考虑多工序设备权重的资源协同综合调度算法

周 伟 谢志强*
(哈尔滨理工大学计算机科学与技术学院 哈尔滨 150080)

摘 要：针对多品种、小批量复杂产品综合调度研究中，没有考虑加工较多工序设备上的工序间调度空隙会对调度结果产生重要影响的问题，该文提出考虑多工序设备权重的资源协同综合调度算法。该算法在综合调度中首先提出多工序设备和工序权重值的定义，其次提出以权重值为主的调度策略，提高了工序纵向连续加工的紧密度；最后提出最佳调度时刻的调整策略，提高了工序横向并行优化的力度。实验结果表明，该算法在提高综合调度设备整体利用率和减少复杂产品时间成本等方面，具有更优性。

关键词：综合调度；资源协同；设备；多工序；权重
中图分类号：TP278 **文献标识码**：A **文章编号**：1009-5896(2022)05-1625-11
DOI: 10.11999/JEIT211366

Resource Collaborative Integrated Scheduling Algorithm Considering Multi-process Equipment Weight

ZHOU Wei XIE Zhiqiang
(*School of Computer Science and Technology, Harbin University of Science and Technology, Harbin* 150080, *China*)

Abstract: In order to solve the problem that scheduling gaps between processes on equipment with many processes will have important influence on scheduling results in the integrated scheduling research of multi-variety and small batch complex products, a resource cooperative integrated scheduling algorithm considering the weight of equipment with many processes is proposed. In the integrated scheduling, the definition of multi-process equipment and process weight value is proposed for the first time, and then the scheduling strategy based on weight value is proposed to improve the tightness of longitudinal continuous processing. Finally, the optimal scheduling time adjustment strategy is proposed to improve the intensity of horizontal parallel optimization. Experimental results show that the algorithm has better performance in improving the overall utilization rate of the integrated scheduling equipment and reducing the time cost of complex products.

Key words: Integrated scheduling; Resource collaboration; Equipment; Multi-process; Weight

1 引言

调度问题作为制造系统的关键环节，一直受到学术界的广泛关注。高效的调度系统既可以降低产品制造的各项成本，又可以提高企业的经济效益、社会效益等。随着计算机技术的发展和社会需求的变化，制造行业的产品调度也随之改变[1–15]。例如，王卓昊等人[1]构建了一套分布式抽取、转换再加载的任务调度框架，优化了复杂任务的调度执行时间；胡致远等人[2]提出了非工作保持型的多节点联合调度模型，保障了业务确定的可达性需求；王萍等人[3]针对动态负载环境下的高效分布式资源分配问题，提出了一种预留—重用联合的Q学习型半持续调度算法等。但是产品制造的多样化和人们需求的个性化，使得小批量、复杂工艺的产品调度问题日益突出。不同于传统工业中的流水调度(flow-shop)和车间调度(job-shop)问题，谢志强[16]提出了第3类产品制造调度模式，即在具有树形复杂结构的多品种、小批量产品中，将产品加工和装配进行同步处理的综合调度，并针对综合调度开展了深入研究，已取得了较丰富的成果[17–26]。

例如，文献[17]针对柔性集成调度问题，提出了基于网络设备协作的改进人工蜂群算法；为解决综合调度智能优化算法中，编码方式遗漏最优解的问题，文献[18]提出了动态优先级约束表的编码方

收稿日期：2021-11-30；改回日期：2022-01-18；网络出版：2022-02-18
*通信作者： 谢志强 xiezhiqiang@hrbust.edu.cn
基金项目：国家自然科学基金(61772160, 61602133, 61672179)
Foundation Items: The National Natural Science Foundation of China (61772160, 61602133, 61672179)

法，保证了初始种群个体的可行性和完备性；文献[19]为解决资源服务器状态动态变化对工作流的调度问题，基于时间窗曲线模型和平均利用率模型提出了考虑资源状态动态反馈的算法等。

但是，在以“工序”为优化对象的研究中，如果算法以长路径为研究主线，那么当相同设备出现短路径上叶节点工序较早开始加工的情况时，首次适用调度策略就会失效，会在工序之间产生不可利用的空闲时间；在以后续工序排序为主线的算法中，加工工序较多的设备上会形成工序序列间的众多加工空隙；在以设备驱动并行为主线的算法中，当后续工序路径长度相同时，如果两个工序都是叶节点工序，那么短用时策略就会失效，纵向优化调度效果不佳，加工工序较多的设备上也会出现很多空闲时间段。

近年来综合调度中以“设备”为优化对象的相关研究较少，只有文献[24]的关键设备工序紧凑调度算法和文献[25]的关键设备工序紧凑的动态调度算法。二者的相同之处是：均将加工用时最长的设备定义为关键设备，然后在关键设备上应用拟关键路径方法，最后应用最佳适应方法在设备空闲时间段插入独立工序。不同之处在于：在文献[24]的算法中关键设备和关键路径是固定的，而文献[25]的算法中关键设备和关键路径是根据部分工序加工完成情况动态调整的。但是，当关键设备上加工工序的前序工序不是独立工序时，两种算法都会在关键设备上产生很多无法利用的空闲时间段，进而拉长了串行工序之间的空隙。

针对目前多品种、小批量复杂产品综合调度中，加工工序较多的设备在调度时出现较多空闲段的问题，本文提出了考虑多工序设备权重的资源协同算法。本文的主要贡献有：

(1)在综合调度领域中拓展了特殊设备的定义范畴：在柔性设备、批处理设备、关键设备、加工中心之外，定义了多工序设备；

(2)算法全面考虑了综合调度领域中的相关因素，如复杂产品工艺树结构、工序自身属性、工序之间的约束关系、设备加工情况等，在此基础上，提出了权重值策略和最佳调度时刻策略；

(3)算法以“多工序设备”作为优化对象，优先调度多工序设备上竞争相对紧张的工序，既提高了多工序设备上工序连续加工的力度，又联动其紧后工序及后续工序的尽早开始加工，有效地减少了产品的加工时间，进而优化了综合调度的整体效果。

2 问题及建模

假设综合调度系统中有m台加工设备，共需加工完成复杂产品的n道工序，每个工序自身加工用时已知，记为d_i，具体要求如下：

(1)每个工序具有工序序号、对应加工设备序号和自身加工用时的属性；

(2)设备加工工序时，具有时间确定性和加工连续性；

(3)除了叶节点工序，其他任何一道工序可以被加工的充分必要条件是其所有紧前工序均加工完毕；

(4)所有设备上最后一道工序加工完成的时间为产品的总加工用时。

综合调度的优化目标为合理调度各个设备上的工序，最小化复杂产品最大加工用时；约束条件为除了叶节点工序具有紧后工序、根节点工序具有紧前工序以外，其他所有节点工序均存在紧前紧后工序约束关系。

建立问题模型如下：

优化目标

$$\mathrm{Min}\sum_{i=1}^{n}T_i=\sum_{i=1}^{n}W_i+\alpha_i\mathrm{ST}_i,\alpha\geq 0 \tag{1}$$

$$\mathrm{s.t.}\ \mathrm{ST}_{i+1}-\mathrm{ST}_i\geq d_i \tag{2}$$

$$\text{设备资源：}P_{(i+1)m}-P_{im}\geq 0 \tag{3}$$

各符号代表的含义如下：T_i：复杂产品总加工用时；W_i：第i个工序的权重值；ST_i：第i个工序开始加工时间；P_{im}：第m个设备上正在加工的第i个工序。其中，式(1)表示优化目标为使得复杂产品总体加工用时T_i最少，包括工序的权重值W_i和最佳调度时间$\alpha_i\mathrm{S}_i$等因素，α为第i个工序开始加工的调节系数；式(2)表示第$(i+1)$个工序(组)开始加工的时刻必须大于等于第i个工序加工结束的时刻；式(3)表示在同一设备m的某一时刻，第i个工序具有加工连续性，只有其加工结束后，其后续的第$(i+1)$个工序才拥有开始加工的设备资源。

3 算法设计思想

3.1 相关定义

定义1 工序优先级。根据“减少并行工序加工时间”的原则，文献[26]针对复杂产品工艺树中的加工工序提出了工序优先级问题，即将工序调度的优先顺序定义为工序的优先级[26,27]。假设产品加工工艺树有n层，则将根节点工序的优先级定义为1；根节点工序的所有后裔节点工序的优先级定义为2，同层工序节点作为兄弟节点；以此类推，直到第n层的所有节点的优先级定义为n。定义根节点工序的优先级最低，第n层上工序的优先级最高。

定义2　设备优先级。假设综合调度系统有m台加工设备序列$\{\boldsymbol{M}_1, \boldsymbol{M}_2, \cdots, \boldsymbol{M}_m\}$，定义加工工序数量最多的设备优先级最高，加工工序数量次多的设备优先级次之，以此类推，加工工序数量最少的设备优先级最低为1，允许存在相同的设备优先级。

定义3　约束度。以某一工序为中心，将与其直接相邻的所有具有紧前紧后约束关系的工序数量之和，定义为此工序的约束度。

定义4　权重值。将工序的层优先级、设备优先级和工序约束度的数值之和定义为此工序的权重值。

定义5　最佳调度时刻。对于某一工序而言，将其紧前工序(组)加工结束的时刻定义为此工序的最佳调度时刻。

3.2 算法描述

现有综合调度算法的调度过程多以复杂产品工艺树中工序的路径、是否为叶节点工序以及工序的加工用时等因素为考量，较少考虑调度过程中设备的制约条件。与其他综合调度算法不同，本文充分考虑了多工序设备的优化对综合调度总体优化效果的重要作用，设计了层优先级和权重值的排序策略以及最佳调度时刻的调整策略。

层优先级策略是综合调度中的经典策略，主要作用是通过缩短并行工序加工时间的方式优化调度效果。优势主要体现在：(1)约束力强的紧前工序具有优先调度的先决条件；(2)以“层”为单位，优先调度的加工工序可以在不同的设备上同时加工。

权重值策略是算法的核心策略，通过提高多工序设备的利用率带动综合调度设备总体利用率的提高。优势主要体现在：(1)遵循了综合调度中“设备忙”的原则；(2)首次将工序的紧前紧后约束度作为考量因素，提高了多工序设备上工序的衔接度，促进综合调度效果的总体优化。

最佳调度时刻调整策略，有效地提高了设备并行处理的力度。优势主要体现在：(1)对于约束度不强的叶节点工序，具有在对应设备上最早开始加工的优势；(2)对于约束度较强的非叶节点工序，减少了其在设备间的串行空隙。

算法具体描述如下：

步骤1　根据复杂产品工艺树的结构特征，确定工艺树的层序，计算各个工序的层优先级；

步骤2　根据复杂产品工艺树的属性特征，计算各个工序的设备优先级、约束度和权重值；

步骤3　确定层优先级最高的工序；

步骤4　判断层优先级最高的工序是否唯一，如果是，则直接调度，如果否，则转步骤5；

步骤5　判断工序的权重值是否相同，如果是，则转步骤6，如果否，则按照权重值由大到小依次调度工序；

步骤6　根据最佳调度时刻调整策略，调整工序在对应设备上的最佳调度时刻；直到根节点工序调度结束，算法框架流程图如图1所示。

3.3 数据标准化处理

复杂产品工艺树结构可以通过虚拟根节点的方式构造更加复杂的复杂产品，因此产品的加工规模和复杂程度不尽统一；同时，由于制造系统中相关设备的情况也不尽相同，所以在采用本文算法进行权重值计算前，需要将工序的层优先级、设备优先级和工序约束度的具体数值进行标准化处理，目的是使不同量纲级别的数据经过计算后得到的结果不会出现偏差。

本文选取在数据处理中常用的Z-score标准化方法[28]，如式(4)所示

$$x^* = (x - \mu)/\sigma \tag{4}$$

其中，μ为数据的平均值，σ为数据的标准差。

3.4 算法复杂度分析

假设复杂产品的加工工序数、加工设备数和每道工序的自身加工用时等信息均已知，分别设为n, m, d_i，则本文算法复杂度分析如下：

将复杂产品的加工工艺图转化为工艺树，树中节点对应工序，约束关系对应箭头指向，确定根节

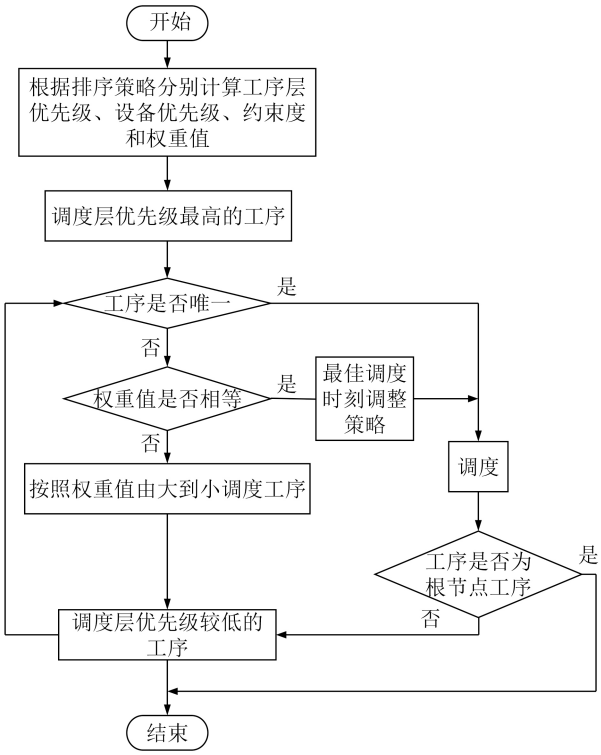


图 1 算法框架流程图

点为最后一道完成加工的工序，其优先级为1；依据产品工艺中严密的紧前紧后约束关系，计算所有节点的层优先级的时间复杂度为$O(n)$。在工序集、设备集已知的前提下，计算各个工序的设备优先级、约束度和权重值的时间复杂度均是$O(n)$。在设备空闲时间段上查找调度的最佳时刻，一道工序需要与其紧前工序(之和)的加工结束时刻进行对比，最坏情况是一道工序既要与其层优先级次之的所有工序进行对比，又要与其相同设备上的所有加工工序进行对比，所以时间复杂度为$O(n^2)$。而将数据进行标准化处理的过程中，仅在求标准差的过程中涉及了数据的平方计算，其时间复杂度为$O(n^2)$。

由上述分析可得，本文算法的时间复杂度为$\max\{O(n), O(n^2)\}=O(n^2)$。算法表述如表1所示。

4 实例分析

本文算法具有普适性，对其他算例亦有较好的调度效果。为便于读者理解本文算法，现举例说明，如图2所示，假设复杂产品A加工工艺树由12道工序组成，分别在4台设备上进行加工。

根据本文算法，首先计算各设备优先级，如表2所示。其次分别计算各个工序对应的层优先级、约束度、权重值和最佳调度时刻，如表3所示。

调度过程如表4所示，调度甘特图如图3所示，总加工用时为30工时。

表 1 考虑多工序设备权重的资源协同综合调度算法

```
Weight-value (A)
输入： Input S(A)，S(M);
输入： Set(P) Compute Layerpriority to A_i;
输入： Set(M) Compute Machine to M_i;
输入： Set(C) Compute Constraint to A_i;
输入： Set(W) Compute Weight-value to A_i;
输出： Output Dispath(A)
    For each node A_i∈A do
      If unique A_i to Max(P_i) then
        dispath A_i ;
      Else if
        {
          If same Weight-value A_i then
            optimal scheduling time adjustment(A_i);
        Else if
          dispath A_i by Weight-value Descending order
        }
    End if
    If root node process A_i
      Exit;
    Else if
      i--;
    End if
```

5 算法对比分析

5.1 以工序为研究对象的算法对比分析

为阐述本文算法的更优性，现选择文献[20]中的复杂产品B作为实例进行分析阐述，如图4所示。再选择文献[20]紧密衔接工序组联动的算法、文献[21]考虑串行工序紧密度的择时算法与本文算法进行对比分析。

针对复杂产品B加工工艺图，采用本文算法工序调度顺序为{A3, A2, A1, A4, A10, A5, A11, A12, A6, A18, A7, A8, A13, A20, A19, A21, A9, A10, A15, A22, A24, A23, A7, A25, A17, A26, A27}，调度甘特图如图5所示，总加工用时为27工时。

采用文献[20]紧密衔接工序组联动的算法，工序的调度顺序为{A10, A12, A1, A2, A3, A4, A5, A7, A6, A8, A9, A11, A13, A14, A15, A18, A19, A20, A21, A22, A23, A24, A25, A16, A17, A26, A27}，调度甘特图如图6所示，总加工用时为28工时。

采用文献[21]考虑串行工序紧密度的择时算法，初始调度方案为{A27, A26, A17, A16, A15, A9, A8, A6, A4, A3, A1}，在此基础上按照{A25, A23, A21, A20, A18, A14, A13, A11, A10, A24, A22, A7, A2, A12, A19, A5}的顺序进行调整。因算法采用逆序调度，在保证各工序约束条件的前提

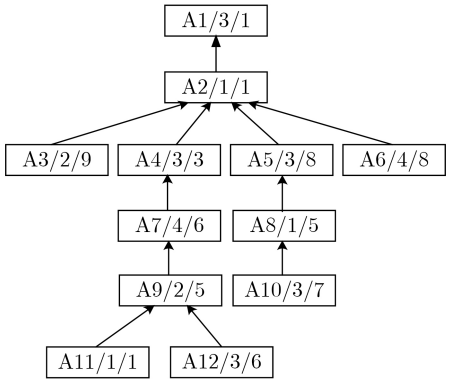


图 2 复杂产品A加工工艺树

表 2 复杂产品A设备优先级统计

设备号	加工工序数量	设备优先级
M_1	3	2
M_2	2	1
M_3	5	3
M_4	2	1

表 3　复杂产品A工序权重值统计

工序号	设备优先级	设备优先级数据标准化	层优先级	层优先级数据标准化	约束度	约束度数据标准化	权重值	最佳调度时刻
A1	3	1.063410138	1	–1.85933936	1	–0.729324957	–1.52525418	t=29
A2	2	–0.096673649	2	–1.183215957	5	2.771434838	1.491545233	t=28
A3	1	–1.256757436	3	–0.507092553	1	–0.729324957	–2.493174946	t=11
A4	3	1.063410138	3	–0.507092553	2	0.145864991	0.702182577	t=17
A5	3	1.063410138	3	–0.507092553	2	0.145864991	0.702182577	t=20
A6	1	–1.256757436	3	–0.507092553	1	–0.729324957	–2.493174946	t=0
A7	1	–1.256757436	4	0.169030851	2	0.145864991	–0.941861593	t=11
A8	2	–0.096673649	4	0.169030851	2	0.145864991	0.218222194	t=13
A9	1	–1.256757436	5	0.845154255	3	1.02105494	0.609451759	t=6
A10	3	1.063410138	5	0.845154255	1	–0.729324957	1.179239435	t=6
A11	2	–0.096673649	6	1.521277659	1	–0.729324957	0.695279052	t=0
A12	3	1.063410138	6	1.521277659	1	–0.729324957	1.855362839	t=0

表 4　复杂产品A调度过程

调度时刻	M_1 设备	M_2 设备	M_3 设备	M_4 设备
t=0～t=1	加工工序A11	–	加工工序A12	加工工序A6
t=1～t=6	–	–	工序A12加工完毕	加工工序A6
t=6～t=8	–	加工工序A9	加工工序A10	工序A6加工完毕
t=8～t=11	–	工序A9加工完毕	加工工序A10	–
t=11～t=13	–	加工工序A3	工序A10加工完毕	加工工序A7
t=13～t=17	加工工序A8	加工工序A3	–	工序A7加工完毕，加工结束
t=17～t=18	工序A8加工完毕	加工工序A3	加工工序A4	–
t=18～t=20	–	工序A3加工完毕，加工结束	工序A4加工完毕	–
t=20～t=28	–	–	工序A5加工完毕	–
t=28～t=29	工序A2加工完毕，加工结束	–	–	–
t=29～t=30	–	–	工序A1加工完毕，加工结束	–

下，逆序调整调度甘特图如图7所示，总加工用时为31工时。

5.2　算法结果对比分析

针对复杂产品B加工工艺图，本文算法的总加工用时最少，之所以本文算法更优，主要是因为：

文献[20]的算法优先调度“紧密衔接工序(组)中的工序”，忽略了约束度不高的工序(组)的相对位置对调度结果的影响，因而在工序串行调度过程中产生了空闲时间段。例如在图6中，设备M_2在t=17～t=24时刻出现较长时间的空闲，设备M_2在整个加工过程中共计有10个工时的空闲，多于图5中设备M_2的7个工时的空闲。

文献[21]算法在用择时调度策略确定工序加工开始时间点时，没有充分考虑到多工序设备的加工和利用情况，因而影响了整体调度效果。例如在图7中，

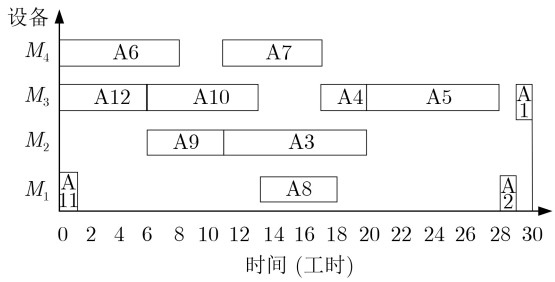


图 3 复杂产品A加工甘特图 30工时

设备M_2在t=0～t=2，t=5～t=7和t=14～t=18，t=19～t=24时刻一直处于空闲状态；而在图5中，工序A2，A3，A8，A24，A25比图7中的开始加工时间分别提前了7个工时、8个工时、8个工时、10个工时和6个工时，不仅提高了多工序设备M_2上工序连续加工的紧密度，而且设备利用率也提高了14.6%。

5.3 以设备为研究对象的算法对比分析

下面再将本文算法分别与同类代表性论文——文献[24]和文献[25]的算法进行对比分析，结果同样证明了本文算法的更优性。

(1)与文献[24]关键设备工序紧凑的算法对比。在文献[24]中，将加工工序用时最长的设备定义为关键设备，在本文中，将加工工序数量最多的设备定义为多工序设备。那么，本文有必要针对多工序设备上工序(组)总加工用时(下述简称前者)与关键设备上工序(组)总加工用时(下述简称后者)的3种不同情况进行详细对比分析。

(2)前者小于后者的对比分析。在复杂产品B加工工艺树中，本文算法中的多工序设备为M_2，设备M_2对应加工的8个工序(组)为{A2, A3, A8, A11, A12, A21, A24, A25}，总用时15工时；文献[24]算法中关键设备为设备M_1，对应加工的6个工序(组)为{A1, A6, A13, A20, A22, A27}，总用时16工时。

采用文献[24]的算法，工序调度顺序为{A1, A2, A3, A4, A5, A6, A7, A8, A9, A10, A11, A12, A13, A14, A15, A16, A17, A18, A20, A19, A21, A22, A23, A24, A25, A26, A27}，调度甘特如图8所示，总加工用时41工时。

当多工序设备上工序(组)总加工用时小于关键设备上工序(组)总加工用时，本文算法更优。

对比分析甘特图图5和图8：在图5中，设备

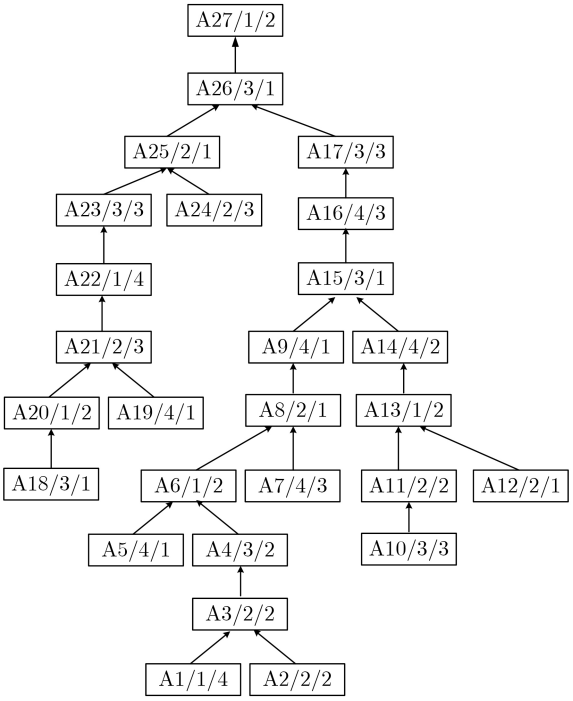


图 4 复杂产品B加工工艺树

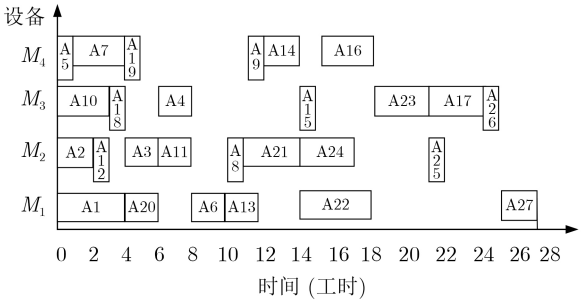


图 5 本文算法调度复杂产品B工艺树甘特图 27工时

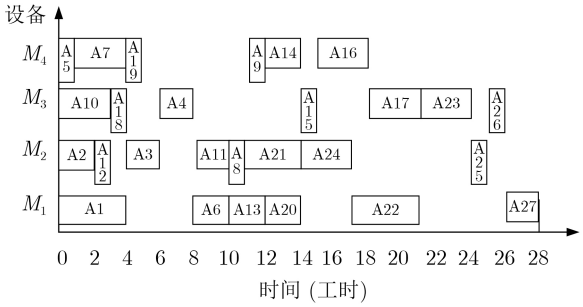


图 6 文献[20]算法调度复杂产品B甘特图 28工时

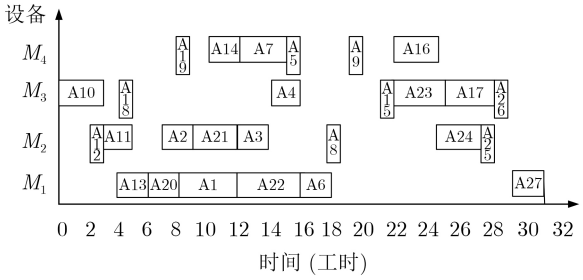


图 7 文献[21]算法调度复杂产品B逆序甘特图 31工时

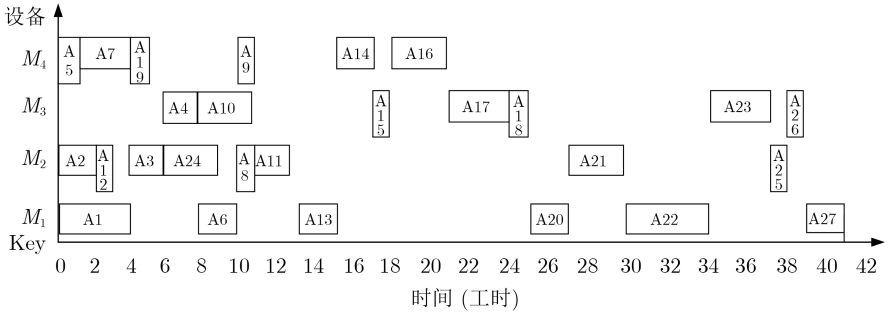


图 8 文献[24]算法调度复杂产品B甘特图 41工时

M_4上的总加工用时为18工时、设备M_3上的总加工用时为25工时、多工序设备M_2上的总加工用时为22工时、设备M_1上的总加工用时为27工时；在图8中，设备M_4上的总加工用时为21工时、设备M_3上的总加工用时为39工时、设备M_2上的总加工用时为38工时、关键设备M_1上的总加工用时为41工时。在图8中设备M_2上在加工过程中出现了共计23个工时的空闲，明显多于图5中的7个工时的空闲时间段。设备利用率对比分析如表5所示，本文算法设备总体利用率相对提高率提高了20.6%。

(3)前者大于等于后者的对比分析。由关键设备的判定条件可知，当前者大于等于后者时，对于同一复杂产品工艺树而言，多工序设备和关键设备为同一个设备。下面，本文以图9所示的复杂产品C工艺树为例进行对比分析。

在复杂产品C加工工艺图中，因设备M_3加工工序数量最多：6个，所以多工序设备为M_3，又因设备M_3上加工工序(组)的加工用时最长：160工时，所以在本例中，设备M_3既是多工序设备又是关键设备。

采用本文算法工序调度顺序为{A19, A18, A17, A11, A13, A14, A12, A15, A16, A8, A5, A6, A9, A10, A7, A3, A4, A2, A1}，调度甘特图如图10所示，总加工用时为200工时。

采用文献[24]的算法工序调度顺序为{A19, A17, A11, A12, A5, A6, A2, A18, A14, A15, A8, A13, A7, A9, A3, A16, A10, A4, A1}，调度甘特图如图11所示，总用时240工时。

当多工序设备上工序(组)总加工用时大于等于关键设备上工序(组)总加工用时，本文算法更优。

表 5　前者小于后者时两种算法的设备利用率对比分析

	M_1 利用率(%)	M_1相对提高率(%)	设备总体利用率(%)	总体利用率相对提高率(%)	产品B加工用时
本文算法	59.3	20.3	60.9	20.6	27
关键设备工序紧凑算法	39		40.3		41

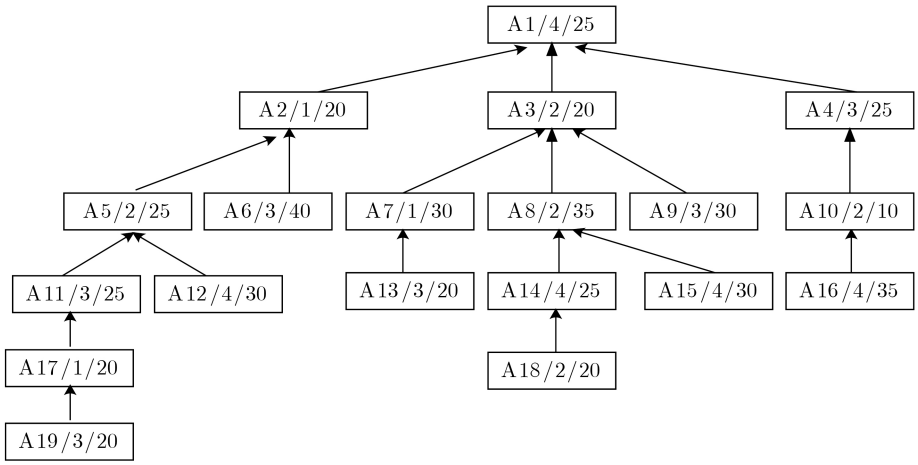


图 9 复杂产品C加工工艺树

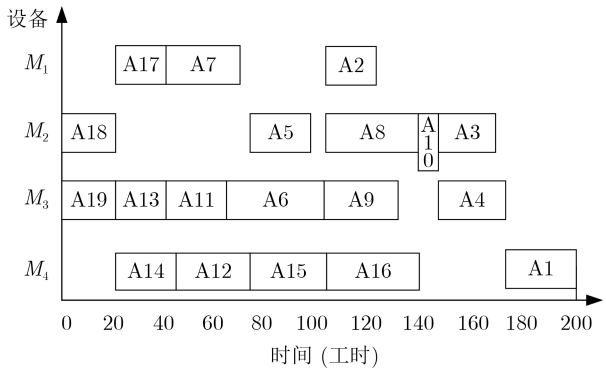


图 10 本文算法调度复杂产品C甘特图 200工时

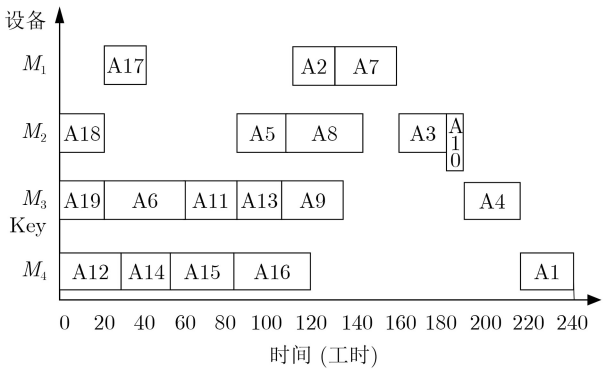


图 11 文献[24]算法调度复杂产品C甘特图 240工时

对比分析调度甘特图图10和图11：在图10中，多工序设备M_3的总加工用时为175工时，工序A13作为叶节点工序使用本文算法的最佳调度时刻调整策略，在t=20时刻开始加工，因此其紧后工序A7较图11中的开始加工时刻提前了90个工时；同时设备M_3也仅仅在t=135时刻到t=150时刻之间出现了15个工时的空闲时间段。在图11中，关键设备M_3的总加工用时为215工时，设备M_3在t=135时刻到t=190时刻的55个工时内，一直处在空闲状态。而在设备利用率方面，文献[24]设备总体利用率为59.6%，本文算法设备总体利用率为72.4%，相对提高了12.8%。

(4)与文献[25]关键设备工序紧凑的动态调度算法对比。文献[25]算法的主要特点是“动态”，即在“拟关键路径+最佳适应调度时刻”的基础上，将最先确定的关键产品部分加工后，再根据设备上对应工序总加工用时的长短动态调整关键产品和关键设备。下面，本文以文献[25]中的复杂产品A和B实例为例，如图12所示，与本文算法进行对比分析。结果同样表明，本文算法更优。

采用本文算法，工序调度顺序为{A1, A4, A3, B1, A2, A5, B2, B3, A6, A7, B4, B5, A9, A8, A10, B6, B7, B8, A11}，调度甘特图如图13所示，总加工用时为235工时。

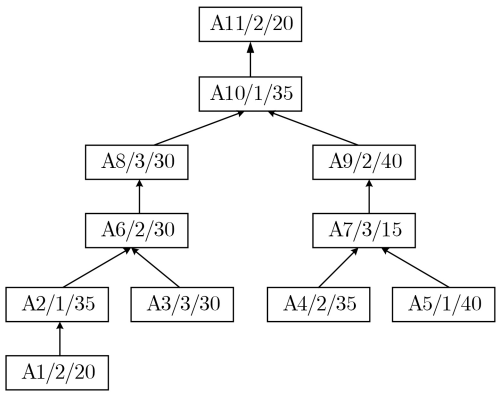
由文献[25]原文可知，在t=55时刻动态调整关键设备，在t=130时刻动态调整关键产品，在t=55时刻调度效果达到最佳，调度甘特图如图14所示，总加工用时为255工时。

对比分析调度甘特图13和图14：在图13中，多工序设备M_2上的总加工用时为235工时，在t=0时刻，设备M_1、设备M_2和设备M_3均开始加工工序；在图14中关键设备M_2上的总加工用时为255工时，设备M_1从t=0时刻t=20时刻处在空闲状态。本文算法中叶节点工序B2在图13中在t=0时刻开始加工，那么与工序B2具有紧后约束关系的工序(组)也较早开始加工，例如工序B3在图13中较在图14中提前了100个工时，所以本文算法通过优先调度多工序设备上的叶节点工序，联动了其具有较高紧后约束关系的工序(组)也实现了较早开始加工，进而减少了复杂产品的总加工用时。设备利用率对比分析如表6所示，本文算法较文献[25]算法的设备总体利用率相对提高了10.3%。

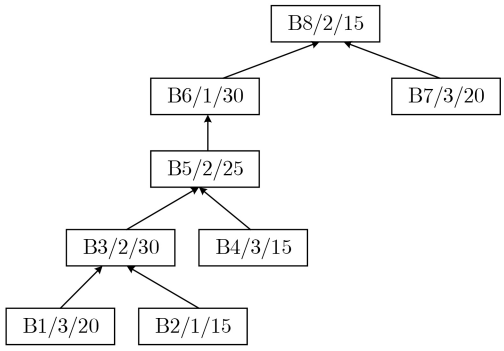
5.4 本文算法、文献[24]算法和文献[25]算法结果对比分析

通过与文献[24]和文献[25]的算法对比分析，之所以本文算法更优，主要是因为：

(1)本文算法采用了“层优先策略”纵横优化的思想，而文献[24]和文献[25]的算法均采用的“拟



(a) 文献[25]原文实例复杂产品A加工工艺树



(b) 文献[25]原文实例复杂产品B加工工艺树

图 12 文献[25]原文实例复杂产品A和B加工工艺树

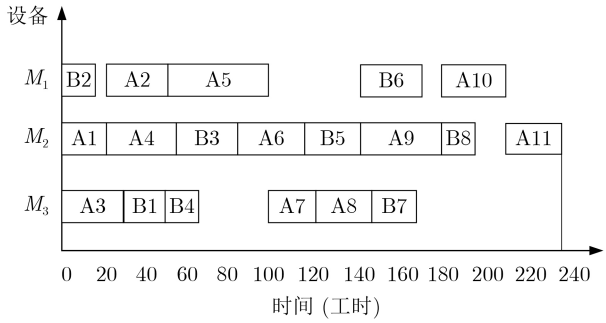


图 13 本文算法加工文献[25]原文实例甘特图 235工时

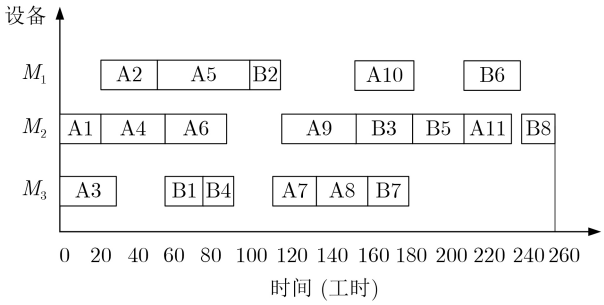


图 14 文献[25]在t=55时刻加工甘特图 255工时

表 6　本文算法和文献[25]算法设备利用率对比分析

	设备M_2的空闲时间总和	M_2利用率(%)	M_2利用率相对提高率(%)	设备总体利用率(%)	设备总体利用率相对提高率(%)	复杂产品加工总工时
本文算法	20	91.5	7.2	86.6	10.3	235
关键设备工序紧凑的动态调度算法	40	84.3		76.3		255

关键路径”策略是纵向优化的思想，从而拉长了串行工序的加工空隙。例如在图11中，设备M_1在t=0～t=20时刻、t=40～t=110时刻，设备M_2在t=20～t=85时刻，设备M_3在t=135～t=190时刻，设备M_4在t=120～t=215时刻均出现了连续且大段的空闲时间；在图14中，工序B2在t=100时刻才开始加工，比在图13的开始加工时刻延迟了100个工时。

(2)本文算法采用了“权重值策略”，在优先考虑多工序设备加工情况的基础上，又全面考虑了加工工序间的各种约束关系。而文献[25]的算法中只考虑了“仅具有唯一紧前、紧后关系的相关工序和独立工序”，也就是仅考虑了约束度小于等于2的节点工序情况，这必然会影响约束性强的工序，导致工序间的紧密衔接性较弱。例如在图13中，因工序B2的优先调度，既利用了设备的空隙又使其紧后工序B3较早开始调度，从而联动了工序B3的后续工序(组)也得到了较早的调度，有效地减少了设备上的加工空隙，缩短了综合调度总体加工用时。

(3)本文算法采用了“最佳调度时刻调整策略”，首先保证了工序(组)紧前关系的约束条件，而在设备空闲时间段查找并插入相关工序则是完全利用设备的调度空隙，充分弥补了文献[24]和文献[25]的算法因加工工序拉伸或者后移而产生的串行工序间的空隙。文献[24]算法采用的“最佳适应调度”策略，当层优先级较低的叶节点工序不是“独立工序”时，“工序紧凑”的策略则会失效。例如图11中设备M_3上的叶节点工序A13，因其不是独立工序，不能调整到最佳时刻开始加工，因而影响了综合调度的整体效果。

6　结论

本文在综合调度中，以多工序设备为优化对象，以最小化复杂产品总加工用时和提高设备整体利用率为优化目标，在全面考虑制造系统中产品工艺、加工工序、设备运行情况和工序约束关系等诸多因素基础上，提出了考虑多工序设备权重的资源协同综合调度算法。在纵向优化方面，以优先级和权重值为主的排序策略，有效地减少了工序间的加工空隙；在横向优化方面，最佳调度时刻策略有效地提高了设备利用率，因此算法进一步优化了综合调度效果。同时，也为深入研究复杂产品的综合调度问题拓展了思路，具有一定的理论和实际意义。

参 考 文 献

[1] 王卓昊, 杨冬菊, 徐晨阳. 基于ISE算法的分布式ETL任务调度策略研究[J]. 计算机科学, 2019, 46(12): 1–7. doi: 10.11896/jsjkx.190100023.

WANG Zhuohao, YANG Dongju, and XU Chenyang. Research on distributed ETL tasks scheduling strategy based on ISE algorithm[J]. *Computer Science*, 2019, 46(12): 1–7. doi: 10.11896/jsjkx.190100023.

[2] 胡致远, 胡文前, 李香, 等. 面向业务可达性的广域工业互联网调度算法研究[J]. 电子与信息学报, 2021, 43(9): 2608–2616. doi: 10.11999/JEIT200583.

HU Zhiyuan, HU Wenqian, LI Xiang, *et al.* Research on wide area industrial internet scheduling algorithm based on service reachability[J]. *Journal of Electronics & Information Technology*, 2021, 43(9): 2608–2616. doi: 10.11999/JEIT200583.

[3] 王萍, 陆岩, 王帅, 等. 一种基于预留-重用联合的C-V2X通信Q学习型半持续调度算法[J]. 电子与信息学报.待发表. doi: 10.11999/JEIT210543.

WANG Ping, LU Yan, WANG Shuai, *et al.* A reservation and reuse combined Q-learning semi persistent scheduling for C-V2X communication[J]. *Journal of Electronics & Information Technology*. To be published. doi: 10.11999/JEIT210543.

[4] WEN Hubin. On the innovative practice of intelligent manufacturing technology in precision mold design and manufacture[C]. 2021 8th International Conference on Machinery, Mechanics, Materials, and Computer Engineering, Huhhot, China, 2021(27): 82–85. doi:10.25236/MMMCE.2021.

[5] 邓世龙, 赵士杰, 赵雪娇, 等. 多核并行实时仿真平台多任务调度管理系统研究与设计[C]. '21全国仿真技术学术会议论文集, 贵阳, 2021: 48–52. doi: 10.26914/c.cnkihy.2021.019139.

DENG Shilong, ZHAO Shijie, ZHAO Xuejiao, *et al.* The design of multi-task management system based on multi-core parallel real-time simulation platform[C]. Conference on '21 National Conference on Simulation Technology, Guiyang, China, 2021: 48–52. doi: 10.26914/c.cnkihy.2021.

019139.

[6] SHI Hongyan and JING Dongxu. Research on the construction and key technologies of cloud manufacturing platform for intelligent manufacturing[C]. The 2021 10th International Conference on Applied Science, Engineering and Technology, Qingdao, China, 2021: 51–55. doi: 10.2991/AER.K.210817.012..

[7] 苏命峰, 王国军, 李仁发. 边云协同计算中基于预测的资源部署与任务调度优化[J]. 计算机研究与发展, 2021, 58(11): 2558–2570. doi: 10.7544/issn1000-1239.2021.20200621.
SU Mingfeng, WANG Guojun, and LI Renfa. Resource deployment with prediction and task scheduling optimization in edge cloud collaborative computing[J]. *Journal of Computer Research and Development*, 2021, 58(11): 2558–2570. doi: 10.7544/issn1000-1239.2021.20200621.

[8] 唐伦, 魏延南, 谭颀, 等. H-CRAN网络下联合拥塞控制和资源分配的网络切片动态资源调度策略[J]. 电子与信息学报, 2020, 42(5): 1244–1252. doi: 10.11999/JEIT190439.
TANG Lun, WEI Yannan, TAN Qi, *et al*. Joint congestion control and resource allocation dynamic scheduling strategy for network slices in heterogeneous cloud Raido access network[J]. *Journal of Electronics & Information Technology*, 2020, 42(5): 1244–1252. doi: 10.11999/JEIT190439.

[9] 赵铖泽, 张延华, 李萌, 等. 无人机辅助的物联网系统中计算资源优化调度研究[C]. 第十四届全国信号和智能信息处理与应用学术会议论文集, 北京, 2021: 184–190. doi: 10.26914/c.cnkihy.2021.002967.
ZHAO Chengze, ZHANG Yanhua, LI Meng, *et al*. Computing resource optimization and allocation for internet of thing systems assisted by unmanned aerial vehicle[C]. The 14th National Conference on Signal and Intelligent Information Processing and Applications, Beijing, China, 2021: 184–190. doi: 10.26914/c.cnkihy.2021.002967.

[10] KHORSAND R and RAMEZANPOUR M. An energy-efficient task-scheduling algorithm based on a multi-criteria decision-making method in cloud computing[J]. *International Journal of Communication Systems*, 2020, 33(9): e4379. doi: 10.1002/dac.4379.

[11] 唐红亮, 吴柏林, 胡旺, 等. 基于粒子群优化的地震应急物资多目标调度算法[J]. 电子与信息学报, 2020, 42(3): 737–745. doi: 10.11999/JEIT190277.
TANG Hongliang, WU Bolin, HU Wang, *et al*. Earthquake emergency resource multiobjective schedule algorithm based on particle swarm optimization[J]. *Journal of Electronics & Information Technology*, 2020, 42(3): 737–745. doi: 10.11999/JEIT190277.

[12] DONG Hui. Research on the transformation and upgrading path of traditional industries based on intelligent manufacturing[C]. 2020 8th International Education, Economics, Social Science, Arts, Sports and Management Engineering Conference, Chengdu, China, 2020(64): 142–146. doi: 10.25236/IEESASM.2020.027.

[13] 赵宝锋, 赵倩倩, 赵均, 等. 事件驱动的联产空分装置月调度系统[C]. 2020中国自动化大会(CAC2020)论文集, 上海, 2020: 290–295. doi: 10.26914/c.cnkihy.2020.043608.
ZHAO Baofeng, ZHAO Qianqian, ZHAO Jun, *et al*. Event-driven Monthly Scheduling System for Multiple Air Separation Units[C]. Conference on 2020 China Automation Conference (CAC2020), Shanghai, China, 2020: 290–295. doi: 10.26914/c.cnkihy.2020.043608.

[14] CHUA P C, MOON S K, NG Y T, *et al*. Prediction of production performance in smart manufacturing using multivariate adaptive regression spline[C]. ASME 2021 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Virtual, Online, 2021. doi: 10.1115/DETC2021-69632.

[15] 李小林, 司佳佳, 尹传传, 等. 考虑工件释放时间和柔性维护的单机调度问题[J/OL]. 计算机集成制造系统. http://kns.cnki.net/kcms/detail/11.5946.TP.20210621.1722.004.html, 2021.
LI Xiaolin, SI Jiajia, YIN Chuanchuan, *et al*. Single-machine scheduling problem considering jobs' release times and flexible maintenances[J/OL]. Computer Integrated Manufacturing System. http://kns.cnki.net/kcms/detail/11.5946.TP.20210621.1722.004.html, 2021.

[16] 谢志强. 工件间有约束的复杂产品工序调度研究[D]. [博士论文], 哈尔滨理工大学, 2009.
XIE Zhiqiang. Study on operation scheduling of complex product with constraint among jobs[D]. [Ph. D. dissertation], Harbin University of Science and Technology, 2009.

[17] XIE Zhiqiang, YANG Dan, MA Mingrui, *et al*. An improved artificial bee colony algorithm for the flexible integrated scheduling problem using networked devices collaboration[J]. *International Journal of Cooperative Information Systems*, 2020, 29(01n02): 2040003. doi: 10.1142/S0218843020400031..

[18] GAO Yilong, XIE Zhiqiang, LIU Xinyang, *et al*. Integrated scheduling algorithm based on the priority constraint table for complex products with tree structure[J]. *Advances in Mechanical Engineering*, 2020, 12(12): 1–15. doi: 10.1177/1687814020985206.

[19] 谢志强, 邵侠. 一种考虑资源状态动态反馈的云计算调度算法[J]. 北京理工大学学报, 2019, 39(6): 624–631,637. doi: 10.15918/j.tbit1001-0645.2019.06.012.
XIE Zhiqiang and SHAO Xia. A scheduling algorithm for the cloud computing system with the dynamical feedback of

server state[J]. *Transactions of Beijing Institute of Technology*, 2019, 39(6): 624–631,637. doi: 10.15918/j.tbit1001-0645.2019.06.012.

[20] 谢志强, 滕宇峥, 杨静. 紧密衔接工序组联动的综合调度算法[J]. 自动化学报, 2011, 37(3): 371–379. doi: 10.3724/SP.J.1004.2011.00371.
XIE Zhiqiang, TENG Yuzheng, and YANG Jing. Integrated scheduling algorithm with no-wait constraint operation group[J]. *Acta Automatica Sinica*, 2011, 37(3): 371–379. doi: 10.3724/SP.J.1004.2011.00371.

[21] 谢志强, 张晓欢, 高一龙, 等. 考虑串行工序紧密度的择时综合调度算法[J]. 机械工程学报, 2018, 54(6): 191–202. doi: 10.3901/JME.2018.06.191.
XIE Zhiqiang, ZHANG Xiaohuan, GAO Yilong, *et al*. Time-selective integrated scheduling algorithm considering the compactness of serial processes[J]. *Journal of Mechanical Engineering*, 2018, 54(6): 191–202. doi: 10.3901/JME.2018.06.191.

[22] 谢志强, 辛宇, 杨静. 基于设备空闲事件驱动的综合调度算法[J]. 机械工程学报, 2011, 47(11): 139–147. doi: 10.3901/JME.2011.11.139.
XIE Zhiqiang, XIN Yu, and YANG Jing. Integrated scheduling algorithm based on event-driven by machines' idle[J]. *Journal of Mechanical Engineering*, 2011, 47(11): 139–147. doi: 10.3901/JME.2011.11.139.

[23] 谢志强, 辛宇, 杨静. 可回退抢占的设备驱动综合调度算法[J]. 自动化学报, 2011, 37(11): 1332–1343. doi: 10.3724/SP.J.1004.2011.01332.
XIE Zhiqiang, XIN Yu, and YANG Jing. Machine-driven integrated scheduling algorithm with rollback-preemptive[J]. *Acta Automatica Sinica*, 2011, 37(11): 1332–1343. doi: 10.3724/SP.J.1004.2011.01332.

[24] 谢志强, 李树生, 刘胜辉. 关键设备工序紧凑的调度算法[J]. 哈尔滨理工大学学报, 2003, 8(1): 37–41,45. doi: 10.3969/j.issn.1007-2683.2003.01.011.
XIE Zhiqiang, LI Shusheng, and LIU Shenghui. A scheduling algorithm based on key equipment compact procedures[J]. *Journal of Harbin University of Science and Technology*, 2003, 8(1): 37–41,45. doi: 10.3969/j.issn.1007-2683.2003.01.011.

[25] 谢志强, 郑翘, 刘胜辉. 关键设备工序紧凑的动态调度算法[J]. 哈尔滨理工大学学报, 2003, 8(2): 50–53. doi: 10.3969/j.issn.1007-2683.2003.02.016.
XIE Zhiqiang, ZHENG Qiao, and LIU Shenghui. A dynamic scheduling algorithm based on key equipment's compact procedures[J]. *Journal of Harbin University of Science and Technology*, 2003, 8(2): 50–53. doi: 10.3969/j.issn.1007-2683.2003.02.016.

[26] XIE Zhiqiang, YANG Guang, and TAN Guangyu. An algorithm of JSSP with dynamic collection of job with priority[C]. The International Technology and Innovation Conference 2006, Hangzhou, China, 2006: 398–402. doi: 10.1049/cp:20060793.

[27] 谢志强, 杨静, 杨光, 等. 可动态生成具有优先级工序集的动态Job-Shop调度算法[J]. 计算机学报, 2008, 31(3): 502–508. doi: 10.3321/j.issn:0254-4164.2008.03.016.
XIE Zhiqiang, YANG Jing, YANG Guang, *et al*. Dynamic job-shop scheduling algorithm with dynamic set of operation having priority[J]. *Chinese Journal of Computers*, 2008, 31(3): 502–508. doi: 10.3321/j.issn:0254-4164.2008.03.016.

[28] LUOR D C. A comparative assessment of data standardization on support vector machine for classification problems[J]. *Intelligent Data Analysis*, 2015, 19(3): 529–546. doi: 10.3233/IDA-150730.

周　伟：女，1979年生，博士生，研究方向为计算机集成制造.
谢志强：男，1962年生，教授，博士生导师，研究方向为企业智能计算和调度优化、数据库和知识工程.

责任编辑：余　蓉