Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 81 (2019) 647–652

www.elsevier.com/locate/procedia

52nd CIRP Conference on Manufacturing Systems

Integrative simulation of information flows in manufacturing systems

Sebastian Thiede[a,*], Marc-André Filz[a], Bastian Thiede[a], Niels L. Martin[a], Jakob Zietsch[a], Christoph Herrmann[a]

[a]*Technische Universität Braunschweig, Institute of Machine Tools and Production Technology, Chair of Sustainable Manufacturing and Life Cycle Engineering, Langer Kamp 19b, 38106 Braunschweig, Germany*

* Corresponding author. Tel.: +49-531-7152; fax: +49-531-5842. *E-mail address:* s.thiede@tu-braunschweig.de

Abstract

Factories consist of different elements (e.g. machines) that are connected through material, energy and information flows. For material and energy flows, methods and tools to support their planning and operation are already available. In contrast to that, from manufacturing perspective there is a strong demand for augmenting support to consider information flows as well. This is of specific importance given the current developments in context of digitalization, with more complex and dynamic IT- architectures. Against this background, the paper presents an approach to simulate information flows (besides material/energy) in manufacturing systems which is also applied in a case study.

Keywords: Simulation; Information flow; Cyber-physical production systems, Industrie 4.0; OPC UA; MQTT

1. Introduction

Without a question, digitalization is one of the big change drivers in manufacturing. Under terms like industry 4.0, smart factory/manufacturing or industrial internet of things a diversity of technologies, methods and tools have been developed and partly already implemented [1]. In the sense of cyber physical production systems this includes innovative approaches for data gathering and processing, for data analytics combined with (simulation based) forecasting as well as for real time control integration and decision support systems [2].
Those solutions typically aim at improving the flexibility and productivity (and therewith the profitability) in manufacturing [3]. Thus, material (e.g. part processing and transport) and energy flows (e.g. energy demand from machines and technical building services) are directly affected – however, the nature of digital solutions lead to significant changes of information flows as well [4].

Fig. 1 underlines the dynamic interaction of material, energy and information flows in manufacturing systems. Information flows are necessarily involved for triggering and monitoring activities in manufacturing, thus directly influencing material and energy flows. This also includes direct connections of information and resulting energy flows, e.g. energy demand of

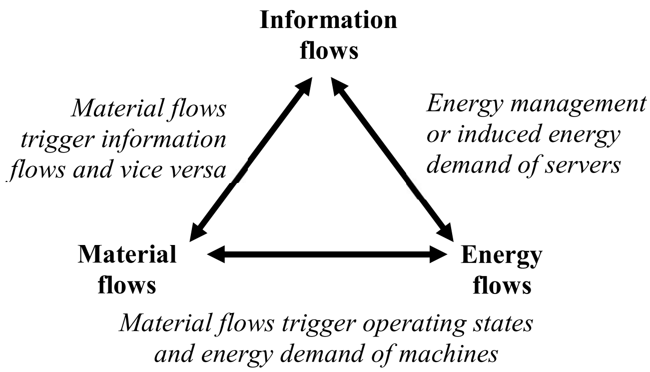

Fig. 1. Interdependencies of information, material and energy flows.

10.1016/j.procir.2019.03.170

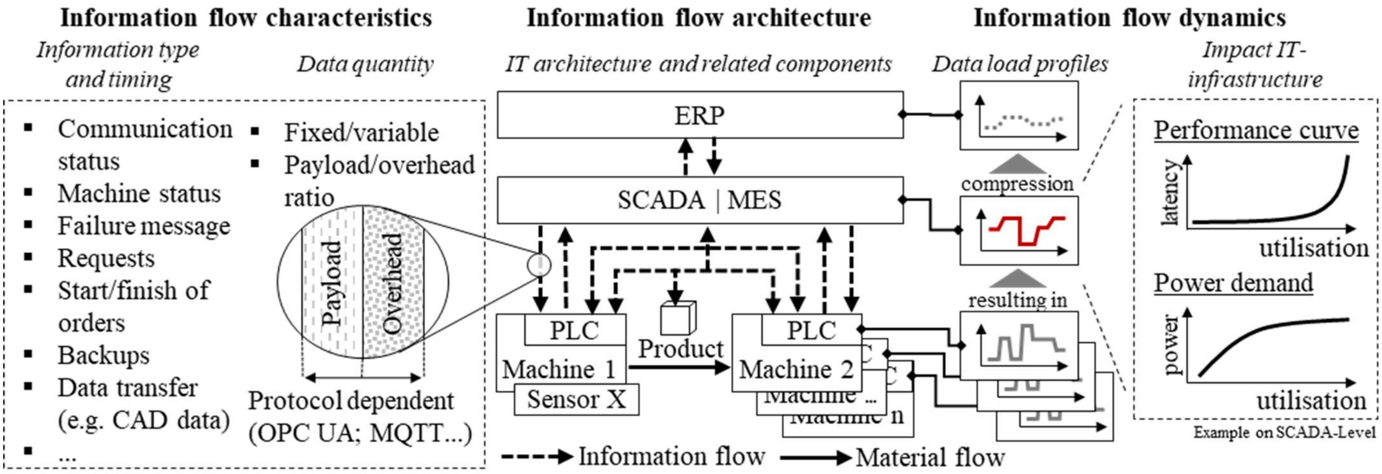

Fig. 2. Technical framework in context of information flows in manufacturing systems.

IT-infrastructure like servers or energy related demand side management. As the examples show, these connections are dynamic in nature and determined by time-dependent interactions of all manufacturing system elements. These dynamics need to be considered while planning and controlling digitalized manufacturing systems – this includes production machines and technical building services but also IT-hardware and software [5].

However, nowadays the focus in manufacturing system planning and control lies mainly on material and partly on energy flows. One reason for that is the lack of appropriate methods and tools to deal with the complexity and dynamics of information flows from production engineering perspective. Against this background, a simulation based approach is proposed in this paper. It allows an integrative consideration of material, energy and information flows in manufacturing systems.

2. Technical background

2.1. Framework

Fig. 2 shows an overview of the necessary technical background regarding information flows in manufacturing systems. The displayed *information flow architecture* follows the classical concept of the automation pyramid with an ERP system on enterprise level, SCADA/MES system on monitoring/supervisory control level which is connected to Programmable Logic Controllers (PLC) of different machines respectively processes that receive data from various sensors. Nowadays, these hierarchical architectures are not that rigid anymore; PLCs are able to share data among each other and with products following the material flow leading to more decentralized (and even more complex) information flow structures [6].

Information flow characteristics are determined based on information type, timing and the data quantity. Table 1 shows some examples for information flow types allocated to fixed or variable data volume as well as their timing characteristics. The data quantity can be subdivided in a protocol-dependent payload and overhead portion. Due to their relevance, some background on communication protocols will be given in the next section.

Table 1. Type of information flows in manufacturing systems.

Data quantity / Timing	Fixed data volume	Variable data volume
Periodic/ regular *Triggered by time steps, protocol settings*	• Status heartbeat • Sensoring data flow	• E.g. regular backup runs
Sporadic/ irregular *Triggered by events, e.g. failures, order start or finish*	• Failure codes	• Status order data after finish • Recipe parameters before start • Conventional data transfer

As indicated above, *information flow dynamics* are an important issue: single information flows of e.g. PLCs are connected to manufacturing system dynamics (e.g. operation of machines to fulfill production orders, failures) and result in (partly accumulating) information load profiles on higher system levels. This has direct impact on the utilization and therewith performance (e.g. latency) as well as energy demand [7] of IT architecture. If utilization reaches critical areas (e.g. due to wrong dimensioning), manufacturing itself might be negatively affected since operation data is not or just delayed available.

2.2. Communication protocols

In order to exchange data or information, like web sites, E-Mails, videos or machine states, computers and networks components rely on established, standardized or custom protocols. Protocols define, how two or more participants exchange information. This can include either (or both) hardware or software requirements. Each protocol focuses on specific aspects, e.g. how a message is formatted, who sends information first or the requirements for electromagnetic shielding of copper wires. In order to successfully exchange data, one or more protocols are required. Especially in internet communication, a stack of protocols is required to send to and receive data from different computer networks. This applies equally to local networks as machines and decentralized periphery are.

In order to categorize protocols by their function, different reference models have been developed. One successor is the Open Systems Interconnection (OSI) Basic Reference Model, standardized as ISO/IEC 7498-1, which defines seven protocol layers [8]. For communication purposes, these layers can be occupied by different protocols and form a stack. Each layer n serves an upper layer $n + 1$. For each layer, the transported information from upper layer is the payload. Each layer typically introduces overhead, which is required for the process of communication itself, e.g. time gap between two transmissions to avoid collisions or cyclic redundancy checks to ensure the integrity of messages. From a user or machine perspective, the information of interest is the payload. Everything else is considered as overhead. It is yet required and generates load for involved network components. In everyday life, the so called internet protocol suite is the most common one. It involves the layers one to four and includes protocols like 1000BASE-T, MAC, Ethernet, PPP, IP, and TCP.

As indicated, protocols define how data or information is exchanged. Depending on the layer, typically for layers above or equal to layer four, connections between systems are stateful. This means, connections might be established or closed. Protocols relying on authentication, like HTTPS require a prior exchange of encryption keys and/or certificates, which might even need to be checked by a third party authority. This can introduce and excessive overhead by means of transmitted data and computation time. In further discussion, a ready to transmit state is assumed. The amount of overhead introduced by non-transmitting states is considered as neglectable in contrast to per-transmission overhead due to the need for long lasting connections and number of transmissions. It should be noted, that initial peaks may lead to critical delays and cannot be neglected generally.

2.3. Protocols for communication processes in manufacturing

For process automation, many protocols have been developed. Examples for such protocols are Modbus, EtherCAT, Profibus, Profinet, AS-i. Each one of them addressing specific requirements, like low latency, low overhead, low jitter and use of or integration in existing infrastructure, explosion protection etc. In context of the proposed framework, two newer protocols, namely Message Queuing Telemetry Transport (MQTT, [9]) and OPC Unified Architecture (OPC UA, [10]) are put into focus, as two enabling protocols for Industry 4.0 and Internet of Things (IoT).

OPC UA is a standardized protocol for machine-to-machine communication, especially in context of production systems. It overcomes drawbacks from previous OPC specifications, like operating system dependency, and introduces new features like security features or heartbeats. As open standard, OPC UA fosters the interoperability of different suppliers.

MQTT on the other hand is standardized as the Internet of Things protocol by OASIS [11] for machine-to-machine communication. The specification is publically available [10] with wide acceptance. The protocol itself introduces a minimum amount of overhead (minimum of $2\,byte$ per message) and thereby claims to be especially suitable for battery driven devices. In contrast to many protocols, MQTT does not rely on a Client-Server-Architecture (like OPC UA), but on an observer pattern using a broker for message transmission.

3. Conceptual framework

3.1. Structure of simulation

In research and industry, simulation is an established method for illustrating parts of the real world over time or imitating those [12]. Within the context of manufacturing, simulation is used for the support of different tasks. It is often applied for layout design, planning, analysis and optimization of manufacturing systems [13]. Moreover, a manufacturing simulation can support the analysis of a system with a special focus on cause-effect relationships and makes the occurring behavior visible [12, 13].

Two major paradigms in simulation are static and dynamic simulation, where a static simulation is time independent while a dynamic simulation is changing over time. In the context of manufacturing, discrete event simulation is often used. Within this paradigm, passive entities can trigger actions at a discrete time point [13, 14]. In addition, decentralized modeling of a system is realized by an agent-based simulation. This approach spotlights on decentralized modelling of individual entities with different behavior [15].

Due to individual entity behavior and their direct or indirect interaction, the simulation architecture is modelled as an agent-based approach. [15]. The structure of the simulation considers two agent populations. The *product agents* are modelling the specific behavior of the products manufactured within the production system. They consist of different state charts and are characterized by different parameters, such as product identification and processing time at different machines. After completion of a production process, the agent is sent to the next process step. After finishing all process steps, the product agent moves to a storage.

Furthermore, the *machine agents* model the behavior of different machines within the manufacturing system. Each machine agent is built of a generic process flow expressing the different machine states off, ramp-up, idle and processing with a machine specific time. After a machine has been ramped-up, it switches into idle state. When a product arrives at the machine, it changes its state into processing for the manufacturing process. Moreover, every machine agent has two functions to model the different information flows described in section 2. One function occurs periodically after a defined period of time. Another function is triggered for irregular events such as “production start”, “production finish” or “failure”. Specific communication events occur between the agents in order to control the process properly. Here, focus is on aligning product and machine properties.

On the main level of the simulation architecture, all information flows are aggregated and stored. In addition, the information flows can be visualized over time and are divided in this case into payload and overhead.

3.2. Modelling of information flows

For a detailed analysis of information flows in terms of behavior and communication process, specific tools are already available, e.g. Riverbed Modeler [16] or NetSim [17]. Tools like these are able to simulate relevant aspects. This might even include time delays through computation, collisions in wired connections or hand over of mobile phones in GSM networks. However, the level of gained detail is unreasonable in context of the proposed framework. Therefore, simplifications of actual protocol behavior have been conducted. For a simplified model, following assumptions have been conducted:

- All transmissions are instantaneous. The bandwidth of transmitting medium is infinitely high. This assumption respects the comparable high data exchange cycle times.
- The computational time for all network components is neglectable. I.e, there is no time for preparation required and queried results are always immediately available.
- All resources are unlimitedly available. This assumption summarizes the previous ones and generalizes them. In situations of high loads for instance, there is no negative or disturbing effect observable or modelled.
- To compromise these assumptions, actual transmissions moments will differ from planned transmissions due to stochastic influence.

Depending on the use case, e.g. robot motion, transmission and computational times must not be neglected.

Higher level protocols, like OPC UA or MQTT, often build upon TCP. TCP itself uses IP and in case of a local Ethernet based network, the protocol stack can be summarized. Ethernet packets, or bits on wire, have a maximum size of $1538\ byte$[*]. The introduced overhead is $12\ byte$ for the Ethernet packet, $18\ byte$[†] for the Ethernet Frame, $20\ byte$ for IP and $20\ byte$ for TCP, leaving space for $1460\ byte$ of payload. If the size of the payload exceeds this limit, the message is split into multiple transmissions. TCP furthermore requires the receiver to acknowledge the transmission with a response without any payload. The simplified amount for data to be transmitted, including message and response, can be summarized to:

$$n_{TCP}(n) = 2 \cdot 78\ \text{byte} \cdot \text{ceil}\left(\frac{n}{1460\ \text{byte}}\right) + n \qquad (1)$$

where n is the amount of payload to transmit, and n_{TCP} is the total size which is actually transmitted. $78\ byte$ result from protocol overhead. Since TCP requires an acknowledgement for each transmission, the amount of overhead doubles for each required transmission of maximum $1460\ byte$.

Due to increasing importance of security and integrity, $128\ bit$ AES[‡] based TLS[§] encryption as optional feature for certain protocols [18]:

$$n_{TLS(AES128)}(n) = n_{TCP}\left(25\ \text{byte} + 16\ \text{byte} \cdot \text{ceil}\left(\frac{n}{16\ \text{byte}}\right)\right) \qquad (2)$$

For the already introduced protocol MQTT following equation is derived, which is applicable for transmission from publisher to subscribers with no additional Quality of Service. The protocol does not explicitly define the data format of the payload and must be specified by the user. This could either be single basic type variables in byte wise representation, complex data types (e.g. serialized or binary formatted objects) or another form of representation, like JSON[**] or XML[††]. In this case, only single variables of basic data types are taken into account. Furthermore, TLS encryption is used as defined by (2).

$$n_{MQTT}(\text{Topic}, \text{Variable}) = n_{TLS(AES128)}\big(2\ \text{byte} + \text{Length}(\text{Topic}) \cdot 1\ \text{byte} + \text{sizeof}(\text{Variable})\big) \qquad (3)$$

In case of OPC UA communication, multiple messages are transmitted. A setup is assumed where the PLC of machines act as OPC UA servers and the supervisory system (SCADA) queries information as client from machines, using OPC UA subscription mechanism. I.e. the client sends a message and queries updated values of the subscription and the server answers with all updates related to the subscription. Hereby, at least four TCP messages (one query from client to server with one acknowledgement and one answer from server with acknowledgement) are transmitted. Following equation was derived from own measurements with non-constant values.

$$n_{\text{OPC UA,no Security}}(\text{Variables}) = n_{TCP}\Big(86\ \text{byte} + 116\ \text{byte} + \sum_i \big(22 + \text{sizeof}(\text{Variables}_i)\big)\Big) \qquad (4)$$

OPC UA supports multiple forms of security and authentication. For instance, in case of using OPC UA security policy Basic128Rsa15 with signed and encrypted messages following:

$$n_{\text{OPC UA,signed and encrypted}}(\text{Variables}) = n_{TCP}\Big(116\ \text{byte} + 182\ \text{byte} + \sum_i \big(28 + \text{sizeof}(\text{Variables}_i)\big)\Big) \qquad (5)$$

The definitions (2-4) inherently show the protocol stack which is used for transmission. TLS utilizes TCP. Consequentially, the data transmit (n and the TLS overhead) are the payload of TCP (compare (2)). This can be tracked down to higher levels as well.

3.3. Visualization of information flows in Sankey diagrams

Compared to energy or material, information is an even more abstract measure which is difficult to grasp without a proper interface. The choice of visualization depends substantially on the recipient and the type of information to be presented. In the context of manufacturing systems, one way of

[*] Assumption: Network elements do not limit frame size; no VLAN-Tagging
[†] Jumbo packets are not considered.
[‡] Advanced Encryption Standard
[§] Transport Layer Security
[**] JavaScript Object Notation
[††] Extensible Markup Language

representing flows is a Sankey diagram. First introduced to analyze the thermal efficiency of steam engines, it found its way to network or lifecycle engineering and was extended amongst other to material, energy, people and value flow [19]. Reducing complexity while showing relations and interactions in processes and networks as well as being able to provide quantitative and qualitative information with a customizable level of detail established the Sankey diagram as a preferred form of visualization for industrial engineers. Its standardized form is facilitating the comprehension of the connection between different types of flows.

4. Case study

4.1. Description of use case

The concept of the integrative simulation of information flows is oriented towards electronics manufacturing and therefore has a realistic (fixed) cycle time of 20 seconds per part. The defined use case is exemplary applied to a production system of sixteen interlinked processes. After each sequence of four processes, the products go into a buffer. After finishing all process steps, the products are stored.

The simulation model of the use case is realized within the simulation software AnyLogic®, which is a hybrid simulation environment to combine different simulation paradigms such as discrete-event, dynamic systems and agent based simulation. Before running the simulation, all necessary parameters are read from an Excel® spreadsheet.

In order to represent time and event based information flows in manufacturing systems, all machine communicate cyclically with a SCADA or/and MES. For this use case, a normal distributed cycle period of one second is defined. Moreover, an event based communication occurs in case of “production start”, “production finish” or “machine failure”.

Within this paper, five different cases (shown in Table 2) are simulated differentiated by cycle time, amount of periodic and sporadic triggers as well as protocol type (OPC UA or MQTT), whereas “Hybrid” is an equal divided and random combination of MQTT and OPC UA protocols for the process steps. The manufacturing system in *case 4* represents a big data approach (more sensors per machine) and *case 5* a change in production outcome (change of production parameter).

Table 2. Overview of selected simulation cases.

Case	Protocol	Cycle time	Periodic triggers	Sporadic triggers
1	OPC UA	20 s	20	10
2	MQTT	20 s	20	10
3	Hybrid	20 s	20	10
4	Hybrid	20 s	40	10
5	Hybrid	30 s	20	10

4.2. Results

During the simulation, all information flows were determined for every machine and the total information flow was aggregated over all machines. Fig. 3 represents the total information flow for the simulation runs of *case 1 and 3* over time. The figure intuitively reveals the dynamic behavior of information flows as well as the significant difference of case 1 and case 3 in terms of absolute values and volatility. In order to analyze the five cases more quantitatively, different assessment criteria are used:

- The *average value* shows the arithmetic mean of the total information flow which is a good indicator for the regular order of magnitude of data rates.
- Since information flows are not constant, the *standard deviation* helps to assess the volatility of the considered information flow.

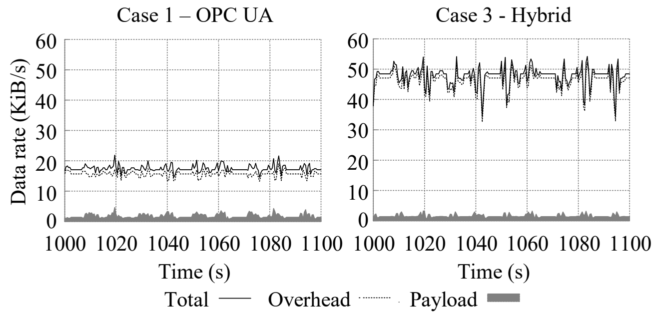

Fig. 3. (a) Total information flow with OPC UA protocol; (b) Total information flow with Hybrid protocol.

- *Maximum (peak) values* indicate critical areas of utilization (e.g. through aggregation of single information flows) which may serve as important value for IT infrastructure dimensioning
- The *payload/overhead ratio* can serve as indicator for the efficiency of data transmission while it shows the proportion of the value adding information (from production perspective) within the total information flow.

For each case, a simulation was run over a period of 3600 seconds. To get a more detailed overview of the simulation outcome, the results for *case 1 to 5* are shown in Table 3.

Table 3. Results of simulation for cases 1 to 5.

Case	Peak value [KiB/s]	Average value [KiB/s]	Standard deviation [KiB/s]	Payload/ overhead ratio
1	22.78	17.45	0.97	11.83%
2	87.36	72.75	3.97	1.89 %
3	57.87	47.76	2.99	3.52 %
4	91.91	77.55	4.47	3.49 %
5	57.79	48.04	2.49	3.32 %

Regarding the peak of total information flow, *case 4* shows the highest peak of 91.91 KiB/s while *case 1* has the lowest with 22.78 KiB/s. Comparable to this, the average information flow shows a similar behavior for *case 2 and 4*. Over all cases it should be considered, that average total information flows vary between 17.45 and 77.55 KiB/s. Coming from *case 1*, the average information flow for *case 4* is almost five times bigger. The payload/ overhead ratio specifies the payload share of the total information flow for *case 1* of 11.83%. For case 2 – 4, the share is remarkably lower. To gain even more insight into the characteristics of information flows, Fig. 4 shows an exemplary Sankey diagram of information flows. It underlines the (protocol-dependent) high relevance of overhead shares in information flows. Based on the simulation results, the specific type and form of the Sankey flow representation (e.g. number of packages, amount of content, or events per second) can be adapted according to the analysis objectives.

Besides these dedicated indicators for dynamically assessing the information flows in manufacturing systems, material and energy flows were simulated as well. Please note that *cases 1-4* all result in the same production output whereas for *case 5* production related parameters were changed which directly influenced all types of flows. As stated earlier, the impact of information flows in terms of disturbing production due to high latencies or energy demand of servers has been neglected in this first simulation studies.

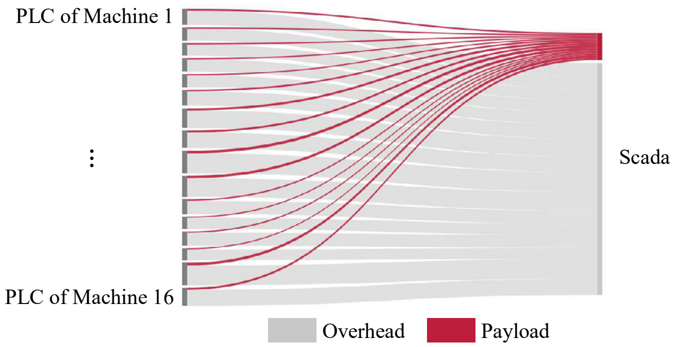

Fig. 4. Resulting Sankey diagram of all information flows of case 1.

5. Discussion and outlook

The first studies based on the integrative simulation approach underline the necessity to consider information flows next to material and energy flows for planning and control of manufacturing system. Information flows can be highly volatile and are strongly influenced by data gathering strategies, configuration of IT hardware and protocols but also the operation of the manufacturing system itself. Knowing the dynamic behavior of information flows provides valuable decision support to set up efficient and reliable IT infrastructure but also enables production engineers to more comprehensively foresee the impact of their decisions, e.g. when configuring smart manufacturing systems.

It needs to be noted, that the presented case study itself is of course relatively simple, small and the configuration of cases quite selective (e.g. comparison of OPC UA or MQTT is not necessarily fair - it could be also implemented in different ways and individually optimized for the protocols). It should just serve as example to illustrate functionalities and potential takeaways from the simulation approach itself. However, reality is likely to be more complex in terms of interacting systems elements and flows and even more calls for those advanced simulation approaches. Further work will focus on more and more detailed modelling of protocols and also the integration of interactions between IT component utilization and performance as well as power demand. In addition, more specific case studies should be conducted which allow the comprehensive assessment of the impact of smart manufacturing solutions.

Acknowledgements

This paper evolved of the research project “Synus” (Methods and Tools for the synergetic conception and evaluation of Industry 4.0-solutions) which is funded by the European Regional Development Fund (EFRE | ZW 6-85012454) and managed by the Project Management Agency NBank.

References

[1] Kang HS *et al.* “Smart manufacturing: Past research, present findings, and future directions,” *Int. J. Precis. Eng. Manuf. Technol.*, vol. 3, no. 1, pp. 111–128, Jan. 2016.

[2] Thiede S. “Environmental Sustainability of Cyber Physical Production Systems,” *Procedia CIRP*, vol. 69, pp. 644–649, 2018.

[3] Kagermann H, Helbig J, Hellinger A and Wahlster W. “Recom-mendations for implementing the strategic initiative INDUSTRIE 4.0; final report of the Industrie 4.0 Working Group,” 2013.

[4] Matt C, Hess T and Benlian A. “Digital Transformation Strategies,” *Bus Inf Syst Eng*, vol. 57, no. 5, pp. 339–343, 2015.

[5] Herrmann C, Schmidt C, Kurle D, Blume S and Thiede S. “Sustainability in manufacturing and factories of the future,” *Int. J. Precis. Eng. Manuf. Technol.*, vol. 1, no. 4, pp. 283–292, Oct. 2014.

[6] Vogel-Heuser B, Bauernhansl T and Hompel M ten. *Handbuch Industrie 4.0 Bd.4*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017.

[7] Barroso LA, Clidaras J and Hölzle U. “The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines, Second edition,” *Synth. Lect. Comput. Archit.*, vol. 8, no. 3, pp. 1–154, Jul. 2013.

[8] ISO/IEC 7498-1:1994. “Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model. International Standard.” International Organization for Standardization, Geneva, Switzerland, 1996.

[9] MQTT. “https://www.mqtt.org,” *Official web presence*. .

[10] OASIS, “https://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html,” *MQTT Version 3.1.1*, 2014. .

[11] OASIS. “https://www.oasis-open.org,” *Open Source*. .

[12] Banks J, Carson JS, Nelson BL and Nicol DM. *Discrete-event system simulation*. Upper Saddle River, NJ, 2010.

[13] Negahban A and Smith JS. “Simulation for manufacturing system design and operation : Literature review and analysis,” *J. Manuf. Syst.*, vol. 33, no. 2, pp. 241–261, 2014.

[14] Mourtzis D, Doukas M and Bernidaki D. “Simulation in Manufacturing : Review and Challenges,” *Procedia CIRP*, vol. 25, no. Figure 1, pp. 213–229, 2014.

[15] Borshchev A and Filippov A. “From System Dynamics and Discrete Event to Practical Agent Based Modeling: Reasons, Techniques, Tools,” in *Proceedings of the 22nd International Conference of the System Dynamics Society*, 2004, vol. 66, no. 11.

[16] Riverbed Technology. “Riverbed Modeler.” San Francisco, USA.

[17] Tetcos. “NetSim Pro.” Bangalore, India.

[18] RFC Editor. “RFC 8446. The Transport Layer Security (TLS) Protocol Version 1.3. Request for Comments.” 2018.

[19] Schmidt M. “The Sankey Diagram in Energy and Material Flow Management Part I : History,” vol. 12, no. 1, 2008.