

用于半导体制造中 Q-time 管理的深度强化学习

Harel Yedidsion
Prafulla Dawadi
David Norman
Emrah Zarifoglu

应用材料公司
AI/ML 团队

摘要

队列等待时间约束 (QTC) 是指一个批次在流程中两个流程步骤之间允许等待的时间限值。在半导体制造中, 超过该时间限值的批次会导致良率损失, 需要返工, 或予以报废。QTC 很难进行排程, 因为批次需要一直等到最后步骤的加工有可用产能时, 才能放行进入第一个流程步骤。然而, 精确计算是否有足够产能的计算成本很高。在本研究中, 我们提出一种深度强化学习 (RL) 方法来管理放行批次进入队列等待约束。我们分析了该强化学习方法的性能, 并将其与七种基线解决方案进行比较。我们的实证评估显示, 该强化学习方法在队列等待超时次数和完工时间等五项性能指标上均优于基线解决方案, 而需要的在线计算时间可忽略不计。

1 简介

半导体制造晶圆厂中的批次排程就相当于可重入作业车间调度问题。在这种情况下尽可能缩短完工时间, 就是我们所知的 NP 困难排程问题 (Pinedo 2012)。

任何排程决策都需要考虑诸多限制条件, 比如批次的关键比率、交付日期、优先级, 以及工作台的加工时间、批处理要求、资质、停机时间、瓶颈利用率等等。在实践中, 实时派工 (RTD) 规则采用启发式逻辑来综合所有限制条件和考虑因素, 对每个队列中的批次进行优先级排序。然而, 完全依靠派工可能导致很多批次出现队列等待超时 (Cho 2014)。

队列等待时间约束 (QTC) 是指两个加工步骤之间 (例如湿法工作台与炉管设备之间) 的最大时间限值。在湿法工作台完成加工的批次, 必须在一定时间间隔之内开始在炉管设备上加工, 以免发生氧化或受到污染等化学反应, 从而导致晶圆良率降至需要返工甚至批次完全报废的水平。

为防止 QTC 超时, 采用队列等待时间管理系统 (QMS), 通过一个闸前 (pre-gate) 步骤——湿法工作台步骤的上游虚拟步骤, 来管理放行批次进入队列等待时间约束。QMS 系统会考虑晶圆厂中的所有 QTC, 并生成一份放行计划, 决定在每个时间步每种零部件类型有多少批次将放行进入队列等待时间循环。区分 QMS 与派工的基本原理是, 当一个机台出现空闲时, 派工逻辑必须非常快速地运行, 而 QMS 解决方案可以从容进行计算。这种区别让 QMS 系统可以在每个固定时间间隔异步运行, 并生成一份放行计划。

图 1 显示了我们将本文中采用的路线。它包括一个虚拟的闸前步骤和两个工艺步骤, 它们之间有一个 QTC。批次一直在闸前步骤等待, 直至被放行进入队列等待时间约束, 然后开始湿法工作

台和炉管步骤的加工。完成湿法工作台步骤后，必须在队列等待时间限值之内开始炉管步骤。

有几种不同的方法来控制批次放行，每一种在超出时限、完工时间和计算复杂性方面各有优劣。诸如固定队列的看板法等保守的启发式方法 (Scholl and Domschke 2000) 能够用简单的在线计算尽可能减少超时次数，但确定最佳队列长度以尽可能缩短完工时间颇具挑战 (Kopp et al. 2020)。更复杂的方法采用了如约束优化 (Cho 2014) 或混合整数规划 (Klemmt and Mönch 2012) 等技术，来准确计算实现最短完工时间而不会导致任何超时的排程，但由于问题的组合复杂性，对于大型实例来说，找到最优解并验证其没有超出队列等待时限是难以计算的。

在本文中，我们提出了一种基于深度强化学习 (RL) 的方法来解决队列等待时限问题，它能够在运行时快速生成近似最优解，并能够用于任何湿法工作台和炉管设备派工规则。我们将该方法与七种其他启发式方法进行了比较，结果显示它在完工时间和队列等待超时次数方面优于所有其他方法。



图 1：队列等待时间管理系统的队列等待时限 (QTC)。

1.1 强化学习

强化学习 (RL) 是一套机器学习技术，已成功应用到多个领域，比如电脑游戏 (Wurman et al. 2022)、复杂游戏 (Silver et al. 2017)、机器人科学 (Hanna and Stone 2017, Park et al. 2020) 以及控制 (Cui et al. 2021)。

强化学习问题由一个智能体和一个环境构成。智能体进行的动作会影响环境并改变其状态。每一个动作后，智能体会转变到下一个状态，并收到一个奖励。随着时间推移，智能体就会学到一种最大化累计奖励总额的行为策略。环境通常以马尔科夫决策过程 (MDP) 的形式来说明 (Puterman 2014)。

我们期望在本文中解决的队列等待时间管理问题可以建模为一个离散时间、 T 长度有限的 MDP，即一个元组 $M = (S, A, P, R, \rho^0, \gamma, T)$ ，其中 S 是状态集合， A 是动作集合， $P: S \times A \times S \rightarrow R^+$ 是转移概率分布， $R: S \times A \rightarrow R$ 是奖励函数， $\rho^0: S \rightarrow [0, 1]$ 是初始状态分布， γ 是贴现系数， T 是时间范围。

一个解决策略是将状态映射到动作的概率分布 $\pi: S \times A \rightarrow [0, 1]$ 。为了找到解决策略，我们训练了一个强化学习智能体，学习将预期回报最大化的策略 $E_\tau \sum_{t=0}^T \gamma^t R(s^t, a^t)$ ，其中， $\tau := (s^0, a^0, s^1, a^1, \dots)$ 表示轨迹， $s^0 \sim \rho^0$ ， $a^t \sim \pi(s^t)$ ， $s^{t+1} \sim P(s^t, a^t)$ 。

2 相关成果

曾采用过多种计算方法来控制队列等待时限。在 (Scholl and Domschke 2000) 论文中，采用了将队列保持在固定规模的看板法。(Klemmt and Mönch 2012) 作者提出了一种混合整数规划方法，将问题分解为具有最大作业数的可管理子问题。(Choung et al.) 提出约束优化的解决方法，而 (Mason et al. 2007) 开发出了遗传算法。(Kim et al. 2020) 提出了一种监督学习方法，训练一个深度神经网络来预测批次派工，从而最大化机台利用率。训练该模型所需的标记数据由领域专家通过模拟获得。

近年来，强化学习已日益应用于生产规划和控制 (Panzer et al. 2021)，特别是在半导体晶圆厂排程的以下研究应用中：

(Lin et al. 2019) 使用深度 Q 网络算法 (DQN) 为优化完工时间选择派工规则。(Park et al. 2019) 使用 DQN 为晶圆厂的机台设置变更进行排程。

(Park et al. 2021, Waschneck, et al. 2018, Zhang et al. 2021, Zhou et al. 2020, and Shi et al. 2020) 均使用深度强化学习进行派工，以最小化完工时间、生产周期和生产周期偏差等指标。但是，这些论文中都没有考虑队列等待时限。

(Altenmüller et al. 2020) 使用 DQN 进行了带队列等待时限的任务派工，但他们没有将该方法与看板法 (Scholl and Domaschke 2000) 等既有的队列等待管理系统进行比较。

据我们所知，本研究是首个开发深度强化学习智能体专用于控制队列等待时间管理系统的成果。

3 解决方法

在这部分中，我们阐述了基本解决框架和强化学习方法的组成部分，状态空间、动作空间、奖励函数以及我们使用的强化学习算法。

3.1 强化学习训练和策略更新

在训练期间，智能体进行一个动作，模拟器就应用该动作并模拟一个未来的时间步。然后，智能体收到一个新的状态观察和一个奖励。在我们的案例中，状态转变是确定性的。状态-动作-奖励顺序被保存，并且强化学习算法定期使用该经验来更新代表该策略的神经网络的权重。采用该策略来选择下一个动作。策略更新旨在最大化该时间范围内累积的奖励。

3.2 测试

待学习曲线稳定，策略不再改善后，我们保存该策略并用其在已知和未知混合环境中测试该强化学习智能体的性能。参见附录 A 查看该强化学习算法的学习曲线示例。

3.3 状态观察

在每一个时间步，智能体都收到一个状态观察。观察状态由以下部分组成：

- 晶圆厂特性：步骤加工时间，队列等待时限。
- 晶圆厂观察：每个步骤和每个机台加工的批次数。
- 队列等待时间观察：成功的批次数，超时的批次数，在制的批次数。
- 产能观察：完成所有在制品 (WIP) 的时间估计

状态特征被标准化为介于 $[0,1]$ 的值，并连成一个观察向量。

3.4 动作空间

在每一个时间步，智能体可决定放行或不放行一个批次。该智能体可以放行 N 类零部件中的一类的一个批次。因此，该智能体能够选择 0 到 N 之间的一个离散动作。选择动作 0 不放行任何批次，选择动作 a_i 则放行类型 $Part_i$ 的一个批次。

3.5 奖励函数

我们设计了一个确定性的奖励结构，鼓励智能体尽可能减少队列等待超时次数，同时优化完工时间和成功批次数。

3.6 强化学习算法 - PPO

近端策略优化 (PPO) 算法 (Schulman et al. 2017) 是一种广受欢迎的深度强化学习算法, 它采用策略梯度法以 on-policy 的方式来训练一个随机策略。而且, 它还采用了行动者评论家 (actor critic) 方法。在本论文中, 我们采用了 StableBaselines3 的 PPO 执行 (<https://stable-baselines3.readthedocs.io/en/master/modules/ppo.html>), 并将其默认参数调整为适合队列等待时间管理系统应用 (请参阅附录中的表 2 了解 PPO 参数设置的更多详情)。

4 评价

4.1 基线智能体

我们将强化学习智能体的性能与七种基线智能体的性能进行了比较。

1. **看板智能体**: 由 (Scholl and Domaschke 2000) 在论文中提出, 此智能体的宗旨是在炉管区的前面将队列保持在固定规模。如果队列中的批次数小于设定的队列规模, 则该智能体放行一个批次。设定的队列规模按照能够加工且不会引起任何队列等待时限超时的最大队列规模计算。
2. **基于产能的智能体**: 该智能体在每一个时间步计算炉管区的现有可用产能。产能估算基于炉管区中的在制品 (WIP), 包括当前工序。如果产能小于预定的阈值, 该智能体就会放行一个批次。该智能体与看板智能体有些类似, 但它允许更精细的控制, 因为它并不限于固定规模的队列。
3. **频率智能体**: 考虑到每个时间步放行一个批次的限制, 该智能体将计算炉管区的统一加工时间, 并以最接近该加工时间的频率放行批次。
4. **随机智能体**: 在每个时间步, 随机智能体用相等概率在放行一个批次和不放行一个批次之间随机选择一个动作。
5. **始终放行智能体**: 始终放行智能体会在每个时间步放行一个批次, 无论当前状态如何。该智能体提供了最短完工时间的范围 (尽管以多次出现超时为代价)
6. **从不放行智能体**: 与始终放行智能体相反, 该智能体从不放行一个批次。该智能体用于评估由初始条件导致的超时次数, 并提供了最小超时次数的可能范围。
7. **基于 Q 学习的智能体** 该智能体使用 Q 学习算法来学习将当前状态映射到动作的策略。Q 学习 (Sutton and Barto 2018) 是一种典型的强化学习算法, 通过使用

$$Q(s^t, a^t) = Q(s^t, a^t) + \alpha [R^t + \gamma \max_a Q(s^{t+1}, a) - Q(s^t, a^t)],$$

迭代更新 Q 值, 来学习优化的动作-价值函数, 其中 $Q(s^t, a^t)$ 是动作 a^t 在 s^t 状态的价值, γ 是折扣因子, α 是学习率。我们执行了这个强化学习算法, 以根据另一种强化学习算法对 PPO 的性能进行基准测试。

4.2 实验设置

在本节中, 我们将阐述实验框架的实施详情。

4.2.1 模拟器和环境设置

我们设计了一个定制模拟器, 来模拟一个有队列等待时限的晶圆厂。该模拟器允许我们灵活定义晶圆厂环境, 设定任何数量的机台、机台区以及多种零部件类型的批次。我们为每一种零部件类型定义了相应的工艺路线, 即一组加工步骤。每一个步骤都分配给一个机台区, 并有其自己的加工

时间。路线中的任何一对步骤之间都可能存在队列等待时限。模拟器支持放行多种零部件类型的批次。

用于评价的实验设置与图 1 中的类似，其中该系统有一个闸前步骤、一个湿法工作台和两个炉管设备。闸前步骤的加工时间为 0 分钟，并控制进入湿法工作台队列的批次放行。湿法工作台有 20 分钟的连续加工时间，两台炉管设备有不同的加工时间（600 分钟和 700 分钟）。每个机台的产能为一次加工一个批次。批次在闸前步骤开始，必须经过湿法工作台和炉管加工，以完成其工艺路线。湿法工作台步骤和炉管设备步骤之间有一个队列等待时限，设定为 200 分钟。这意味着任何在湿法工作台完成加工的批次必须在 200 分钟之内开始在炉管设备上加工方视为成功，否则即为队列等待时限超时。我们遵循了 (Altenmüller 2020) 中的常规，即如果超过时限，该超时批次仍继续进行到路线结束。

请注意，在实际的半导体工厂中，湿法工作台和炉管一次可能会加工多个批次。通过在闸前步骤放行多个而非单个批次，我们的方法也能够扩展适用于这种情况。

4.2.2 派工规则

对于炉管区，我们采用了基于队列等待时间的派工规则，按照批次的剩余队列等待时间来排序，因此最邻近队列等待时限超时的批次将首先排程。对于湿台区，我们采用了先进先出 (FIFO) 的派工规则。

4.2.3 预备

在每次实验开始时，机台区前面的队列为空。为避免这些不现实的统一初始条件，并让系统进入稳定生产状态，我们设置了一个预备阶段。每一个实验都有不同的预备阶段，其中我们会按给定的时间步数随机将批次放行进入系统，以创建不同的初始条件。我们为培训和测试采用了不同的预备期（分别为 3 个和 6 个时间步）以预防强化学习算法中的过拟合。因为预备阶段不考虑队列等待时限，所以放行的批次在加工时无法保证没有超出该时限。在结果部分，我们没有将预备期放行批次导致的超时视为超时。

4.2.4 回合终止条件

每一个实验（或回合）有最多 100 个时间步，每一个时间步有 100 分钟。智能体在每一个时间步能够做一个动作。之后，该强化学习智能体收到一个观察结果，反映该系统在时间步结束时的状态。当 100 个时间步结束，或至少 10 个批次完成路线（以先发生者为准），回合终止。

4.2.5 强化学习集成、训练和测试

为了训练和评价强化学习智能体，我们将模拟器集成在 OpenAI Gym 中 (Brockman et. al 2016)。OpenAI Gym 是一个开源的 Python 库，通过提供一个标准 API 进行学习算法与环境之间的通信，用于开发和对比 RL 算法。强化学习智能体接受了 5000 个回合的训练。

对于测试，我们让每个智能体控制闸前步骤 30 个回合。每个回合都有在预备阶段随机创建的初始在制品。我们为每个智能体设定了相同的随机种子，这样它们将面临相同的 30 个随机生成的环境。我们对每个智能体的评价指标进行平均，并在表 1 中进行总结。

4.3 结果

在这部分中，我们对实证评价的结果进行了分析。我们对智能体进行了五个指标的比较，其中包括：

- 平均每个回合超时数，
- 平均每个回合成功数，
- 完成加工 10 个批次的平均完工时间分钟数，及回合长度的最大时间（包括预备阶段），
- 炉管区的平均利用率，即计算炉管设备加工时间在回合总时间中所占比例，
- 以及完成批次的平均周期时间分钟数。

从不智能体超时次数最小（平均 0.97），从预备阶段创建。其他智能体（产能、看板、Q 学习和 PPO）也能够达到最小的超时次数。在表 1 中，我们列出了从每项输入中减去 0.97 后的调整后超时次数。

在那些得到最小超时次数的智能体中，PPO 的成功次数最高，平均 11.83 次成功。

始终智能体的完工时间最短，为 31.33 时间步，PPO 与之相近，平均为 32.1 时间步。请注意，始终智能体的最短完工时间以多次超时为代价，因为始终智能体的超时次数最高，平均 29.9 次。

与始终智能体的 0.94 相比，PPO 的炉管设备利用率达到了 0.93 接近最优，且周期时间也接近最优，仅比始终智能体的周期时间多 1%。

用于训练强化学习智能体的平均累积奖励本身不是一个指标，但它有助于智能体的性能排名。

在计算时间方面，给出状态观察后，PPO 法能快速做出下一个动作的决策，而且与约束优化等精确求解方法不同，我们认为时间不会随问题规模（零部件类型数、批次数、机台数）而呈指数级增长。但是，我们确实认为训练时间会随状态和动作空间增加而增加。

表 1: 实证评价结果。每一项输入在 30 轮评价中平均。绿色背景的单元格表明始终和从不智能体在特定指标方面实现的最佳可能值。黄色背景的单元格着重标示在特定指标方面实现最佳值的智能体。

方法	指标					
	超过最低限值的超时次数	成功次数	完工时间 (时间步)	奖励	利用率	生产周期 (分钟)
始终	29.9	2.4	31.33	-30.83	0.94	2182
随机	14.1	3.13	31.47	-14.96	0.93	2235
从不	0	1.97	93	-1.59	0.1	--
频率	2.9	7.77	33.3	-3.32	0.87	2324
看板	0	9.03	55.4	-0.51	0.56	3194
产能	0	10.03	46.43	-0.32	0.63	2855
Q 学习	0	11.03	33.4	-0.09	0.88	2303
PPO	0	11.83	32.1	0	0.93	2207

5 结论

在本研究中，我们开发了一个深度强化学习智能体，它能够高效控制队列等待时间管理系统实现优化性能，最小化超时次数，同时保持接近最优的完工时间，在部署中无须运行耗时的计算。我们的方法优于七种其他经测试的基准方法，其中包括看板法。

未来研究的方向包括更准确地研究表现真实世界晶圆厂条件的更复杂环境。例如，考虑更多零件、更长路线、分批、专用机台以及多个队列等待时限等。

附录 A - PPO 参数和学习曲线

表 2 详细列出了本研究中使用的 PPO 参数的值。在超参数调整阶段之后选择这些值，其中默认 PPO 参数值在两个方向上逐步修改，直至找到更好的值，或如果没有找到更好的值，则保留默认值。

表 2: PPO 参数和值列表。

参数名称	参数值
batch_size	50
gae_lambda	0.99
learning_rate	0.00025
n_steps	50
pi net_arch	[8,10]
vf net_arch	[8,10]
activation_fn	ReLU

图 2 给出了 PPO 算法的学习曲线。在 50 万时间步后，累计回合奖励稳定在 0。

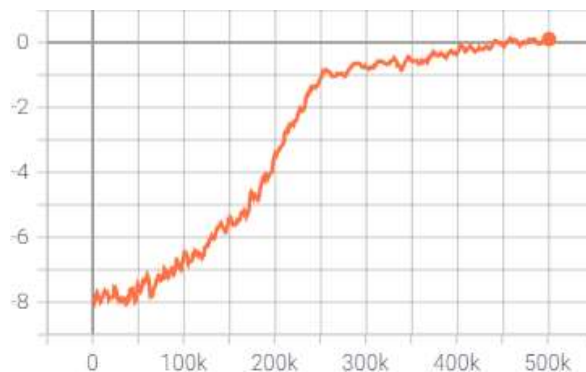


图 2: 从 Tensorboard 提取的 PPO 学习曲线。

参考文献

- Altenmüller, T., Stüker, T., Waschneck, B., Kuhnle, A. and Lanza, G., 2020 *Reinforcement learning for an intelligent and autonomous production control of complex job-shops under time constraints*. *Production Engineering*, 14(3), pp.319-328.
- Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. (2016). *Openai gym*. ArXiv Preprint ArXiv:1606.01540.
- Cho, L., Park, H.M., Ryan, J.K., Sharkey, T.C., Jung, C. and Pabst, D., 2014 *Production scheduling with queue-time constraints: Alternative formulations*. In IIE Annual Conference. Proceedings (p. 282). Institute of Industrial and Systems Engineers (IISE).
- Choung, Y.I., Jun, K.S., Han, D.S., Jang, Y.C., Lee, T.E. and Leachman, R.C., 2002 *Design of a scheduling system for diffusion processes*. In SMOMS (pp. 1-6).
- Cui, J., Macke, W., Yedidsion, H., Goyal, A., Urieli, D. and Stone, P., 2021 *Scalable Multiagent Driving Policies for Reducing Traffic Congestion*. In Proceedings of the 20th International Conference on Autonomous Agents and Multi Agent Systems (pp. 386-394).
- Hanna, J.P. and Stone, P., 2017, February. *Grounded action transformation for robot learning in simulation*. In Thirty-first AAAI conference on artificial intelligence.
- Kim, H., Lim, D.E. and Lee, S., 2020 *Deep learning-based dynamic scheduling for semiconductor manufacturing with high uncertainty of automated material handling system capability*. *IEEE Transactions on Semiconductor Manufacturing*, 33(1), pp.13-22.
- Klemmt, A. and Mönch, L., 2012, December. *Scheduling jobs with time constraints between consecutive process steps in semiconductor manufacturing*. In Proceedings of the 2012 winter simulation conference (WSC) (pp. 1-10). IEEE.
- Kopp, D., Hassoun, M., Kalir, A. and Mönch, L., 2020, December. *Integrating critical queue time constraints into SMT2020 simulation models*. In 2020 Winter Simulation Conference (WSC) (pp. 1813-1824). IEEE.
- Lin, C.C., Deng, D.J., Chih, Y.L. and Chiu, H.T., 2019 *Smart manufacturing scheduling with edge computing using multiclass deep Q network*. *IEEE Transactions on Industrial Informatics*, 15(7), pp.4276-4284.
- Mason, S.J., Kurz, M.E., Pohl, L.M., Fowler, J.W. and Pfund, M.E., 2007 *Random keys implementation of NSGA-II for semiconductor manufacturing scheduling*. *International Journal of Information Technology and Intelligent Computing*, 2(3).
- Panzer, M., Bender, B. and Gronau, N., 2021 *Deep Reinforcement Learning In Production Planning And Control: A Systematic Literature Review*. ISSN: 2701-6277.
- Park, I.B., Huh, J., Kim, J. and Park, J., 2019 A reinforcement learning approach to robust scheduling of semiconductor manufacturing facilities. *IEEE Transactions on Automation Science and Engineering*, 17(3), pp.1420-1431.
- Park, J., Chun, J., Kim, S.H., Kim, Y. and Park, J., 2021 *Learning to schedule job-shop problems: representation and*

- policy learning using graph neural network and reinforcement learning*. International Journal of Production Research, 59(11), pp.3360-3377.
- Park, J.S., Tsang, B., Yedidsion, H., Warnell, G., Kyoung, D., Stone, P. and Sony, A.I., 2020 *Learning to improve multi-robot hallway navigation*. In Proceedings of the Conference on Robot Learning, (pp. 1883-1895). PMLR.
- Pinedo, M.L., 2012 *Scheduling* (Vol. 29). New York: Springer.
- Scholl, W. and Domaschke, J., 2000 *Implementation of modeling and simulation in semiconductor wafer fabrication with time constraints between wet etch and furnace operations*. IEEE Transactions on Semiconductor Manufacturing, 13(3), pp.273- 277
- Puterman, M.L., 2014 Markov decision processes: discrete stochastic dynamic programming. John Wiley & Sons.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A. and Klimov, O., 2017 Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.
- Shi, D., Fan, W., Xiao, Y., Lin, T. and Xing, C., 2020 *Intelligent scheduling of discrete automated production line via deep reinforcement learning*. International journal of production research, 58(11), pp.3362-3380.
- Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A. and Chen, Y., 2017 *Mastering the game of go without human knowledge*. nature, 550(7676), pp.354-359.
- Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.
- Wurman, P.R., Barrett, S., Kawamoto, K., MacGlashan, J., Subramanian, K., Walsh, T.J., Capobianco, R., Devlic, A., Eckert, F., Fuchs, F. and Gilpin, L., 2022 *Outracing champion Gran Turismo drivers with deep reinforcement learning*. Nature, 602(7896), pp.223-228.
- Waschneck, B., Reichstaller, A., Belzner, L., Altenmüller, T., Bauernhansl, T., Knapp, A. and Kyek, A., 2018, April. *Deep reinforcement learning for semiconductor production scheduling*. In 2018 29th annual SEMI advanced semiconductor manufacturing conference (ASMC) (pp. 301-306). IEEE.
- Zhang, C., Song, W., Cao, Z., Zhang, J., Tan, P.S. and Chi, X., 2020 *Learning to dispatch for job shop scheduling via deep reinforcement learning*. Advances in Neural Information Processing Systems, 33, pp.1621-1632.
- Zhou, L., Zhang, L., & Horn, B. K. (2020). *Deep reinforcement learning-based dynamic scheduling in smart manufacturing*. Procedia Cirp, 93, 383-388.

作者简介

HAREL YEDIDSION 是应用材料公司 AI/ML 团队的研究科学家。他在以色列本古里安大学工业工程与管理系获得博士学位，并曾在得克萨斯大学奥斯汀分校计算机科学系进行博士后研究。他的研究方向包括多智能体系统、机器人科学和强化学习。他的电子邮件地址为 harel_yedidsion@amat.com。

PRAFULLA DAWADI 是应用材料公司 AI/ML 团队的数据科学家。他在华盛顿州立大学普尔曼分校电气工程与计算机科学学院获得计算机科学博士学位。他的研究方向包括数据科学、机器学习系统和强化学习。他的电子邮件地址为 prafulla_dawadi@amat.com。

DAVID NORMAN 是应用材料公司一名杰出的科研人员。他在明尼苏达大学获得数学博士学位。他的方向包括应用数学优化、机器学习和强化学习来排程和规划制造和供应链管理中的问题。他的电子邮件地址为 david_norman@amat.com。

EMRAH ZARIFOGLU 是应用材料公司自动化产品部的研发负责人。他的研究和方向着重于云计算、人工智能、机器学习、供应链、半导体制造、运筹学、模拟等领域。他拥有得克萨斯大学奥斯汀分校运筹学和工业工程博士学位。他的电子邮件地址为 emrah.zarifoglu@amat.com。