

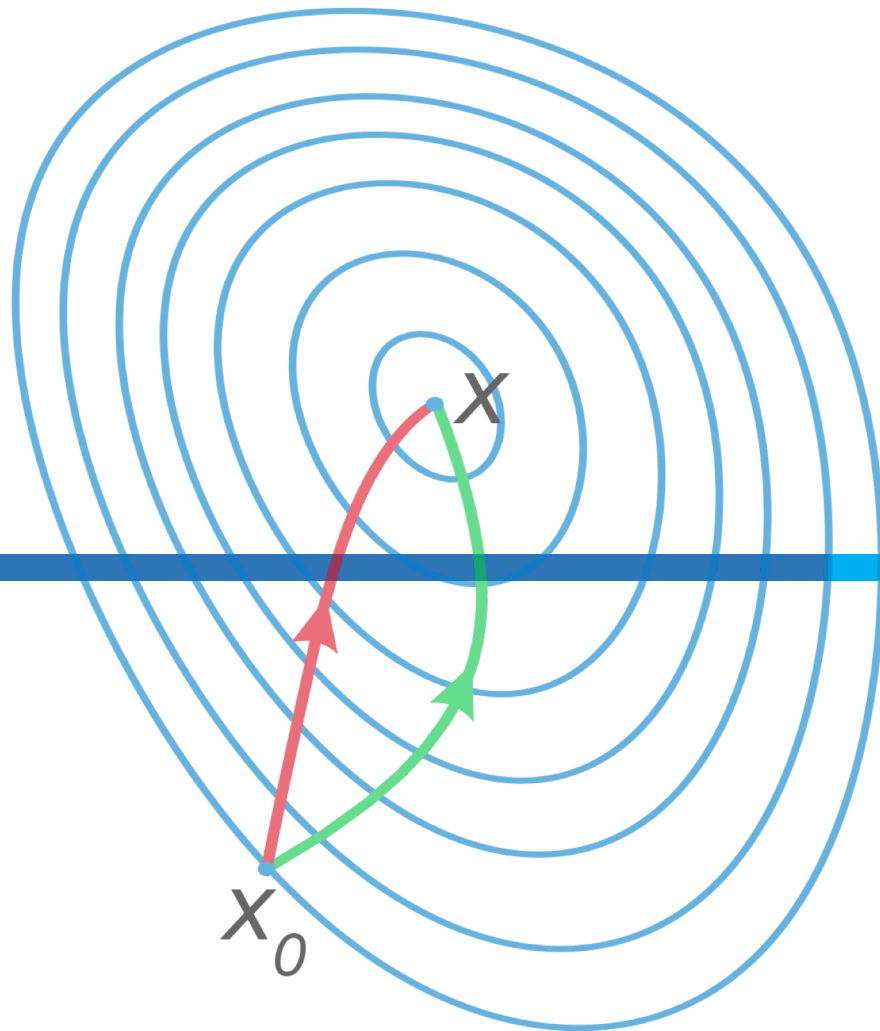
最优化方法

大作业

计算机学院

余皓然

2023/4/6

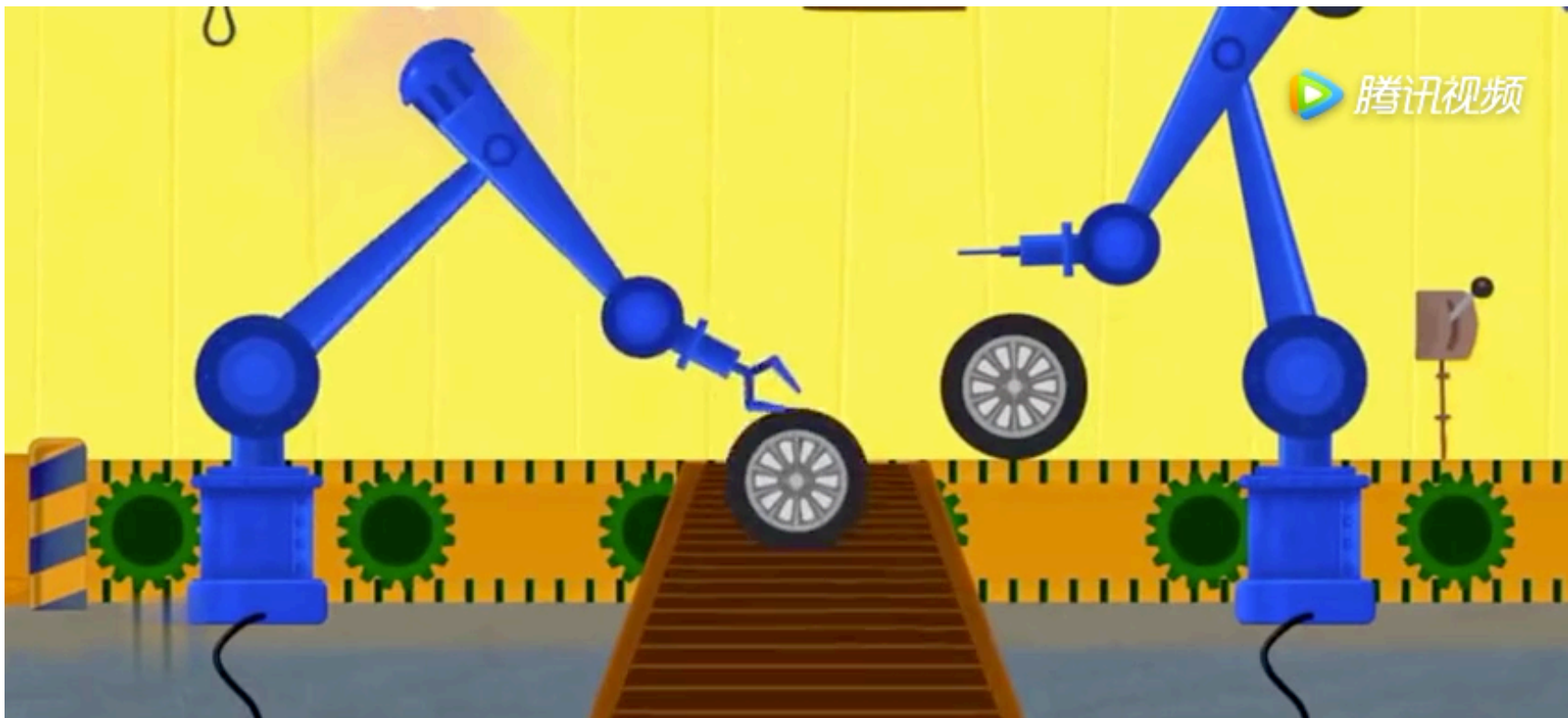


实验课

1. 置换流水车间调度问题
2. 拓展题
3. 大作业要求

问题描述

流水车间调度 (Flow Shop Scheduling)



m 台机器依次加工1个工件（1辆车）的例子

问题描述

m台机器加工n个工件的流水车间调度

每个工件需要依次经过 m 道工序，每道工序要求不同的机器加工，每个机器同时只能加工一个工件，一个工件不能同时在不同的机器上加工。每个工件在每台机器上的加工时间是给定的。

求 n 个工件在每台机器上的加工顺序，最小化总完工时间（完成最后一个工件加工的时刻 减 开始对第一个工件加工的时刻）

问题描述

- 已知
 - 有 n 个工件需要在 m 台机器上流水加工
 - 每个工件的**加工顺序相同**，从第一台机器至最后一台机器
 - 每个工件均在**0时刻释放**
 - 每个工件在每台机器上**只加工一次**
 - 一个工件**不能同时**在不同的机器上加工
 - 每个机器同时**只能加工一个工件**
 - 各个工件在各个机器上的加工时间（共 nm 个数字）
- 目标
 - 给出调度方案（即 n 个工件的排列顺序），使**总完工时间最小**
- 要求
 - 使用**模拟退火算法**进行求解

问题描述-例子

- 3台机器
- 3个工件
- 加工时间表

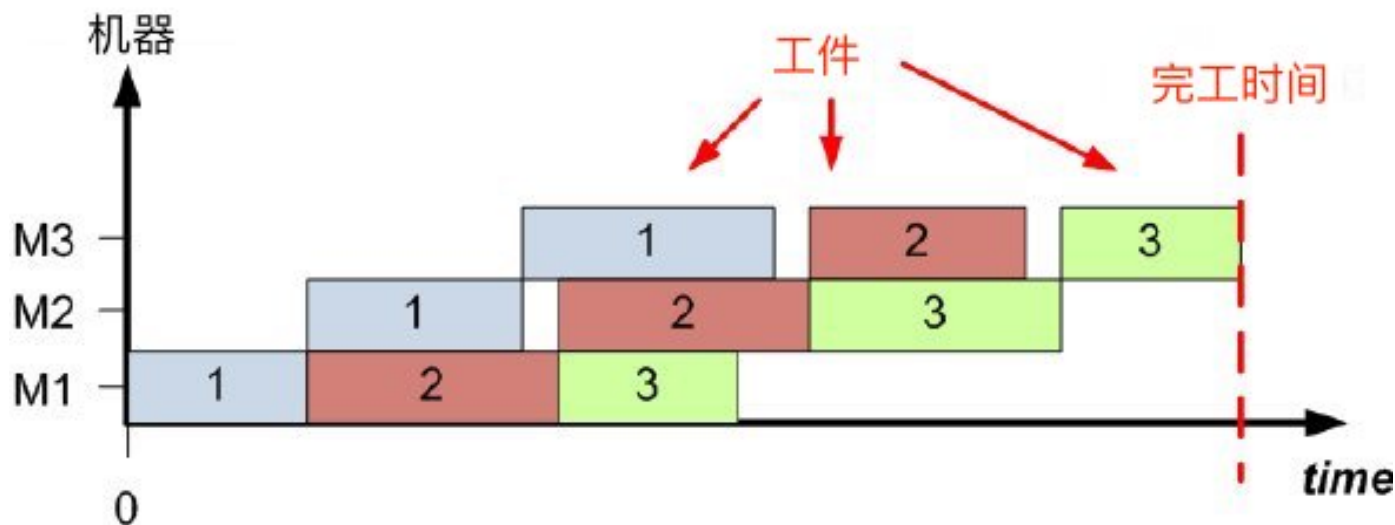
工件	机器1	机器2	机器3
工件1	5	7	9
工件2	9	9	7
工件3	5	9	5

问题描述-例子

- 3台机器
- 3个工件
- 加工时间表

工件	机器1	机器2	机器3
工件1	5	7	9
工件2	9	9	7
工件3	5	9	5

调度方案为1→2→3时的加工情况（甘特图）



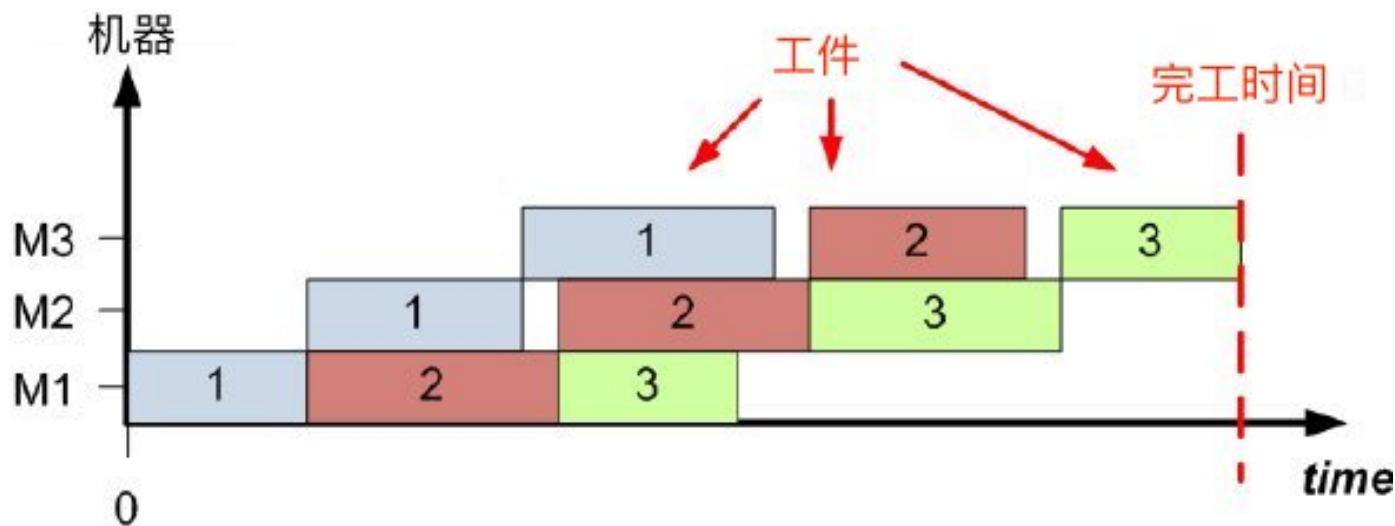
注意：这只是例子，工件1不是必须放在第一个进行加工

问题描述-例子

- 3台机器
- 3个工件
- 加工时间表

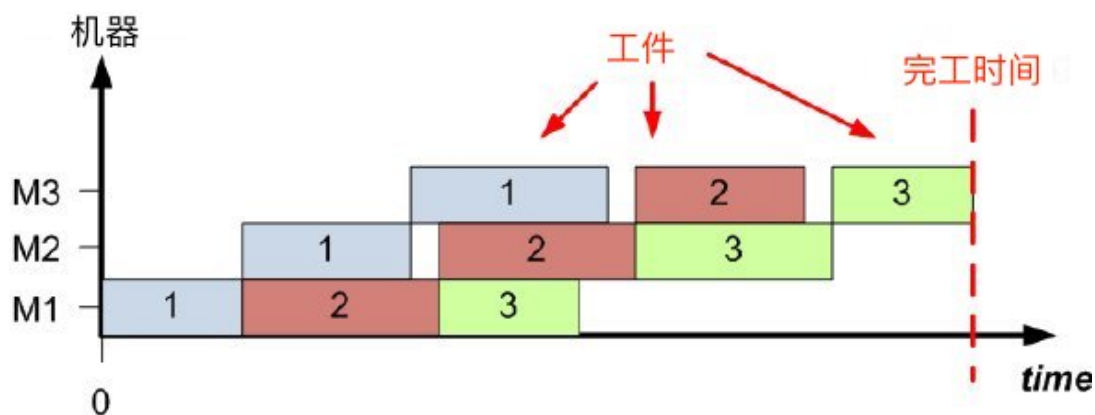
工件	机器1	机器2	机器3
工件1	5	7	9
工件2	9	9	7
工件3	5	9	5

调度方案为1→2→3时的加工情况（甘特图）



此调度方案对应时间为37

问题描述-例子



这里我们把解空间限定为由 n 个工件的各种排列顺序构成的 $|n!|$ 大小的集合



如果允许在不同机器上用不同的顺序加工工件，是否可以进一步减少完工时间？

比如在上例中，M1和M2的顺序是 $1 \rightarrow 2 \rightarrow 3$ ，M3的顺序是 $2 \rightarrow 1 \rightarrow 3$

PFSP与NPFSP

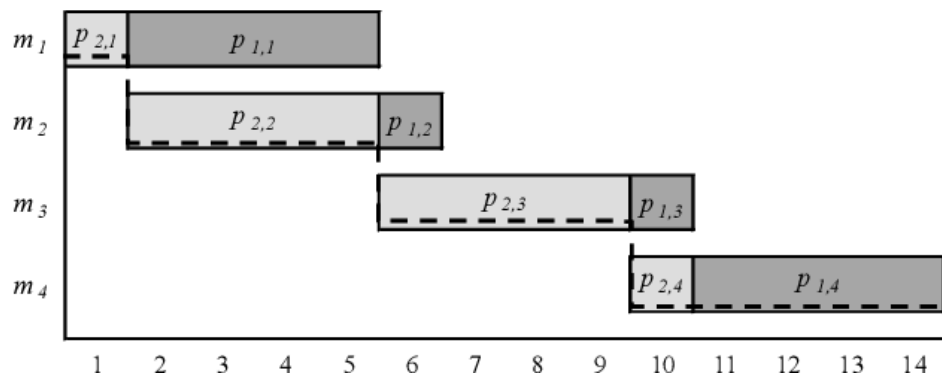
置换流水车间调度问题（PFSP）：把解空间限定在 n 个工件各种排列顺序构成的集合

非置换流水车间调度问题（NPFSP）：允许不同机器用不同的顺序加工工件

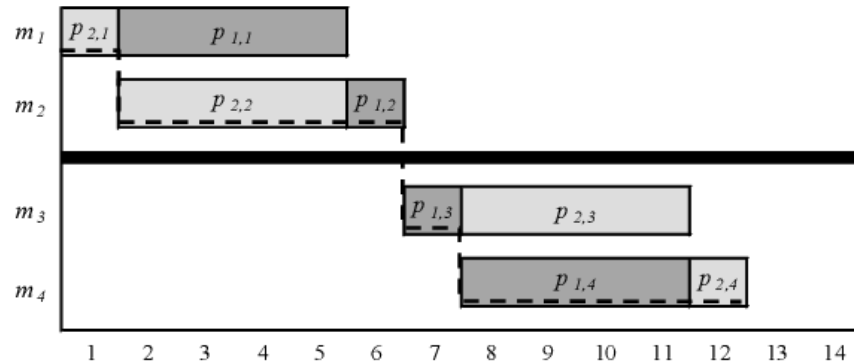
PFSP与NPFSP

置换流水车间调度问题（PFSP）：把解空间限定在n个工件各种排列顺序构成的集合

非置换流水车间调度问题（NPFSP）：允许不同机器用不同的顺序加工工件



permutation

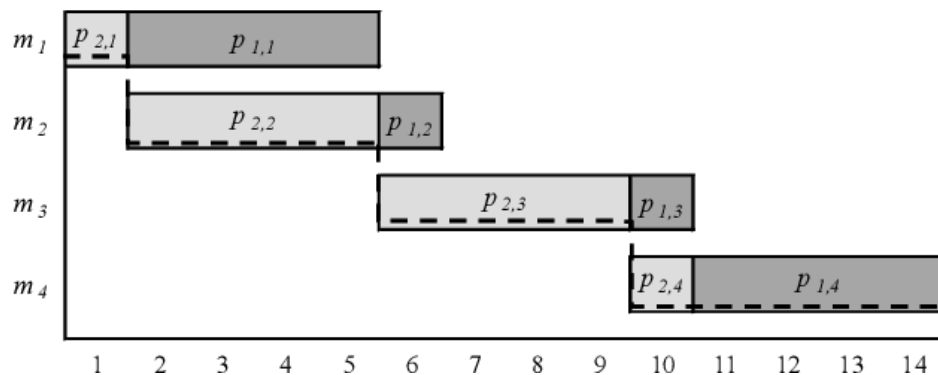


non-permutation

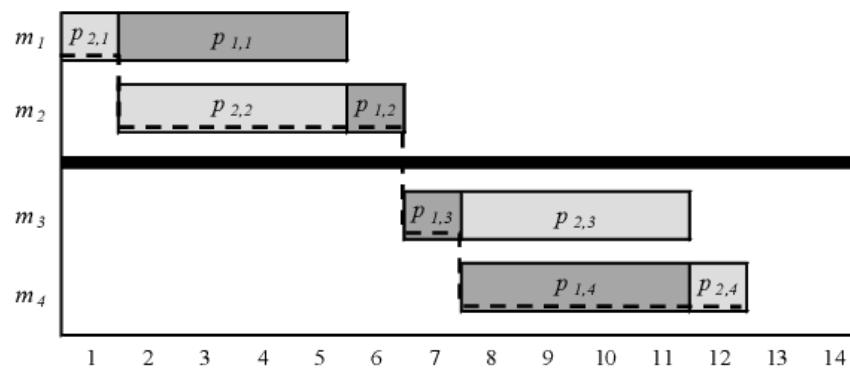
PFSP与NPFSP

置换流水车间调度问题（PFSP）：把解空间限定在n个工件各种排列顺序构成的集合

非置换流水车间调度问题（NPFSP）：允许不同机器用不同的顺序加工工件



permutation



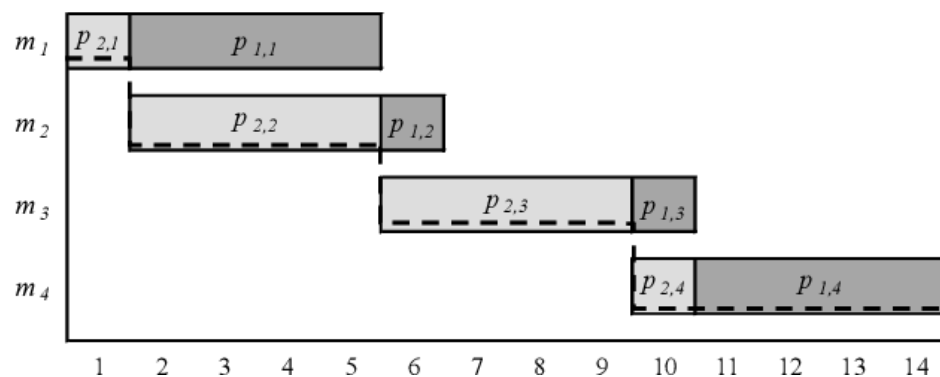
non-permutation

令机器3加工工件2 与 机器4加工工件1
同步进行，缩短时间

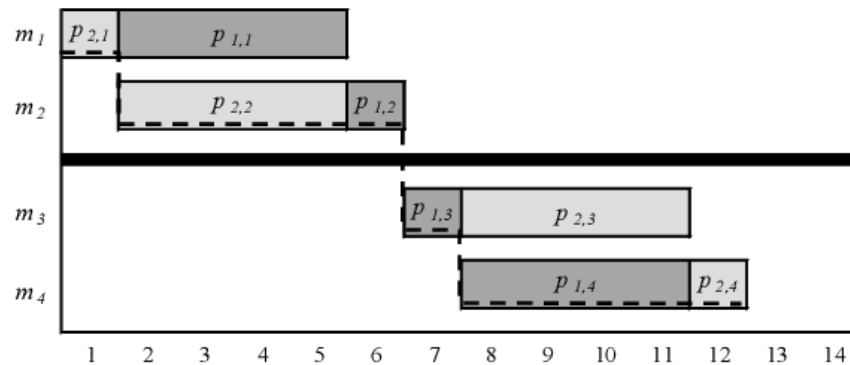
PFSP与NPFSP

置换流水车间调度问题（PFSP）：把解空间限定在n个工件各种排列顺序构成的集合

非置换流水车间调度问题（NPFSP）：允许不同机器用不同的顺序加工工件



permutation



non-permutation

大作业只考虑置换流水车间问题
(即解空间更小的情况)

用例说明

- 每个用例的第一行为两个数字：工件数 n ，机器数 m
- 接下来有 n 行，每行包含 m 组数字（机器序号，加工时间）

- instance 0

+++++

11 5

0 375 1 12 2 142 3 245 4 412

0 632 1 452 2 758 3 278 4 398

0 12 1 876 2 124 3 534 4 765

0 460 1 542 2 523 3 120 4 499

0 528 1 101 2 789 3 124 4 999

0 796 1 245 2 632 3 375 4 123

0 532 1 230 2 543 3 896 4 452

0 14 1 124 2 214 3 543 4 785

0 257 1 527 2 753 3 210 4 463

0 896 1 896 2 214 3 258 4 259

0 532 1 302 2 501 3 765 4 988

+++++

第一行表示有**11**个工件，**5**个机器

第二行表示第一个工件在机器**0**的加工时间为**375**单位时间，在机器**1**的加工时间为**12**单位时间，在机器**2**的加工时间为**142**单位时间，在机器**3**的加工时间为**245**单位时间，在机器**4**的加工时间为**412**单位时间

这个示例的最优加工时间为**7038**单位时间

- optimum result: 7038

实验课

1. 置换流水车间调度问题

2. 拓展题

3. 大作业要求

无等待约束

无等待置换流水车间调度问题（**no-wait** permutation flow shop scheduling problem）

- 在原问题基础上额外加上**无等待约束**，即任意工件相邻的两个加工工序之间不允许有等待

Job	Process time on			
	M1	M2	M3	M4
1	2	4	8	10
2	3	4	7	11
3	2	6	9	12
4	1	5	9	13
5	3	6	8	14

例子

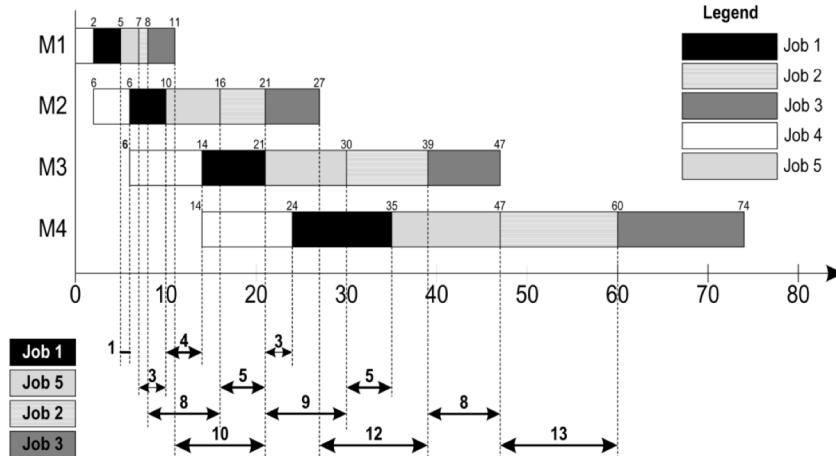
无等待约束

Job	Process time on			
	M1	M2	M3	M4
1	2	4	8	10
2	3	4	7	11
3	2	6	9	12
4	1	5	9	13
5	3	6	8	14

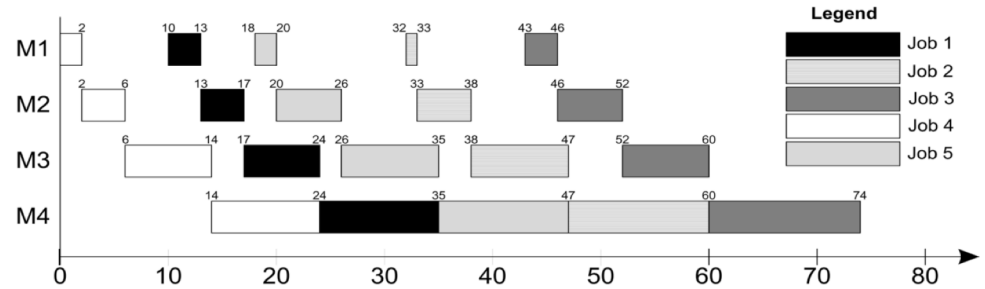
无等待置换流水车间调度问题（**no-wait** permutation flow shop scheduling problem）

- 在原问题基础上额外加上**无等待约束**，即任意工件相邻的两个加工工序之间不允许有等待

加工顺序都是4→1→5→2→3



不满足无等待约束



满足无等待约束

问题的解空间没变，但完工时间的计算方式、最优解、甘特图可能会变

无等待约束

Job	Process time on			
	M1	M2	M3	M4
1	2	4	8	10
2	3	4	7	11
3	2	6	9	12
4	1	5	9	13
5	3	6	8	14

无等待置换流水车间调度问题（**no-wait** permutation flow shop scheduling problem）

- 在原问题基础上额外加上**无等待约束**，即任意工件相邻的两个加工工序之间不允许有等待

应用场景包括化学药品加工（避免化学物在等待中挥发）、炼钢（避免已经加热的钢件在等待中冷却）、食品生产（避免未装罐食品在等待中变质）

实验课

1. 置换流水车间调度问题
2. 拓展题
3. 大作业要求

评分标准

- 代码（20分）
 - 有效性（14分）（在报告中明确给出14个例子的调度方案、加工完成时间、甘特图）
 - 代码注释（6分）有适当注释，增加可读性

不允许直接调用模拟退火程序包，代码要有算法过程

- 报告（50分）
 - 具体要求见报告模版说明（严格按得分点给分）

提交要求

- 截止时间

- 5月28日23:59（第十四周周日）
- 过期补交0分

- 提交方式

- 乐学平台提交
- 分两个文件上传
 - 报告（把word转成PDF格式再提交，不要交word格式）
 - 代码（压缩包形式）

- 严禁抄袭

- 抄袭：与任何他人、网络上（如CSDN）的内容明显相似
- 抄袭则大作业0分（挂科）：如果A抄B，则A和B两人大作业都是0分
- 2021年10人挂科/2022年15人挂科，大多为抄袭或缺交