

半导体制造被公认为当前最复杂的制造过程之一,主要包括 4 个阶段:晶圆加工(即前端工艺,常常被称作半导体生产线)、中试、装配(即后端工艺)以及最终产品测试。其中,半导体生产线作为半导体制造最昂贵、最复杂也是最耗时的部分,具有高度不确定性、多重入、混合加工方式、设备负载不均衡等明显区别于其他制造业的显著特点,其控制与调度问题得到学术界与工业界普遍关注。批加工设备调度是半导体生产线调度中的重要问题之一,对半导体生产线的性能有重要影响。批加工设备是半导体制造系统中的一类特殊设备,该类设备可同时加工数卡工件,相应地,其调度具有自己的特点。本章以批加工设备调度为对象,介绍批加工设备调度、基于 ACO 的批加工设备调度方法、基于柔性 ACO 的批加工设备调度方法以及基于 ACO-ICSA 的批加工设备调度方法。

5.1 概述

在半导体生产线的氧化(oxidation)、扩散(diffusion)与淀积(deposition)等加工区域存在一类并行批量加工设备,即批加工设备(Batch Processing Machines, BPM)。批加工设备与非批加工设备的并存,导致了半导体生产线上加工流的不平稳。在批加工设备处,只要工件待加工工序所使用的设备上的工艺菜单相同,就可以组批加工。批加工设备的加工时间与一起加工的工件数目无关,一般是非批加工设备加工时间的 5~10 倍。最大加工批量,即设备加工能力,是批加工设备的重要参数,组批工件的数量不能超过批加工设备的最大加工批量。批加工设备一旦开始加工,就不能再增加或减少工件,并且工件加工是非抢占式的,也就是说,只有正在加工的工件完成加工后,才可加工下一批工件。因此,批加工设备的调度对半导体生产线的制造性能有很重要的影响,从而得到了普遍重视。

综上,批加工设备的加工满足以下特征:

(1) 在批加工设备处,只有同类工件才可以同时加工,并且同时加工的各工件当前工序使用的设备上的工艺菜单相同。

(2) 批加工设备的加工时间与一起加工的工件的数目是无关的,这个时间一般是非批加工设备加工时间的 5~10 倍。

(3) 最大加工批量,即设备加工能力,是批加工设备的重要参数,组批工件的数量不能超过批加工设备的最大加工批量。

(4) 批加工设备一旦开始加工,就不能再增加或减少工件,并且工件加工是非抢占式的,也就是说,只有正在加工的工件完成加工后,才可加工下一批工件。

带有批加工设备的简化半导体生产线模型如图 5-1 所示,生产线上可能存在多台设备能够完成工件的某些工序,这些设备共享一个缓冲区,当设备出现空闲时,从相应的缓冲区中取出合适的工件进行加工。

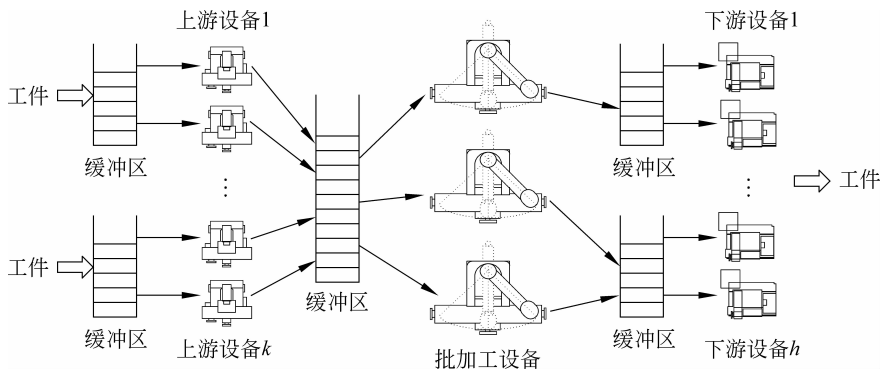


图 5-1 带有批加工设备的半导体生产线简化模型

批加工设备调度主要需要解决两个问题：一是工件如何组批，二是确定批次加工优先级。目前,批加工设备调度已成为半导体制造调度领域的研究热点之一,其研究经历了不同阶段:单台批加工设备调度、并行批加工设备调度与带有批加工设备的半导体生产线调度。其中,单台批加工设备调度与并行批加工设备调度主要是基于数学规划、计算智能、离散事件仿真技术与启发式规则的,其调度目标是最小化工件总的加权延迟或总的等待成本;带有批加工设备的半导体生产线调度的研究则主要是基于启发式规则的,动态决定批加工设备的加工批量与加工次序。

5.2 批加工设备调度描述

5.2.1 单台批加工设备调度

1. 基于启发式规则的单台批加工设备调度

1) 问题描述

这里介绍加工不同产品类型工件的批加工设备调度问题。具体的调度问题描述如下。

(1) 存在 m 类不同产品类型的工件,只有相同产品类型的工件才能并批加工。

(2) 对于不同产品类型的工件,设备的最大加工能力可能不同;使用 K_j 表示设备加

工属于产品类型 j 的工件时的最大加工批量。

(3) 属于产品类型 j 的工件的加工时间是随机分布的, 平均值为 $1/\mu_j$, 并且服从累积分布函数 F_j 。

(4) 假定所有的整定时间都已包括在加工时间中, 即当先后加工的工件产品类型不同时, 不发生整定时间或成本。

(5) 调度是非抢占式的, 即只有当正在加工的工件完成加工后, 才能开始下一批工件的加工。

(6) 属于产品类型 c_j 的每个工件的单位时间等待成本为 c_j , 对于所有的 $j, c_j > 0$ 。

(7) 假定设备的缓冲区是无限制的。

(8) 假定当前待加工的工件是已知的, 并且将来不会有工件到来。

(9) 调度的目标是 minimized 单位时间长期平均等待成本。也就是说, 如果设所有的属于产品类型 j 的工件的单位等待成本 $c_j = 1$, 其目标即为最小化工件的平均等待时间以及最小化平均等待的工件的数目。

2) 问题求解

假设批加工设备前有 n 个待加工工件, n_j 表示属于产品类型 j 的工件的数目, 批加工设备的最大加工批量为 B 。则存在以下两个命题。

命题 1 在调度方案中, 如果除了每种产品类型工件的最后一批工件以外, 所有的批都是满足最大加工批量的, 则存在优化调度方案。

证明: 对于属于产品类型 j 的工件, 任意选取工件组成批量为 $\lceil n_j/B \rceil$ 的一批满足最大加工批量的工件和批量为 $n_j - \lceil n_j/B \rceil B$ 的一批不满足最大加工批量的工件。由于相同产品类型的工件的单位时间等待成本是相同的, 因此如何将工件组批对获得优化调度方案是没有影响的。

命题 2 使用下述过程获得优化调度方案。

步骤 1: 对于属于每种产品类型 j 的工件, 任意组成批量为 $\lceil n_j/B \rceil$ 的满足最大加工批量的批以及一个批量为 $n_j - \lceil n_j/B \rceil B$ 的非最大加工批量的批。

步骤 2: 按照 $W_{B_k} \mu_j$ 降序排列 $\sum_{i=1}^m \lceil n_j/B \rceil$, W_{B_k} 是属于产品类型 j 的第 k 批组批工件的等待成本之和, $1/\mu_j$ 是属于产品类型 j 的工件的平均加工时间。当批满足最大加工批量时, $W_{B_k} = c_j B$; 否则, $W_{B_k} = (n_j - \lceil n_j/B \rceil B) c_j$ 。

2. 基于动态规划的单台批加工设备调度

1) 问题描述

仍然以加工不同产品类型工件的批加工设备调度问题为研究对象。具体的调度问题描述如下。

(1) 存在 n 个待加工工件, 这些工件属于 m 类不同的产品类型。

(2) 属于产品类型 j 的工件数目为 n_j , 加工时间为 p_j 。

(3) 批加工设备的最大加工批量为 B 。

(4) 只有属于相同产品类型的工件才能够组批加工, 每批工件的数目不能超过批加工设备的最大加工批量。

(5) 属于产品类型 j 的工件 i 的交货期为 d_{ij} , 假定同种产品类型的工件按其交货期进行非降排列, 即 $d_{1j} \leq d_{2j} \leq \dots \leq d_{n_j}$ 。

(6) C_{ij} 表示调度方案中属于产品类型 j 的工件 i 的完工时间。

(7) 属于产品类型 j 的工件 i 的延迟表示为 $T_{ij} = \max[0, C_{ij} - d_{ij}]$ 。

(8) 调度的目标是工件总加权延迟最小, 工件总加权延迟表示为 $\sum T_j = \sum_{j=1}^m \sum_{i=1}^{n_j} \omega_j T_{ij}$, 其中 ω_j 是产品类型 j 的权重, $\sum_j \omega_j = 1$ 。

2) 问题求解

在此问题中存在以下几个命题。

命题 1 如果工件类型 m 与批加工设备的能力 B 是任意的, 则带有不同类型工件的单台批加工设备的最小化工件总加权延迟调度问题是 NP 难问题。

命题 2 在静态带有不同类型工件的单台批加工设备的调度中, 是存在优化调度方案的, 如果能够保证调度方案中除每类工件的最后一批工件外, 其他组批都是满足最大加工批量的。

命题 3 如果同种类型工件组批工件都是按照连续次序进行组批的, 则存在优化调度方案。也就是说, 对于所有同类工件的组批, 存在

$$\max_{ij \in B_{kj}} \{d_{ij}\} \leq \max_{ij \in B_{k+1,j}} \{d_{ij}\}$$

其中, B_{kj} 表示调度方案中属于产品类型 j 的工件的第 k 批组批。

命题 4 在任一优化调度方案中, 如果属于产品类型 j 的工件的组批 B_{kj} 排在属于产品类型 t 的工件的组批 B_{st} 之前, 则满足以下条件:

$$p_j \leq p_t; \quad |B_{st}| \leq |B_{kj}|; \quad d_{kj} \leq d_{st}$$

基于以上命题, 在组批时使用贪婪方式, 即按照工件的交货期的非降次序连续选择工件。设 $k_j = \lceil n_j / B \rceil$ 为按贪婪方式对属于产品类型 j 的工件进行组批获得的批数, $f(\omega_1, \omega_2, \dots, \omega_m)$ 表示属于产品类型 j 的工件的组批 1 到组批 ω_j 中 $\sum T_i$ 的最小值。则可以定义以下动态规划方程用于组批工件的排序:

$$\text{边界条件: } f(0, 0, \dots, 0) = 0;$$

$$\text{如果 } \omega_j > k_j \text{ 或 } \omega_j < 0, f(\omega_1, \omega_2, \dots, \omega_m) = \infty$$

$$\text{优化目标: } f(k_1, k_2, \dots, k_m)$$

递推关系: 考虑属于产品类型 j ($j=1, 2, \dots, m$) 的工件的组批 B_{1j} 到组批 $B_{\omega_j, j}$ 的部分调度方案。在应用命题 4 之前, 设 $A(\omega_1, \omega_2, \dots, \omega_m) = \{B_{\omega_j+1, j} | j=1, 2, \dots, m\}$ 是所有满足命题 3 条件的在 $B_{\omega_j, j}$ 之后的组批的集合。设 $\sigma(B_{st})$ 表示使用命题 4 得到的组批 B_{st} 之后的所有组批的集合。那么对于以上部分调度方案, 应用命题 3 和命题 4 后, 满足条件的在该调度方案之后的组批集合表示为

$$\phi(\omega_1, \omega_2, \dots, \omega_m) = A(\omega_1, \omega_2, \dots, \omega_m) \setminus (A(\omega_1, \omega_2, \dots, \omega_m) \cap S(\omega_1, \omega_2, \dots, \omega_m))$$

$$S(\omega_1, \omega_2, \dots, \omega_m) = \bigcup_{j=1}^m \sigma(B_{\omega_j+1, j})$$

其中, $S(\omega_1, \omega_2, \dots, \omega_m)$ 表示被命题 4 剔除的组批的集合。

设 $\Gamma(\omega_1, \omega_2, \dots, \omega_m)$ 表示属于工件类型 s 的组批的集合, 因此 $B_{\omega_s+1, s} \in \phi(\omega_1, \omega_2, \dots, \omega_m)$ 。这种递推关系可以表示为式(5-1)所示:

$$\begin{aligned} f(\omega_1, \omega_2, \dots, \omega_m) = & \min_{q \in \Gamma(\omega_1, \omega_2, \dots, \omega_{q-1}, \dots, \omega_m)} \{f(\omega_1, \omega_2, \dots, \omega_m)\} \\ & + \sum_{iq \in B_{\omega_q, q}} \max\{0, C_{iq} - d_{iq}\} \end{aligned} \quad (5-1)$$

其中, 工件 iq 的完工时间 C_{iq} 定义为

$$C_{iq} = \sum_{k=1}^m \omega_k p_k$$

这种递推关系及时构造了部分调度方案。在最小化表达式(5-1)中, 右边第二项等价于组批调度方案中最后调度的工件的 $\sum T_i$ 值。由于属于相同产品类型的批具有相同的加工时间, 则权值 ω_j 与各类产品的加工时间 p_j 就足以用于计算 $\sum T_i$ 。另外, 每个工件的集合 $\sigma(B_{st})$ 的计算可以在开始进行递推 $O(m^2 B)$ 之前进行, 因为需要找到属于产品类型 m 的第一个未被调度的组批, 并且使用命题 4 与所有已调度的批进行比较, 这种比较每批最多进行 B 次。

$\prod_{j=1}^m (k_j + 1)$ 表示按照命题 4 不存在主导关系。其中 k_j 是属于产品类型 j 的工件的组批数目。已知 $k_j = \lceil n_j / B \rceil$, 则可以得到

$$\prod_{j=1}^m (k_j + 1) = \prod_{j=1}^m \left(1 + \lceil \frac{n_j}{B} \rceil\right) \leq \prod_{j=1}^m \left(2 + \lceil \frac{n_j}{B} \rceil\right) \leq \left(2 + \lceil \frac{n}{mB} \rceil\right)^m$$

作为问题的上界。最后一项不等式是非常有用的, 它给出了第三项乘积的最大值是满足约束 $\sum_j n_j = n$ 的。因此对于给定的 $\sigma(B_{st})$, 比较的次数是 $O(mB)$ 。显然, 在最坏情形下, 动态规划的时间复杂性为

$$O\left(mB \left(2 + \frac{n}{mB}\right)^m\right)$$

因此, 当产品类型 m 与批加工设备能力 B 是固定的情况下, 这种动态规划算法具有多项式复杂性。另外, 这种算法的复杂性与工件的数目无关, 取决于组批的数目。但是, 动态规划算法在解决大规模问题(例如存在许多产品类型)时, 需要很大的计算量。

5.2.2 并行批加工设备调度

在半导体制造生产线上, 最常使用的概念是设备群, 即功能类似的多台设备聚集在一处, 共同完成类似的加工任务。这些设备又称作并行加工设备。半导体制造生产线上的批加工设备, 如氧化炉管等, 也往往是并行批加工设备。本节主要以此为对象, 详细介绍遗传算法与启发式方法相结合的半导体生产线并行批加工设备调度。

1. 问题描述

以加工不同产品类型工件的并行批加工设备调度问题为研究对象。具体的调度问题描述如下。

- (1) 工件具有不同的到达时间。
- (2) 优化目标是 minimized 总加权延迟。
- (3) 属于相同产品类型的工件有相同的加工时间。
- (4) 不同产品类型的工件不能并批加工。
- (5) 并行批加工设备中所有的批加工设备都是完全相同的。
- (6) 设备一旦开始加工,就不能中断(即非抢占式的)。

定义如下变量:

- (1) 共存在 f 种不同的产品类型, n 个待调度工件, 并行设备中存在 m 台相同设备。
- (2) 产品类型 j 的工件 i 的完工时间表示为 C_{ij} , 权值表示为 W_{ij} 。
- (3) 产品类型 j 的第 k 批工件的批量表示为 B_{kj} 。
- (4) 产品类型 j 有 n_j 个待调度工件, $\sum_{j=1}^f n_j = n$ 。
- (5) 产品类型 j 的工件 i 的到达时间表示为 r_{ij} 。
- (6) 产品类型 j 的工件 i 的交货期表示为 d_{ij} 。
- (7) 产品类型 j 的工件的加工时间表示为 p_j 。
- (8) 批加工设备的加工能力为 B 个工件。
- (9) 工件的加权延迟表示为 $w_{ij} T_{ij} = w_{ij} (C_{ij} - d_{ij})^+$, 定义 $x^+ = \max(x, 0)$ 。
- (10) 使用 $\alpha|\beta|\gamma$ 定义形式, 这个问题就可以定义为

$$P_m \mid r_j, \text{batch, incompatible} \mid \sum w_j T_j$$

2. 问题求解

并行批加工设备调度的求解方式可以分为两种: 一种是先将待加工工件组批, 然后将组批工件分派给各个批加工设备, 最后再为每个设备的待加工批次排序, 即三层结构; 另一种是先将工件分派到各批加工设备, 然后在各个设备处将工件组批并排序, 即两层结构。

在这两种结构的求解方式中, 均采取遗传算法与启发式方法相结合的方法, 其中遗传算法负责工件或组批向设备上的分派, 启发式方法则用于将工件组批以及组批工件的排序, 具体的求解过程如图 5-2 所示。

1) 用于工件组批的启发式方法

步骤 1: 在时刻 t , 选择时间窗口 Δt , 并且从各产品类型待调度工件集合 $M(j, t, \Delta t)$ 选出最大数目可供调度的工件集合 $M(j, t, \Delta t, \text{thres})$ 。在选择过程中要计算各种产品类型待调度工件的参数 I_{ij} , 并按降序排列。其中 $M(j, t, \Delta t)$ 和 $M(j, t, \Delta t, \text{thres})$ 分别表示

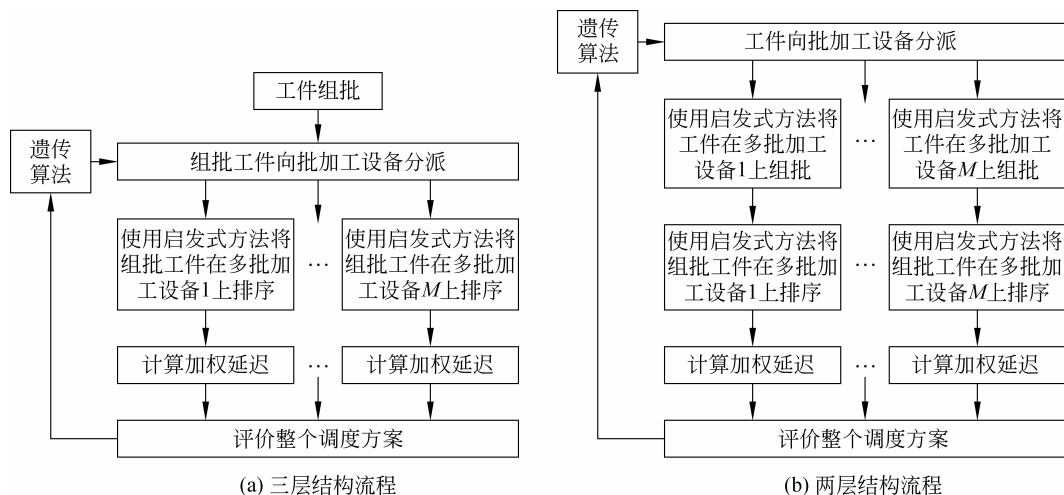


图 5-2 遗传算法与启发式方法相结合的并行批加工设备调度流程

为式(5-2)和式(5-3):

$$M(j, t, \Delta t) = \{ij \mid r_{ij} \leq t + \Delta t\} \quad (5-2)$$

$$M(j, t, \Delta t, \text{thres}) = \{ij \mid ij \in M(j, t, \Delta t) \text{ 且 } \text{pos}(ij) < \text{thres}\} \quad (5-3)$$

其中, $\text{pos}(ij)$ 的计算是基于 I_{ij} 的, 在 t 时刻 I_{ij} 的表示如式(5-4)所示:

$$I_{ij} = \left(\frac{w_{ij}}{p_j} \right) \exp\left(-\frac{(d_{ij} - p_j + (r_{ij} - t))^+}{k \cdot \bar{p}} \right) \quad (5-4)$$

其中, k 是预测系数, \bar{p} 是剩余未调度工件的平均加工时间。使用 I_{ij} 对工件集合 $M(j, t, \Delta t)$ 中的工件进行非降序排列, 取每一个到达 thres 工件之前的所有工件组成工件集合 $M(j, t, \Delta t, \text{thres})$ 。

步骤 2: 对工件集合 $M(j, t, \Delta t, \text{thres})$ 中的工件按产品类型并按 I_{ij} 的降序排列, 考虑批加工设备的加工批量 B , 开始组批。

这样就完成了基于启发式方法的工件组批过程。该方法既可应用于三层结构解决方案中的第一阶段——将所有满足条件的待调度工件组批, 也可应用于二层结构解决方案中的第二阶段——将各批加工设备前待调度工件分别组批。

2) 用于组批工件排序的启发式方法

各组批工件排序的优先级计算如式(5-5)所示:

$$\text{TWT}(B_j) = \sum_{ij \in B_j} w_{ij} (t^* + p_j - d_{ij})^+ + \sum_{kj \in M-B_j} w_{kj} (t^* + p_j + p_i + p^* - d_{ki})^+ \quad (5-5)$$

其中, $t^* = \max_{ij \in B_j}(t, \max(r_{ij}))$ 是组批 B_j 开始加工的时间。 p^* 表示如式(5-6)所示:

$$p^* = \frac{1}{2B_j} \sum_{ki \in M-B_j} p_i \quad (5-6)$$

在式(5-5)中, 第一项代表选中组批工件的加权延迟, 第二项代表其他工件的总加权延迟的估计值, 工件 i 只有在组批 B_j 加工完成并经过 p^* 延迟后才能开始加工。

延迟 p^* 意味着工件 i 并不能在组批 B_j 加工完成后立即开始加工, 因为可能已调度其他工件在工作 i 之前开始加工。 p^* 取决于时间窗口内未调度工件的总的数量。如果未调度工件数量较多, 则 p^* 的值较大; 如果未调度工件数量较少, 则 p^* 的值较小。为所有的组批工件计算 $TWT(B_j)$ 值后, 具有最小值的组批工件将被优先加工。

3) 用于工件分派或组批工件分派的遗传算法

遗传算法在并行批加工设备调度中用于将工件分派到各个批加工设备(两层结构中)或将组批工件分派到各个批加工设备(三层结构)。具体使用的遗传算法计算流程如下。

(1) 编码(coding): 在编码中以工件或组批工件为对象。如果存在 k 个工件或组批工件, 在分派中将使用下述表达式 $c = (m_1 m_2 \cdots m_{k-1} m_k)$ 。其中 m_j 表示用于加工工件或组批工件的设备。例如, $c = (1 \ 10 \ 4 \ 4 \ 4 \ 4)$ 意味着工件或组批工件 1 将在设备 1 上加工, 工件或组批工件 2 将在设备 10 上加工, 其他的工件或组批工件将在设备 4 上加工。这样的表达式称作单染色体。遗传算法就是由这样的一组染色体构成的。

(2) 初始化(initialization): 将上述染色体初始化, 即赋初值, 将 k 个工件或组批工件随机分派到一组设备 $\{1, 2, \dots, m\}$ 上。

(3) 父代染色体选择: 在父代染色体选择中, 采取轮盘赌方法进行。染色体被选作父代染色体的概率与其适应度成正比。遗传算法的适应度函数是工件总加权延迟。

(4) 交叉: 使用单点交叉法。按照染色体的适应度, 随机选择两个父代染色体, 并随机选择交叉点。如果将两个父代染色体分别表示为 $c_1 = (m_{11} \ m_{12} \ \cdots \ m_{1k-1} \ m_{1k})$ 和 $c_2 = (m_{21} \ m_{22} \ \cdots \ m_{2k-1} \ m_{2k})$, 交叉点选择在位置 s , 则交叉后获得两个子代染色体 $c_3 = (m_{11} \ \cdots \ m_{2s} \ \cdots \ m_{2k})$ 和 $c_4 = (m_{21} \ \cdots \ m_{1s} \ \cdots \ m_{1k})$ 。

(5) 变异: 按照事先定义的概率随机将后代的染色体中的一个基因改变为另一台加工设备。

(6) 选择新一代染色体: 将经过选择、交叉、变异操作的新的染色体种群与原来的染色体种群组合在一起, 选出新的适应度较好的染色体种群, 继续进行迭代, 生成新的后代。

(7) 停止迭代条件: 停止迭代既可以通过定义迭代次数来实现, 也可以通过定义种群中各染色体的适应度函数值之间的差值来实现。如果染色体的适应度函数值的差值小于给定的阈值, 则可以停止迭代过程。

5.2.3 带有批加工设备的半导体生产线动态调度

带有批加工设备的半导体生产线动态调度主要是基于启发式规则的。在这里主要介绍两种带有批加工设备的半导体生产线启发式调度规则: 最小加工批量规则与批加工设备调度规则。

1. 最小加工批量规则

最小加工批量规则常常用于评价其他批加工设备调度启发式规则的性能。当工件的到达时间呈泊松分布时,最小加工批量规则通常比只考虑系统当前状态的规则性能要好。该规则具体描述如下。

设 C 是批加工设备的最大加工批量, L 为排队等待加工的工件数目。最小加工批量规则指的是只有当排队队列中至少有 S 个工件时 ($S \leq C$), 才可以开始加工一批工件。如果批加工设备是空闲的, 而排队工件数目小于 S 时, 设备将继续保持空闲状态。其中, $S=1$ 和 $S=C$ 是最小加工批量规则的两种特例。 $S=1$ 意味着每次到来一个新工件就要装载到设备上开始加工, $S=C$ 意味着只有排队工件数满足最大加工批量才可以开始向设备装载工件开始加工。

最小加工批量规则的优点是计算量小, 并且决策中需要的实时信息少, 非常容易实现, 因此在实际的半导体生产线上容易被管理者接受。但是, 最小加工批量规则有许多不足。

首先, 如何决定合适的最小加工批量, 特别是在多品种、生产率变化多端的生产环境下, 没有确定的标准。其次, 最小加工批量规则没有考虑生产线的运作状态以及其决策对整个系统的影响, 因此, 如果最小加工批量的选择小于批加工设备的能力或需要额外的空闲时间以等待组批, 该规则可能会导致设备的能力浪费。最后, 最小加工批量规则是典型的局部决策方法, 没有考虑整个生产线的性能。下面介绍既考虑了生产线当前状态又考虑了即将到来的工件及设备负载的批加工设备调度规则。

2. 批加工设备调度规则

批加工设备调度规则 (Scheduling Rule for Batch Processing Machines, SRB) 既考虑了批加工设备本身的负载情况, 又考虑了其下游设备的负载情况, 还考虑了即将到来的工件 (即上游设备的工件加工情况)。批加工设备调度规则由两部分组成: 一是工件组批规则, 二是组批排队规则。

首先定义如下变量。

i : 批加工设备。

j : 批加工设备 i 的缓冲区中排队的工件, $j=1, 2, \dots, J$ 。

l : 生产线上的工件, 即 WIP。

k : 批加工设备 i 的上游设备, $k=1, 2, \dots, K$ 。

h : 批加工设备 i 的下游设备, $h=1, 2, \dots, H$ 。

m : 批加工设备 i 可完成的工艺菜单, $m=1, 2, \dots, M$, 只要工件待加工工序使用的工艺菜单相同, 就可组批一起加工。

B_i : 批加工设备 i 的加工能力, 即最大加工批量。

t_{im} : 批加工设备 i 的工艺菜单 m 的加工时间。

l_j^m : 缓冲区中使用批加工设备 i 的工艺菜单 m 完成待加工工序的工件。

N_j^{im} : 缓冲区中使用批加工设备 i 的工艺菜单 m 完成待加工工序的工件的数目。

ω : 批加工设备 i 的下游设备可完成的工艺菜单, $\omega=1, 2, \dots, W$ 。

t_{hw} : 批加工设备 i 的下游设备 h 的工艺菜单 w 的加工时间。

t_k^{im} : 正在批加工设备 i 的上游设备 k 加工或刚刚完成加工, 下一工序使用工艺菜单 m 的紧急工件到达批加工设备 i 缓冲区所需时间。

t_j^{iw} : 工件 j 在批加工设备 i 的缓冲区等待加工时间。

1) 工件组批规则

批加工设备一旦由忙状态转变为空闲状态, 或者由维护或维修状态转变为可用状态, 就要选择合适的工件开始加工。在批加工设备调度中, 一个重要的约束是组批工件不能超过设备的最大加工批量, 即加工能力。因此, 当缓冲区中使用相同工艺菜单的工件数超过最大加工批量时, 要确定工件如何组批, 以提高半导体生产线性能。另外, 当缓冲区中使用相同工艺菜单的工件数小于最大加工批量时, 还要考虑正在上游设备加工的下一步加工将要使用此批加工设备的工件或已完成加工还未运送至相应缓冲区的工件, 以达到批加工设备时间利用率与产能利用率的折中。

具体的工件组批过程如下。

第一步: 确定批加工设备对应的缓冲区中是否有紧急工件, 如果有紧急工件, 转第二步; 否则, 转第三步。

如果 t 时刻各工件的理论剩余加工时间与实际剩余加工时间的差值大于等于 0, 如式(5-7)所示:

$$[(F_j \times RPT_j) - (D_j - t)] \geq 0 \quad \text{则} \quad x_j = 1 \quad (5-7)$$

说明该工件极有可能拖期, 因此要作为紧急工件, 在任何设备上具有最高的加工优先级, 以满足准时交货的要求。

式(5-7)中:

t : 调度决策点时刻, 即批加工设备 i 开始进入空闲状态或可用状态的时刻。

D_j : 工件 j 的交货期。

RPT_j : 工件 j 的剩余净加工时间。

F_j : 工件 j 的生产周期倍增因子, 即工件 j 的平均加工周期与工件净加工时间的比值。

x_j : 急件标志, 如果工件 j 是紧急工件, 则 $x_j=1$; 否则, $x_j=0$ 。

第二步: 判断紧急工件是否使用相同的工艺菜单。若存在多个紧急工件使用相同的工艺菜单, 判断其数目, 如果大于等于最大加工批量, 直接选出最紧急的满足最大加工批量的紧急工件并批; 如果小于最大加工批量, 检查是否还有使用相同的工艺菜单的非紧急工件, 若满足条件的非紧急工件数目较多, 按照工件等待时间越长越优先的原则选出若干个非紧急工件以满足最大加工批量。然后进入组批加工优先级判断;

如果

$$\sum_{j=1}^J x_j |_m \geq B_i$$

则