Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 81 (2019) 962–967

www.elsevier.com/locate/procedia

52nd CIRP Conference on Manufacturing Systems

A Framework for Pilot Line Scale-up using Digital Manufacturing

Malarvizhi Kaniappan Chinnathai*, Zeinab Al-Mowafy, Bugra Alkan, Daniel Vera, Robert Harrison

Automation Systems Group, WMG, University of Warwick, CV4 7AL, West Midlands, UK

* Corresponding author. Tel.: +44 7831001684. *E-mail address:* m.kaniappan-chinnathai.1@warwick.ac.uk

Abstract

Pilot lines are essential test-beds for process and product validation before the establishment of production lines. However, there is a lack of well-defined methodology for pilot line scale-up. To better support this transition, Virtual Models can be integrated with Discrete-Event Simulation (DES) models for potential production-line configurations. However, the validation of the developed models is hardly possible due to the absence of a physical counterpart. Therefore, this paper proposes a framework to increase the accuracy of the DES scale-up models with Virtual Modelling tools and Ontology. Subsequently, a test-case is used to explain the concept.

© 2019 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/3.0/)
Peer-review under responsibility of the scientific committee of the 52nd CIRP Conference on Manufacturing Systems.

Keywords: Digital manufacturing; Pilot line; Discrete-Event Simulation;

1. Introduction

The challenges faced by today's manufacturing industries are fueled by the increased product variety, rapid changes in technology, reduced time-to-market and shortened product life-cycle [1]. To cope up with the reduced time-to-market, firstly, it is important to achieve reduced time-to-volume i.e. to advance from the conceptual phase to full-volume production with increased thrust. During the conceptual phase, it is not uncommon for manufacturing industries to produce prototypes for purposes such as testing and validation of product, process and resource design. As it is crucial to achieve a successful transition from design phase to time-to-volume, it is essential to use pilot lines to identify potential disturbances prior to commissioning of the line [2]. A myriad of issues actually arise in early design phases and are not detected until commissioning; anticipating these issues before commissioning of production lines can ensure successful upscaling that can provide a competitive market advantage [3,4,5]. A successful scale-up project significantly reduces the time-to-market which consequently enables the industry to secure more revenue by dominating the market [5]. Although a plethora of articles have been published pertaining to the identification and management of disturbances and issues that could be faced during the up-scaling procedure [2,6,7], there is still lack of a robust methodology to enable the scale-up process in a smooth way. To support the transition from planning phase to full-volume, however, simulation and modelling is identified as one of the enabling technologies [8,4,6].

The concept of digital manufacturing has previously been found to support the manufacturing system and detect potential disturbances and issues affecting the line [3]. For this purpose, there are several commercial tools available, however, the underlying principles and techniques on which they function varies widely. In this paper, two simulation methods i.e. Virtual Modelling and Discrete-Event Simulation (DES) are identified and integrated with pilot line data to support the scale-up process. For several years, DES has been widely used for

2212-8271 © 2019 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/3.0/)
Peer-review under responsibility of the scientific committee of the 52nd CIRP Conference on Manufacturing Systems.
10.1016/j.procir.2019.03.235

supporting manufacturing industries [9]. DES finds use in identifying and analysing potential scale-up scenarios with input data from the pilot line [8]. However, as a standalone tool, DES does not have the capability to analyse the feasibility of the modelled scenarios of the future manufacturing line; this could potentially result in a situation where the solution offered through simulation might not actually be possible to realise. In specific, the assumption of station processing time values of potential production line models due to the absence of real system could lead to misleading results. To overcome this drawback, DES software module can be integrated with a Virtual Modelling software that models the kinematics, geometry and the logical behavior of the workstation resources. Commercially available PLM suites offer this capability to integrate multi-level software modules, but their implementation, training and license cost is exacting [10]. Moreover, there is requirement for the integration of heterogenous software tools within the overarching concept of digital factory [11].

1.1. Summary

From the above-mentioned discussion, the key points can be summarized as follows: i) the use of digital software modules can support the upscaling phase ii) DES software, if used as a standalone module, is not smart enough to identify whether the assumed station process times for future scenarios is feasible or not and iii) the integration of heterogenous digital software modules is aligned with the concept of digital factory.

1.2. Key contribution

Therefore, the core benefits of this paper are twofold i) proposal of an approach for integration of data from Virtual Modelling tools with an ontology software to calculate station process time such that the accuracy of the DES models are improved and ii) supporting the transition from pilot line to full-scale, subsequently shortening the time-to-market.

2. Literature review

2.1. Digital manufacturing

The notion of using simulation tools for manufacturing is not a new one. The software tools differ in their method and level of detail with which they model the system. This review briefly touches on production line modelling, namely Discrete-Event Simulation and workstation modelling referred to as Virtual Modelling.

2.1.1 Difference between DES and Virtual Model

Amongst the available tools for modelling the production line for operational research, Discrete-event Simulation is identified to be the most popular one [12,13] . Conventionally, DES is used for operational phase analysis, but its benefits can be exploited during the early stages of production as well [14]. The benefits of employing DES during early design stage include layout planning, material handling design, etc. and during the operational phase for scheduling and operational policies, and real-time control. However, in DES, analyses are performed by modelling the system with higher level of abstraction with the process and workstation level detail not included in the model; the focus is on detailing the production line and product flow. On the other hand, Virtual Modelling tools are used to model and analyse the system at the workstation or machine-level. They encompass information about the kinematic model (geometry and joint), behavior model (transition and states) and the reference coordinate system [3]. Moreover, they can be used to analyse ergonomics, collision detection, validation of PLC codes and design planning [11].

2.1.2 Benefits of integrating DES and Virtual Model

The primary benefit of integrating the Virtual Modelling tool with DES is to support the production-line level model in DES with the workstation-level details such as station processing times, breakdown information, robot motion time, human performance modelling, energy consumption and layout modifications [15,11].

Several commercial PLM suites have software modules that perform Virtual Modelling and DES. Additionally, these modules are present on an integrated platform that supposedly allows the sharing of data in a seamless way and thereby realizing the integration of Virtual Models and DES models. Although PLM tools have this capability, the tools are not affordable for SMEs due to i) cost of training and license ii) cost of changing infrastructure to adapt to the PLM environment iii) replacing any existing specialized software with the PLM toolset and the cost of implementation of PLM [16,17]. Moreover, from the view of digital factory, it is difficult to integrate PLM tools with heterogenous software and databases [18].

2.2. Summary

An analysis of articles about digital manufacturing indicates the following: i) quantification of the benefit of integrating heterogenous digital tools and ii) the lack of knowledge on the benefits and the procedure for integration of Virtual Modelling with DES to successfully support smooth transition from planning to full-volume. Therefore, in this research study, the authors propose an approach to support the transition from pilot phase to full-scale production by leveraging the integration of Virtual Modelling tool and DES.

3. Methodology

The research presented in this paper is primarily aimed at upscaling of assembly systems. The core idea of this research article is to share relevant workstation data from Virtual Modelling tool and the existing pilot line with an ontology tool to generate a list of station process times. The station process time data is necessary for ensuring that results of DES are

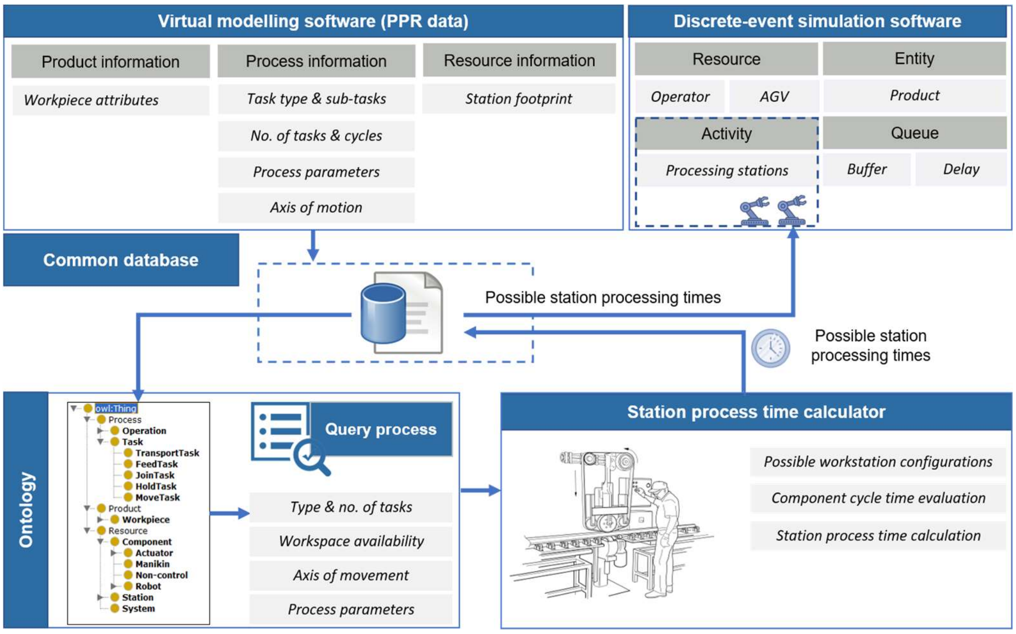

Fig. 1. Methodology.

realistic. The assumption of the process times could lead to situations where the models have workstation process times that might be too high or too low for the considered workstation configuration which could adversely affect the simulation results. The information sharing between the different tools is achieved using common database. From Fig 1, the common database model is a centralized database model which has been created in a way to support the integration of different software modules. In this paper, a common database scheme was designed to store the Virtual Model information in 3 tables: Product, Process and Resource. Each table has columns to represent the considered parameters and their respective IDs. A relational mapping between the Virtual Model and ontology classes facilitates provision of data for query and inserts the calculated results back into the tables. The common database model allows automating the integration between the virtual modelling software, ontology software and DES. The key concepts of the methodology can be explained as i) integration of Virtual Model with Ontology and ii) station process time calculator.

3.1 Integration of Virtual Model with Ontology

Virtual Modelling tools have capability to store information about product, process and resource at the workstation level; this information can also be shared with other tools. Within the context of this paper, a manufacturing resource comprises of system, station and component with increasing level of detail. A component is defined as the basic unit of a system that can be sub-divided into elements [19]. As an example, a robot can be considered as a component and the drives and motors of this component are the elements. The data from the existing pilot line serves as the crucial input for creating the Virtual Model. Table 1 shows the data intended to be used by the ontology model.

It is important to note that a significant proportion of this data is obtained from the existing pilot line. The task types are decomposed as shown in Table 1, however, the inspection and testing operations are not included within the scope of this research [20]. The axis of motion of the system resources in the existing line essentially enables removing components that have less axis of motion from future workstation configurations. This helps eliminating options that have less productivity than the existing components in pilot line, with the underlying assumption that an increase in the axis of motion, i.e. from a 3D gantry to six-axis robot, increases the productivity.

Table 1: Input data for ontology

Data type	Description
Workpiece attributes	Product features that are necessary to filter system resources that can perform the assembly.
Task type	Five types of tasks are considered: move, hold, transport, feed and join.
No. of tasks	The number of tasks that are performed in a workstation.
No. of cycles	The total number of cycles to perform an operation at the workstation.
Sub-tasks	The sub-task corresponds to the specific actions that are executed to achieve a task
Process parameters	Process parameters represent the accuracy, repeatability, force requirement, torque etc, for carrying out an operation.
Station footprint	The dimensions of the workstation that helps determine the available space to configure the workstation.
Axis of motion	The degree of freedom of the ‘current resources’ that are used in the pilot line/virtual model

Ontology is defined an explicit specification of a conceptualization and the development of ontology enables the sharing of common understanding of a domain between people and application systems [21]. The idea behind the use of the ontology model is that the process parameters, task type,

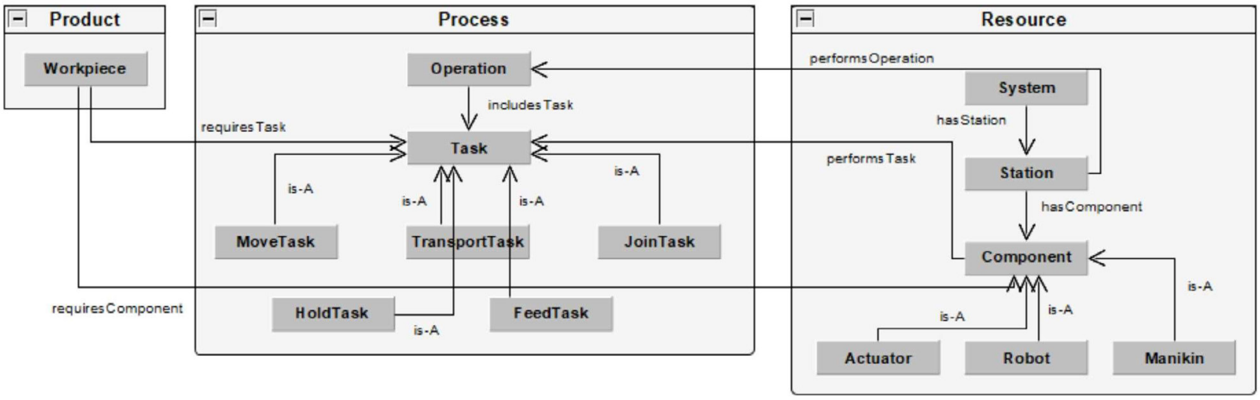

Fig. 2. Ontology definition in protégé.

number of tasks, axis of motion and station footprint data can be used to filter an available catalogue of assembly equipment to find those that meet the requirements. In this paper, the software protégé [22] is used to define the ontologies and three classes, namely product, process and resource as shown in Fig. 2. Three types of assembly components are considered within the scope of this paper: actuator, manikin and robot. Although, the proposed approach is suitable for all the assembly components and task type considered, this research article will focus on the 'move components' and 'hold components'. A catalogue of components can be created, either in the database or protégé, which consists of potential assembly equipment that are at the disposal of the industry.

Following the definition of ontology, a query operation using SPARQL on the generated equipment list will enable identifying the components that can i) perform the required number and type of tasks ii) fit within the available workspace iii) able to satisfy the process parameters and iv) have the required axis of motion. From the resulting list of components, the next step is to calculate processing time of the workstation when the selected components are used. Essentially, the station process time is expected to vary with component and the method of calculation is explained in the next section.

3.2 Station process time calculator

The station process time calculator (Fig 3) considers the type of task, either 'move' or 'hold', and the selected components for each are listed as *[M_1, M_2 . . . M_n]* and *[H_1, H_2 . . . H_k]*, where 'n' is the total number of selected components for 'move' task and 'k' is the total number of selected components for 'hold' task. From Fig 3, the 'sub-task level' shows the sub-tasks performed for a pick and place operation, wherein two tasks 'move' and 'hold' are involved. The information in the sub-task level are acquired from the virtual model. The motion times for the sub-tasks of each of components *[M_1, M_2 . . . M_n]* and *[H_1, H_2 . . . H_k]* are calculated with data from different sources: physics-based model of the component that can calculate the motion time, experience-based motion time, machine-learning from previous projects, motion time from component datasheet or from virtual modelling software. Additionally, it is important to understand the distance that actuators are displaced by during the 'move' and 'hold' tasks

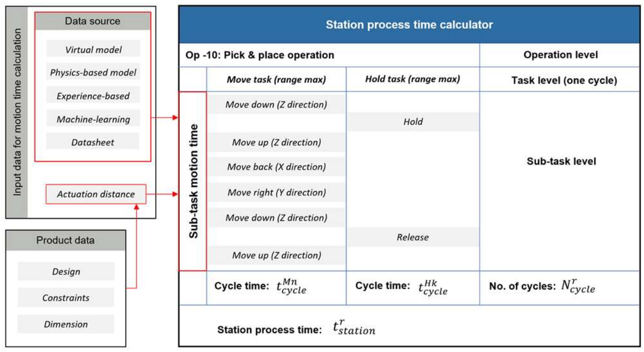

Fig. 3. Station process time calculation.

to calculate the motion time. Essentially, the product dimensions, design and constraint details can be translated to dimensional values in the Virtual Model that provides the necessary data for actuator displacement distance. The letters

$$t^{M_n}_{cycle} = \sum_{z=1}^{j} t^{z}_{motion,\,M_n} \tag{1}$$

$$t^{H_k}_{cycle} = \sum_{t=1}^{m} t^{t}_{motion,\,H_k} \tag{2}$$

'j' and 'm' represent the total number of sub-tasks for the 'move' task and 'hold' task respectively. The motion times of the sub-tasks for component M_1 is represented as *[$t^1_{motion,M1}$, $t^2_{motion,M1}$. . . $t^j_{motion,M1}$]* and the motion times of sub-tasks for component H_1 is represented as *[$t^1_{motion,H1}$, $t^2_{motion,H1}$ . . . $t^m_{motion,H1}$].* Similarly, the motion time of the sub-tasks for each of 'n' components for 'move' task and 'k' components for 'hold' task can be calculated. Following the calculation of motion time, the cycle time for the components performing 'move' task and 'hold' task can be calculated using Equations 1 and 2 respectively. To find the total cycle time, t_{cycle}, the cycle time for the 'move' task and 'hold' task should be added together. Therefore, each of the component performing 'move' task will be added with each of the component performing 'hold' task that will result in n*k cycle time values. This is then multiplied with the total number of cycles per operation, N^r_{cycle}, to obtain the station processing time, $t^r_{station}$. It is assumed that each station performs one operation and the total number of stations is represented as $N_{station}$ and 'r' is an index that represents the station number. This list of station processing times for each operation performed in the production line is stored in the common database and readily available for performing analyses in DES. Typically, in DES software, the station process time is a parameter that does not have any rules to determine whether the time is a feasible one or not. Integration with the database allows only the verified time values to be used in DES and subsequently improves the accuracy of the model. There is a choice of different station process time values stored in the database for each workstation and it provides the user the flexibility to choose process time according to certain criteria.

4. Case study

The proposed methodology is applied to a battery module assembly case. The station that is considered is the 'cell loading station', where '18650 battery cells' are picked up by a three-axis gantry with vacuum gripper and placed in a battery module. The station model is created in a virtual modelling toolset called 'VueOne' developed in the Automation Systems Group, University of Warwick. The software has two platforms that enable creation and definition of the component and station. The components such as gripper and gantry unit are the actuators that are associated with logical behavior. On the other hand, the station frame is considered as non-control component due to the absence of a logical behavior.

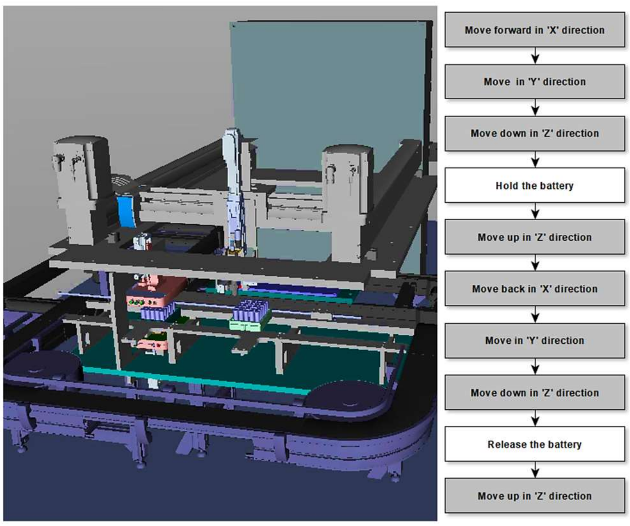

Fig. 4. Model in VueOne with process sequence.

The model that is created in VueOne and the process sequence of the sub-tasks are shown in Fig. 4. The coloured boxes represent the 'move' sub-tasks and the white boxes represent the 'hold' sub-tasks. In this example, the number of 'move' sub-tasks 'j' equals 8 and the number of 'hold' sub-tasks 'm' equals 2. The data from Virtual Model are represented in Table 2.

To demonstrate the methodology, potential components were queried from the VueOne component library to identify those components that meet the requirements in Table 2. For the 'move' task, a total of nine gantries were queried and four were found suitable. For the 'hold' task, a total of 53 grippers were queried and nine were found suitable.

4.1 Cycle time calculation

The motion time for the 'move' task is calculated for the four selected gantries. The gantries should perform 'eight' sub-tasks, the motion time of which is obtained from the gantry datasheets. A summation of the motion time results in four cycle time values. Similarly, the 'hold' task comprises of 'two'

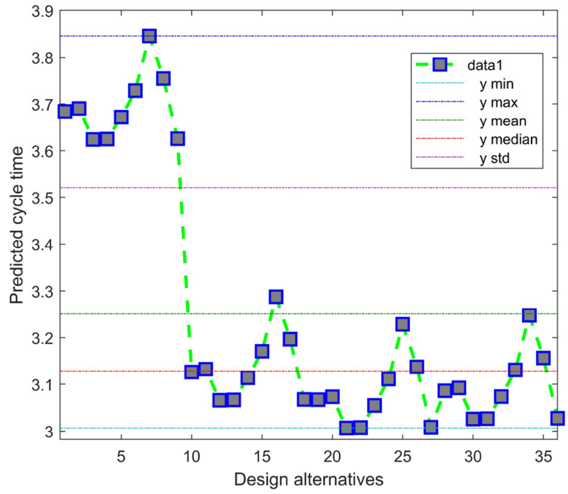

Fig. 5. Case study results.

sub-tasks, the motion times are calculated from the gripper datasheet and summed up to obtain nine cycle time values. This results in 'four' cycle time values for the gantries and 'nine' cycle time values for the grippers. The total cycle time is calculated by adding the 'move' and 'hold' cycle time values for identified components which results in a total of '36' cycle time values which are illustrated in the plot in Fig. 5. This provides decision support for choosing the best combination of components depending on the cycle time requirements. For

Table 2: Data from Virtual Model

Data type	Values		
Working range required (in mm)	X	Y	Z
	750	450	300
Workspace availability required (in mm)	2000	1500	1000
Positioning accuracy required (in mm)	0.5	0.5	1
Number of cycles	100		
Axis of motion	3		
Drive type	Electric		
Payload (in gram)	45		

example, from Fig. 5, the combinations 21, 22 and 27 have very less cycle time values and could be considered as candidates for the new workstation configuration. Since for considered case, the operation has 100 cycles, the cycle time values are multiplied by 100 to obtain the station processing time values.

4.2 Integration with DES

The resulting station process time values are stored in the common database. The line level model of the pilot line is created in DES using the commercially available tool provided by Lanner group called 'Witness'. The pilot line consists of eight workstations and the process time for seven workstations are assumed, whereas for workstation 1 which is the test case of cell loading station, the process time values are retrieved from the common data base using 'in-built' functions available in 'Witness'. Thereby, the cell loading station has more realistic process-time values that are obtained by the integration between VueOne and protégé. The station process time data can be linked with other decision supporting criteria such as cost, machine breakdown information etc. for multi-criteria decision making.

5. Future work and discussion

The proposed methodology is demonstrated for a pick and place operation, but it can be extended to other types of operations as well. Although the primary focus in this research was calculation of the cycle time of 'actuators' like gantry and grippers, the methodology is applicable for robots and digital human models. Additional work will be done to apply the proposed methodology to robotic stations and manual workstations. The methodology primarily targets improving the functionality of the existing stations by replacing the components. However, the changes in layout configuration of the workstations are not considered. The authors plan to perform further analysis in DES by incrementing the station quantity and performing layout modifications and integrating it with the workstation level analysis achieved in this paper. This will provide a holistic view of the scale-up from workstation as well as production line level. One major limitation of the approach is that the motion time values from data sources in Fig. 3, may not be accurate. Moreover, for simple processes the calculations for cycle and process time performed in this paper can be approximated to be close to the real, however, for complex processes this may not be the case. More work needs to be done in this area to enrich the data sources in Fig. 3 with better and realistic component motion time values by employing machine learning techniques.

6. Conclusion

This article presents an approach to demonstrate the integration of a virtual modelling tool with an ontology model to calculate the station process time. Additionally, the common database stores the station process time which can be accessed by the DES software as and when necessary. This essentially improves the accuracy of the DES model with more realistic time values that are significant to perform meaningful production line analysis. Therefore, the integration of workstation level model using Virtual Modelling software with a line-level model using DES software is proposed to support the upscaling process. It is envisioned that the decision-support provided by the methodology can significantly reduce the time-to-volume and ultimately result in cost and time savings.

Acknowledgements

The proposed approach is part of ongoing research studies through the project AMPLiFII 2 (Innovate UK). The primary objective of the research is to guide the scale-up of the battery module assembly facility, present at WMG, University of Warwick, by leveraging the pilot line data and integration of digital manufacturing tools.

References

[1] ElMaraghy HA. Flexible and reconfigurable manufacturing systems paradigms. Int. J. Flex. Manuf. Syst., 17(4): 261–276, 2005.
[2] Almgren H. Pilot production and manufacturing start-up: The case of Volvo S80. Int. J. Prod. Res., 38(17): 4577–4588, 2000.
[3] Liu Z, Suchold N and Diedrich C. Virtual Commissioning of Automated Systems, 2008.
[4] Colledani M, Tolio T and Yemane A. Production quality improvement during manufacturing systems ramp-up. CIRP J. Manuf. Sci. Technol., 23:197–206, 2018.
[5] Terwiesch C and Bohn RE. Learning and process improvement during production ramp-up. Int. J. Prod. Econ., 70(1):1–19, 2001.
[6] Klocke F, Stauder J, Mattfeld P and Müller J. Modeling of Manufacturing Technologies During Ramp-up. Procedia CIRP, 51:122–127, 2016.
[7] Doltsinis SC, Ratchev S and Lohse N. A framework for performance measurement during production ramp-up of assembly stations. Eur. J. Oper. Res., 229(1), 85–94, 2013.
[8] Chinnathai MK, Alkan B, Vera D and Harrison R. Pilot to Full-Scale Production: A Battery Module Assembly Case Study. Procedia CIRP,2018, vol. 2018.
[9] Jahangirian M, Eldabi T, Naseer A, Stergioulas LK and Young T. Simulation in manufacturing and business: A review. Eur. J. Oper. Res., 203(1):1–13, 2010.
[10] Vezzetti E, Violante MG and Marcolin F. A benchmarking framework for product lifecycle management (PLM) maturity models. Int. J. Adv. Manuf. Technol., 71(5): 899–918, 2014.
[11] Kuehn W. Digital factory: integration of simulation enhancing the product and production process towards operative control and optimisation. Int. J. Simul., 7(7): 27–39, 2006.
[12] Negahban A and Smith JS. Simulation for manufacturing system design and operation: Literature review and analysis. J. Manuf. Syst., 33(2): 241–261, 2014.
[13] Robinson S. Discrete-event simulation: A primer. Discrete-Event Simulation and System Dynamics for Management Decision Making, John Wiley & Sons, Ltd, 2014, pp. 10–25.
[14] Kampker A, Kreisköther K, Hehl M, Gillen S and Rothe M. Discrete event simulation approach considering scalable systems and non-expert users in the early phase of production planning for electric powertrains. International Conference on Industrial Technology and Management (ICITM), 2017, pp. 139–144.
[15] Ghani U, Monfared R and Harrison R. Integration approach to virtual-driven discrete event simulation for manufacturing systems. Int. J. Comput. Integr. Manuf., 28(8): 844–860, 2015.
[16] Neumeyer S, Exner K, Kind S, Hayka H and Stark R. Virtual Prototyping and Validation of Cpps within a New Software Framework. 2017.
[17] Soto-acosta P, Placer-maruri E and Perez-gonzalez D. A case analysis of a product lifecycle information management framework for SMEs. Int. J. Inf. Manage., 36: 240–244, 2016.
[18] Zhang J and Agyapong-Kodua K. Integrated Ontologies in Support of Factory Systems Design. IFAC-PapersOnLine, 48(3): 2095–2102, 2015.
[19] Lohse N. Towards an ontology framework for the integrated design of modular assembly systems. no. May. 2006.
[20] Chinnathai MK, Alkan B and Harrison R. Convertibility Evaluation of Automated Assembly System Designs for High Variety Production. Procedia CIRP, 2017, vol. 60.
[21] Lumsden J, Hall H and Cruickshank P. Ontology definition and construction , and epistemological adequacy for systems interoperability : A practitioner analysis. 2011.
[22] Musen MA. The protégé project: a look back and a look forward. {AI} Matters, 1(4): 4–12, 2015.