

面向快速响应的赛汝生产系统构建模型与方法

湛荣鑫¹ 李冬妮¹ 马涛² 李俊杰² 吴延昭² 殷勇³

摘要 当前市场环境具有多品种、变批量、产品生命周期短的波动特点, 赛汝生产系统 (Seru production system, SPS) 是一种目前广泛应用于电子制造行业的新型生产方式, 其具有优良的重构性和响应能力, 适合于应对波动市场环境. 本文提出了一个两阶段的赛汝生产系统构建问题模型, 两个阶段分别是 Seru 构建和 Seru 调度, 并证明了这两个问题模型均是 NP 难的, 结合模型分析给出了相应的精确/近似算法. 实验结果表明, 在波动市场环境下按照本文模型与方法构建出的赛汝生产系统其工人利用率始终保持在较高水平, 系统具有较强的重构性能和响应能力.

关键词 赛汝生产系统, 响应能力, 重构性, 波动市场

引用格式 湛荣鑫, 李冬妮, 马涛, 李俊杰, 吴延昭, 殷勇. 面向快速响应的赛汝生产系统构建模型与方法. 自动化学报, 2022, 48(12): 2922–2930

DOI 10.16383/j.aas.c190731

Configuration Model and Approach of a Seru Production System for Quick Response

ZHAN Rong-Xin¹ LI Dong-Ni¹ MA Tao² LI Jun-Jie² WU Yan-Zhao² YIN Yong³

Abstract Variable product types, fluctuating production volumes, and short product life cycles are the main characteristics of the current market environment. Seru production system, which has excellent reconfigurability and responsiveness, has been widely used in electronic manufacturing industry. A two-stage model of configuring a Seru production system is proposed in this paper. The complexity and property of the model is analyzed and related exact or approximation algorithms are given. The computational result shows that the configured Seru production systems have high level performance of utilization and reveal strong reconfigurability and responsiveness.

Key words Seru production system (SPS), responsiveness, reconfigurability, volatile markets

Citation Zhan Rong-Xin, Li Dong-Ni, Ma Tao, Li Jun-Jie, Wu Yan-Zhao, Yin Yong. Configuration model and approach of a Seru production system for quick response. *Acta Automatica Sinica*, 2022, 48(12): 2922–2930

自 1980 年代以来, 电子信息产业蓬勃发展, 产品生命周期显著缩短, 电子产品目前的平均生命周期仅为 6 个月^[1]. 因此, 面对更为多样化的产品种类需求、更短的产品生命周期要求以及由此带来的市场环境波动性和不确定性^[2–4], 传统的生产系统如流水线或丰田制造系统难以获得很好的表现, 亟需一种具有优异快速响应能力的新型生产系统^[5]. Seru 是日式生产单元, 由 Seru 构成的赛汝生产系统 (Seru production system, SPS) 作为一种新兴的生产模

式广泛应用于日本的电子制造行业. 与传统生产系统相比, SPS 最突出的优势在于其应对不确定市场需求的快速响应能力, 而之所以可以实现快速响应, 主要原因就是 Seru 在物理结构上是可重构而不是固定的, 并由此可以带来诸如降低成本、缩短生产时间、缩短生产准备周期、缩短劳动时间等方面的显著收益^[2, 6–14].

目前, 已有很多学者围绕 SPS 构建问题展开研究, 这类研究大多将 SPS 构建分成了两个阶段, 分别是 Seru 构建和 Seru 调度. Seru 构建大多分为两类模型, 一类是研究如何将现有流水线拆分为若干个 Seru; 另一类是给定若干数量的 Seru, 主要考虑工人的分配问题^[15]. Seru 调度决策如何将产品批次分配到已构建好的 Seru 中去^[16–21]. 例如, Liu 等^[22]研究了在 SPS 中多能工的培训和分配问题, 建立了最小化总培训成本和平衡多能工之间劳动时间的多目标问题模型. Liu 等^[23]考虑了优化经济和环境指标的 SPS 构建模型. Lian 等^[24]提出了考虑 Seru 内部和 Seru 间多能工劳动力平衡分配的 SPS 构建模型.

收稿日期 2019-10-22 录用日期 2020-04-06

Manuscript received October 22, 2019; accepted April 6, 2020

内蒙古自治区重大基础研究开放课题 (GZ2018KF001), 国家自然科学基金 (61763046) 资助

Supported by Major Basic Research and Open Project of the Inner Mongolia Autonomous Region (GZ2018KF001) and National Natural Science Foundation of China (61763046)

本文责任编辑 张俊

Recommended by Associate Editor ZHANG Jun

1. 北京理工大学计算机学院智能信息技术北京市重点实验室 北京 100081 中国 2. 内蒙古第一机械集团有限公司 包头 014000 中国 3. 同志社大学商学院 京都 201101 日本

1. Beijing Laboratory of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, Beijing 100081, China 2. Inner Mongolia First Machinery Group Co., Ltd., Baotou 014000, China 3. Graduate School of Business, Doshisha University, Kyoto 201101, Japan

已有研究大多有一个共同的假设: 一旦 Seru 构建完成, 就不再会发生变化. 然而, SPS 应该具备随着生产需求变化动态进行 Seru 构建和拆除的能力. 此外, 相关研究还大多假定工人都是全能工 (即具有一个或多个产品所需要的所有工序的加工能力). 实际中, 工人很难全部被培训为全能工且存在技能水平上的差异. 另一方面, 纯粹由全能工构成的 SPS 也不一定是最优的^[25-26]. 因此, 在考虑工人具有不同的技能组合和技能水平的情况下, 如何组织工人构建 SPS 是一个重要的研究问题.

基于以上考虑, 本文考虑如下的 SPS 场景并设计了两阶段的 SPS 构建模型, 分为 Seru 构建 (Seru configuration, SC) 和 Seru 调度 (Seru scheduling, SS). 当新订单到来时, 有恰当数量的 Seru 被构建并完成分配给各自的生产任务. 当每个 Seru 完成其生产任务时立刻拆解. 因此, 整个 SPS 可以动态调整其自身结构, 调整的对象包括 Seru 的数目、分配给 Seru 的工人以及每个工人负责的工序等. 具体而言, 两阶段描述如下. 1) Seru 构建阶段: 在同一生产周期内, 所有的工人具有同样的初始剩余工作时间. 工人具有不同的技能组合以及在不同的工序上具有不同的技能水平, 不同的工人在相同的工序上也具有不同的技能水平. 首先在不超出每个工人工作时间上限的约束下, 为订单找出一个可行的工人-工序分配方案, 并在依照这个分配方案, 考虑每个 Seru 工人之间的配合从而构建 Seru. 在本阶段提出了工人-工序映射模型和优化 Seru 内部工人工作时间平衡的工人-Seru 分配模型; 2) Seru 调度阶段: 一个工人在 Seru 构建阶段中可能会被分配参与多个 Seru 的生产任务, 而任一工人同一时刻只能参与一个 Seru 的生产, 因此 Seru 间存在调度问题, 基于此提出了最小化 SPS 最大完工时间 (Makespan) 的 Seru 调度模型.

本文的主要贡献如下: 1) 考虑了工人并非全部是全能工的 SPS 构建问题, 即工人具有不同的技能组合和技能水平; 2) 本文提出的 SPS 模型是动态可重构的, 考虑了 Seru 从构建到拆解的全生命周期. 本文其余部分组织如下: 第 1 节介绍了问题模型和数学模型, 第 2 节给出了每个阶段的模型性质分析、相应的精确/近似算法以及算法分析, 第 3 节介绍了实验设计、实验结果以及分析, 第 4 节给出了本文结论.

1 问题模型

本文抽象出的问题模型包含两个阶段: Seru 构建和 Seru 调度, 基于此两阶段问题抽象出问题

模型.

1.1 问题假设

本文研究的 SPS 构建问题假设描述如下:

1) SPS 内工人总数是确定的, 所有的工人都是多能工, 且存在技能组合和技能水平上的差异. 所有工人在同一生产周期内具有同样的初始剩余工作时间.

2) 一个订单中只包含一个产品类型, 订单的信息 (产品类型和需求) 只有在到来时才能被获知.

3) 一个 Seru 只生产一种产品类型, 一种产品类型可以被 SPS 中的多个 Seru 生产.

4) 一个工人可以被分配到多个 Seru 中去.

1.2 符号列表

与本文问题相关的符号变量定义如下所示.

1) 索引

w 工人索引 ($w = 1, \dots, W$).

i 工序索引 ($i = 1, \dots, I$).

c 产品类型索引 ($c = 1, \dots, C$).

d 订单索引 ($d = 1, \dots, D$).

s Seru 索引 ($s = 1, \dots, S$).

p 可供构建 Seru 的场地索引 ($p = 1, \dots, P$).

2) 系统变量

T_d^w 当订单 d 到来时, 工人 w 的剩余工作时间.

t_w^i 工人 w 加工工序 i 时的单位加工时间.

O_w 工人 w 能够加工的工序集合.

$$x_w^i = \begin{cases} 1, & \text{工人 } w \text{ 可以加工工序 } i \\ 0, & \text{其他} \end{cases}$$

X_d^w 工人 w 在参与生产订单 d 时参与工序种类上限.

f_{w_1, w_2, i_1, i_2}^d 若工人 w_1 和 w_2 加工同一个订单 d 的某两道相邻工序 i_1 和 i_2 , f_{w_1, w_2, i_1, i_2}^d 是这两道工序上这两个工人单位加工时间之差的绝对值. 如果某工人加工相邻的多道工序, 则将该工人在这些工序上的时间进行累加并等同于同一工序单位时间用于本变量计算. $f_{w_1, w_2, i_1, i_2}^d = 0$ 表示工人 w_1 和 w_2 加工同一个订单 d 的某两道相邻工序 i_1 和 i_2 时, 相邻工序 i_1 和 i_2 之间是平衡的, 没有等待或者空闲时间产生.

t_d 订单 d 的到来时间.

n_d 订单 d 的需求量.

\mathcal{I}_d 生产订单 d 的所需工序集合.

\mathcal{W}_d 订单 d 到来时, 当前空闲可以参与加工的工人集合.

\mathcal{S}_d 完成订单 d 所构建的 Seru 集合.

pt_s Seru s 从构建到拆解的存在时间.

ct_s Seru s 完成生产任务拆解的时刻.

3) 决策变量

n_d^{wi} 工人 w 生产订单 d 参与工序 i 加工任务数量, $n_d^{wi} \in \mathbb{N}$.

n_{w_1, w_2, i_1, i_2}^d 工人 w_1 和 w_2 生产订单 d 共同加工的产品数量, 且工人 w_1 参与的是工序 i_1 生产任务, 工人 w_2 参与的是工序 i_2 生产任务, 工序 i_1 与工序 i_2 是相邻工序.

kt_s^p Seru s 在场地 p 的构建时间.

1.3 图表模型及约束条件

本节对 SC 问题和 SS 问题分别建模并给出约束条件.

1) 对 SC 问题的工人-工序映射步骤建模, 这是一个约束满足问题, 其模型描述如下.

是否存在一种分配方案 n_d^{wi} , 能够满足如下约束:

$$\sum_{w \in W_d} x_w^i n_d^{wi} = \sum_{i \in I_d} n_d, \quad \forall i \quad (1)$$

$$\sum_{i \in I_d} n_d^{wi} \times t_w^i \leq T_d^w, \quad \forall w \quad (2)$$

$$\sum_{n_d^{wi} > 0} 1 \leq X_d^w, \quad \forall w \quad (3)$$

其中, 式 (1) 表示对于任一订单 d , 其所需的任一工序 i 的数量应与该 SPS 中所有能加工该工序的工人实际执行工序 i 的数量相等; 式 (2) 确保分配给任一工人的工作量不会超过该工人初始剩余工作时间; 式 (3) 表示参与任一订单 d 生产的任一工人 w 所参与加工的工序种类不能超过限制 X_d^w .

2) 对 SC 问题的工人-Seru 分配步骤建模, 其模型描述如下.

定义 E_d 为加工某一产品类型包含所有可能的加工路径的图, 并结合费用流图的基本定义, 为 E_d 在第一道工序前设置虚拟始点, 在最后一道工序后设置虚拟终点. 如图 1 所示, 订单 1 需要 2 道工序, 分别是工序 1 和工序 2, 工人 1 和工人 2 能够加工工序 1, 工人 3 和工人 4 能够加工工序 2. 则所有代表加工路径的虚线、所有工人节点、始点和终点共同构成 E_1 .

对于任一订单 d :

$$\min \sum_{(w_1, w_2) \in E_d} f_{w_1, w_2, i_1, i_2}^d \times n_{w_1, w_2, i_1, i_2}^d \quad (4)$$

需满足如下约束:

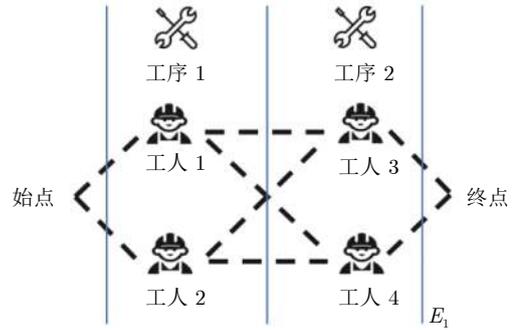


图 1 订单 1 的潜在加工路径 E_1

Fig.1 Potential processing paths E_1 for order 1

$$\sum_w (n_{w, w_1, i_1, i_2}^d - n_{w_1, w, i_1, i_2}^d) = -n_d, \quad \text{若 } w_1 \text{ 为始点} \quad (5)$$

$$\sum_w (n_{w_2, w, i_1, i_2}^d - n_{w, w_2, i_1, i_2}^d) = n_d, \quad \text{若 } w_2 \text{ 为终点} \quad (6)$$

$$n_{w_1, w_2, i_1, i_2}^d \leq n_d^{w_1 i_1}, \quad \text{若工人 } w_1 \text{ 加工工序 } i_1 \quad (7)$$

$$n_{w_1, w_2, i_1, i_2}^d \leq n_d^{w_2 i_2}, \quad \text{若工人 } w_2 \text{ 加工工序 } i_2 \quad (8)$$

$$n_{w_1, w_2, i_1, i_2}^d \geq 0 \quad (9)$$

其中, 式 (4) 为该阶段的目标函数, 表示通过选择不同的工人构建 Seru 从而最小化订单 d 的所有两两相邻工序之间的单位加工时间之差的绝对值; 式 (5) 和式 (6) 表示为订单 d 所构建 SPS 的生产量应与订单 d 的需求量一致; 式 (7) 和式 (8) 表示工人 w_1 和 w_2 共同生产订单 d 且工人 w_1 参与的是工序 i_1 生产任务, 工人 w_2 参与的是工序 i_2 生产任务, 工序 i_1 与工序 i_2 是相邻工序, 此时他们的共同生产任务量不能超过第一阶段时所决策出的生产任务量 $n_d^{w_1 i_1}$ 和 $n_d^{w_2 i_2}$; 式 (9) 表示产出量必须大于等于 0.

对 SS 问题建模, 其模型描述为

$$\min \max_{1 \leq s \leq S} ct_s \quad (10)$$

需满足如下约束:

$$ct_s = \min_{1 \leq p \leq P} kt_s^p + pt_s, \quad \forall 1 \leq s \leq S \quad (11)$$

$$kt_s^p \geq ct_{s'}, \quad \forall 1 \leq s, s' \leq S, \text{ 若 } s' \text{ 已构建, } s \text{ 未构建, 且 } s, s' \text{ 存在工人占用} \quad (12)$$

$$kt_s^p \geq kt_{s'}^p + pt_{s'}, \quad \forall 1 \leq s, s' \leq S, \forall 1 \leq p \leq P, \text{ 若 } s' \text{ 已构建, } s \text{ 未构建} \quad (13)$$

其中, 式 (10) 为该阶段的目标函数, 表示最小化该 SPS 中 Seru 的最大完工时间; 式 (11) 计算的是任

一 Seru s 完成其生产任务的最早时刻; 式 (12) 保证需要同一工人参与生产的两个不同 Seru 在同一时刻不能同时存在; 式 (13) 保证在任一时刻, 一块场地上只能存在一个 Seru.

2 模型分析及算法

本节给出 SC 问题和 SS 问题相关模型性质分析及算法.

对于 SC 问题来说, 其中涉及到工人-工序映射建立和工人-Seru 分配两个步骤. 本节对于这两个步骤分别进行了模型性质分析并给出了精确算法来求解.

定理 1. 工人-工序映射问题是 NP 难的.

证明. 本文将经典的 NP 难问题——划分问题 (Partition problem, PP)^[27] 归约到建立工人-工序映射问题, 从而证明建立工人-工序映射问题也是 NP 难问题.

划分问题描述如下: 有一个有限集合 U , 该集合中每个元素 u 有一个权重 $h(u)$, 决策目标是希望找到一个 U 的子集 U' 使得 $\sum_{u \in U'} h(u) = \sum_{u \in U \setminus U'} h(u)$.

对于任一划分问题的实例, 考虑构建如下建立工人-工序映射问题的实例: 目前只有一个订单 d 到来, 其产品类型需要两道工序 i_1 和 i_2 来生产. SPS 中共有 $|U|$ 个工人可以参与加工, 且每个工人均有能力加工工序 i_1 和 i_2 . 任一工人 w 在这两道工序上的单位加工时间为 $1/h(u)$. 对于任一工人来说, 参与的工序种类限制 $X_d^w = 1$ 且初始剩余工作时间为 1. 订单 d 的需求量为 $\sum_{u \in U} h(u)/2$. 此时, 显然当构建的建立工人-工序映射问题的实例能够求解时, 对应的划分问题实例也能得到求解.

由此, 建立工人-工序映射是 NP 难的. \square

定理 2. 工人-Seru 分配问题可以被多项式时间内求解.

证明. 工人-Seru 分配问题模型与最小费用流问题模型是一样的, 如图 1 所示. 最小费用问题是一个可以被多项式时间内求解的经典问题, 因此工人-Seru 分配问题也可以被多项式时间内求解. \square

推论 1. SC 问题是 NP 难的.

证明. 由定理 1 和定理 2 可知, SC 问题是 NP 难的. \square

算法 1. SC 算法

步骤 1. 初始化每道工序上的工人-工序映射列表, 将所有可以加工该道工序的工人放入对应列表中, 产品生产量设置为 0, n_d^{wi} 设置为 0.

步骤 2. 工人排序: 对于每道工序, 按照单位加工时间非减的顺序对列表进行排序.

步骤 3. 依次检验每道工序, 若工人-工序映射列表为空, 返回 False; 否则将该工序任务分配给工人-工序映射列表中首个工人, 并扣除该工人剩余加工时间, 数值为该工人在该道工序上的单位加工时间, 对应 n_d^{wi} 增加 1.

步骤 4. 检测至最后一道工序后, 产品生产量增加 1, 若产品生产量小于该订单产品需求量, 返回步骤 3; 否则, 记录并输出 n_d^{wi} .

步骤 5. 利用网络单纯型法计算工人-Seru 分配, 记录并输出 n_{w_1, w_2, i_1, i_2}^d .

对于 SC 算法, 其流程中最复杂的算法循环需要先对订单需求量 n_d 进行 1 次循环, 再对工人数量 W 进行嵌套的 3 次循环, 因此 SC 算法的时间复杂度为 $O(n_d W^3)$.

目前已有许多多项式时间算法或求解器可以对最小费用流问题进行求解, 网络单纯形法是其中一个常用的算法. 因此, SC 算法的步骤 5 采用这一算法对工人-Seru 分配问题进行求解, 使用版本为 Python 库中 NetworkX 内置的网络单纯型法. 算法较为经典, 因此细节不再赘述.

定理 3. Seru 调度问题是 NP 难的.

证明. 本文将已被证明是 NP 难问题的并行多机调度问题 (Parallel machine scheduling, PMS) 归约到提出的 Seru 调度问题.

并行多机调度问题描述如下: 对于给定的一系列任务 $J = \{1, \dots, j, \dots\}$, 任一任务 $j \in J$ 有相应的到来时间 $rt(j)$ 、处理时间 $pt(j)$ 、完工时间 $ft(j)$. 这些任务可以在 M 台机器上加工且不允许抢占. 决策目标是找出一个调度方案使得最大完工时间最小.

任一并行多机调度问题实例等同于如下 Seru 调度问题实例:

令 $t_d = rt(j)$, $pt_s = pt(j)$, $ct_s = ft(j)$, 显然当 Seru 调度问题的实例可以被求解时, 对应的并行多机调度问题实例也可以被求解. 因此, Seru 调度问题是 NP 难的. \square

针对这一 NP 难的 Seru 调度问题, 本文提出了近似算法如下:

算法 2. SS 算法

步骤 1. 初始化: 将所有待调度 Seru 放入未调度列表, 将已调度列表设置为空, 将时刻设置为 0.

步骤 2. 当未调度列表不为空时, 按照 pt_s 非增原则对未调度列表中 Seru 进行排序; 否则, 跳至步骤 6.

步骤 3. 若存在 n 个空闲场地, 则将未调度列表中前 n 位的 Seru 从未调度列表移动至已调度列表.

步骤 4. 对新放入已调度列表中的 n 个 Seru 进行检测, 若与其他任一已调度列表中 Seru 存在工人占用, 则将其重新移动至未调度列表. 完成对 n 个 Seru 的检测和操作后, 将时刻增加 1.

步骤 5. 如果有新的 Seru 需要被调度, 则添加其至未调度列表, 检测并移除已调度列表中所有完成加工任务的 Seru, 返回步骤 2.

步骤 6. 停止, 输出结果.

对于 SS 算法, 其流程中最复杂的算法循环需要对 Seru 的数目 S 进行 2 次循环, 因此 SS 算法的时间复杂度为 $O(S^2)$.

定理 4. SS 算法的近似比为 3.

证明. 首先, 定义最优 Makespan 为 C_{\max}^* , 定义 $\sum \delta$ 为 SS 算法得出的生产过程中由于工人占用所造成的 Seru 之间非并行的等待时间之和. 对于所有已构建的 Seru 来说, C_{\max}^* 有 3 个非常简单的下界:

$$C_{\max}^* \geq \frac{\sum_{s=1}^S pt_s}{P} \quad (14)$$

$$C_{\max}^* \geq pt_{s'}, \quad \forall s' \quad (15)$$

$$C_{\max}^* \geq \sum \delta \quad (16)$$

式 (14) 表示 C_{\max}^* 至少要 and 每个场地上所构建的 Seru 的平均存在时间一样长; 式 (15) 表示 C_{\max}^* 至少要 and 任意一个 Seru 的存在时间一样长. 式 (16) 表示 C_{\max}^* 至少要 and 生产过程中所有由于工人占用所造成的 Seru 之间非并行的等待时间一样长.

假设 s' 为 Seru 调度算法计算出的某个周期内 SPS 构建的最后一个 Seru, 系统的 Makespan 为 C_{\max} , 则 $kt_{s'}^p$ 为该 Seru 构建时刻, $pt_{s'}$ 为该 Seru 的存在时间. 显然, $C_{\max} = kt_{s'}^p + pt_{s'}$. 此时, 对于 s' 之前构建的所有 Seru 来说, 若在 $kt_{s'}^p$ 之前, 场地应只存在 $\sum \delta$, 否则 $kt_{s'}^p$ 有希望被提前. 在这样的状态下, 所有 Seru 的平均最大完工时刻应为 $\sum_{s=1}^S pt_s/P + \sum \delta$, 而 s' 按照 SS 排序应为存在时间最短的 Seru, 因此它的开工时间一定在该时刻之前. 由此, 结合式 (14) ~ (16), 可以得到:

$$\begin{aligned} C_{\max} &= kt_{s'}^p + pt_{s'} \leq \\ &\frac{\sum_{s=1}^S pt_s}{P} + \sum \delta + pt_{s'} \leq \\ C_{\max}^* + C_{\max}^* + C_{\max}^* &= 3C_{\max}^* \end{aligned} \quad (17)$$

由此, 得证 Seru 调度算法的近似比为 3. \square

3 实验与分析

为了检验 SPS 的性能表现, 本文进行了多组实

验, 波动市场采用不同的参数组合来模拟. 仿真实验采用 Python 语言实现, 运行在 3.10 GHz Core i5-2400 CPU, 4 GB RAM 的 PC 机上. 共有 3 750 组用例被测试. 以下分别介绍实验设计、算例展示、实验结果和相关分析.

3.1 实验设计

本文中, 通过对多样的产品类型和波动变化的需求量两个维度来对波动市场进行描述. 具体来说, 每种产品的生产需要多道工序, 因此设置了不同复杂程度的产品类型组合以不同的级别表示来区分, 分别为 1, 3, 5, 7, 9. 每个级别表示该测试用例下到来的多个订单所包含的不同产品类型数量上限. 每个测试用例实际生成的产品类型数量采用范围从 1 到该级别数值的均匀分布来生成. 每个订单的需求量服从截断正态分布, 平均值 μ 以及波动系数 $cf = \sigma/\mu$ 用于体现市场波动. μ 设置为: 10, 20, 30, 40, 50, cf 设置为: 0.1, 0.3, 0.5, 0.7, 0.9. 对于上述波动市场的参数设置如表 1 所示.

表 1 波动市场的参数描述
Table 1 Parameters of volatile markets

参数	取值范围
产品类型级别 (k)	1, 3, 5, 7, 9
产品类型生成方式	$\sim U[1, k]$
产品需求量均值 (μ)	10, 20, 30, 40, 50
产品需求量波动系数 (cf)	0.1, 0.3, 0.5, 0.7, 0.9
产品需求量生成方式	$\sim N[\mu, (\mu \times cf)^2]$

工人的技能设置采用 Chaining strategy 的方式来设置^[28]. 本实验的其余参数设定如表 2 所示.

表 2 算例的其余参数描述
Table 2 Parameters of test problems

算例参数	取值范围
SPS 工人数量	15
工人初始剩余工作时间	1500
工人在任一工序上的单位加工时间	$\sim U[8, 12]$, 向下取整
订单数量	$\sim U[5, 15]$, 向下取整
相邻到来订单的时间间隔	$\sim E(1/30)$

因此, 共有 5 (产品类型级别) \times 5 (μ) \times 5 (cf) = 125 组不同参数组合产生, 依据中心极限定理^[29], 每个参数组合下产生 30 个算例, 因此共有 3 750 个算例被测试.

3.2 算例展示

为了更清晰地展示模型性质, 本节选取一个简

单的包含 5 个工人的 SPS 构建算例来展示. 其中, 所有工人的初始剩余加工时间均为 200, 工人在不同工序上的单位加工时间如表 3 所示, 该生产周期内的订单信息如表 4 所示, 工人在不同工序上的加工优先次序为单位加工时间短的工人排列在前.

表 3 工人在不同工序上的单位加工时间
Table 3 Processing time of each worker on different operations

工人/工序	工序 1	工序 2	工序 3	工序 4	工序 5
工人 1	10	—	—	—	9
工人 2	8	10	—	—	—
工人 3	—	12	8	—	—
工人 4	—	—	9	9	—
工人 5	—	—	—	11	12

表 4 订单信息
Table 4 The information of orders

订单	到来时间	产品类型 (所需工序标号集合)	需求量
订单 1	0	{1, 2}	15
订单 2	40	{3, 4, 5}	11

具体地, 在该算例中共会有 2 个订单到来, 订单 1 会在时刻 0 到来, 加工该类型产品需要工序 1 和工序 2 工两道工序, 其需求量为 15, 订单 2 会在时刻 40 到来, 加工该类型产品需要工序 3, 工序 4, 工序 5 三道工序. 订单的详细信息在该订单到来时刻才会被工厂获知. 通过第 1 阶段算法计算可以很容易得出该算例下工人在不同工序上的加工优先次序, 如在加工工序 1 时, 共有工人 1 和工人 2 可以参与加工, 选择优先顺序为优先选择工人 2, 其次选择工人 1.

在决策订单 1 的生产任务分配时, 工序 1 和工序 2 均会优先分配给工人 2, 而工人 2 的生产能力不足以完成整个订单 1 的生产, 因此需要构建多个 Seru 来共同完成订单 1 的生产. 同理, 对于生产订单 2 的生产, 可根据第 1 阶段和第 2 阶段的算法计算得出所构建 Seru 具体信息如表 5 所示. 具体来说, 为生产订单 1 构建 Seru 1 和 Seru 2, 其中 Seru 1 为只有工人 2 参与加工的单人物台, Seru 2 为工人 1 和工人 3 共同配合加工, 工人 1 负责工序 1, 工人 3 负责工序 2. Seru 1 产出产品 11 件, 存在时长为 198. Seru 2 产出产品 4 件, 存在时长为 58. 为生产订单 2 构建 Seru 3 和 Seru 4, 其中 Seru 3 有工人 3、工人 4、工人 5 参与加工, 工人 3 负责工序 3, 工人 4 负责工序 4, 工人 5 负责工序 5. Seru 4 有工人 4 和工人 5 参与加工, 工人 4 负责工序 3 和工序

4, 工人 5 负责工序 5. Seru 3 产出产品 9 件, 存在时长为 125. Seru 4 产出产品 2 件, 存在时长为 48.

表 5 为生产 3 个订单所构建的 Seru 展示
Table 5 Serus that are configured for orders

订单	订单1		订单2	
Seru	Seru 1	Seru 2	Seru 3	Seru 4
工序 1	工人 2	工人 1	—	—
工序 2	工人 2	工人 3	—	—
工序 3	—	—	工人 3	工人 4
工序 4	—	—	工人 4	工人 4
工序 5	—	—	工人 5	工人 5
产品产出量	11	4	9	2
存在时长	198	58	125	48

经第 3 阶段算法计算, 4 个 Seru 的调度结果如图 2 所示. Seru 1 和 Seru 2 不存在工人占用, 可以并行构建. 订单 2 在时刻 40 到来, 此时 Seru 1 和 Seru 2 仍未完成各自生产任务, Seru 2 和 Seru 3 均需工人 3 参与加工, 因此此时 Seru 3 不能被构建, 先行构建 Seru 4. 在时刻 58, 工人完成 Seru 2 的生产任务被释放, 而 Seru 3 和 Seru 4 之间有共同需要工人 4 和工人 5 参与加工, 因此 Seru 3 仍不能被构建, 需等待 Seru 4 加工完毕. 直到时刻 88, Seru 3 可以被构建.

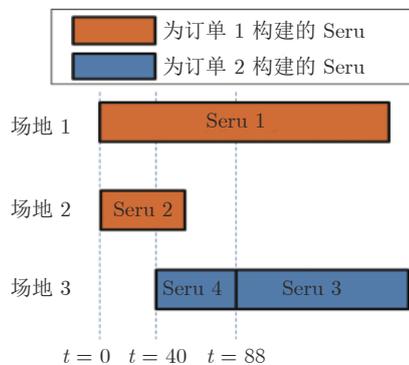


图 2 为订单 1 和 2 所构建的 4 个 Seru 甘特图
Fig.2 Gantt chart for 4 Serus

这个算例展示了 Seru 间可能频繁发生的并行构建与工人占用, 由此可观察到 SPS 具备根据市场变化灵活调整其结构的能力.

3.3 实验结果与分析

本节给出了 4 组不同实验的结果展示与分析, 分别是: 1) 波动市场环境下, 不同的产品类型参数设置对 SPS 中 Seru 数目、工人利用率、订单完成时间的影响; 2) 波动市场环境下, 不同的产品需求量

设置对 SPS 中 Seru 数目、工人利用率的影响; 3) 波动市场环境下, 不同的市场波动系数设置对工人利用率的影响.

3.3.1 产品类型参数设置的影响

图 3 展示了随着产品类型级别上升, SPS 所构建的 Seru 平均数目的变化情况. 如图 3 所示, 当产品类型级别从 1 升至 9, Seru 平均数目从 2.48 上升到了 10.62, 且整体呈上升趋势. 图 4 展示了随着产品类型级别上升, 工人平均利用率的变化情况. 如图 4 所示, 随着产品类型级别从 1 升至 9, 工人利用率在 79.20% 和 85.14% 之间波动, 基本趋于稳定. 图 3 和图 4 体现了 SPS 在应对同一生产周期内到来的产品类型多样的订单时, 为了高效完成生产任务 (如维持较高的工人利用率), 可以通过构建和拆解不同数量、结构的 Seru 从而动态调整系统整体结构来应对不同的市场环境. 这也是 SPS 可以快速响应多种不同类型订单的重要原因.

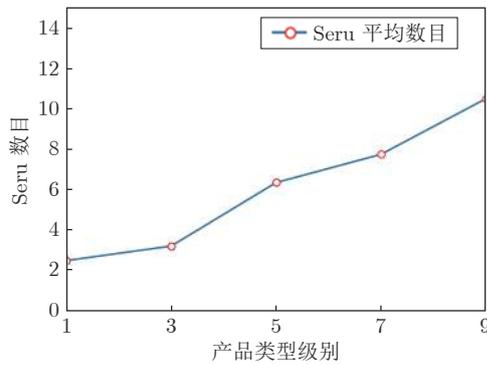


图 3 Seru 数目随产品类型级别变化趋势
Fig.3 The number of Serus versus product types

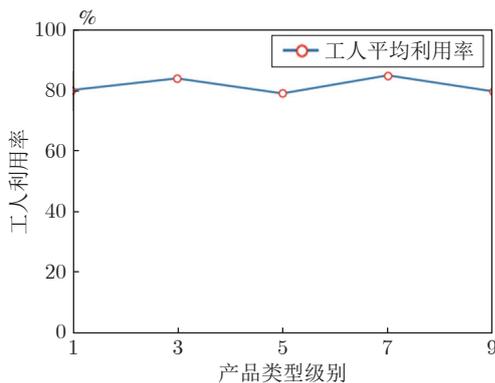


图 4 工人利用率随产品类型级别变化趋势
Fig.4 The utilization of workers versus product types

图 5 展示了 Makespan 随着产品类型级别上升的变化情况. 可以看到, 随着产品类型级别从 1 升至 9, Makespan 呈整体下降趋势, 从 2445 下降到

了 2022. 这是一个较为反常的现象, 因为随着产品需求种类增多 (即产品类型级别上升), 一般而言需要更长的完工时间. 通过跟踪实验细节发现, 对于产品类型级别较低的测试用例, 需要的工序较为单一, 从而使得参与加工的工人较为固定, 生成的 Seru 由于互相之间工人的占用, 很难并行构建, 因此需要的完工时间较长. 而当产品类型级别提高后, 产品类型增多, 工序种类数目上升, 可以参与加工的工人数量也随之上升, 这导致了更多的 Seru 可以被构建且互相之间发生工人占用情况的概率下降, 因此 SPS 整体并行性得到了提高, 从而获得了更短的 Makespan.

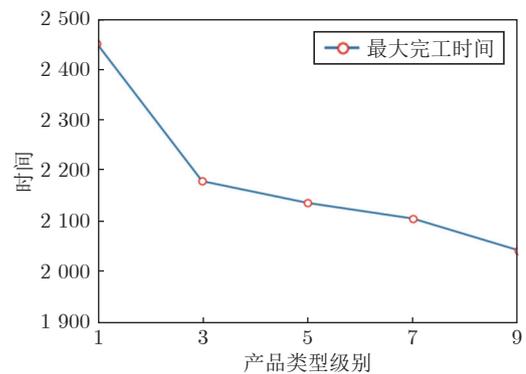


图 5 订单最大完工时间随产品类型级别变化趋势
Fig.5 The makespan versus product types

3.3.2 产品需求量设置的影响

图 6 展示了 SPS 所构建的 Seru 平均数目随着订单需求量均值 μ 上升的变化情况. 在图 6 中, 随着订单需求量均值 μ 从 10 升至 50, Seru 平均数目从 10.18 上升到了 14.50. 图 7 展示了工人平均利用率随着订单需求量均值 μ 上升的变化情况. 如图 7 所示, 工人利用率稳定保持在 77.16% 附近. 图 6 和图 7 相结合体现出, 当市场需求上升时, SPS 的工人

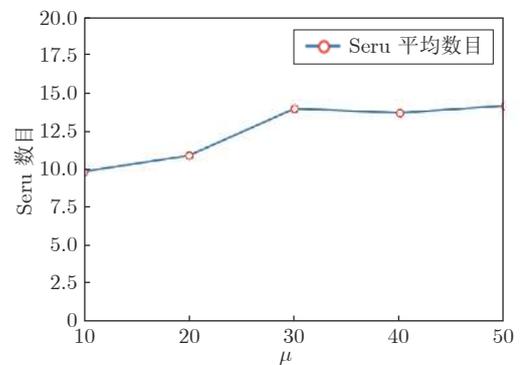


图 6 Seru 数目随订单需求量变化趋势
Fig.6 The number of Serus versus product volumes

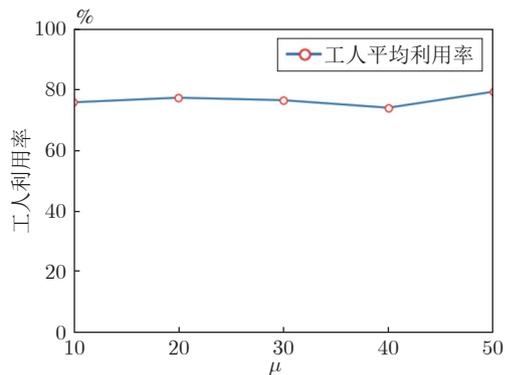


图7 工人利用率随订单需求量变化趋势

Fig.7 The utilization of workers versus product types

平均利用率仍能保持在稳定的较高水平。

3.3.3 市场波动系数设置的影响

图8展示了工人平均利用率随着市场波动系数 cf 上升的变化情况。从图8可以看出,随着 cf 从 0.1 升至 0.9,工人平均利用率保持在 80.61% 附近。这一结果体现了随着市场的逐步恶化(市场从稳定到剧烈波动),SPS 可以保持较高的工人平均利用率。

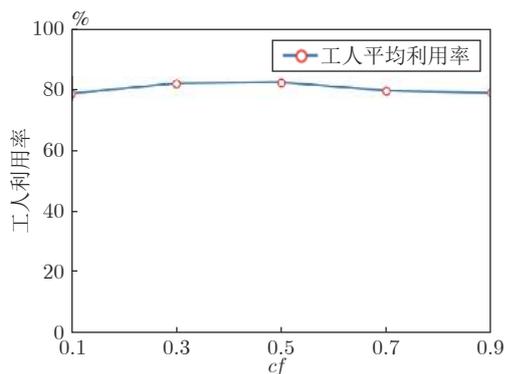


图8 工人利用率随市场波动变化趋势

Fig.8 The utilization of workers versus cf levels

以上实验结果验证了面对产品类型多样且产品需求变化大的典型波动市场环境,SPS 具有稳定高效的性能表现,其可重构性使其具有了良好的响应能力。另一方面,在不同参数的设置下,工人利用率始终维持在较高水平,实验结果从侧面验证了所提出模型的有效性。

4 结束语

本文针对波动市场环境下 SPS 构建问题,从实际情况出发,考虑了系统中工人是多能工且具有不同的技能组合和技能水平的情况,提出了一种 SPS 三阶段构建模型与相应方法。构建出的 SPS 具有较强的重构能力和良好的响应能力。具体来说,首先,

针对当前到来订单中产品的每道工序,为其分配有足够生产能力(具备相应技能和足够的工作时间)的工人;其次,安排不同的工人之间配合来生成 Seru,使得工人之间的配合最为平衡,最大化工人利用率;最后,考虑不同 Seru 间的工人占用问题,对 Seru 进行调度,最小化加权完工时间。在对模型性质分析的基础上,本文提出了相应的精确算法、在线调度算法对模型进行求解。实验结果验证了所构建 SPS 的优良性能。

本文所提出的 SPS 构建模型和方法具有较强的现实意义。围绕这一研究方向,在今后的工作中,会从以下几个方面进行更深入的探索:1)考虑更大规模、更为复杂的波动市场环境下 SPS 构建问题,设计高效的智能优化算法求解;2)对于生产系统来说,重要的优化指标有很多,比如最大化工人利用率、最小化成本、最小化库存、最小化完工时间等等,考虑建立 SPS 构建问题的多目标决策模型及设计相应的多目标优化算法。

References

- 1 Yokoi K. *Yokoi Style of Sales*. Akashi City: Pencom Publication, 2014. (in Japanese)
- 2 Yin Y, Stecke K E, Swink M, Kaku I. Lessons from Seru production on manufacturing competitively in a high cost environment. *Journal of Operations Management*, 2017, **49-51**(1): 67-76
- 3 Gong J J, Irikura N, Suzuki S. New product development operational performance and its influential factors: Cross-national comparison between Japan and China. *Asian Journal of Management Science and Applications*, 2014, **1**(3): 237-260
- 4 Kambara H. Outsourcing and performance in Japanese manufacturing firms. *Asian Journal of Management Science and Applications*, 2016, **2**(4): 376-393
- 5 Wang J W, Dou R L, Muddada R R, Zhang W J. Management of a holistic supply chain network for proactive resilience: Theory and case study. *Computers and Industrial Engineering*, 2018, **125**: 668-677
- 6 Stecke K E, Yin Y, Kaku I, Murase Y. Seru: The organizational extension of JIT for a super-talent factory. *International Journal of Strategic Decision Sciences*, 2012, **3**(1): 106-119
- 7 Yin Y, Kaku I, Stecke K E. *The Evolution of Seru Production Systems Throughout Canon*. Edinburgh: Neilson Journals Publishing, 2008.
- 8 Yin Y, Stecke K E, Swink M, Kaku I. *Integrating Lean and Agile Production Paradigms in a Highly Volatile Environment with Seru Production Systems: Sony and Canon Case Studies*. Yamagata University, Yamagata, Japan, 2012.
- 9 Yin Y, Stecke K E, Li D N. The evolution of production systems from industry 2.0 through industry 4.0. *International Journal of Production Research*, 2018, **56**(1-2): 848-861
- 10 Stecke K E, Yin Y, Kaku I. Seru production: An extension of just-in-time approach for volatile business environments. *Analytical Approaches to Strategic Decision-Making: Interdisciplinary Considerations*. Hershey: Business Science Reference, 2014. 45-58
- 11 Zhang X L, Liu C G, Li W J, Evans S, Yin Y. Effects of key enabling technologies for Seru production on sustainable performance. *Omega*, 2017, **66**: 290-307
- 12 Yu Y, Tang J F. Review of Seru production. *Frontiers of Engineering Management*, 2019, **6**(2): 183-192
- 13 Yu Yang, Tang Jian-Fu. *Seru Production Mode*. Beijing: Science Press, 2018.

(于洋, 唐加福. Seru生产方式. 北京: 科学出版社, 2018.)

- 14 Yu Y, Wang J W, Ma K, Sun W. Seru system balancing: Definition, formulation, and exact solution. *Computers and Industrial Engineering*, 2018, **122**: 318–325
- 15 Wu Xu-Hui, Du Shao-Feng, Hao Hui-Hui, Yu Yang, Yin Yong, Li Dong-Ni. A line-Seru conversion approach by means of cooperative coevolution. *Acta Automatica Sinica*, 2018, **44**(6): 1015–1027
(吴旭辉, 杜劲峰, 郝慧慧, 于洋, 殷勇, 李冬妮. 一种基于协同进化的流水线向Seru系统转化方法. *自动化学报*, 2018, **44**(6): 1015–1027)
- 16 Liu C G, Stecke K E, Lian J, Yin Y. An implementation framework for Seru production. *International Transactions in Operational Research*, 2014, **21**(1): 1–19
- 17 Liu C G, Li W J, Lian J, Yin Y. Reconfiguration of assembly systems: From conveyor assembly line to Serus. *Journal of Manufacturing Systems*, 2012, **31**(3): 312–325
- 18 Yu Y, Gong J, Tang J F, Yin Y, Kaku I. How to carry out assembly line–cell conversion? A discussion based on factor analysis of system performance improvements. *International Journal of Production Research*, 2012, **50**(18): 5259–5280
- 19 Yu Y, Tang J F, Sun W, Yin Y, Kaku I. Combining local search into non-dominated sorting for multi-objective line-cell conversion problem. *International Journal of Computer Integrated Manufacturing*, 2013, **26**(4): 316–326
- 20 Yu Y, Tang J F, Sun W, Yin Y, Kaku I. Reducing worker(s) by converting assembly line into a pure cell system. *International Journal of Production Economics*, 2013, **145**(2): 799–806
- 21 Yu Y, Sun W, Tang J F, Kaku I, Wang J W. Line-Seru conversion towards reducing worker(s) without increasing makespan: Models, exact and meta-heuristic solutions. *International Journal of Production Research*, 2017, **55**(10): 2990–3007
- 22 Liu C G, Yang N, Li W J, Lian J, Evans S, Yin Y. Training and assignment of multi-skilled workers for implementing Seru production systems. *The International Journal of Advanced Manufacturing Technology*, 2013, **69**(5–8): 937–959
- 23 Liu C G, Dang F, Li W J, Lian J, Evans S, Yin Y. Production planning of multi-stage multi-option Seru production systems with sustainable measures. *Journal of Cleaner Production*, 2015, **105**: 285–299
- 24 Lian J, Liu C G, Li W J, Yin Y. A multi-skilled worker assignment problem in Seru production systems considering the worker heterogeneity. *Computers and Industrial Engineering*, 2018, **118**: 366–382
- 25 Wang Y, Tang J F. Is full skill the best configuration for Seru production system? In: Proceedings of the 2018 Chinese Control and Decision Conference (CCDC). Shenyang, China: IEEE, 2018.
- 26 Wang Y, Tang J F. Cost and service-level-based model for a Seru production system formation problem with uncertain demand. *Journal of Systems Science and Systems Engineering*, 2018, **27**(4): 519–537
- 27 Chopra S, Rao M R. The partition problem. *Mathematical Programming*, 1993, **59**(1–3): 87–115
- 28 Jordan W C, Graves S C. Principles on the benefits of manufacturing process flexibility. *Management Science*, 1995, **41**(4): 577–594
- 29 Aczel A, Sounderpandian J. *Complete Business Statistics*. (7th edition). New York: McGraw-Hill Companies, 2008.



湛荣鑫 北京理工大学计算机学院博士研究生. 主要研究方向为赛汝生产与智能优化.

E-mail: bitzrx@163.com

(**ZHAN Rong-Xin** Ph.D. candidate at the School of Computer Science, Beijing Institute of Technology. His research interest covers Seru production and intelligent optimization.)

His research interest covers Seru production and intelligent optimization.)



李冬妮 北京理工大学计算机学院教授. 主要研究方向为智能优化与仿真计算, 智能制造及数字孪生. 本文通信作者. E-mail: ldn@bit.edu.cn

(**LI Dong-Ni** Professor at the School of Computer Science, Beijing Institute of Technology. Her research interest covers intelligent optimization and simulation, smart factory and digital twin. Corresponding author of this paper.)

Her research interest covers intelligent optimization and simulation, smart factory and digital twin. Corresponding author of this paper.)



马涛 研究员级高级工程师, 特种车辆及其传动系统智能制造国家重点实验室副主任. 主要研究方向为数字化及智能制造应用基础技术.

E-mail: matao@nmgyj.com

(**MA Tao** Senior engineer in research fellow level, and associate director of the State Key Laboratory of Smart Manufacturing for Special Vehicles and Transmission Systems. His research interest covers basic technology for digital and intelligent manufacturing applications.)

His research interest covers basic technology for digital and intelligent manufacturing applications.)



李俊杰 内蒙古第一机械集团有限公司研究员级高级工程师. 主要研究方向为车辆动力辅助系统和自动装填系统工艺技术.

E-mail: 13337199371@163.com

(**LI Jun-Jie** Senior engineer in research fellow level at Inner Mongolia First Machinery Group Co., Ltd.. His research interest covers vehicle power assist system and technology of automatic loading system.)

His research interest covers vehicle power assist system and technology of automatic loading system.)



吴延昭 内蒙古第一机械集团有限公司高级工程师. 主要研究方向为冲压自动化, 智能化生产.

E-mail: wuyanzhaolishiqi@126.com

(**WU Yan-Zhao** Senior engineer at Inner Mongolia First Machinery Group Co., Ltd.. His research interest covers stamping automation and intelligent production.)

His research interest covers stamping automation and intelligent production.)



殷勇 日本同志社大学商学院教授. 主要研究方向为赛汝生产与工业 4.0.

E-mail: yyin@mail.doshisha.ac.jp

(**YIN Yong** Professor at the Graduate School of Business, Doshisha University, Japan. His research interest covers Seru production and Industry 4.0.)

Industry 4.0.)