ENABLING THE DIGITAL FACTORY THROUGH THE INTEGRATION OF DATA-DRIVEN AND SIMULATION MODELS

Olivia J. Pinon*, Dennis JL Siedlak*, Dimitri Mavris*
***Aerospace Systems Design Laboratory (ASDL)**
School of Aerospace Engineering
Georgia Institute of Technology, Atlanta, GA, 30332-0150, USA

Keywords: *Digital factory, Digital twin, Data-driven modeling, Disruption simulation, Production scheduling*

Abstract

The digital factory relies on the development of advanced integrated analysis and simulation capabilities that better utilize the vast amounts of data produced by production systems. This paper discusses the integration of data-driven and simulation models in the context of a partially automated aircraft assembly process, with the goal to analyze and assess the impact of disruptions on the production system and identify strategies to best recover from these disruptions.

1 Motivation

Manufacturing is a very established industry and yet factories are continuously evolving and becoming more complex and dynamic. Today, large numbers of sensors gather process data (time, energy consumption, environmental conditions, etc.) throughout the factory. This digital revolution, often defined as Industry 4.0 and driven by advances in big data analytics, artificial intelligence, robotics, augmented reality and additive manufacturing, will change the nature of design, manufacturing and the interactions between designers, manufacturers, suppliers, customers, and physical industrial assets [3, 13].

1.1 The Digital Factory

While the concept of a digital factory is not new (it has been introduced in the literature as early as 1950), it is now, more than ever, within reach of being achieved. Enabling the concept of the digital factory, however, is not without challenges. In particular, it requires the development of advanced integrated analysis and simulation capabilities that better utilize the vast amounts of data produced. As discussed by William P. King, chief technology officer of the Digital Manufacturing and Design Innovation Institute (DMDII), "manufacturing generates more data than any other sector of the economy" [15]. In a digital world, data is a strategic asset. If properly captured and used, data has the potential to unlock new sources of revenues [3] and enable significant improvements in innovation, product design, operational effectiveness, reliability, time-to-market, customer satisfaction, and sustainability. While the benefits and competitive edge brought by the digital thread/digital twin, and big data technologies in particular, are widely discussed and fully acknowledged within the community [23, 8], achieving these benefits is not without challenges.

As highlighted by Baur and Wee [3], "many manufacturing companies have deep expertise in their products and processes, but lack the expertise to generate value from their data." Hence, while many seamlessly integrated hardware, software and technology-based services exist that enable the digital factory concept and the generation of data at any level of detail desired, improved frameworks and platforms that organize data-analytic thinking and facilitate data-driven decision making across the many facets of the business are still needed [24].

1.2 A Paradigm Shift

To be effective, the digital factory needs to go beyond tracking and collecting data, and instead focus on transforming this data into intelligence [28] to help make plants more transparent and truly proactive. Doing so requires a shift in the way we approach and model factories. Traditionally, data is collected from the real factory and analyzed to help identify important process parameters, inefficiencies, etc. A separate effort then consists in building simulation models to help predict the state of the factory under different scenarios and eventually support decision making. To unleash the real value of data, one needs to enable the link between data and the digital twin (Figure 1), i.e. integrate data-driven models with simulation models. Doing so will enable the information and knowledge contained within the data to be integrated within the manufacturing system [9].

1.2.1 Opportunities for Machine Learning Applications

The rise of automation has enabled huge amounts of data to become available. For all purposes, this data, due its size and the many variables it captures, cannot be analyzed using traditional (manual) techniques. However, the continuously increasing amount of data available, as well as its high dimensionality and variety (sensor data, environmental data, machine tool parameters, etc.) [6, 27] lends itself particularly well to the use of machine learning algorithms [27, 20] and the development of data-driven models for prediction, detection, classification, regression, or forecasting [5, 12]. As such, machine learning algorithms have successfully been applied to support predictive manufacturing [16, 17], manufacturing process monitoring and control [7, 2, 1, 25, 11], etc.

1.2.2 Modeling of Production Systems

The integration of data and information with simulation models is traditionally a challenging, time-consuming task [10]. As discussed by Fowler and Rose, the time to "design, collect information/data, build, execute, and analyze simulation models" represents a grand challenge that tends to inhibit the extensive use of simulation in production process design [10]. While machine learning approaches have been implemented to support production scheduling [18, 26, 21, 14], the integration of data-driven predictive models with simulation models is rare. Equally lacking is the development of data-driven system level models or the implementation of knowledge-based approaches that take advantage of real-time sensor data to automatically generate simulation models and inform production systems and scheduling [29, 22, 4].

This research explores leveraging upfront data analysis and lower complexity simulation models to reduce the problem-solving cycle and support more expansive studies. In particular, the present paper discusses the integration of data-driven predictive and simulation models in the context of a partially automated aircraft assembly process with the goal to assess the impact of disruptions on the production system and identify how events such as quality assurance (QA) and repair processes can be best handled.

Section 2 describes the system of interest to this study. Section 3 discusses the approach developed, while Section 4 provides additional context to its implementation. Section 5 presents some results from the simulation study. Finally, Section 6 provide some concluding remarks.

2 System of Interest

This research focuses on a partially automated assembly process of a large commercial aircraft. The automated drilling systems is composed of two types of drillers:

- Drillers A are mounted on rails to sweep

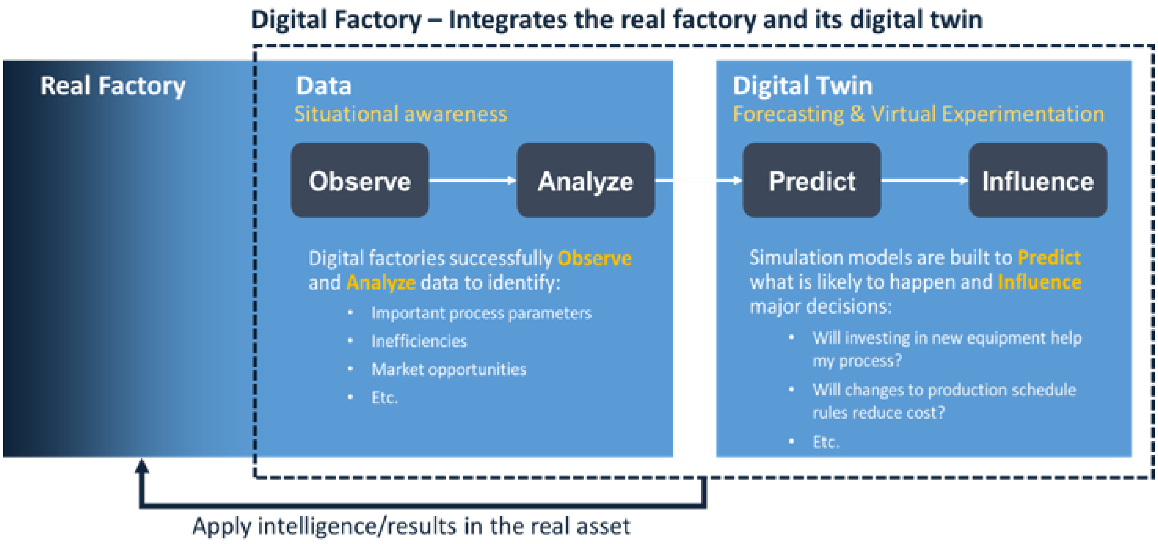

Fig. 1 : Transforming Data into Intelligence

around the circumference of the fuselage join, drilling and fastening as they go

- Drillers B move across the side of the fuselage, joining the upper shell of the fuselage section to the lower corresponding section

Both drillers are provided custom drilling profiles for each hole type and material stack-up configurations.

Manual assembly processes are also taking place alongside the drilling tasks with multiple technicians installing various components/elements throughout the section of the vehicle.

The data leveraged for this research consists of:

- Automated drilling data for 750,000 hole-by-hole entries for 325 aircraft, including:
  - Hole and process information (diameter, material, drill only or drill, countersink, fill)
  - Process parameters (XYZ hole location, stack thickness, tool life, etc.) and measured values/process outputs (proved diameter, error messages, etc.)
- Production data, including:
  - Actual and scheduled start and end timestamps for 250 aircraft at the position considered,
  - All manual assembly processes alongside drilling tasks
  - “Baseline” assembly processes and non-conformances, quality shakes, emergent removals, etc.

While significant efforts were required to clean and format the data prior to conducting any type of analysis, these efforts are beyond the scope of this paper and thus not discussed herein.

3 Approach

The approach developed focused on integrating into a digital testbed, predictive data analytics, an hybrid simulation model and process mining heuristics with the goal to provide predictive and impact assessment capabilities and support more informed scheduling decision making.

This approach, illustrated in Figure 2, is broken down into three major tasks:

- **Task 1 - Automated Drilling Cycle Time and Process Analysis:** The purpose of this task is to develop a cycle time prediction model for individual drilling job using the

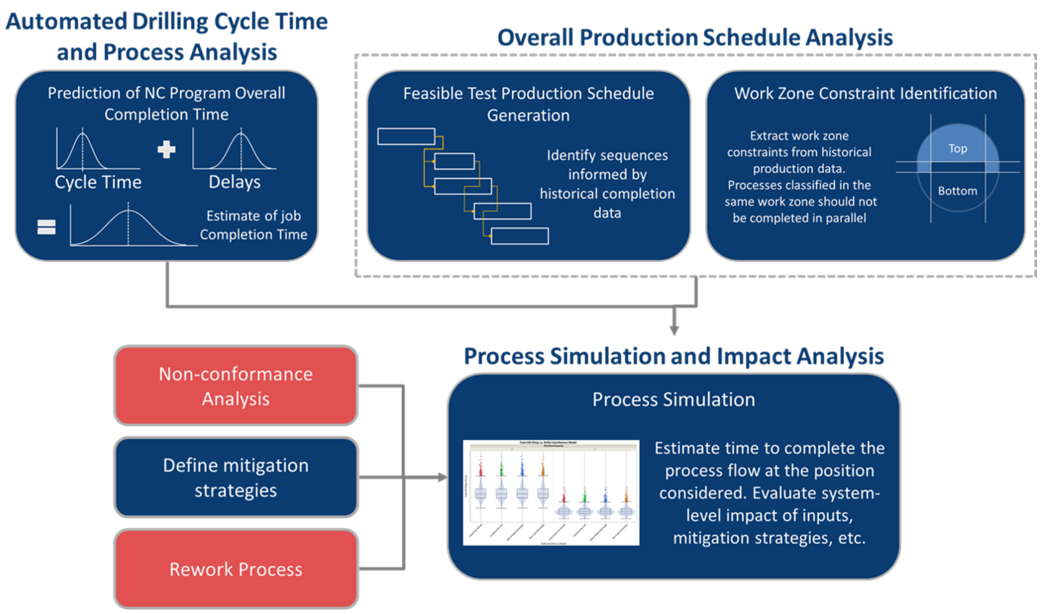

Fig. 2 : Overall Approach

vast amount of automated drilling data collected by the system.

- **Task 2 - Production Schedule Analysis and Baseline Production Schedule Generation:** The purpose of this task is to leverage process mining heuristics to stochastically generate a baseline production schedule based on the actual completion of previous aircraft. Such capability is particularly important in this context due to the absence of actual "baseline" from the raw data.

- **Task 3 - Process Simulation and Impact Analysis:** The purpose of this task is to develop a unified simulation model which integrates both a drilling and a schedule model. The unified model is an enhanced agent-based model with process mining heuristics added to simulate realistic interactions between manual processes (removing rivets, inspections) and automated processes (drilling holes) within the aircraft assembly line.

3.1 Task 1 - Automated Drilling Cycle Time & Delay Classification and Disruption Analysis

The purpose of this task is two-fold: 1) Develop data-driven models to predict the cycle time of individual drilling jobs and, 2) Identify the sources of common disruptions and delays.

3.1.1 Predictive Modeling of Drilling Cycle Time

Multiple predictive modeling techniques and approaches were investigated. A first approach, which focused on developing an adaptive regression model to predict the cycle time of individual drilling jobs, was first attempted. This approach, illustrated in Figure 3, integrated Random Forest models for error message predictions within a Neural Network model to predict drilling cycle time.

A second approach focused on developing a multi-layered perceptron (MLP) using the Levenberg-Marquardt algorithm for weight optimization was then implemented that provided better model predictive capabilities. This particular model was trained using the automated drilling data provided by the system (Figure 4). The model used 144 parameters, including ma-

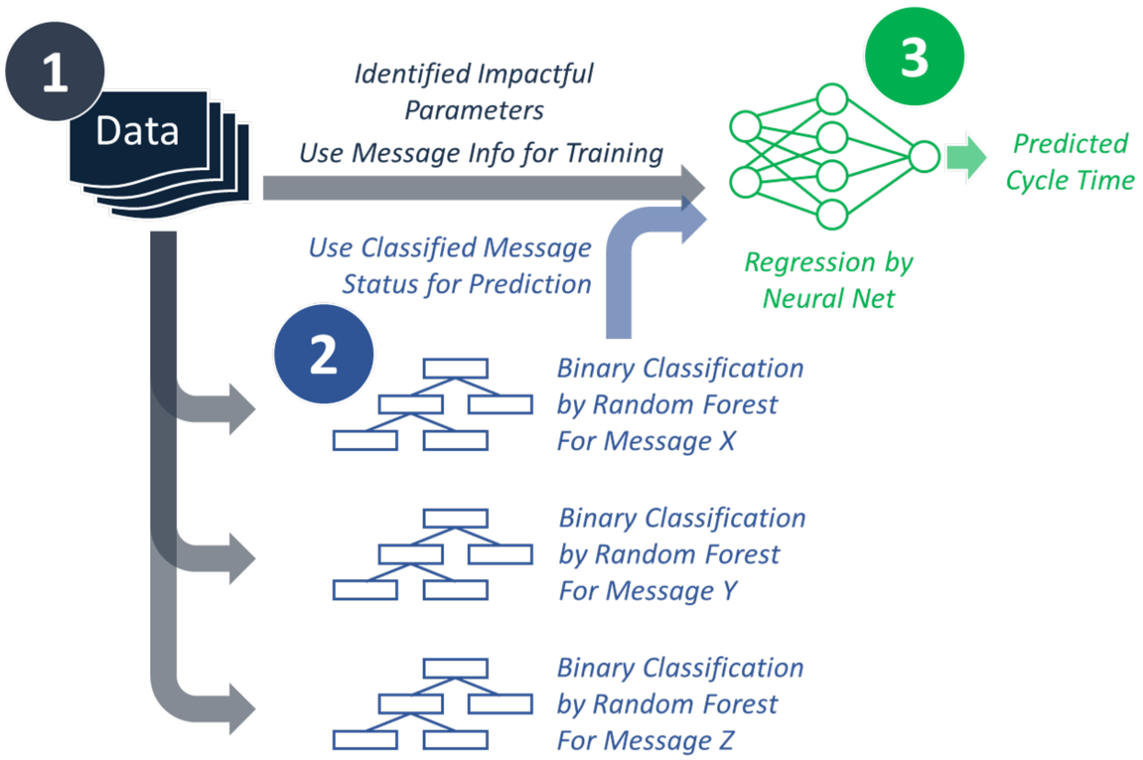

Fig. 3 : Adaptive Regression Model

chine number, remaining tool life, mechanical properties of the processing layers, and machine settings and had 20 to 30 hidden nodes. It predicted the cycle time of individual drilling jobs with a R^2 value of ~ 0.9 for both training and validation samples (Figure 5).

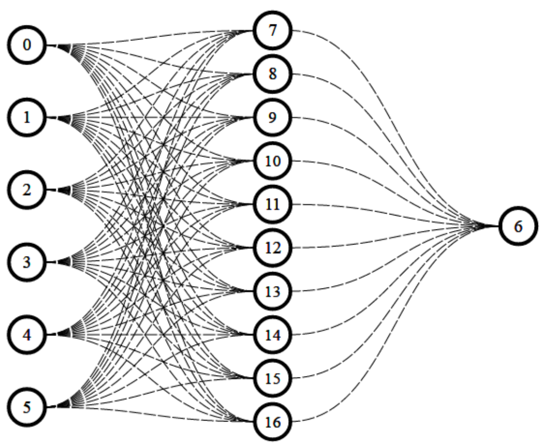

Fig. 4 : Model architecture: the model developed as a similar topology but uses 144 input nodes and 20 $\sim$ 30 hidden nodes

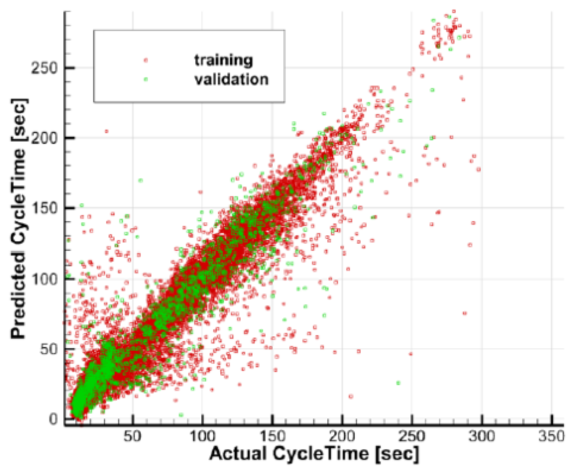

Fig. 5 : Actual vs. Predicted Cycle Time

3.1.2 Identification of Source of Delays

Gaps between drilling jobs were also studied to facilitate the modeling of the overall drilling process. In particular, the following sources of delays were identified and integrated in the model developed as part of Task #3:

- Shift change

- Tool change: either due to a change in process or a lack of remaining useful tool life
- Work zone interferences: the relative possible collocation of the drillers provides opportunities for disruptions. Work zone interferences between Drillers A and B, or between two drillers A at the top of the fuselage, for example, are known to create delays. Similarly, interactions between drillers and manual tasks are also cause for disruptions
- Other delays unexplained by drilling data alone (part shortages, rework, maintenance, etc.)

3.2 Task 2 - Production Schedule Analysis and Baseline Production Schedule Generation

The value of simulation in production analytics is well understood; however, the time and cost to develop a simulation model is sometimes prohibitive. This is particularly true when the process to be simulated lacks a clear precedence flow, as is the case for the problem at hand. Hence, developing a model to represent such process and, ultimately provide a testbed to examine improvements, is extremely challenging.

When faced with a lack of baseline, it is particularly important to identify the overarching goal of the simulation model. If the goal is to examine the logic of how the order of jobs to be completed is chosen, then a very detailed simulation model may be required. In the context of this study, where the focus is on identifying how events (such as QA and repair processes) impact the overall production flow, the simulation model is intended to provide a representative case to account for these events. The following three pieces of data/information are needed for the simulation:

1. Representative process order
2. Job constraints: jobs that can not be worked on at the same time
3. Job completion times

Identifying and codifying this information is commonly a very time consuming process which makes completing such simulation studies a challenge. Fortunately, there exists a wide variety of information available from a myriad of data sources throughout the factory. The following sections discuss how data analytics is leveraged to automatically extract the data required for the simulation study.

3.2.1 Process Flow Identification

The first piece of information to extract from production data is a potential process flow. However, unlike well-defined, automated processes, the assembly process of interest does not follow a strict precedence network. In other words, the assembly schedule changes from one vehicle to the next: in some cases, jobs completed near the beginning of the process are completed near the middle or end on a later vehicle. Therefore, this assembly process does not lend itself to the use of more traditional process mining techniques - techniques that attempt to identify decision branches in processes such as those found in call centers.

Yet, because the simulation only requires a representative job schedule (extracted baseline), the flow selected by the algorithm does not need to perfectly conform to process constraints. To arrive at an average impact of a selected strategy, a number of potential schedules can then be simulated.

The data used to determine the representative schedule contains the start and end times for every job competed for a sub-assembly process for approximately 100 vehicles. Approximately 700 jobs are completed per vehicle during this overall process.

The resulting schedule needs to be represented as a precedence network that can then be executed as a Simio process flow. Simio (SImulation Modeling framework based on Intelligent Objects) [19] was chosen as the modeling software for this work because 1) its object-oriented nature supports the automated generation of models, 2) the integration between the model and data-driven predictive models can be achieved using Simio's API, and 3) it supports distributed computing.

The algorithm discussed below aims at identifying a job ordering as well as precedence relationships. As mentioned above, the goal is not to identify *the* precedence network, since the assembly process does not follow a unique one, but rather identify one that is representative enough of the overall process.

The algorithm operates by analyzing historical sequences to identify the typical sequence position for each job. The first step is to identify the processes that occur in the majority of the sequences and their corresponding median process time and sequence location. These jobs are then initially ordered based on their median sequence location. As a first cut, precedence relations are added between jobs that are completed before another in 95% of the historical schedules provided. For example, if a fitup operation is finished before a drilling operation 95% of the time, then a precedence relationship is added.

With these "firm" predecessors set, softer, "preference" constraints are added based on a weighted probability draw. This is where the stochasticity of the process comes about. The probability that job i is completed before job j is calculated by Equation 1. This is intended to break apart jobs that commonly occur close to each other in the sequence. Therefore, if a job often occurs before another that is close in the sequence, it has a higher probability of being assigned a precedence relationship. If a pair fails the probability check, then no precedence relationship is added.

Once the initial set of precedence relationships are identified, the job start and end times are assigned. The start and end times are entered by technicians during job execution, so there may be some variability and errors that must be accounted for. The jobs being ordered based on their median historical sequence position, the start times are set as the maximum of their predecessors' end times. As some cycles are generated from the initial precedence identification, this process is repeated multiple times to get a representative spread of the start times.

$$\frac{\text{Percent of the time Job}_i \text{ occurs before Job}_j}{\text{Median sequence position between Job}_j \text{ and Job}_i} \tag{1}$$

Process Sequence Preference Probability

While this provides a good first cut at a representative sequence, additional data can be considered that add more constraints and eventually lead to a more representative schedule. First, if a job has never been completed in parallel with another job but appears in parallel in the generated schedule, then they can be separated by adding a precedence relationship between them. In such instances, jobs are separated such that the job with the earlier historical sequence position is executed first. Additionally, work zone constraints can be considered. Hence, if two jobs require access to the same work zone, then they cannot be completed in parallel and must be separated. The identification of work zones is discussed in the following section. Following the addition of these new constraints, the process start times are recalculated and the schedule can be used by Simio.

In summary, the process to generate test production schedules is as follows:

1. Identify "baseline" processes (those that occur in 95% of the provided schedules)

2. Identify a duration for each baseline job as the median of the previous completion times excluding weekends

3. Identify the median sequence position for each baseline job

4. Order the jobs based on their median sequence position for the schedule to be generated

5. Identify, for each job, all other jobs that "always" (95% of the time) occur before the current. Add precedence relationships between these jobs to enforce this ordering.

6. Further add "preference" predecessors based on Equation 1
7. Assign start and end times based on the defined predecessors
8. Separate parallel tasks based on historical and work zone constraints
9. Recompute start and end times to take into consideration the newly added constraints

3.2.2 Work Zone Constraint Identification

During the assembly process, multiple technicians are installing various components/elements throughout the section of the vehicle. However, jobs that are in the same area of the vehicle cannot be processed at the same time due to physical constraints or foreign object debris (FOD) considerations. Identifying work zone constraints is thus critical to the modeling exercise as two jobs that have to be done in the same work zone cannot be executed in parallel.

These work areas, however, are not nicely laid out in the process information and consequently can not be readily integrated in the model. Fusing information available about historical process completions, type of parts/components to be installed as well as their physical locations can help address this issue. Hence, information from the historical process completions can be combined with information about the parts and components to be installed to get a sense of what those work areas are. Because job completion information provides information about tasks that have been completed in parallel in the past, it is fair to assume that these tasks do not interfere with each other.

The physical location of the components in the vehicle and their associated jobs are also available. Therefore, jobs that have associated parts can be tagged with a physical location. The goal is to then divide the vehicle into work zones that can be used to constrain the process.

The work zone separations are identified using an optimization algorithm. Figure 6 shows an example work zone separation. The current vehicle has a temporary floor separating the upper and

$$p_{accept} = e^{\frac{\Delta f}{T}} \tag{2}$$

Probability of Accepting Worse Solution

lower sections, so work can be completed independently in the upper and lower halves. Within each upper or lower section, the optimization algorithm selects the location of the work zone dividers. The goal of the optimization algorithm is to minimize the number of jobs that are classified in the same work zone and occur in parallel. In other words, the optimizer helps ensure that jobs that are being completed in parallel are in different work zones.

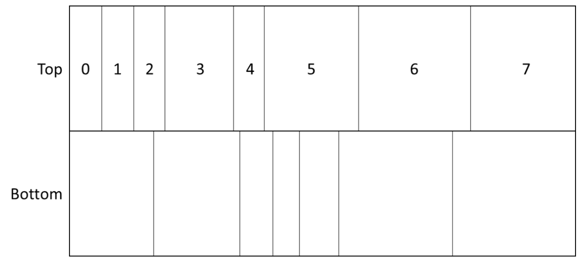

Fig. 6 : Example Work Zone Separators

The optimization is driven by a Simulated Annealing (SA) algorithm. The dividers are initially evenly spaced. Then, during each iteration, the algorithm randomly selects a divider to move. The selected divider can move anywhere between the two adjacent dividers with a 5 foot buffer zone to ensure that the created zone is about wide enough for a technician to complete a task. The objective function is then evaluated and the new solution is either accepted or rejected based on Equation 2, where Δf is the change in objective function and T is the current annealing temperature. As the problem begins to "cool", the temperature is dropped to reduce the probability of accepting a worse solution.

Figure 7 presents an example of the optimization's progress. With the regularly spaced divisions from the first iterations, there are about 55,000 overlapping jobs. Following the optimization, the algorithm identifies a solution with about 37,000 overlapping jobs. This solution can then be used to both separate jobs when generating the schedule and enforce constraints in the simulation model.

Hence, with the work zones identified, the generated schedule can be updated to observe the work zone constraints (step 8 in the schedule generation process). This is accomplished by checking the work zones that each task occupies in the schedule. If tasks that are planned in parallel occupy the same zone, then they are separated by adding a precedence relationship between them and the schedule start times are recalculated.

The simulation model discussed in Section 3.3.3 also enforces these constraints during the execution of a schedule. Each work zone is modeled as a seizable resource that is required to process a task. As such, if a task attempts to start but its required work zone is occupied, then it will wait to start until the work zone becomes available.

3.3 Task 3 - Process Simulation and Impact Analysis

The overarching simulation model integrates a scheduling model (generated schedule and identified work zone constraints) and a drilling model that includes the individual drilling cycle time predictive models. As such, the model leverages the two sources of data discussed in Section 2:

- Drilling data that is used to generate the prediction models for the driller cycle times as well as identify the drilling tasks (and drillers) that belong to each individual NC program.
- Production data, which includes location and time information regarding both manual tasks and NC programs, expected and actual durations and other specific tasks characteristics such as description, type, etc

The characteristics of each model are discussed below.

3.3.1 Scheduling Model Characteristics

The scheduling model combines the input of a job schedule that is generated stochastically based on the identified baseline production schedule discussed in Section 3.2. Hence, the model stochastically varies the processing time of the tasks within the pre-generated baseline production schedule based on information from real schedules. The model can also evaluate which tasks can be completed in parallel based on:

- Required predecessors (processes within the same line that have to be completed)
- Work zone constraints, which are spatial-dependent, and determine whether jobs can be completed in parallel or not

The model also has the potential ability to reschedule tasks based on disruptions happening within the system.

3.3.2 Drilling Model Characteristics

The drilling model simulate sequences of NC programs and has the following characteristics:

- It integrates the predictive models for drilling cycle time developed in Task #1
- It allows for work-zones to be simulated. Hence, each driller has working space assigned with respect to well-defined spacing rules
- It allows for two types of interference to be captured:
  - Work-zone interference: between Driller A and Driller B)
  - Driller A interference at the top of the fuselage (with variable threshold added in inches)
- It allows for different interferences rules to be enforced
  - Continue with Line: This model benefits the machine that uses repetitive tasks, as long as it was the first one repeating it, which means that every time a driller used a resource, and will re-use that resource in the next task,

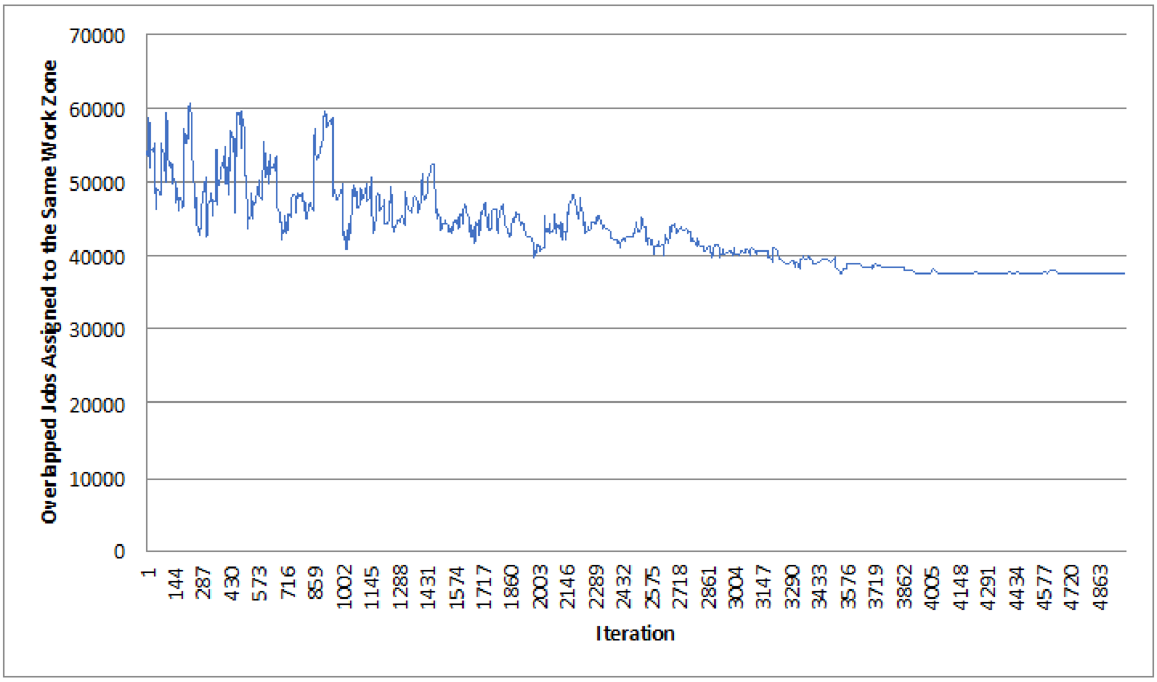

Fig. 7 : Work Zone Optimization Progress

it will have priority for that resource, halting other possible takers until it finishes all successions with that particular resource

– First Come First Served (FCFS) : Will assign a resource to any driller waiting for one as soon as one is released, no matter if the driller who just released needs the same resource for the next task

– Driller A/Driller B Prioritization: Will give priority to either Driller A or Driller B

3.3.3 Overarching Model Characteristics

The overarching model simulates realistic interactions between manual processes (removing rivets, inspections) and automated processes (drilling holes) within the assembly line considered. As such, the main purpose of this model is to serve as a platform where:

- Disruptions within the schedule involving manual and automated tasks can be implemented and analyzed to determine their impact
- Each of the sub-components can be used in isolation to make more specialized analysis either at the schedule or the driller level
- More data can be used to simulate more complex factory behaviors

The overarching model thus benefits from the capabilities of both Scheduling and Drilling models. The Scheduling model, instead of determining the tasks times via regressions, generates tasks (NC program sequences) that the Drilling model has to complete. The Drilling model then signals the end of a tasks so that the Scheduling model can proceed and carry on with the schedule. Both drilling interferences within the Drilling model and work zone constrains within the Scheduling model are taken into account.

3.3.4 Data Flow and Overall Architecture

The data flow, overall architecture and computing languages/platforms leveraged are illustrated in Figure 8. The generated schedule is output into a format that can be read into the overarching model. The overarching model in Simio evaluates a schedule based on precedence relationships while allowing for stochasticity in the

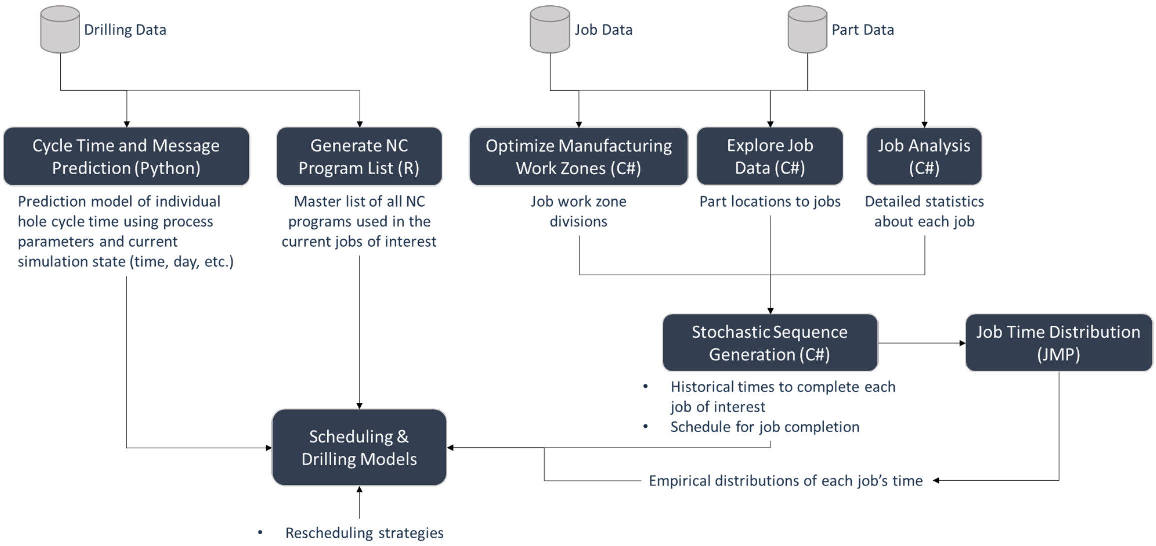

Fig. 8 : Data Flow, Overall Architecture and Computing Languages/Platforms Leveraged

process time and job success. Simple triangular distributions are fit to the historical job completion times to provide the process times for the model. For those jobs that require a work zone because they have a component/element/part defined, the simulation will seize the corresponding work zone resource. In this way, the simulation can enforce the work zone constraints if the schedule needs to be modified during the simulation run.

4 Implementation

One question driving the development of the simulation model involved analyzing the impact of a job failure and subsequent non-conformance repair on the overall process. Figure 9 presents an overview of the simulation flow when a QA failure is encountered. In this figure, Job 231 fails QA and must go through an engineering analysis and repair process. While this is ongoing, Job 232 is required to wait for 231 to be reworked, thereby delaying the process. However, during this time, other jobs that are planned later in the schedule (302, 305, 315, and 325 in this instance) can be completed as they are not tied to Job 231's precedence relationship. As such, the simulation decides to complete these tasks while the rework on 231 is being completed to help minimize the impact of that failure.

When investigating the impact of rework and alternative mitigation strategies, the two research questions of interest are:

1. Research Question #1: How should I prioritize the list of quality problem fixes that must pass through engineering before being resolved?

2. Research Question #2: When a disruption occurs, what strategies best mitigate the impact of the disruption?

Research question #1 focuses on identifying how quality issues should be prioritized through non-conformance review. There are only so many engineers who can recommend and approve non-conformance fixes, so it is possible that quality issues can build up and lead to significant delays in the process. As such, different ranking heuristics can be identified to best mitigate these problems. The data about the non-conformance review was not available at the time of this study, so this question will be investigated in future studies.

Research question #2 focuses on finding how various strategies to select tasks to complete while waiting for the QA analysis impact the overall completion time of the schedule. As such, various strategies are investigated including first

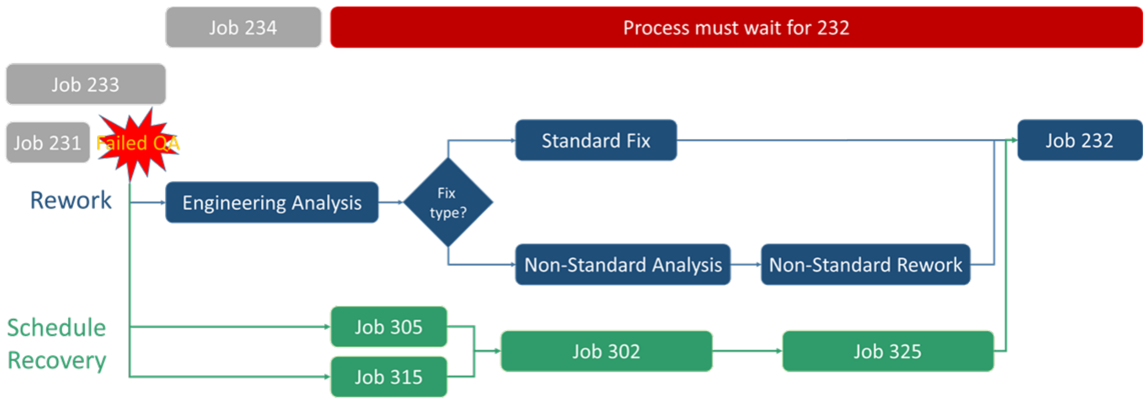

Fig. 9 : Simulation Schedule Recovery Process

in/first out, shortest available job, longest available job, and processing the job with the most successors first.

For both research questions, the primary metric of interest is the process time required to complete the schedule. The simulation is run for multiple replications (because the process times and QA failures are stochastic) and for multiple testing schedules (because the schedule generation itself is also stochastic). To demonstrate the analysis provided by this approach, sample results from these studies are presented in the following section.

5 Results

The simulation results presented in this section investigate different strategies to employ to recover from a non-conformance. Using the process described in Figure 2, simulation experiments are conducted to test the following strategies to select tasks to complete while waiting for a non-conformance inspection to be completed:

- Average processing time/schedule location: This rule favors completing longer tasks that are earlier in the schedule in an attempt to free resources for smaller jobs once the non-conformance review is completed

- Earliest in schedule: Favors tasks purely based on when they happen in the schedule to try to complete predecessors for as many upcoming tasks as possible

- Largest average duration: Favors long tasks to free resources later on

- Smallest average duration: Favors shorter tasks to complete as many as possible during the rework cycle

- Most successors: Favors tasks that have the most successors to open up more possibilities to complete future tasks

Upon encountering a non-conformance, the simulation uses the rule selected for that run to choose task(s) to complete during the non-conformance review. Figure 10 shows the impact of the selection rule choice on the overall process flow time. This figure is presented as a Simio Measure of Risk & Error (SMORE) plot. This is similar to a standard box-and-whiskers plot except that the brown and light blue boxes centered around the mean and quantiles denote the statistical confidence of those measures. The lighter blue bars extending above the whiskers show the histogram of raw results. From Figure 10, the two strategies targeting longer duration tasks for completion take longer to finish the overall process than the other three strategies. Figure 11, which shows the number of tasks completed during the rework process, provides a possible explanation for this impact. By targeting long duration tasks, fewer tasks can be completed during the rework process (as expected). Hence, it appears that this overall process benefits more from completing more short duration tasks than

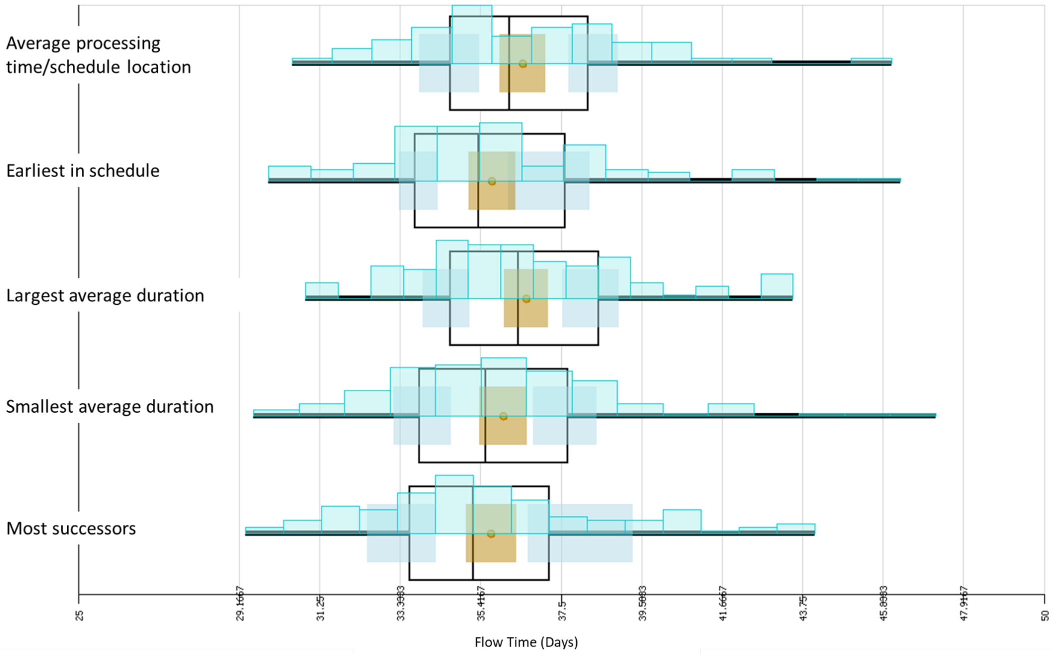

Fig. 10 : Impact of Selection Rule on Overall Process Flow Time

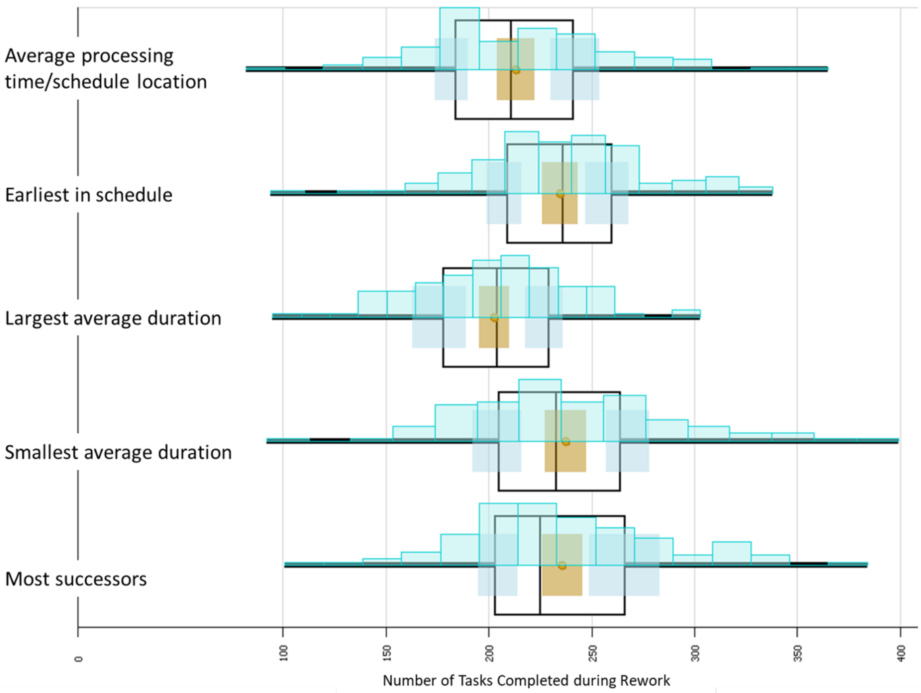

Fig. 11 : Impact of Selection Rule on the Number of Tasks Completed During Rework Periods

fewer, long duration tasks. However, selecting to complete the longest duration tasks first does produce the smallest maximum processing time. Consequently, there may exist a hybrid strategy that would allow to complete many short duration tasks while preventing having long tasks near the end of the process.

These results demonstrate the nature of the information that can be generated by integrating multiple data sources, data analytics, machine learning, process mining techniques, and simulation. While not mature enough to be used operationally, this process is intended to provide a starting point for future, more focused analytics and simulation studies.

6 Concluding Remarks

The present paper discussed the integration of data-driven and simulation models in the context of a partially automated aircraft assembly process to 1) assess the impact of job failure and subsequent non-conformance repair on the overall process, and 2) test strategies to best recover from such disruptions. In particular, this paper has discussed and/or demonstrated:

- The utilization of statistical methods and prediction techniques to identify predictive features to be included in the cycle time prediction model
- The investigation of interaction between both automated and manual jobs to identify relationships and disruptions to the production system
- The creation of a data-driven, robust schedule analysis tool to identify most commonly occurring sequences of jobs from actual schedule completion information
- The creation of a work zone identification tool to infer spatial constraints from schedule completion information as well as part installation requirements
- The integration of both data-driven and simulation models to:
  - Evaluate critical tasks that significantly impact the overall flow
  - Identify promising recovery rules and/or schedules

While not mature enough to be used on the shop floor, this study has nonetheless contributed to demonstrate the value of integrating data-driven and simulation models as a means to obtain more visibility on how to best recover from disruptions.

7 Acknowledgments

The authors would like to acknowledge Boeing Research & Technology for funding this research. In particular, the authors would like to thank Jonathan Fulton for sharing his expertise and providing support and feedback during the development of the approach presented herein. Finally, the authors would like to acknowledge Dr. WoongJe Sung, David Solano and Burak Bagdatli for their significant contributions to this work.

8 Copyright Statement

References

[1] Abellan-Nebot J and Subiron F. A review of machining monitoring systems based on artificial intelligence process models. *International Journal of Advanced Manufacturing Technology*, Vol. 47, pp 237–257, 2010.

[2] Alaeddini A and Dogan I. Using bayesian networks for root cause analysis in statistical process control. *Expert Systems with Applications*, Vol. 38, pp 11230–11243, 2011.

[3] Baur C and Wee D. Manufacturing's next act. Technical report, McKinsey Digital, 2015.

[4] Bergmann S and Strassburger S. Challenges for the automatic generation of simulation models

for production systems. *Proc SCSC '10 Proceedings of the 2010 Summer Computer Simulation Conference*, Ottawa, Ontario, Canada, July 2010. Society for Computer Simulation International.

[5] Choudhary A, Harding J, and M.K. T. Data mining in manufacturing: A review based on the kind of knowledge. *Journal of Intelligent Manufacturing*, Vol. 20, pp 501–521, 2009.

[6] Davis J, Thomas E, Graybill R, Korambath P, Schott B, D, Wang J, and Wetzel J. Smart manufacturing. *Annual Review of Chemical and Biomolecular Engnieering*, Vol. 6, No 141-160, 2015.

[7] Dey S and Stori J. A bayesian network approach to root cause diagnosis of process variations. *International Journal of Machine Tools & Manufacture*, Vol. 45, pp 75–91, 2005.

[8] ElMaraghy H and ElMaraghy W. Smart adaptable assembly systems. *Proc Procedia CIRP 44*, 2016.

[9] Elovici Y and Braha D. A decision theoretic approach to data mining. *IEEE Transactions on Systems, Manufacturing, and Cybernetics, Part A, Systems and Humans*, Vol. 33, No 1, pp 42–51, 2003.

[10] Fowler J. W and Rose O. Grand challenges in modeling and simulation of complex manufacturing systems. *Simulation*, Vol. 80, No 9, pp 469–476, 2004.

[11] Guh R.-S. Integrating artificial intelligence into on-line statistical process control. *Quality and Reliability Engineering International*, Vol. 19, pp 1–20, 2003.

[12] Harding J. A, Shahbaz M, Srinivas, and Kusiak A. Data mining in manufacturing: A review. *Journal of Manufacturing Science and Engineering*, Vol. 128, No 4, pp 969–976, 2006.

[13] Hartmann B, King W. P, and Narayanan. Digital manufacturing: The revolution will be virtualized. Technical report, McKinsey Digital, 2015.

[14] Huet A. Optimization and analysis aid via data-mining for simulated production systems. *European Journal of Operational Research*, Vol. 173, pp 827–838, 2006.

[15] Kelly R. The digital-manufacturing revolution: How it could unfold. Technical report, McKinsey Digital, 2015.

[16] Lee J, Lapira E, Bagher B, and Kao H.-a. Recent advances and trends in predictive manufacturing systems in big data environment. *Manufacturing Letters*, Vol. 1, pp 38–41, 2013.

[17] Lee J, Lapira E, Yang S, and Kao H.-A. Predictive manufacturing system - trends of next-generation production systems. *Proc 11th International Federation of Automatic Control (IFAC) Workshop on Intelligent Manufacturing Systems*, Sao Paulo, Brazil., May 2013.

[18] Li X and Olafsson S. Discovering dispatching rules using data mining. *Journal of Scheduling*, Vol. 8, pp 515–527, 2005.

[19] LLC S. Simio, version 8.141, 2016. Sewickley, Pennsylvania.

[20] Pham D and Afify A. Machine learning techniques and their applications in manufacturing. *Part B: Journal of Engineering Manufacture*, Vol. 219, pp 395–412, 2005.

[21] Priore P, Gomew A, Pino R, and Rosillo R. Dynamic scheduling of manufacturing systems using machine learning: An updated review. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 28, pp 83–97, February 2014.

[22] Soderberg R, Warmefjord K, Carlson J. S, and Linkdkvist L. Toward a digital twin for real-time gepmetry assurance in individualized production. *CIRP Annals - Manufacturing Technology*, Vol. 66, pp 137–140, 2017.

[23] Tambe P. Big data investment, skills and firm value. *Management Science*, Vol. 60, No 6, 2014.

[24] Uhlemann T. H.-J, Schock C, Lehmann C, Freiberger S, and Steinhilper R. The digital twin: Demonstrating the potential of real time data acquisition in production systems. *Proc Procedia Manufacturing - 7th Conference on Learning Factories, CLF 2017*, Vol. 9, pp 113–120, 2017.

[25] Weese M, Martinez W, Megahed F. M, and Allison J.-F. L. Statistical learning methods applied to process monitoring: An overview and perspective. *Journal of Quality Technology*, Vol. 48, No 1, pp 4–27, 2016.

[26] Witkowski T. Data-based scheduling production with simulation and optimization methods. *Proc 7th IEEE International Conference on In-*

telligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2013.

[27] Wuest T, Weimer D, Irgens C, and Thoben K.-D. Machine learning in manufacturing: Advantages, challenges, and application. *Production & Manufacturing Research*, Vol. 4, No 1, pp 23–45, 2016.

[28] Zoeller K. Digital factory is key to increasing manufacturing capacity say industrial automation north america 2016 exhibitors. Industrial Technology, 2016.

[29] Zou J, Chang Q, Lei Y, and Arinez J. Production system performance identification using sensor data. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 48, No 2, pp 255–264, February 2018.