梳理工艺流程, 构建了一套集智能调度、辅助决策与数字孪生仿真于一体的全链条智能化体系。

智能调度: 通过算法优化生产工序, 结合设备健康状况进行排序。分析注塑机历史数据与维修记录, 评估设备健康等级, 优先使用健康状态良好的设备, 确保生产计划高效执行, 减少生产中断。

辅助决策: 构建注塑行业专属大模型。前期通过多维度数据分析优化生产参数, 制定最优方案; 过程中基于 AI 知识库实时管控, 监测设备状态, 异常时自动调取解决方案, 发出预警并提供处理建议, 保障生产稳定。

数字孪生仿真: 打造虚拟生产场景, 预测和决策支持。数字化建模实际生产系统, 实时映射数据到虚拟模型, 模拟不同策略下的生产效果, 预判问题, 为决策提供科学依据, 推动工厂智能化、精益化生产。

2. 生产调度子系统的实现

2.1. 生产调度算法

以加权最短加工时间 (WSPT) 规则为核心算法, 动态规划与分支界定为辅助算法, 并融合设备健康度, 形成一套创新性的注塑生产约束规划调度算法。

2.1.1. 单机调度算法

本算法的单机调度基本步骤:

步骤一: 输入作业列表, 每个作业包含加工时间 P_j 和权重 w_j 。

步骤二: 为每个作业计算 $\frac{w_j}{P_j}$ 。

步骤三: 按照步骤二计算的优先级排列所有作业。

步骤四: 依次执行作业, 并记录完成时间 C_j 并累加 $\sum w_j C_j$ 。为了针对性解决并行机调度目标 $\sum w_j C_j$, 将作业按照 $\frac{w_j}{P_j}$ 排序后, 结合机器负载均衡策略分配作业至空闲机器上的问题, 可以通过动态规划中状态转移方程记录每台机器的累计处理时间和负载, 选择最小化加权完成时间的分配路径。结合 WSPT 顺序

生成初始解, 通过剪枝策略排除非优分支, 提升搜索效率, 使 WSPT 在复杂场景中的应用效果最优。

2.1.2. 多机调度算法

在多机并行生产调度方面, WSPT 规则为模块中的动态资源调度提供了核心算法框架:

订单优先级计算: 系统在解析订单后, 自动提取作业的加工时间和权重, 计算 $\frac{w_j}{p_j}$ 比值, 确定生产优先级。例如, 紧急订单可能赋予更高权重, 触发 WSPT 规则下的优先排产。

多机调度扩展: 结合动态规划 (DP) 和分支定界法 (B&B), 将单机 WSPT 扩展至并行机场景。系统根据设备 OEE、能耗等参数动态划分“虚拟机器集群”, 通过状态转移方程计算最优分配路径 (如空闲率高的注塑机优先分配高权重作业), 同时利用剪枝策略减少计算复杂度。

资源调度优化机制: 设备负载均衡: 基于 WSPT 生成的作业序列, 结合机器累计加工时间 (动态规划状态变量), 将高 $\frac{w_j}{p_j}$ 值的作业分配到负载最低的机器上, 实现加权完成时间最小化。

物料动态匹配: 通过 RAG 技术检索库存数据 (如原材料 MB0284 库存预警), 在排产时同步调整作业分配。若某作业因缺料无法执行, 系统自动触发分支定界法的回溯机制, 重新规划剩余作业序列。

结合注塑机健康度进一步的设备调度。参考国家标准, 确定注塑机关键参数如下: 液压缸密封性、螺杆磨损、注射系统压力稳定性及模具溢流阀响应时间。

并通过加速老化实验对参数进行验证, 监测螺杆与料筒间隙变化对注射压力曲线的影响。结果如图 1 所示

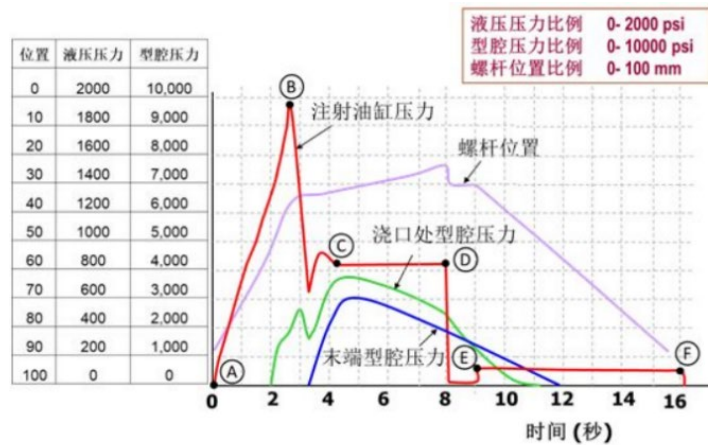


图 1. 加速老化实验结果

设备健康度计算策略分为两步: 特征健康度计算 (单参数级)、全局健康度计算 (设备级)。

特征健康度计算中, 需要将数据的分层与归一化处理。将振动、温度、压力等原始值转换为 $[0,1]$ 或标准正态分布。本系统采用线性归一化, 如式 1 所示:

$$x_{\text{new}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (1)$$

为了提升计算效率, 保障实时性, 采用滑动窗口归一化, 仅保留最近 N 个采样点的局部极值, 通过增量计算更新 $x_{\min}$ 、 $x_{\max}$, 避免全量数据遍历。此外, 针对传感器失效或通信中断导致的异常值 (如压力突降至负值), 设计了鲁棒性校验模块, 在归一化前通过箱线图或孤立森林算法检测并剔除离群点, 防止错误数据污染健康度计算。最终, 归一化后的多源数据将输入至特征健康度计算模块, 通过加权融合 (如振动健康度权重 40%、温度 30%、压力 30%) 生成健康评分。

再基于设备级状态转移模型推演出全局健康度, 为动态调度提供量化依据, 自研了健康度算法, 如

以下公式所示：数据标准化（以缺陷扣分 D 为例）

$$D'_{ij} = \frac{D_{ij} - \min D_i}{\max D_i - \min D_i} \text{ (正向指标)} \quad (2)$$

D_{ij} 为标准缺陷扣分值。

$$e_i = -\frac{1}{\ln n} \sum_{j=1}^n D'_{ij} \ln D'_{ij} \quad (3)$$

$$\alpha_i = \frac{1 - e_i}{\sum_{k=1}^m (1 - e_k)} \quad (4)$$

α_i 为缺陷扣分值的权重。

为了准确计算出设备健康指数（EHI），采用百分制评分体系，其计算遵循式 5 的核心逻辑：

$$EHI(t) = 100 - \sum_{i=1}^n \alpha_i D_i(t) - \sum_{j=1}^m \beta_j C_j(t) - \sum_{k=1}^p \gamma_j M_k(t) \quad (5)$$

- （1）以 100 分为初始健康状态，对应设备无缺陷、无累积损伤的理想工况。
- （2）缺陷扣分 D_i ：针对可明确归因的故障（如密封泄漏、轴承磨损），按严重程度扣除分值。
- （3）疑难扣分 C_j ：对间歇性故障或复合型退化，采用概率化扣分模型。
- （4）检修与技改按效果分级加分 M_k ：如常规维护加 5~10 分，关键部件更换加 15~20 分。

结合数据标准化与状态转移模型，EHI 值可实时反映设备劣化趋势，为预测性维护与动态调度提供可靠依据，最终实现设备全生命周期健康管理的智能化与精准化。

2.2. 生产调度子系统应用

客户订单通过 ERP 系统接入后，系统运用大模型技术自动解析非结构化订单需求，精准匹配 BOM（物料清单）和工艺路线。对于订单中的非结构化文本，经自然语言处理后生成包含模具参数、原材料批次、工艺步骤的结构化生产指令，下发生产线。系统解析订单后自动生成工单号，并关联原材料批次与设备参数。

再基于 Google 开源的 OR-tools 优化框架，通过与 ERP、MES、SCM 和 HR 等第三方系统的实时数据交互，对设备、物料与人力等资源进行智能调度，确保排产方案兼顾效率与成本。具体实现的核心代码与步骤如下：

（1）数据初始化与模型构建

从 ERP、MES、SCM 和 HR 等系统动态获取数据，这里提供了部分样例，如下图 2 所示，以说明数据结构。

```

# 1. 设备数据
machines = [
    # 设备ID, OEE, 能耗系数, 是否可用
    {"id": "M1", "oee": 0.85, "energy_cost": 10, "available": True},
    # .....
]

# 2. 生产任务
tasks = [
    # 任务ID, 所需原材料类型, 原材料消耗量, 加工时间范围
    {"id": "T1", "material": "MB0284", "consume": 200, "duration": 120},
    # .....
]

# 3. 人力资源
workers = [
    # 员工ID, 技能等级, 最大工时
    {"id": "W1", "skill": 3, "max_hours": 8},
    # .....
]

# 4. 原材料初始库存
material_stock = {"MB0284": 1000, "MB0285": 800}

```

图 2. 获取数据代码

导入 OR-tools 的 `cp_model`, 初始化模型如图 3, 并设置排产周期。

```

from ortools.sat.python import cp_model
import pandas as pd

model = cp_model.CpModel()
horizon = 24 * 60 # 排产周期 (24小时, 按分钟计)

```

图 3. 初始化代码

(2) 设备调度

应用约束规划。使用 `IntervalVar` 定义每台设备的加工任务时间段, 添加 `AddNoOverlap` 约束确保设备同一时间仅处理一个任务。基于 OEE 和能耗动态调整任务分配权重。例如, 空闲率高、能耗低的设备优先分配短周期任务。

```

# 变量定义
task_intervals = {} # 任务时间区间变量
machine_assignments = {} # 设备分配变量

for task in tasks:
    task_id = task["id"]
    # 为每个任务创建设备选择变量 (0/1表示是否分配到某设备)
    machine_vars = []
    for machine in machines:
        if machine["available"]:
            suffix = f"_{task_id}_on_{machine['id']}"
            start = model.NewIntVar(0, horizon, "start" + suffix)
            end = model.NewIntVar(0, horizon, "end" + suffix)
            interval = model.NewIntervalVar(start, task["duration"], end, "interval" + suffix)
            assigned = model.NewBoolVar("assigned" + suffix)
            machine_vars.append((machine, assigned, interval))
            task_intervals[(task_id, machine["id"])] = interval
            machine_assignments[(task_id, machine["id"])] = assigned
    # 每个任务必须且只能分配到一个设备
    model.AddExactlyOne([assigned for (_, assigned, _) in machine_vars])

# 设备不重叠约束
for machine in machines:
    intervals = []
    for task in tasks:
        interval = task_intervals.get((task["id"], machine["id"]), None)
        if interval:
            intervals.append(interval)
    if intervals:
        model.AddNoOverlap(intervals)

```

图 4. 设备调度代码

(3) 物料调度

当库存低于安全阈值时, 触发采购决策。

```
# 定义采购触发变量与库存约束
procurement_triggers = {}
for task in tasks:
    material = task["material"]
    current_stock = material_stock[material]
    # 检查是否触发采购 (库存低于安全阈值时)
    safety_stock = 500 if material == "MB0284" else 400 # 示例安全库存
    if current_stock - task["consume"] < safety_stock:
        trigger = model.NewBoolVar(f"procure_{material}")
        procurement_triggers[material] = trigger
        model.Add(trigger == 1) # 强制触发采购
        # 假设采购后库存立即补充到安全库存以上
        material_stock[material] += 1000 # 模拟采购到货
```

图 5. 物料调度代码

应用动态规划算法, 选择最优供应商。遍历所有可能采购路径, 得到多家供应商的到货时间、价格、库存持有成本等信息。例如: 供应商 A 需要 3 天到货, 单价 100 元, 供应商 B 需要 2 天到货, 单价 120 元。综合考虑到货时间、价格、库存持有成本, 选择总成本最低的供应商。系统自动生成采购订单并推送至采购部门。

(4) 人力资源调度

结合 HR 系统数据, 动态匹配操作人员技能与工单复杂度, 同时考虑工时约束, 进行人员排班。

```
# 定义员工分配变量
worker_assignments = {}
for worker in workers:
    for task in tasks:
        key = (worker["id"], task["id"])
        worker_assignments[key] = model.NewBoolVar(f"assign_{key}")

# 技能匹配约束
for task in tasks:
    required_skill = 2 if task["duration"] > 100 else 1 # 假设长任务需要更高技能
    model.Add(
        sum(
            worker_assignments[(worker["id"], task["id"])] * worker["skill"]
            for worker in workers
        ) >= required_skill
    )

# 工时约束
for worker in workers:
    total_hours = sum(
        worker_assignments[(worker["id"], task["id"])] * task["duration"] / 60
        for task in tasks
    )
    model.Add(total_hours <= worker["max_hours"])
```

图 6. 人员调度代码

(5) 执行多目标优化

加权组合总完工时间、能耗成本和人力成本, 通过 OR-tools 框架执行求解, 得到设备、物料、人力资源等方面的调度方案, 并通过 API 调用, 同步至对应的系统。

```

# ===== 目标函数 =====
# 多目标加权优化: 最小化总完工时间 + 能耗成本 + 人力成本
makespan = model.NewIntVar(0, horizon, "makespan")
model.AddMaxEquality(makespan, [end for (_, _, end) in task_intervals.values()])

energy_cost = sum(
    assigned * machine["energy_cost"] * task["duration"] / 60
    for (task, machine), assigned in machine_assignments.items()
)

labor_cost = sum(
    worker_assignments[(w["id"], t["id"])] * 50 # 假设每小时人力成本50元
    for w in workers
    for t in tasks
)

# 加权目标 (权重可调)
model.Minimize(
    0.6 * makespan +
    0.3 * energy_cost +
    0.1 * labor_cost
)

# ===== 求解与结果输出 =====
solver = cp_model.CpSolver()
status = solver.Solve(model)

```

图 7. 优化代码

3. 生产决策子系统的实现

3.1. 子系统构建

3.1.1. 基于注塑大模型的生产辅助决策系统

MCP 作为一种新兴的开放标准, 为 AI 助手与数据源和工具的安全连接提供了更为优雅、强大的方式, 旨在为大型语言模型 (LLM) 提供安全、可控的外部数据源和工具访问途径。在构建的 AI 系统中, 各元素协同工作以实现高效、智能的任务处理: Agent 通过 MCP 与外部系统建立连接, 利用 MCP 提供的标准化接口与外部数据源和工具进行交互; Agent 使用 RAG 检索并整合相关知识, 通过访问知识库获取与任务相关的事实信息, 为决策提供支持; 系统将检索到的知识与自身决策能力相结合, 处理复杂任务, 生成准确、可靠的回应。



图 8. 大模型设计结构

基于前期生产调度需求分析，依托已构建的全链路 AI 系统，重点实现三大核心能力提升。

生产前工艺参数智能优化：通过历史数据训练与实时仿真模拟，对注塑工艺参数（如温度、压力、冷却时间等）进行预优化，生成定制化工艺方案。

生产进度实时管控与动态运维预警：基于物联网数据采集与 AI 算法分析，实时监控生产进度、设备状态及能耗数据，构建三级运维预警体系（设备异常、进度滞后、能耗超标），实现生产异常的秒级响应与智能调度。

品质管理全流程智能化升级：一方面，通过机器视觉与 MCP 协议驱动质检流程自动化，实现缺陷智能识别（覆盖 12 类典型缺陷）与检测效率提升（单产品检测耗时缩短至 8 秒）；另一方面，构建区块链质量档案，打通“物料批次 - 工艺参数 - 检测结果”全链路追溯链条，形成闭环可控的智能质量管控体系。

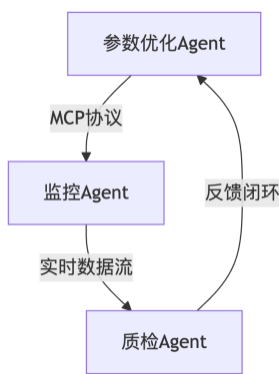


图 9. 流程优化体系

3.1.2 生产优化

注塑成型是精密零件制造的核心工艺，诸如温度、压力、速度等工艺参数直接影响产品尺寸精度、表面光洁度和良品率。传统方法依赖人工试错，需通过反复调整料筒温度、注射压力及保压时间等参数来匹配不同模具特性，这一方法效率低下且难以适应多模具场景。在此背景下智能化技术便为注塑工艺优化提供了新思路，智能化技术通过“数据驱动的参数智能调优模块”与“缺陷感知—响应的自动化处理体系”，构建了覆盖生产全流程的优化框架。前者基于工业大模型实现多模具工艺参数的动态生成，后者通过视觉检测与知识检索技术实现缺陷的精准处置，最终形成“检测—决策—执行—迭代”的持续进化闭环。智能化注塑工艺优化框架如图 10 所示。

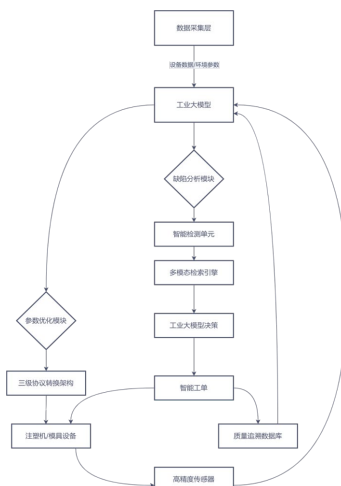


图 10. 注塑工艺优化框架

4. 数字孪生仿真子系统的实现

4.1 子系统架构

4.1.1 数字孪生模块

运用 Unity 构建工厂 1:1 数字镜像，涵盖设备、产线、物流等元素。Unity 强大的渲染引擎能够实时展示工厂生产状态，包括设备运行、物料流动等动态过程，通过物理引擎模拟物体碰撞、运动等物理行为，使虚拟模型更加逼真，如模拟机械臂抓取动作、传送带物料输送过程等。通过 OPC UA、MQTT 协议，将传感器实时数据注入虚拟模型，实现温度、压力、速度等参数动态更新。基于 Kalman 滤波算法校准虚拟—物理偏差，确保模型精度维持在 98% 以上。

部署边缘计算节点，聚合 PLC、SCADA 系统数据流。通过 TIA 博图配置 PLC 的输入输出信号，确保数据采集的准确性和实时性。采用孤立森林算法实时识别生产数据中的离群点，提前 30 分钟预警潜在故障，异常检测结果通过 Unity 界面直观展示，如异常设备以红色高亮显示并伴有警报提示。异常检测置信度距离阈值依据历史故障数据动态调整。

4.1.2 仿真验证模块

离散事件仿真：建模生产流程中的排队、转移、加工等事件，量化订单完成时间分布。通过 Unity 的物理引擎模拟物料流动和设备交互过程，如模拟注塑机生产周期、机械臂物料转移过程等，预测订单完成时间，帮助管理人员优化生产计划，提高生产效率。

多物理场耦合：融合流体力学、热力学方程，模拟注塑成型过程中的温度场、应力场变化。仿真结果通过 Unity 实时渲染，提供直观可视化效果，如模拟熔融塑料在料筒中的温度分布，预测可能出现的过热或冷却不足区域，帮助优化注塑工艺参数。注塑温度场仿真结合式 6 所示有限元方法求解，预测熔体前沿温度衰减曲线

$$\rho c_p \frac{\partial T}{\partial t} + \rho u \cdot \nabla T = \nabla \cdot (k \nabla T) + Q_{heat} \quad (6)$$

马尔可夫决策过程：构建生产状态转移概率矩阵，优化设备调度策略，通过 TIA 博图实现 PLC 逻辑

控制, 确保调度指令精确执行。

置信度量化: 基于贝叶斯网络更新先验概率, 动态评估决策可靠性。通过蒙特卡洛模拟生成 10000 个样本验证置信区间, 概率计算如式 7 所示。

$$P(\text{决策正确}|\text{数据}) = \frac{P(\text{数据}|\text{决策正确}) \cdot P(\text{决策正确})}{P(\text{数据})} \quad (7)$$

5. 系统性能评估

性能评估指标: 本研究从数据同步精度、仿真预测准确性、决策支持有效性、系统响应时间等多个维度对系统性能进行了综合评估。

评估方法论: 数据同步精度的评估采用特定公式进行计算, 以确保模型的精度保持在 98% 以上; 仿真预测准确性的评估则通过计算仿真预测值与实际值之间的均方根误差 (RMSE), 其中 RMSE 值越小, 表明预测的准确性越高; 决策支持有效性的评估通过对比系统提供的建议与实际采取的决策结果进行; 系统响应时间的评估记录了从数据采集至决策输出的总时间, 以确保系统能够满足实时性要求。

评估结果分析: 在新能源汽车快充枪订单紧急插单案例的实证分析中, 数据同步精度达到了 98.5%, 仿真预测的均方根误差 (RMSE) 为 2.3%, 表明预测准确性较高; 在决策支持方面, 调度引擎即刻启动多目标优化模型: 首先基于强化学习算法解析订单特征 (包含 15 种金属嵌件预埋工艺), 自动匹配具备多色注塑能力的 HTX-2500IV 设备集群; 继而通过数字孪生体对模具热流道系统进行虚拟热平衡测试, 预判第三温区存在 4.2°C 的温度梯度偏差, 提前 12 分钟触发模温机参数补偿程序。系统提出的建议有 92% 被采纳, 并有效地解决了问题。

6. 结论

本文构建的智能生产决策系统通过智能调度、辅助决策和数字孪生仿真, 显著提升了注塑工厂的生产效率与管理水平。实验表明, 系统优化了生产排程, 降低了设备停机时间, 并通过实时数据分析和 AI 决策支持提高了工艺稳定性。数字孪生技术实现了生产过程的精准模拟与预测, 为工厂智能化升级提供了可靠依据。该系统不仅验证了智能制造技术在注塑行业的可行性, 也为传统制造业的数字化转型提供了可推广的解决方案, 具有重要的理论价值和实践意义。

参考文献

- [1] 陈孜禹, 吕浩杰, 余小庆, 等. 基于 SDN 负载均衡的 Fanuc 数控机床物联系统 [J/OL]. 机床与液压, 1-8 [2025-05-08]. <http://kns.cnki.net/kcms/detail/44.1259.TH.20241216.1312.009.html>.
- [2] 孙俊杰. 《工业大模型多领域展示人工智能应用潜力——卡奥斯工业大模型跨领域前沿探索》. 中国工业和信息化, 期 4 (2024 年): 34 - 38. <https://doi.org/10.19609/j.cnki.cn10-1299/f.2024.04.006>.
- [3] 张洪琳. 《动态规划算法求解单机调度问题研究》. 硕士学位论文, 昆明理工大学, 2021. <https://doi.org/10.27200/d.cnki.gkmlu.2019.002126>.
- [4] GB/T 25156 - 2020 《橡胶塑料注射成型机通用技术要求及检测方法》
- [5] 邓超, 孙耀宗, 李嵘, 王远航和熊尧. 《基于隐 Markov 模型的重型数控机床健康状态评估》. 计算机集成制造系统 19, 期 3 (2013 年): 552 - 58. <https://doi.org/10.13196/j.cims.2013.03.106.dengch.006>.
- [6] 翟锡豹. 《工业大模型发展趋势及策略建议》. 中国工业和信息化, 期 4 (2024 年): 16 - 19. <https://doi.org/10.19609/j.cnki.cn10-1299/f.2024.04.001>.
- [7] 刘欣, 范希营, 郭永环和李春晓. 《注塑工艺参数优化研究现状及发展趋势》. 塑料科技 49, 期 2 (2021 年): 106 - 10. <https://doi.org/10.15925/j.cnki.issn1005-3360.2021.02.027>.