A Micro-Discrete Event Simulation Environment for Production Scheduling in Manufacturing Digital Twins

Emmanuel Nsiye
Institute for Simulation and Training
University of Central Florida
Orlando, USA
emmanuel.nsiye@ucf.edu

Brian Tse
College of Engineering and Computer Science
University of Central Florida
Orlando, USA
brian.tse@ucf.edu

Tori Wright
College of Engineering and Computer Science
University of Central Florida
Orlando, USA
tori.wright@ucf.edu

Sean Mondesire
School of Modeling, Simulation, and Training
University of Central Florida
Orlando, USA
sean.mondesire@ucf.edu

ABSTRACT

With advancements in automation technologies in manufacturing industries, there is an increasing demand for real-time analytics to aid in performance prediction, production scheduling, and capacity planning. By combining machine learning and automation, Digital Twin models are used for real-time monitoring and optimization of a system. In manufacturing, Digital Twins are the digital representations of the manufacturing plant, connected with real-time data from its physical counterpart. The insights generated by the Digital Twin aid in long-term production analyses and optimized systems planning, such as capacity and maintenance. Capacity, referring to the amount of product that can be produced within a given timeframe, is a challenging metric to capture in complex manufacturing systems accurately. The Digital Twin can address this issue by using machine learning to predict capacity, allowing manufacturers to assess efficiency and prevent machine failure.

This paper introduces a micro-discrete event simulation model that facilitates batch tracking and optimization of job scheduling within a manufacturing facility for capacity planning. The simulation is built upon the OpenAI and Farama Gymnasium framework, combining agent-based modeling and reinforcement learning to effectively schedule product batches by considering equipment availability, downtime, timing constraints, and production yield. With its lightweight and scalable architecture, the simulation is compatible with the real-time connectivity requirement of the Digital Twin model and its ability to evaluate different manufacturing scenarios. By integrating machine learning with the simulation, the simulation parameters can be optimized to meet specific manufacturing capacity planning goals.

ABOUT THE AUTHORS

Emmanuel Nsiye is a Research Assistant at the Institute of Simulation and Training at the University of Central Florida since January 2023. He works at the Human-centered AI Laboratory (HAIL) and graduated from the Data Analytics Master program at the same University in December 2022.

Brian Tse is a Graduate Research Assistant at the Institute for Simulation and Training at the University of Central Florida since March 2023. He works at the Human-centered AI Laboratory (HAIL) and is in the Master of Science in Data Analytics program at the same university.

Tori Wright is a master's student studying Data Analytics at the University of Central Florida with an interest in machine learning and statistics. She works as a graduate research assistant at the Institute for Simulation and Training in the Human-centered AI Laboratory (HAIL), exploring digital twin technology.

Sean Mondesire, Ph.D. is an Assistant Professor at the University of Central Florida's (UCF) School of Modeling, Simulation, and Training (SMST). He is a part of the Knights Digital Twin Initiative, directs the Human-centered Artificial Intelligence Laboratory (HAIL), and co-directs UCF’s Advanced Research Computing Center (ARCC) for high-performance computing. His research specialties are real-time machine learning and big data analytics.

A Micro-Discrete Event Simulation Environment for Production Scheduling in Manufacturing Digital Twins

Emmanuel Nsiye
Institute for Simulation and Training University of Central Florida
Orlando, USA
emmanuel.nsiye@ucf.edu

Brian Tse
College of Engineering and Computer Science
University of Central Florida
Orlando, USA
brian.tse@ucf.edu

Tori Wright
College of Engineering and Computer Science
University of Central Florida
Orlando, USA
tori.wright@ucf.edu

Sean Mondesire
School of Modeling, Simulation, and Training
University of Central Florida
Orlando, USA
sean.mondesire@ucf.edu

INTRODUCTION

In recent years, the manufacturing industry has shifted towards the adoption of digital twin technologies, driven by its ability to simulate operations and systems to facilitate the development of dynamic solutions. These capabilities are attributed to the extensive connectivity features of digital twins, allowing seamless integration with specialized software. This integration ensures efficient functionality of manufacturing operations by tracking and documenting production processes and materials, incorporating data from physical manufacturing operations. Furthermore, the digital twins' ability to simulate real-world systems offers valuable insights to manufacturers by pinpointing bottlenecks and optimizing productivity [Hill, 2022]. The combination of full connectivity and simulation facilitates capacity planning, predictive maintenance, system optimization, and remote monitoring, contributing to improved and proactive management within manufacturing facilities.

Recent innovations in manufacturing attributed to digital twin technologies are exemplified by the BMW iFactory. The BMW automotive factory, developed with Nvidia Omniverse, a platform that allows for the development of 3D tools and applications, provides real-time simulations of the entire automotive plant [Diaz and Jesus, 2023]. This advancement allows plant managers to assess manufacturing processes, closely monitor key performance indicators (KPIs), identify potential issues, and seamlessly integrate artificial intelligence for real-world optimization. With the advent of a digital twin, BMW's process engineers can simulate different scenarios to test for bugs in the plant design. Another use of digital twin technology for manufacturing is in the power industry, where implementing digital twins helps maintain power generation equipment, monitor the overall system, and avoid system failure or external damage [Huang and Jianping, 2021].

The deployment of a digital twin's extensive capabilities can uncover solutions to solving one of the manufacturing industry's most impactful challenges: production scheduling. *Production scheduling* is planning when various tasks get executed in the manufacturing process. When scheduling production, task allocation and resource allocation must be considered. Task allocation determines how duties are assigned to different equipment or operators to produce a product. Resource allocation determines what finite materials are used at each step of manufacturing, which further restricts the flow. Together, they help determine the flow of the manufacturing process, the most efficient operational sequences and equipment usages, and when and by whom limited resources are used.

Production scheduling can be categorized as a stochastic scheduling problem that involves unpredictable characteristics~~,~~ such as varying processing times, diverse due dates, and the potential for conflicting demands. This scheduling can occur on either a short-term or long-term basis, presenting challenges like random weights, dynamic environments, and the occurrence of machine breakdowns that may impede or disrupt processing. One approach for production scheduling is leveraging discrete event simulations, a technique designed to replicate the actions and efficiencies of actual processes within a simulated environment as chronological events that occur in intervals [Allen et al, 2015]. Several solutions utilizing discrete event simulation to handle production scheduling are available; however, production scheduling solutions require dynamism and adaptability; and rely on data to deal with changing market demands and various products.

This work proposes a new simulation that allows customization and execution of manufacturing operations based on user input guidelines. The simulation factors include operational timings, yield probabilities, equipment capabilities, state transitions, order arrivals, fabrication sequences, deadlines, and priorities, opening the door to developing heuristic or Artificial Intelligence optimization to find optimal operations and procedures. The approach uses micro-discrete event simulations built upon the OpenAI and Farama Gymnasium framework (Brockman et al, 2015). The use of Gymnasium will allow combining agent-based modeling and reinforcement learning to iterate through the fabrication process of all product orders and equipment status changes typical of a generalized product. The simulator's output is the number of products produced (*throughput*), equipment assignments (*job schedule*), timings of each product during its fabrication lifespan, and the amount of time each product spent in idle and in-production states. Based on the simulation's customizable job scheduling scheme and the product scheduling protocol, users are able to determine the fabrication capacity of a production operation. Lastly, the simulation is built to be adaptable to live updates from a manufacturing execution system so digital twins can update themselves with the latest operational data and alerts can be inserted to notify when production thresholds and schedules fall out of expected ranges.

BACKGROUND

Historically, attempts to address production scheduling through non-simulation approaches, such as programming, have led some researchers to define the task as NP-hard complex. In other words, once an efficient schedule is produced, solving for subsequent job schedules is reduced to polynomial time (Awad and Abd-Elaziz, 2021). It is imperative to devise a strategy for optimizing the duration required to generate the initial job schedule, aiming to reduce the time taken for scheduling subsequently. Long-term planning for task allocation is challenging in most manufacturing processes, and a constantly changing environment exacerbates the difficulty of this task. While existing research has attempted to address this problem using multiple optimization algorithms under more straightforward scenarios, these approaches do not consider dynamically changing conditions of equipment failures, production uncertainties, or realistic tasks (Ko, Chang, Lee and Kim, 2023). For example, regarding production scheduling, an issue arises when a product stays in queue for a step longer than scheduled (Chuang et al, 2023). A proposed solution to this problem uses *integer programming*, a mathematical model acting as an objective function that seeks to minimize the length of the queue time violation based on differences in process time (Chuang et al, 2023). However, it is important to note that this approach, while addressing certain aspects of the problem, does not fully account for the complexities introduced by equipment failures and their potential impact on job scheduling.

One thing that is overlooked by the mathematical and programming solutions presented is the dynamism inherent in manufacturing. Resource allocation is profoundly influenced by the constantly changing nature of a live manufacturing process, with different jobs requiring varying amounts of resources at different production times. For instance, a common problem in general fabrication is the adherence to a fabrication sequence for each product to ensure successful production. A product represents an unfinished object that must go through a predefined series of operations performed by specific equipment. Products are grouped into batches, where each batch is assigned a piece of equipment to execute the corresponding process on every product in the batch. Each piece of equipment can perform operations according to the fabrication sequence. The challenge arises when scheduling batches to equipment during facility operation, as multiple batches may require simultaneous access to a piece of equipment, leading to the dilemma of determining which batch should be assigned when conflicts arise. A proposed solution to this challenge involves scheduling based on the efficiency of jobs rather than completion time, ensuring the efficient use of resources (Hu and Li, 2023). Another solution leverages the *densest job set first* (DJSF) method, which schedules jobs based on which job has the highest job completion efficiency or density. Before job scheduling, the jobs are packed into sets using an alignment technique such that jobs with similar j*ob completion times* (JCT) are in the same group through the k-means clustering algorithm. As the number of parallel jobs within the groups increases, jobs with the highest job completion efficiency will be scheduled with higher priority. The simulated DJSF scheduling method effectively reduces JCT by maximizing the number of completed jobs per unit of time. Among solutions for scheduling and resource allocation, some have investigated incorporating machine learning combined with statistical analysis to aid in job scheduling. A noteworthy case is in long-term capacity planning, where machine learning is integrated with optimizing equipment capabilities (Ko et al, 2023). In this work, three algorithms were proposed: one to balance the workload of all workstations used in fabrication processes, another to estimate the machine count needed to accomplish all tasks, and a third one that combines the first and second algorithms to optimize the balance of workload by freeing up unneeded machines.

With manufacturing systems producing large amounts of data from different sources, this data serves as a valuable tool that provides insights to enhance the scheduling process and address the dynamism of manufacturing. The *Manufacturing Execution System* (MES) is software used to monitor, manage, and record manufacturing processes. Its utilization has given rise to solutions that focus on integrating the continuous stream of data to solve scheduling challenges. One way to achieve this integration is through simulation, where simulations can be run repeatedly as new information becomes available by implementing an information exchange system between the simulation and the MES. These simulations can be customized based on parameters defined by the user, such as production costs (Kuck et al, 2016). Customization of simulations can also focus on processing times, with the implementation of generic algorithms such as *shortest processing time* (SPT) (Wang et al, 2022). In this application, the SPT algorithm is applied following the completion of the first processing step, with three rules to follow: if there is a lone idle parallel machine and the only workpiece in the waiting queue is newly arrived, the workpiece promptly undergoes processing by entering the idle machine; if there are multiple idle machines on the parallel machine system and only one workpiece has recently joined the waiting queue, the machine will process the workpiece with the shortest processing time among the available idle machines; in the absence of an available idle parallel machine, the workpiece must wait in the temporary storage area. Once an idle machine becomes available, it will select the workpiece with the shortest processing time from the temporary storage area for further processing.

Furthermore, the MES data stream enables the creation of digital twin-based simulations, ensuring rapid, precise, and comprehensive information presentation, coupled with a high degree of flexibility to accommodate the intricacies of multi-variety production lines, varying batch sizes, and diverse product models (Shan et al, 2023). The rationale for digital twin-based simulation lies in its ability to provide swift responses to workshop reconstruction scheduling plans and promptly restructure the production line model. This ensures the absence of conflicts between the reconfigured virtual workshop and the segments undergoing simulation, all made possible by running parallel workstations and resource flow simulations.

Most of the simulations presented above are continuous simulations as they rely on continuous changes in behavior. However, there is another type of simulation called discrete event simulation (DES), which is built on modeling events sequentially and occurring at a given expected time. A primary benefit of DES lies in its capacity to conduct experiments that may be impractical or impossible in actual manufacturing systems. Developing a simulation model contributes to acquiring insights that have the potential to enhance the natural system through gained knowledge (Sharma, 2015). Researchers have sought to apply DES with a digital twin model. In the resulting system, the simulation action depends on prior knowledge collected from the production. Any correlation between events is analyzed, and the logical impact between events and data is well understood to complete mastery (Li et al, 2022). The proposed system can provide forward-looking assessments and recommendations, continuously monitor real-time workshop production conditions, and analyze and forecast future operating conditions in the workshop caused by the creation of new events resulting from scheduling one event. Similarly, there has been an attempt to model production scheduling as a reinforcement learning environment built on integrating a Python-based Discrete Event Simulation library and OpenAI Gymnasium (Lang et al., 2021).

Previous research has demonstrated multiple approaches to solving production scheduling, ranging from programming solutions built on mathematical models to continuous and discrete event simulation models built upon manufacturing data. This work introduces a new simulation– a micro-simulation environment that is small, lightweight, fast, and less complex while implementing principles inherent in discrete event simulations: system state, event list, simulation clock, and statistics (Law et al., 2015).

METHODOLOGY

The presented simulation environment implements the following components of discrete event simulation to build a digital twin model of all possible manufacturing factories successfully:

- ***System State***: The properties of the system, such as the number of equipment and their states and materials.
- ***Event List***: The production flows, such as new orders coming in, those orders being scheduled for production, and moving through the fabrication sequence.

- ***Statistical Counters***: Track all possible production data, such as previous processing times and operational success probabilities.
- ***Simulation Clock***: Time hops to ensure the simulation will go from one event to another.

The environment leverages reinforcement learning to enable the future integration of machine learning and probabilistic modeling for the fabrication process. This setup allows researchers to walk through an operation and measure capacity and throughput. In the rest of this section, the simulation design is presented along with its corresponding components, the look-up tables. An illustration of the execution of the simulation will be shown, accompanied by a runtime analysis, and how Machine Learning optimizes the simulated manufacturing environment.

Simulation Implementation

The simulation is built on the OpenAI Gym and Farama Gymnasium framework for agent-based modeling (Brockman et al, 2015). In this framework, the simulator mirrors the manufacturing operations environment. The environment is fully observable, such that the agent can directly observe its state. The agent deals with an action function that determines what batch processing will be the next discrete event after receiving the current batches and equipment status. To accomplish this, the policy that will be used is based on timestamp and due date; the batches will be sorted based on time and due date, and in ascending order. In the action function, the agent determines the next operation to perform, identifies the pieces of equipment capable of satisfying the operation, and decides if the job should be scheduled if the equipment is available. When a batch does not have an available piece of equipment because the equipment is at capacity while operating on other batches or the equipment is down for maintenance, the batch will be placed on standby and wait until equipment becomes available.

Within the simulation, the agent executes a step to change the environment, in this case, a simulated manufacturing environment. Specifically, each simulation step is the processing of a discrete event, where the environment (manufacturing operations) will receive an action and apply it, operate the processing of a batch, scheduling it to a selected equipment, and simulating its processing by giving the expected output of the operation that is determined by stochastic weighted selection of past probability of success. In addition, the step function updates the system's state by moving to the next operation and to the next timestep and by updating the equipment used to the following probable status for that equipment. The equipment status is determined by implementing a stochastic matrix that this work is calling transition matrix, a square matrix that depicts a Markov chain's transition (Norris, 1998). After the batch is processed and the equipment status is changed, the simulation records the statistics produced by this event through a log file. It returns the new system state by emitting new observations for the actions (events) that are possible for the agent and the equipment status.

Similarly, new batches of products can be introduced to simulate new, incoming orders. Additionally, batches may need to be split if only some of the products in the batch exited an operation successfully. The unsuccessful products will be placed into a new smaller batch, and the operation will be performed again. In contrast, the successful products will remain in their original batch and continue to their next operation. Lastly, the Gymnasium framework was selected because of the capability to integrate reinforcement learning to automate job scheduling decision-making easily. This form of machine learning will be explored in future work to optimize fabrication production sequences.

Simulation Attributes

To make the simulation realistic, all operational outcomes of the manufacturing process are based on historical results gathered from the physical plant, equipment, and products. Gathered data, such as the equipment status change history, equipment yield percentage for operation, and operation timings, are stored in the simulation's Equipment History, Capability, and Operation Tables. Following the Gymnasium's framework, the equipment and Batch tables represent the environment states. In addition, three lookup tables are created to keep track of the system, the events that will be processed, and the statistics of each event: the equipment, batch, and statistics look-up tables. Using these lookup tables, data lookups have reduced runtimes and made the simulator more time efficient, allowing the execution of more simulations in a shorter time. Below are descriptions of the five look-up tables:

- ***Equipment Table***: represents the state of the manufacturing operations or system state by containing live status information on each piece of equipment in the manufacturing plant. For each piece of equipment, information on whether the equipment is available or not, which batch is contained in the equipment, how many products of that batch are being produced, how long the equipment has been in that state, the expected cycle time before completion of that step, and the following status of the batch after its operation is complete. The following batch status is based on the probability of the operation on that device being successful.

- ***Capability Table***: Contains information on which equipment can perform a batch need in its fabrication sequence. This table also defines operational timings and the probability of successful operations, all based on a history of gathered results.
- ***Operation Table***: Lists the fabrication sequence of each product. This listing contains the order of operations that must be executed and the capabilities required for a batch's production.
- ***Batch Table:*** The Batch Table represents the event list; it contains batch information of all the products currently manufactured in the manufacturing plant and the ones already being simulated. This collection of batch information contains the identifier of the batch, the fabrication sequence (operation table) the batch follows, the step of the sequence the batch is on, the current quantity of the batch, the status of the batch (if it is in Production or Stand By), the timestamp (the simulation clock), and the localization of the batch (The equipment it is assigned to).
- ***Equipment History Table***: This look-up table contains equipment history and all the different transitions of equipment status. The table will be updated as the simulation progresses to make the simulation more realistic with new equipment status and information being added.
- ***Statistics Table:*** A collection of statistics about the processing of each event (Scheduling of a batch). It will contain the Batch ID, the operation, the number of products, the time it took, and the yield of the operation.

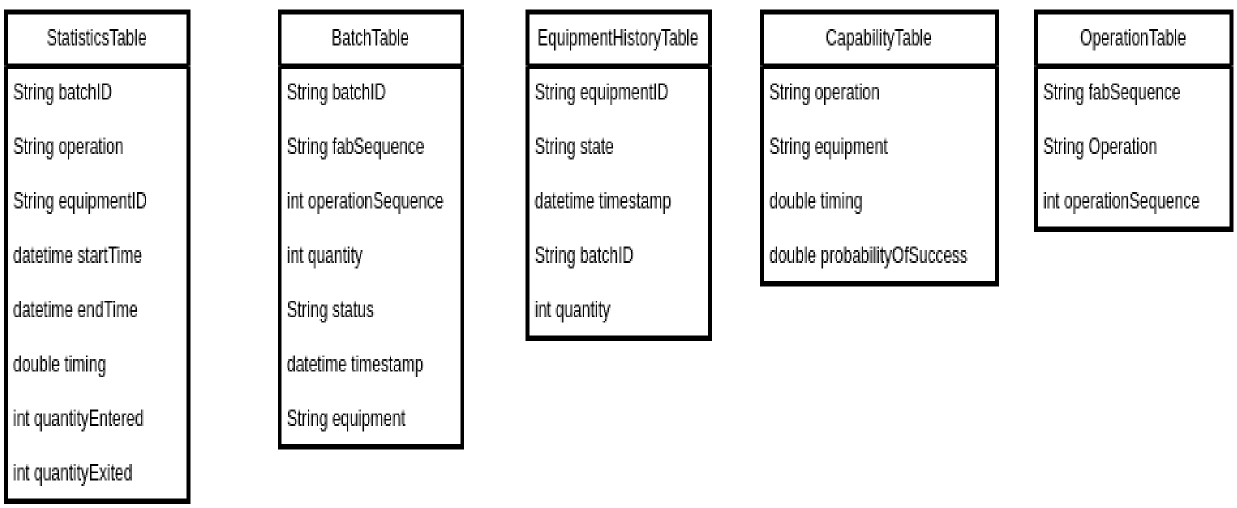

Figure 1. The simulation uses five data look-up tables to track manufacturing states and speed up simulation execution through indexing (hashing) and multi-indexing functionality.

Those look-up tables are initialized at the initialization point of the simulation. Moreover, during the step function and to accomplish the change of the system state, the simulation makes use of helper functions to accomplish changes in the environment:

- **Equipment State Change**: Implements the transition matrix to find the equipment's next state.
- **Logging**: Function that tracks simulation event statistics for the equipment and the batches.
- **Equipment State Change**: Function that is used to approximate the likeliest timing or yield based on the probability distribution for that specific operation.
- **Rework**: Handle the creation of batch-rework.

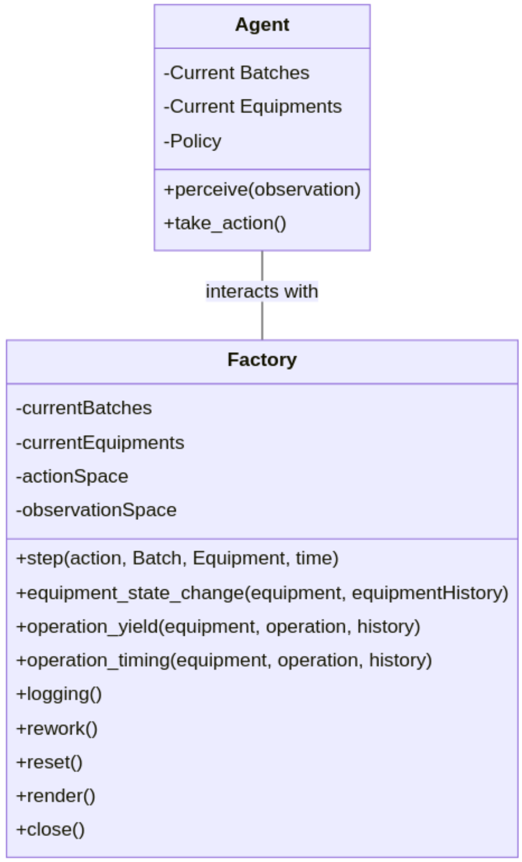

Figure 2. Architecture of the agent and the Factory. It shows the information each of them has available and what functions they have to help in the decision making.

Simulation Procedure

To model the execution of orders, the discrete event simulation iterates through each batch listed to be manufactured in the batch table in ascending order by the batch's current time and due date, updating batch and equipment statuses as manufacturing events are completed. Each batch's current time represents a simulation time. Therefore, during each iteration of the simulation, the agent attempts to schedule a batch and changes the environment until all have completed their entire fabrication sequence. During each timestep, any batch can be assigned to equipment capable of advancing the batch to its subsequent operation. If an available piece of equipment can operate on a batch, then that batch is scheduled on the equipment and set as in a production state; no other batch can be scheduled on that equipment while the operation is performed because the equipment's status is also updated to be in production. If no equipment is available for a batch, the batch is set to a standby state and waits until equipment capable of the batch's operation becomes available. In this case, the batch waits by being queued on equipment until the equipment is done processing that product. Its current time is updated to be the most probable time a capable piece of equipment for the batch becomes available. In cases when multiple lots require the same piece of equipment at the same time, the *first come, first served (FIFO)* scheme is followed to assign the first batch that requests the equipment. FIFO is followed by sorting the batch lookup table in ascending order by its current time and then descending order of priority. With this scheme, batches with the earliest arrival time that request equipment first are guaranteed to be assigned to equipment when it becomes available. In instances of conflict, batches with the highest priority will always be scheduled first. The success of the batch processing is determined by stochastic probability, and if the operation is 100% successful, the batch is moved to the next operation in its fabrication sequence, and the equipment's following status is determined based on stochastic probability. The environment will return comprehensive status updates, presenting valuable information regarding both batches and equipment. This dual-status reporting mechanism is designed to offer a holistic view of the environment's dynamics, facilitating a deeper understanding of the interactions between batches of tasks and the equipment responsible for their execution. The status of the batches will help identify the next event to be processed, and the status of the equipment will help decide if the processing of the next event is possible given the equipment.

The procedure of the job scheduling simulation is described below:

- 1: Initialization
- 2: Action
  - Sort the batches in the Batch lookup table by time and due date in ascending order of current time and descending priority, if defined.
  - Select the next batch in the Batches lookup table that needs to be in a complete state.
  - Identify the next operation by mapping the batch's Operations ID and Operation Sequence to those indexes in the operation table.
  - Identify the equipment pieces capable of performing the batch's next operation. This identification is first done using the operation's capability ID that links to the Capability table's capability ID. With these Capability table entries, the equipment IDs of each capable equipment are from the Equipment table's equipment ID. Lastly, from the matching, Equipment entries are returned to provide the equipment capable of performing the operation.
- 3: Step
  - Case 1: Schedule the batch on the given equipment.
    - Assign the Batch to the defined equipment, update the batch and equipment state.
    - Using the capability table, the likeliest time and yield is estimated based on random probability distributions. Then, the equipment and batch time are updated, and the yield of the operation is given too. With the Batch time being updated, if the Batch is not done on its fabrication sequence this will be a new discrete event entry.
    - The log statistics of the equipment are added to the equipment history.
    - Using the transition matrix and the equipment history, determine the likeliest next state for that piece of equipment, then move the equipment to that state.
    - Log the new equipment state.
    - Based on the yield, determine if some products in the batch might require a rework or not. If that is the case, separate those products into a new batch that will be inserted into the Batch table.
    - Move the batch to the next operation, check if the operation was the last one for the Batch, if it was the Batch can be removed.
    - Check if all the Batches in the Batch table have been processed, if it is then a signal will be changed to show that state.
    - Return the current Batch table, the equipment table, and the state of the Simulation (If it is done or not done).
  - Case 2: Queue the Batch on the given equipment.
    - Assign the batch to equipment.
    - Update the time of the Batch to be the time the equipment will be next available on.
  - Repeat processes 2 and 3 until the Batch table is empty. Then end the simulation.

Asymptotic Analysis

The simulator is designed to be execution time efficient by calculating capacity and throughput assessments as quickly as possible. To create these efficiencies, the lookup tables are initialized once the simulation is initialized, and each lookup table uses indexing and multi-indexing to speed up queries and searches. To evaluate the efficiencies of the approach, asymptotic analysis is used to estimate simulation worst-case runtimes during a simulation step. For the Batch Table, the table will have to be sorted by timestamp, when sorting on a single column pandas uses mergesort, which takes *O(nlogn)* with n corresponding to the number of rows in the dataframe, as such the cost of sorting the table will be *O(nlogn)* After sorting the Batch table, the simulator will iterate through each batch and take *O(n)* for each lookup table's indexed reference. For instance, capability lookups exhibit constant time complexity, denoted as *O(1)*, due to the efficient indexing of both the fabrication sequence and operation sequence. Similarly, searching for capable equipment also operates in constant time (*O(1)*) because the index associated with capability ID directly maps to all equipment pieces capable of performing the operation. When searching for the minimal time until the next capable equipment becomes available, the runtime is *O(n)*, where n is the number of capable equipment. All lookup table updates are also *O(1)* because the indexing on identifier columns allows accessing the element directly based on its key. From the reduction of the total of the sorting runtime, lookups, and table updates, the runtime analysis is *O(nlogn)* per simulation step and a total of $O(n^2 \log n)$ for the entire simulation.

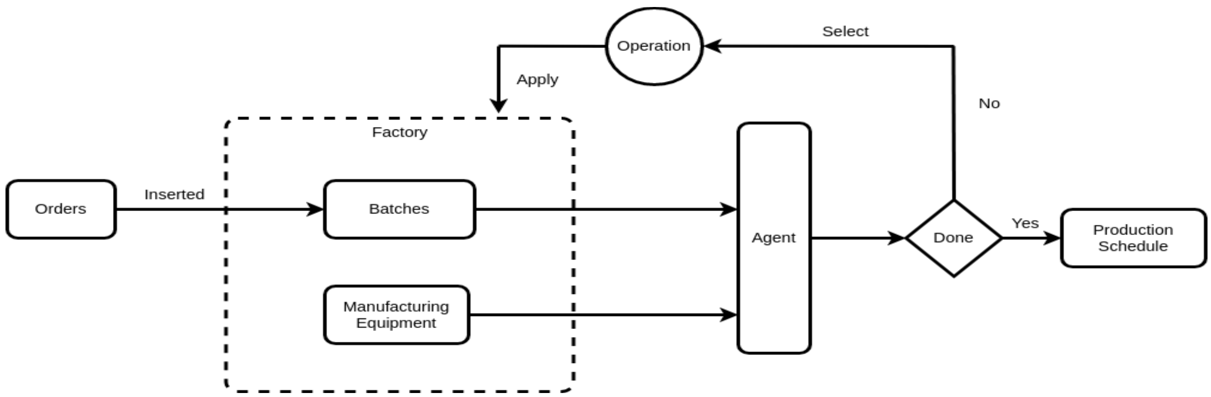

Figure 3. Execution Flow: Orders are inserted into the factory and broken into Batches. The factory contains manufacturing equipment and transfer the list of batches and to the Agent. Based on the information received, the Agent check if there Batches, if not then it Select the operation to be performed and it is applied to the Factory. The processus will be repeated until all the batches have been processed, then the Production Schedule is released.

Machine Learning

With the simulation, users can assess batch operations and calculate the time required to manufacture each batch. However, for the simulation to yield optimal results in terms of fabrication time, scheduling requires optimization, which is made possible by machine learning. In the extension of this work, machine learning will be employed to determine optimal simulation parameters, batch priorities, and batch schedules.

The simulation adheres to the Farama Gymnasium standard, defining elements to create custom reinforcement learning environments. With complete integration, the simulator can use existing reinforcement learning algorithms, such as Stable Baselines's Proximal Policy Optimization (PPO) and Deep Q-learning Network (DQN) (Schulman, 2017) (Alavizadeh, 2021). Future extensions of the work will use these algorithms to develop neural-based models that will use the state of the batches and equipment (the manufacturing plant and process) as the model's inputs and the batch-to-equipment assignment as the outputted action. Machine learning optimization, schedules, and equipment usage will be compared to the FIFO approach described in this work.

Machine Learning will allow production planners to anticipate potential bottlenecks caused by equipment failures and proactively allocate resources based on constraints created by a process requiring more time or resources. Overall, machine learning will help address the main challenges of production scheduling, namely resource allocation and task allocation, resulting in a streamlined production process.

CONCLUSION

Manufacturing environments are dynamic systems where products arrive with different due dates, weights, and constraints, and equipment can experience a downtime unexpectedly. When scheduling production, neglecting any of these parameters can make the process challenging. The paper addresses the issue of production scheduling within generalized manufacturing Digital Twins by providing a customizable and extensible micro-discrete event simulation that simulates manufacturing operations.

The presented simulation environment is built upon OpenAI and Farama Gymnasium framework and simulates task allocation and resource allocation in a manufacturing operation. The simulation leverages historical data to recreate the dynamism of a manufacturing environment: data-driven equipment state changes, data-driven operational timing and yield probabilities, and defined scheduling paths. The simulation's core strength lies in its ability to perform computational operations assessment swiftly. This capability facilitates real-time applications, enabling the simulation to impact production processes immediately. The key to this real-time functionality is the utilization of indexed look-up tables. These tables contain both pre-compiled information about the manufacturing plant and processes and real-time data on ongoing operations within the facility. The simulator is well-suited to support digital twins of manufacturing operations., achieving this by seamlessly integrating real-time connectivity, monitoring, and generating new simulations based on changes to the physical environment. This enables it to assist in real-time predictive maintenance, capacity planning, and equipment scheduling, enhancing the capabilities of digital twins in the manufacturing domain.

While the current version of the simulator represents a significant step forward, it is merely the initial phase of a more ambitious project that extends similar micro-simulation work that is developing digital twins for other domains, including cybersecurity (Schiller et al., 2023) and student learning (Mondesire and Wiegand, 2023), work that aims to build probabilistic models for preventive maintenance (Wright et al., 2024), and work to generate data for manufacturing using GANs (Tse et al, 2024). The next iteration will introduce machine learning capabilities, paving the way for automated equipment planning and scheduling. This advancement aims to optimize production processes further, ensuring efficiency and adaptability in an ever-evolving manufacturing landscape.

REFERENCES

Alavizadeh, H., & Alavizadeh, H. (2021). Deep Q-Learning based Reinforcement Learning Approach for Network Intrusion Detection. ArXiv. /abs/2111.13

Allen M, Spencer A, Gibson A, et al. Right cot, right place, right time: improving the design and organisation of care networks – a computer simulation study. Southampton (UK): NIHR Journals Library; 2015 May. (Health Services a and Delivery Research, No. 3.20.) Chapter 5

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J. & Zaremba, W. (2016). OpenAI Gym (cite arxiv:1606.01540

D. Wang, X. Liu, L. Hu and F. Hu, "Optimization Simulation of Hybrid Assembly Line Production Scheduling in Garment Enterprises," 2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 2022, pp. 1375-1379, doi: 10.1109/ITOEC53115.2022.9734580

Diaz, Jesus, "BMW's new factory doesn't exist in real life, but it will still change the car industry," Fast Company, https://www.fastcompany.com/90867625/bmws-new-factory-doesnt-exist-in-real-life-but-it-will-still-change-the-car-industry (accessed on July 21, 2023).

Hill, T. (2022, January 24). Simulation is a window into the future of Your Manufacturing Operation NIST.https://www.nist.gov/blogs/manufacturing-innovation-blog/simulation-window-future-your-manufacturing-operation#:~:text=Simulation%20can%20be%20used%20to,production%20mix%2C%20schedule%20and%20sequence

Huang, Jianping, "The Application of Digital Twin on Power Industry," IOPscience, https://iopscience.iop.org/article/10.1088/1755-1315/647/1/012015/pdf (accessed on July 21, 2023).

J. Li, N. Zhang, A. Wang, S. Wang and Q. Shan, "Discrete event-driven twin workshop modeling and simulation technology," 2022 IEEE International Conference on Mechatronics and Automation (ICMA), Guilin, Guangxi, China, 2022, pp. 47-52, doi: 10.1109/ICMA54519.2022.9855939.

Kück, Mirko & Ehm, Jens & Freitag, Michael & Frazzon, Enzo & Pimentel, Ricardo. (2016). A Data-Driven Simulation-Based Optimisation Approach for Adaptive Scheduling and Control of Dynamic Manufacturing Systems. Advanced Materials Research. 1140. 449-456. 10.4028/www.scientific.net/AMR.1140.449.

Keyhoon Ko, Seokcheol Chang, Won-Jun Lee, and Byung-Hee Kim. 2023. Machine Learning Powered Capacity Planning for Semiconductor Fab. In Proceedings of the Winter Simulation Conference (WSC '22). IEEE Press, 3263–3274.

Keyhoon Ko, Won-Jun Lee, Seokcheol Chang, and Byung-Hee Kim. Machine Learning Powered Capacity Planning for Semiconductor Fab. 2022 Winter Simulation Conference.

Lang, Sebastian & Kuetgens, Maximilian & Reichardt, Paul & Reggelin, Tobias. (2021). Modeling Production Scheduling Problems as Reinforcement Learning Environments based on Discrete-Event Simulation and OpenAI Gym. IFAC-PapersOnLine. 54. 793-798. 10.1016/j.ifacol.2021.08.093.

Law, A. M., & Kelton, W. D. (2000). Simulation and modeling analysis. McGraw-Hill.

Meurig Beynon and Yun Pui Yung. Agent-oriented Modelling and Simulation for Discrete Event Systems (1992). IEE Colloquium on Discrete Event Dynamic Systems – A New Generation of Modeling, Simulation, and Control Applications.

Mondesire, S. & Wiegand, R. P. (2023). Mitigating Catastrophic Forgetting with Complementary Layered Learning. Electronics - Special Issue: Modeling and Simulation Methods: Recent Advances and Applications, 1(1), 2023.

M. Ahmed Awad and H. Mohamed Abd-Elaziz, "Flexible Job-Shop Scheduling in Smart Manufacturing," 2021 IEEE 7th International Conference on Control Science and Systems Engineering (ICCSSE), Qingdao, China, 2021, pp. 268-272, doi: 10.1109/ICCSSE52761.2021.9545193

Q. Shan, W. Cui, A. Wang and S. Wang, "Digital Twin Simulation System for Reconfigurable Manufacturing Systems," 2023 2nd International Conference on Artificial Intelligence and Computer Information Technology (AICIT), Yichang, China, 2023, pp. 1-4, doi: 10.1109/AICIT59054.2023.10278028.

Schiller, T., Caulkins, B., Wu, A. S., & Mondesire, S. (2023). Security Awareness in Smart Homes and IoT Networks through Swarm-based Cybersecurity Penetration Testing. Information, 14, September 2023.

Schiller, T. & Mondesire, S. (2023). Human-out-of-the-Loop Swarm-based IoT Network Penetration Testing by IoT Devices. In the Annual Modeling and Simulation Conference 2023 (ANNSIM 2023) (p. 2). Hamilton, Ontario, Canada: Society for Modeling and Simulation International (SCS), May 2023.

Schiller, T. & Mondesire, S. (2023). Swarm-based IoT Network Penetration Testing by IoT Devices. In the 21st International Conference on Applied Cryptography and Network Security (ACNS 2023) (p. 5). Kyoto, Japan: Springer (LNCS Series), June 2023.

Sharma, P. Discrete-Event Simulation. Int. J. Sci. Technol. Res. 2015, 4, 136–140.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal Policy Optimization Algorithms. ArXiv. /abs/1707.06347

Tse B., T. Azinord, T. Wright, D. Medina, E. Nsiye, S. Mondesire, “Semiconductor Manufacturing Data Synthesis through GANs,” Advanced Semiconductor Manufacturing Conference (ASMC), Albany, New York, 2024.

V. -T. Chuang, Y. -H. Chang, J. -F. Dang, R. -C. You and S. M. Chang, "Optimizing multi-station scheduling in consideration of equipment flexibility and loading balance in semiconductor wafer fabrication," ASMC 2013 SEMI Advanced Semiconductor Manufacturing Conference, Saratoga Springs, NY, USA, 2013, pp. 41- 44, doi: 10.1109/ASMC.2013.6552772

Wright T., B. Tse, E. Nsiye, T. Azinord, D. Medina, S. Mondesire, “Probabilistic Modeling and Machine Learning for Preventative Maintenance Prediction in Semiconductor Manufacturing,” Advanced Semiconductor Manufacturing Conference (ASMC), Albany, New York, 2024.

Zhiyao Hu and Dongsheng Li, “Improved heuristic job scheduling method to enhance throughput for big data analytics,” IEEE Xplore, https://ieeexplore.ieee.org/document/9552662 (accessed July 12, 2023)