Proceedings of the 2024 Winter Simulation Conference
H. Lam, E. Azar, D. Batur, S. Gao, W. Xie, S. R. Hunter, and M. D. Rossetti, eds.

REAL-TIME TRACKING OF PRODUCTION IN ASSEMBLY OPERATIONS USING AGENT-BASED MODELING AND DIGITAL TWIN TECHNIQUES

Michail Katsigiannis[1], and Konstantinos Mykoniatis[1]

[1]Dept. of Industrial and Systems Eng., Auburn University, Auburn, AL, USA

ABSTRACT

The integration of Digital Twins into manufacturing environments unlocks significant value through a range of benefits. In this work, we demonstrate the digitalization of a physical system. The chosen system is an emulated automotive assembly process, for which we create a virtual representation of both the physical system's state and the product itself using Agent-Based Modeling techniques. To twin the physical and virtual worlds, we leverage real-time data collected from automation systems and Internet of Things devices. The motivation for this work lies behind the need to create Digital Twin models that can be used to integrate bidirectional data flows into industrial simulation models.

1 INTRODUCTION

The concept of Digital Twins (DTs) was originally perceived as a tool to manage a product's entire lifecycle (Grieves 2014). Later, the focus shifted to DTs as digital replicas of physical machines, with the potential to revolutionize manufacturing through constant monitoring and improvements (Stark et al. 2019). Today, the concept of DTs has broadened. They are no longer limited to physical machines but can represent anything – objects, processes, or entire systems – whether they exist in the real world, combine physical and digital aspects, or are purely digital. This is achieved by combining real-time data with computational models and simulation modeling techniques. DTs are increasingly important in various fields due to their ability to optimize performance, predict outcomes, and improve decision-making (Rasheed et al. 2020). They are also used for status monitoring, simulation, and visualization, but there is potential for them to offer value-added services and support autonomous intelligent agents (Ali et al. 2021; Huang et al. 2021). Moreover, integrating DTs into manufacturing processes facilitates a deeper understanding of human interactions within the context of smart manufacturing. By accurately modeling and simulating these interactions, DTs enable manufacturers to improve workflow efficiency, enhance worker safety, and mitigate risks associated with human error (Hou et al. 2021; Bevilacqua et al. 2020)

In industrial operations, DTs have significant roles in production and predictive maintenance (Melesse et al. 2021). Their integration with Model-Based Systems Engineering (MBSE) can enhance system simulation and support Internet of Things (IoT) applications (Madni et al. 2019). They enable data-driven decision-making, complex systems monitoring, and product validation (Botín-Sanabria et al. 2022). Next, they have a wide range of industry use cases, including manufacturing, smart cities, and the automobile industry (Augustine 2020). DT technology uses a range of advanced tools, including the Internet of Things, Machine Learning, and Cloud Computing (Prarthana et al. 2021).

Combining DT and Simulation techniques in manufacturing environments can offer considerable benefits. These include the ability to experiment with new ideas and scenarios without disrupting production (Guerreiro et al. 2020), improved time and cost-effectiveness, and enhanced production planning (Guerra-Zubiaga et al. 2021). The integration of product specification and production and the use of DTs in product development can lead to the creation of high-performance products (Wagner et al. 2019). DTs can also provide diagnostic and prognostic support in complex manufacturing environments (Shen et al. 2023). Integrating DT and manufacturing-focused simulation techniques in a Cyber-Physical System can create platforms that monitor and simulate manufacturing processes (Leng et al. 2021). Lastly, a holistic Digital Twin approach can support the optimization and resilience of factories by considering human factors and interdependencies between production assets (Bécue et al. 2020).

DTs have been integrated into simulation models in various ways. Especially in the case of manufacturing, Bécue et al. (2020) proposed a holistic approach to the creation of DTs for factories of the future, considering many aspects of manufacturing environments. Mykoniatis and Harris (2021) developed a DT emulator of a modular production system using a data-driven hybrid modeling and simulation approach that combines Discrete Event Simulation (DES) and Agent-Based Modeling (ABM). This approach allows for high-fidelity representation of the physical system through near real-time data exchange with Programmable Logic Controllers (PLCs) and can be used during the design phase, including testing different system configurations, verifying system behavior, and predicting performance metrics. The hybrid simulation approach offers advantages over traditional methods by overcoming the limitations of standalone simulation techniques and allowing for better design decisions and performance optimization in production systems. Other examples of integrating DTs into Simulation models consider simulation services for Automated Guided Vehicles (AGVs) (Martínez et al. 2021) and engineering education (Wilking et al. 2023). Furthermore, additional advancements cover the validation of technical systems (Dahmen et al. 2022), the dynamic runtime integration of new models in DTs (Ejersbo et al. 2023), and the application of simulation techniques in creating experimentable DTs for Industry 4.0 (Schluse et al. 2018). Lastly, van der Valk et al. (2020) also contributed to the taxonomy of simulation models in DTs. These studies collectively demonstrate the diverse and innovative ways in which DTs have been integrated into simulation models.

Using simulation models as DTs is also growing as a research directive, mainly focused on DES models, with ABM models or hybrid models covering only a small fraction of the research (dos Santos et al. 2022). ABM refers to modeling approaches in which autonomous agents' interactions are repeatedly simulated within a system. ABM is suitable for simulating dynamic processes that rely on agent interaction, aiming to optimize the collective agent behavior and achieve desired end-states (Macal and North 2009).

In this work, we showcase our initial efforts to digitalize the Tiger Motors Assembly Line, an educational and training manufacturing facility. Launched in 2011, Tiger Motors is an educational and training facility located at the Samuel Ginn College of Engineering, Auburn University. The facility provides students and industry experts with hands-on experience in a simulated assembly line that produces cars made from LEGO® bricks. Since its inception, Tiger Motors has allowed hundreds of students and industry professionals to learn how to apply Lean Manufacturing concepts. Two different products are assembled in this facility: a 234-piece SUV car and a 277-piece Speedster car. Figure 1 illustrates the two cars assembled at Tiger Motors.

Figure 1: The 277-piece Speedster (left) and the 234-piece SUV (right).

One of the primary goals of this facility is to educate students and industry professionals about Lean Manufacturing concepts. Our digitalization efforts consider a multi-phase digitalization plan to transform the Tiger Motors Assembly Line into a demonstration facility for Digital Manufacturing techniques, targeted toward Industry 4.0 technology adoption by Small and Medium Manufacturers (SMMs). The initial phase of this plan considered a hybrid simulation study of the facility's operation, combining ABM and DES techniques (Katsigiannis et al. 2024). That study aimed to assess the effect of the Lean Manufacturing techniques applied in the facility. In this work, we describe the next step, which considers creating a digital model that represents the physical assembly process using real-time data sources to demonstrate and assess the use of DT technologies in assembly operations.

Our key contribution includes a systematic description of the twinning process we implemented for real-time production tracking in assembly operations using ABM. The choice of creating a real-time digital representation of the facility using ABM simulation techniques is based on our future plans to integrate Digital Twin capabilities into the previously developed hybrid simulation model. The modeling choices underlined in this work can be valuable for future research in integrating Digital Twin models into hybrid simulation models.

2 REAL-TIME DATA ACQUISITION

In this section, we describe the assembly line we use as the case study for our digitalization efforts. First, we describe the Tiger Motors Assembly Line, which emulates a high-volume automotive manufacturing assembly plant that produces cars made from LEGO® bricks. Next, we describe the automation systems installed in the facility and the types of data we collect through them. Finally, we describe how we implemented the real-time tracking of entities throughout the system using Internet of Things devices and barcode scanners.

2.1 The Tiger Motors Assembly Line

The Tiger Motor's assembly line is divided into three work cells, each with five workstations. The cars are assembled sequentially, and each car needs to go through all fifteen workstations. The facility operates with a target takt time (available production time divided by customer demand) of 70 seconds and a target cycle time of 60 seconds. A thorough description of the facility's capabilities and operation can be found in our previous work (Katsigiannis et al. 2024). In previous work, we assessed the transition from the Mass Production mode of operation to implementing Lean Manufacturing techniques by developing a hybrid simulation model of the facility. In that work, we evaluated an alternative scenario using the Just-In-Time and Heijunka Lean Manufacturing techniques. In that scenario, the system's lead time was reduced by 47.37%, and the work-in-progress of the workstations decreased by up to 56.73%.

The assembly line can be operated under different production styles. The reasons for allowing for different production archetypes lie within the facility's educational and training goals. Practitioners first go through a production run where they follow the Mass Production mode of operation, where each work cell operates independently according to a Master Production Schedule. Next, the practitioners get hands-on experience regarding Lean Manufacturing concepts by running the facility under a Lean Manufacturing Just-In-Time mode of operation. In this mode of operation, a push system and multiple Lean Manufacturing principles are implemented, such as a leveled schedule based on a Heijunka box. The second workstation of the first manufacturing cell is illustrated in Figure 2.

The assembly process begins at the first workstation of the first work cell. Then, the car under assembly moves through the rest of the work cell's five workstations. The process continues with the cars passing through all the workstations of the second and third work cells. An inspection step is added at the final workstation of each cell, where the cars are placed in front of inspection cameras, and the operators are alerted of potential faults. If the car passes inspection, it is placed in a separate space, called the supermarket, where cars can be batched before being transferred to the next work cell. When cars don't pass inspection,

Figure 2: The second workstation of the first work cell.

a supervisor is called, and they decide whether the car needs to be returned to a prior workstation for rework or if the car is beyond repair. Cars that are deemed to be beyond repair are removed from the system. The third and final work cell is equipped with a conveyor belt that does not stop moving unless there is an assembly problem at one of the workstations or an emergency.

The process of how the cars travel through the assembly line is illustrated in Figure 3. In this figure, the 15 workstations of the assembly line are denoted as W1 through W15, and the supermarkets at the end of the first two work cells are denoted as Supermarket 1 and Supermarket 2, respectively. The three work cells are denoted as Cell 1, Cell 2, and Cell 3, respectively.

2.2 Data Sources

The facility is equipped with an automation system controlled by an Omron Programmable Logic Controller (PLC). The automation system includes Andon Lights at each section, a conveyor belt, inspection cameras, and an Omron Cobra 650 SCARA robot. As part of our digitalization efforts, we have equipped each workstation with an Internet of Things (IoT) device that collects data from barcode scanners and allows for the real-time tracking of the cars in the facility.

All data collected from the assembly operations is stored and processed at a central computer in the facility. This computer is the central hub responsible for connecting to all required networks, running required data collection services and databases, processing the incoming data, and running the simulation software. The computer is connected to two different networks; the first network provides the computer access to the automation network, and the second network provides access to the IoT devices.

The data acquisition from the automation system is performed using a MySQL database. The PLC that controls the automation system is connected to the central computer using a wired Ethernet network. A ladder logic program facilitates the connection to the MySQL database, allowing bi-directional communication. From the PLC, we collect data about the state of the Andon lights at each station and the conveyor belt. The Andon lights are three-colored lightning systems that standardize communication between the operators, supervisors, and maintenance staff. Under normal operation, the light shines a green color. Whenever an operator faces a problem during assembly, they can press a button on their station, and the Andon light for

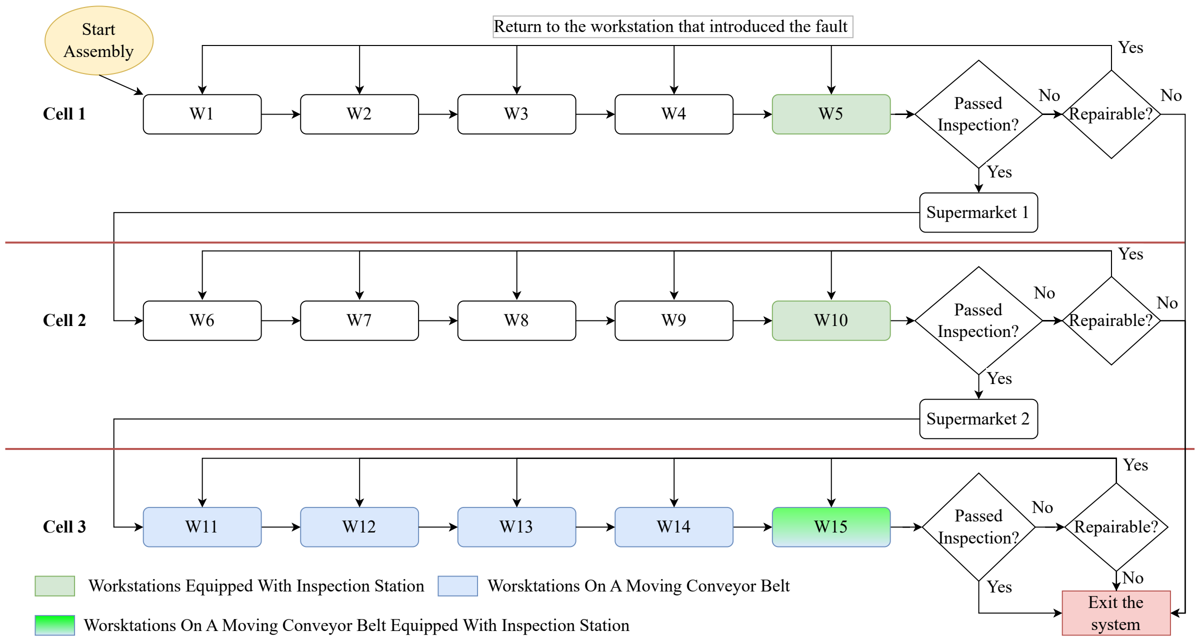

Figure 3: The flow of cars through the assembly line.

this station turns yellow. This indicates to the supervisor that a problem has occurred. If the problem has not been fixed within a certain amount of time, the light turns red. In the case of a red light, the assembly process halts until the issue is resolved. When the issue is resolved, the operator presses the same button they used to bring attention to the problem, and the Andon light turns back to green.

The automation system receives feedback from the model via the same MySQL database, as any changes in the database are accessible to the facility's PLC. By querying the database, the PLC can adjust its programmed variables, changing the timing of the Andon lights and turning the conveyor belt on or off. Furthermore, we are currently integrating the inspection cameras and the SCARA robot installed at the end of the production line into our ABM model. This integration will allow us to experiment with additional ways for the digital model to control the automation system.

2.3 Real-Time Tracking of Products Using Internet of Things

A main requirement for our digitalization plan of the facility was to create the capability to track where each car under assembly is located in the assembly line. We equipped each workstation with a handheld barcode scanner to accommodate this need. When a new car begins the assembly process, the operator at the initiating workstation places a preprinted barcode at the bottom of the car. These barcodes contain a unique code containing information about the specific type of assembled car, akin to the Vehicle Identification Number (VIN) found in automobiles. We used CODE 128 formatting for the barcodes, as this type is compact and can encode a large amount of data. The workstations are also equipped with a card that contains codes that indicate faults that may occur. Whenever an operator recognizes a fault with the car they are currently assembling, they can scan the corresponding code on the card, and the car is tagged with the fault in the developed system. Figure 4 illustrates an example of these barcodes.

The barcodes scanned at each workstation are acquired using IoT devices. These IoT devices combine a microcomputer, barcode scanners, and required communication software. The microcomputer of choice was the Raspberry Pi 4B, running the Raspbian operating system. The reason behind using the Raspberry Pi 4B as the controller of the IoT devices was the low cost, adequate processing power, and networking

SUV CODES		SPEEDSTER CODES		FAULT CODES	
TEXT	BARCODE	TEXT	BARCODE	TEXT	BARCODE
SUV1		SPEED1		FAULT1	
SUV2		SPEED2		FAULT2	
SUV3		SPEED3		FAULT3	

Figure 4: Example of barcodes used to track the cars through the assembly process and the fault codes.

capabilities. More importantly, these microcomputers are supported by online documentation and active communities that provide support when creating new applications. Given the educational and training focus of the facility, we required a solution that allows for the future development of applications by the students and researchers who use the facility during their studies. The IoT devices communicate with the central computer through a separate Wi-Fi network created by a wireless router. Using a separate network provides increased security, as any application that uses the IoT data cannot access the automation network.

The IoT devices are placed on the work cells to facilitate data collection, and the barcode scanners are connected to them via USB. The barcode scanners are installed on the workstations using 3D-printed holders and are within reach of the operators while they are working. Next, we developed software that allows the microcomputer to read the codes scanned by the scanners and recognize which scanner belongs to which workstation. The workstation recognition is enabled using a configuration file created and stored in the device's memory. This configuration file maps each available USB port to a specific workstation. By following this approach, we allow multiple scanners to connect to a single IoT device. Finally, the developed code is set up to run every time the operating system boots, thus allowing for the headless operation of the devices.

The communication between the IoT devices and the facility's central computer is performed using the MQTT communication protocol. The MQTT protocol is based on a publisher and subscriber architecture and uses broker software as the middleware to enable communication. The IoT devices serve as publishers, sending data to the broker. Then, the broker makes the data available to all subscriber software in the network. We installed the Eclipse Mosquitto software, an open-source MQTT broker, on the central computer to serve as the communication middleware. The transmitted data contains the scanned barcode and a timestamp of when the code was scanned. The IoT devices publish this data to separate MQTT topics. We use one topic per workstation, allowing any developed application to collect data directly from specific workstations. Another benefit of this approach is that the topics allow subscriber applications to distinguish between which workstation they received data from.

Subsequently, the operators perform the scanning process manually. The operator scans the associated barcode whenever a new car enters a workstation. If a fault with the car is recognized, the operator pushes a button at the workstation. This button is connected to the PLC and is responsible for changing the status of the Andon lights. In this case, the operator also scans a fault-related barcode with the help of the supervisor. This process is the same for all workstations, with the workstations that are equipped with additional inspection capabilities being the stations to recognize faults more frequently. After the operator finishes their assembly process, they scan the barcode associated with the car once more to signal the end of the process. We illustrate the process flow diagram of this process in Figure 5.

3 AGENT-BASED DIGITAL TWIN MODEL

This section explores how the collected data is transformed into an Agent-Based Digital Twin model (ABDTM), which digitalizes the information and tracks the state of the assembly process. For the

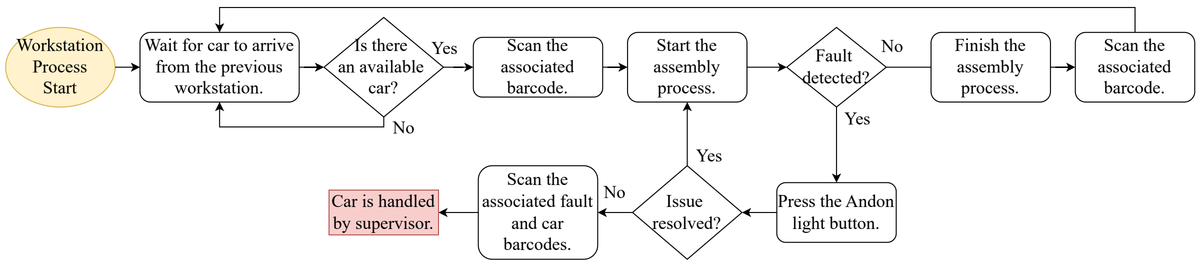

Figure 5: The assembly process at each workstation, including the scanning steps.

development of the ABDTM, we used the AnyLogic 8.8.6 simulation modeling software. The first step of this process is creating a Communication agent that can subscribe to the MQTT topics created by the IoT devices and connect to the automation system MySQL database. Next, we created agents that keep track of each car in the assembly line and agents that keep track of the state of each workstation, namely the Car and Workstation agents.

All agents are created in the Main agent of the AnyLogic model, which is responsible for initiating the processes when the model is run. The Main agent includes the Communication agent, a population of 15 Workstation agents, and an initially empty population of Car agents. The Main agent also contains an event that runs at the model start and is responsible for initiating the Communication and Workstation agents. Finally, we included an additional recurring event with a recurrence time of 1 milliseconds. This event allows the model to update at regular intervals while running, ensuring that the agents' communication is performed promptly. We observed that the synchronization between the model and the data sources was not functioning optimally without including a recurring event.

3.1 The Communication Agent

To receive messages sent by IoT devices, we imported the Eclipse Paho MQTT Java Client library into our model. Using this library, the Communication agent can receive messages from the MQTT broker transmitted by the IoT devices. The Communication agent is responsible for receiving the messages, parsing the contents, and then passing the equivalent messages to the Car agent. If the Communication agent receives a product code for a car not currently in the system, it creates a new Car agent and adds it to the corresponding agent population. If the received message corresponds to a car already being assembled, the Communication agent is responsible for finding this specific car in the population and reading its current state. Depending on the current state of the Car agent, the Communication agent sends a message to the Car and Workstation agents to update their state charts. Finally, if a fault-related code is received, a message is sent to the Car agent to update its state chart, and the fault code is inserted into a corresponding collection element.

The Communication agent is also responsible for querying the MySQL database which contains information about the automation system. Whenever the PLC inserts a new entry in the database, the agent reacts by sending a message to the corresponding Workstation agent to update its state chart. We currently monitor only the state of the Andon lights, but we plan to add additional logic to incorporate the data from the inspection cameras, SCARA robot, and conveyor belt in later phases of our digitalization efforts. The process flow of the logic we implemented for the Communication agent is illustrated in Figure 6.

3.2 The Car Agent

The Car agent is responsible for tracking the location and state of each car assembled in the facility. To achieve real-time tracking of the cars within the facilities, we implemented an Agent-Based state chart that describes the current state of each car. The unique identification number, the car type (Speedster or SUV), and the current workstation for each car are stored in variables within the agents.

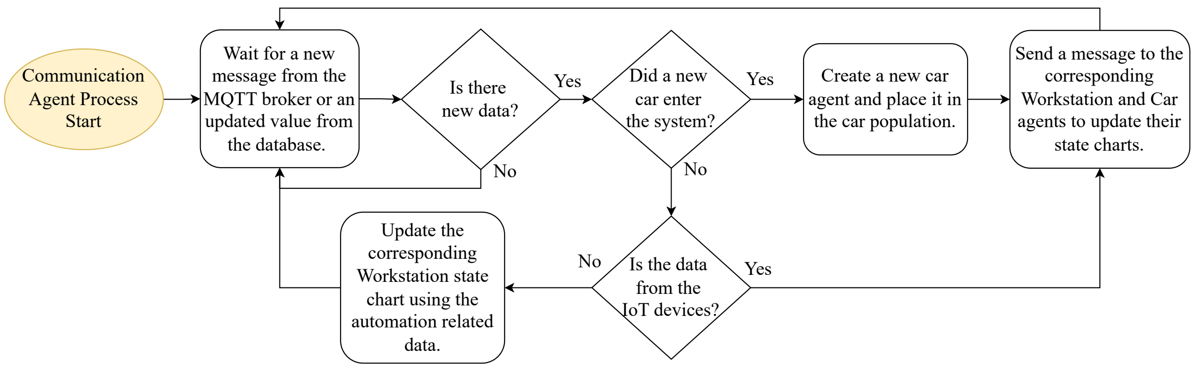

Figure 6: The process flow of the Communication agent's logic.

A new car agent is created every time a new barcode is scanned at the entering workstation of the assembly line. The starting state of the agent is the "assembly_at_station" state, which resides within the "being_made" composite state. This composite state also contains the "wait_in_queue" state, which denotes that the car is queued for assembly as soon as the assigned workstation is free. The transitions between the "assembly_at_station" and "wait_in_queue" states are performed using messages sent by the Communication agent.

A message is sent to the corresponding Car agent whenever a fault is detected during assembly, and the agent enters the "fault_detected" composite state. Within this state, necessary statistics are gathered, and depending on the type of detected error, the car resumes the assembly process, gets transferred to a different workstation, or leaves the system. All detected faults are added using identifiers to a collection within the agent. Finally, whenever a car goes through all fifteen workstations, it transitions to the "finished" state, and we gather important statistics about key performance indicators, such as lead time and types of detected faults. The state chart of the Car agent is illustrated in Figure 7, as it was created using the AnyLogic 8.8.6 software.

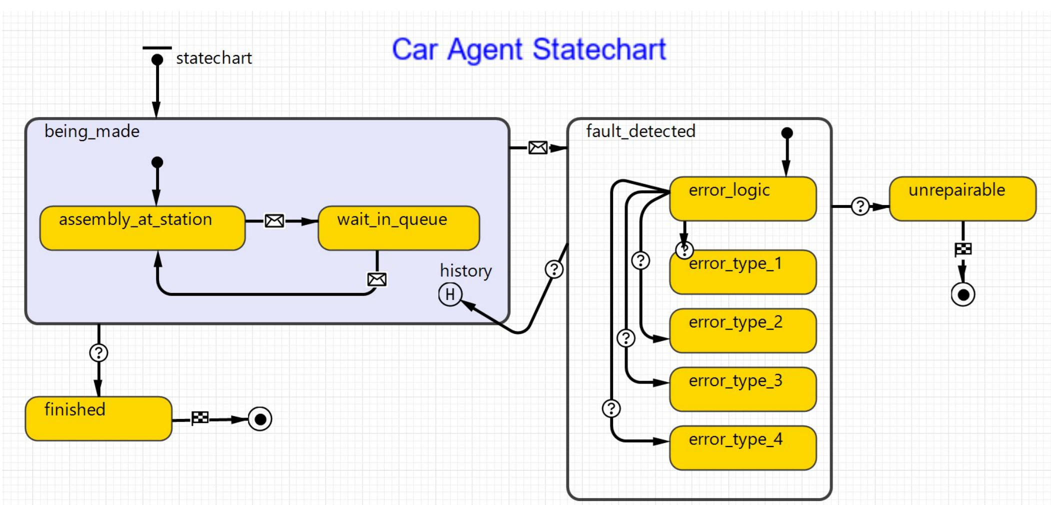

Figure 7: The Agent-Based Modeling state chart of the Car agent.

3.3 The Workstation Agent

The Workstation agent tracks the state of the assembly line's fifteen workstations. When the Model is initiated, the agent starts at the "initialize" state, using data from the MySQL database to determine its starting conditions. After the initialization process, the agent enters one of three states: "green_andon_state," "yellow_andon_state," or "red_andon_state." These states align with the Andon lights positioned at the physical workstations. The "green_andon_state" indicates normal operation, where production can proceed smoothly.

In contrast, the "yellow_andon_state" signifies a cautionary status, prompting attention to a potential issue that requires resolution to prevent escalation to the "red_andon_state." The latter indicates a critical stoppage or malfunction that necessitates immediate action to resolve. The state that corresponds to a green Andon light is a composite state containing the "Idle" and "Busy" states. A Workstation agent is in the "Busy" state whenever a Car agent is assembled at that station and in the "Idle" state whenever no Car agent is assembled.

The agent transitions from the "Idle" to the "Busy" state, and vice versa, by receiving messages from the Car agents. The transitions between the states that correspond to the workstation's Andon lights are performed using messages sent by the Communication agent. Finally, whenever a production run is finished, a message is sent from the Communication agent to each Workstation agent to transition to the final "production_finished" state. In this final state, important statistics regarding key performance indicators are gathered using collected data. These key performance indicators include data about workstation utilization, service times, and downtime due to faults. The state chart of the Workstation agent is illustrated in Figure 8, as it was created using the AnyLogic 8.8.6 software.

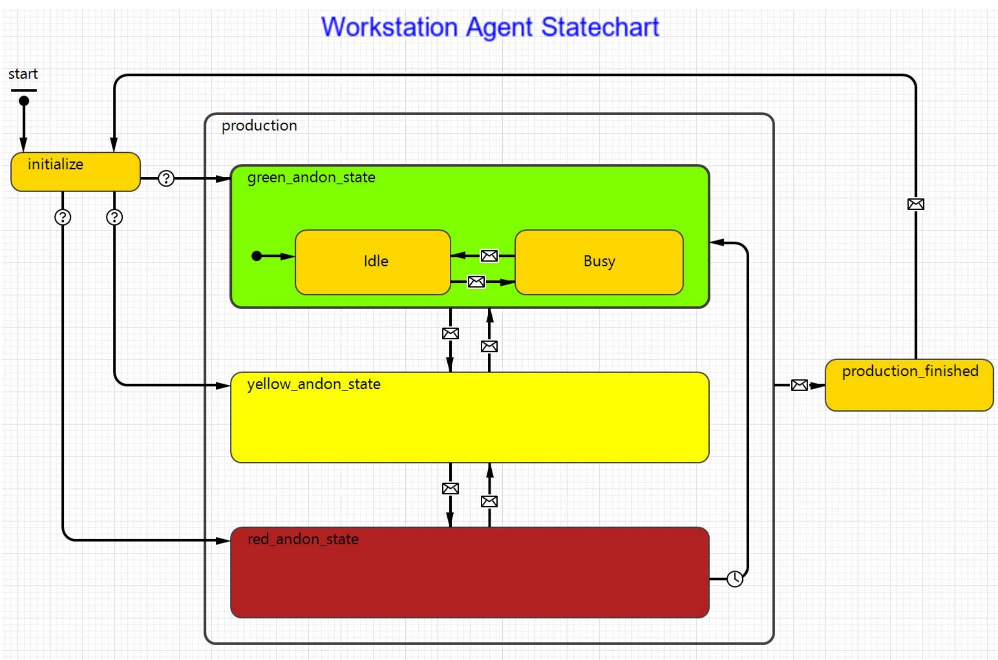

Figure 8: The Agent-Based Modeling state chart of the Workstation agent.

4 CONCLUSIONS AND FUTURE WORK

This work described our efforts to digitalize an educational and training facility that emulates an automotive assembly line using ABM and real-time data. The developed Agent-Based Digital Twin model can achieve bi-directional communication with the facility's automation systems and utilizes Industry 4.0 technologies to create a digital replica of important aspects of the production line. We used real-time data from Internet of Things (IoT) devices and automation systems to track the state of the facility's workstations and the assembled products' locations. By following an ABM approach, the model can be integrated into different simulation models of the facility, enhancing the DT's capabilities. Through this integration, simulation models can gain the capability to run simulation experiments using real-time data as the starting point, as well as an interface that allows the model to directly access and alter the behavior of the physical system based on the experimentation results.

This work is part of a multi-phase digitalization plan aimed at helping Small and Medium Manufacturers (SMMs) embrace Industry 4.0 technology by turning the Tiger Motors Assembly Line into a demonstration facility for Digital Manufacturing methods. The next steps of this work consider verifying and validating the model, creating visualization elements that assist the operators in understanding the system's state during production, and finally, using this system during production runs to assess its effectiveness in improving the practitioners' decision-making.

In the future, we plan to integrate additional assembly process elements into the DT model. These include a SCARA robot that inspects and sorts the cars at the end of the assembly line and an Automatic Storage and Retrieval System. Next, we want to integrate the showcased ABM DT model into the facility's previously created hybrid simulation model. As additional elements are introduced into the DT model, we want to apply data-driven simulation techniques to ensure that the models utilize all available real-time data sources to stay updated, addressing scalability concerns. This can be achieved by creating a high-level semantics-based description of the facility and automatically updating the models based on updated semantic data.

Finally, through the integration of the DT model with the hybrid simulation model, we want to unveil the points of information exchange between the Digital Twin model and the hybrid Simulation model. By understanding these points of information exchange, we anticipate enabling future researchers and simulation practitioners with enhanced capabilities to integrate real-time and bidirectional data sources into their simulation models.

ACKNOWLEDGMENTS

This work was supported by the Interdisciplinary Center of Advanced Manufacturing Systems (ICAMS) funded by the US Army, Grant Army-W52P1J-20–9-3045-Clin 3, awarded to ICAMS from the Industrial Base Analysis & Sustainment Program of the Industrial Base Policy Office of the Secretary of Defense.

REFERENCES

Ali, M. I., P. Patel, J. G. Breslin, R. Harik, and A. Sheth. 2021. "Cognitive Digital Twins for Smart Manufacturing". *IEEE Intelligent Systems* 36(2):96–100.

Augustine, P. 2020. "The Industry Use Cases for the Digital Twin Idea". In *The Digital Twin Paradigm for Smarter Systems and Environments: The Industry Use Cases*, edited by P. Raj and P. Evangeline, 79–105. Cambridge, MA: Elsevier.

Bécue, A., E. Maia, L. Feeken, P. Borchers, and I. Praça. 2020. "A New Concept of Digital Twin Supporting Optimization and Resilience of Factories of the Future". *Applied Sciences* 10(13):4482.

Bevilacqua, M., E. Bottani, F. E. Ciarapica, F. Costantino, L. Di Donato, A. Ferraro *et al*. 2020. "Digital Twin Reference Model Development to Prevent Operators' Risk in Process Plants". *Sustainability* 12(3):1088.

Botín-Sanabria, D. M., A.-S. Mihaita, R. E. Peimbert-García, M. A. Ramírez-Moreno, R. A. Ramírez-Mendoza, and J. d. J. Lozoya-Santos. 2022. "Digital Twin Technology Challenges and Applications: A Comprehensive Review". *Remote Sensing* 14(6):1335.

Dahmen, U., T. Osterloh, and J. Roßmann. 2022. "Modeling Operational Scenarios for Simulation-based Validation of Technical Systems". In *IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, edited by Y. Tingchun, 69–76. New York: Institute of Electrical and Electronics Engineers.

dos Santos, C. H., J. A. B. Montevechi, J. A. de Queiroz, R. de Carvalho Miranda, and F. Leal. 2022. "Decision Support in Productive Processes Through DES and ABS in the Digital Twin Era: a Systematic Literature Review". *International Journal of Production Research* 60(8):2662–2681.

Ejersbo, H., K. Lausdahl, M. Frasheri, and L. Esterle. 2023. "Dynamic Runtime Integration of New Models in Digital Twins". In *2023 IEEE/ACM 18th Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, edited by S. Gatir Yaman, 44–55. New York: Institute of Electrical and Electronics Engineers.

Grieves, M. 2014. "Digital Twin: Manufacturing Excellence Through Virtual Factory Replication". *White Paper* 1(2014):1–7.

Guerra-Zubiaga, D., V. Kuts, K. Mahmood, A. Bondar, N. Nasajpour-Esfahani, and T. Otto. 2021. "An Approach to Develop a Digital Twin for Industry 4.0 Systems: Manufacturing Automation Case Studies". *International Journal of Computer Integrated Manufacturing* 34(9):933–949.

Guerreiro, G., P. Figueiras, R. Costa, M. Marques, D. Graça, G. Garcia *et al*. 2020. "A Digital Twin for Intra-Logistics Process Planning for the Automotive Sector Supported by Big Data Analytics". In *Proceedings of the ASME 2019 International Mechanical Engineering Congress and Exposition*, edited by A. Cuitino, V02BT02A021. New York: American Society of Mechanical Engineers.

Hou, L., S. Wu, G. Zhang, Y. Tan, and X. Wang. 2021. "Literature Review of Digital Twins Applications in Construction Workforce Safety". *Applied Sciences* 11(1):339.

Huang, Z., Y. Shen, J. Li, M. Fey, and C. Brecher. 2021. "A Survey on AI-Driven Digital Twins in Industry 4.0: Smart Manufacturing and Advanced Robotics". *Sensors* 21(19):6340.

Katsigiannis, M., M. Pantelidakis, and K. Mykoniatis. 2024. "Assessing the Transition From Mass Production to Lean Manufacturing Using a Hybrid Simulation Model of a Lego® Automotive Assembly Line". *International Journal of Lean Six Sigma* 15(2):220–246.

Leng, B., H. Sun, G. Si, T. Xia, and H. Wang. 2021. "Digital Twin and Manufacturing Simulation Integrated Platform Embedded in Cyber-Physical System". *Journal of Physics: Conference Series* 1983(1):012117.

Macal, C. M. and M. J. North. 2009. "Agent-Based Modeling and Simulation". In *2009 Winter Simulation Conference (WSC)*, 86–98 https://doi.org/10.1109/WSC.2009.5429318.

Madni, A., C. Madni, and S. Lucero. 2019. "Leveraging Digital Twin Technology in Model-Based Systems Engineering". *Systems* 7(1):7.

Martínez, A., J. Díez, P. Verde, R. Ferrero, R. Álvarez, H. Perez *et al*. 2021. "Digital Twin for the Integration of the Automatic Transport and Manufacturing Processes". *IOP Conference Series: Materials Science and Engineering* 1193(1):012107.

Melesse, T. Y., V. Di Pasquale, and S. Riemma. 2021. "Digital Twin Models in Industrial Operations: State-of-the-Art and Future Research Directions". *IET Collaborative Intelligent Manufacturing* 3(1):37–47.

Mykoniatis, K. and G. A. Harris. 2021. "A Digital Twin Emulator of a Modular Production System Using a Data-Driven Hybrid Modeling and Simulation Approach". *Journal of Intelligent Manufacturing* 32(7):1899–1911.

Prarthana, V., P. Lavanya, N. J. Jain, L. Nagaditya, H. Nagavishnu, and K. Bhargavi. 2021. "Digital Twin Technology: A Bird Eye View". In *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, edited by V. Gunaraj, 1069–1075. New York: Institute of Electrical and Electronics Engineers.

Rasheed, A., O. San, and T. Kvamsdal. 2020. "Digital Twin: Values, Challenges and Enablers From a Modeling Perspective". *IEEE Access* 8:21980–22012.

Schluse, M., M. Priggemeyer, L. Atorf, and J. Rossmann. 2018. "Experimentable Digital Twins—Streamlining Simulation-Based Systems Engineering for Industry 4.0". *IEEE Transactions on Industrial Informatics* 14(4):1722–1731.

Shen, B., X. Xiong, H. Yu, P. Hu, L. Jiang, and H. Cai. 2023. "Online Interactive Simulation with Digital Twins for Collaborative Manufacturing". In *2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, edited by L. O'Conner, 462–467. Los Alamitos, CA: Institute of Electrical and Electronics Engineers Computer Society.

Stark, R., C. Fresemann, and K. Lindow. 2019. "Development and Operation of Digital Twins for Technical Systems and Services". *CIRP Annals* 68(1):129–132.

van der Valk, H., J. Hunker, M. Rabe, and B. Otto. 2020. "Digital Twins in Simulative Applications: A Taxonomy". In *2020 Winter Simulation Conference (WSC)*, 2695–2706 https://doi.org/10.1109/WSC48552.2020.9384051.

Wagner, R., B. Schleich, B. Haefner, A. Kuhnle, S. Wartzack, and G. Lanza. 2019. "Challenges and Potentials of Digital Twins and Industry 4.0 in Product Design and Production for High Performance Products". *Procedia CIRP* 84:88–93.

Wilking, F., M. Behringer, M. Fett, S. Goetz, E. Kirchner, and S. Wartzack. 2023. "Concept of a Modular and System Model Driven Digital Twin for Engineering Education". In *2023 18th Annual System of Systems Engineering Conference (SoSe)*, edited by A. Aïtouche, 135–142. New York: Institute of Electrical and Electronics Engineers.

AUTHOR BIOGRAPHIES

MICHAIL KATSIGIANNIS is a Ph.D. student in the Industrial and Systems Engineering department at Auburn University, AL, USA. He holds an Engineering Diploma (Integrated Masters) in Electrical and Computer Engineering from the Technical University of Crete. His research interests include Modeling and Simulation, Digital Twins, Industry 4.0, Robotics and Automation, and Advanced Manufacturing. His email address is mkatsigiannis@auburn.edu.

KONSTANTINOS MYKONIATIS is an Assistant Professor in the Industrial and Systems Engineering department at Auburn University. He holds an engineering B.Eng. in Production Engineering and Management from the Technical University of Crete and an M.S. and Ph.D. degrees in Modeling and Simulation from the University of Central Florida (UCF). Prior to joining Auburn University, he worked as an Assistant Professor at Wichita State University and as a simulation instructor at the Industrial Engineering and Management Systems at UCF. In addition to his academic positions, he has held professional engineering positions as a modeling and simulation professional in the Business Analytics and Industrial Engineering Division of Universal Orlando Parks and Resorts, the Institute for Simulation and Training, the Institute for Advanced Systems Engineering, and the Public Power Corporation S.A Hellas. His primary research interests are in the areas of hybrid modeling and simulation, advanced manufacturing systems, robotics and automation. His email address is kmykoniatis@auburn.edu.