Industry 4.0 and real-time synchronization of operation and maintenance

J. Vatn
Department of Mechanical and Industrial Engineering
NTNU - Norwegian University of Science and Technology, Norway

In Safety and Reliability - Safe Societies in a Changing World.
Taylor & Francis 2018 ISBN 978-1-351-17466-4. pp 681-686

ABSTRACT: Industry 4.0 represents a trend in manufacturing which includes cyber-physical systems, the Internet of things, cloud computing and cognitive computing. Cyber-physical systems (CPS) refers to smart systems that include engineered interacting networks of physical and computational components. The term digital twin refers to a digital replica of physical assets, processes and systems that can be used in real time for control and decision purposes. The digital twin representation is seen as a prerequisite for effective synchronization of operation and maintenance within the manufacturing industry as well as in other industries. The relation between production plans and activities and actual production can to some extent be described by deterministic. The relation between maintenance plans and activities and the production system availability on the other and requires probabilistic representation. The term stochastic digital twin is therefore introduced. An ambition of Industry 4.0 is to support real-time processing whenever possible. This paper discusses elements of Industry 4.0. A case study is provided to demonstrate these terms and challenges to the mathematical modelling required for optimal synchronization of operation and maintenance.

1 INTRODUCTION

1.1 *Background*

Nowadays Industry 4.0 and digitalization are frequently used terms for the changes that are taking place in industry, civil engineering, transportation, public services and so on. The "4.0" refers to the forth industrial revolution and points to the opportunities communication over the internet gives with respect to real-time control of processes at almost every level. Industry 4.0 and related concepts as cyber-physical systems, internet of things, cloud computing and digital twins give new opportunities for both production and maintenance, but even more important the synchronization and coordination of the two.

A huge number of papers have been published in recent years on Industry 4.0 and cyber-physical systems. Very many of these papers present conceptual descriptions and frameworks, for example (Rojas, Rauch, Vidoni, & Mat 2017, Kacem, Simeu-Abazi, Gascard, Lemasson, & Maisonnasse 2017, Kim & Park 2017). Few papers addressing the modelling aspect of synchronization of production and maintenance in an Industry 4.0 have not been found. Some relevant papers are (Hehenberger, Vogel-Heuser, Bradley, Eynard, Tomiyama, & Achiche 2016, Upasani, Bakshi, Pandhare, & Lad 2017, Cheng, Zhou, & Li 2017).

1.2 *Objective*

The objective of this paper is to clarify basic terms and elaborate on basic elements of Industry 4.0 in relation to real-time synchronization of operation and maintenance. A case study from the railway sector is used to exemplify the concepts.

2 DEFINITIONS AND CONCEPTS

Industry 4.0 is a collective term particularly used in manufacturing to emphasize technologies and concepts of value chain organizations. Further the terms Cyber-Physical Systems, the Internet of Things, Cloud computing and the Digital Twin are often used in relation to Industry 4.0. Although the term originates from the manufacturing industry, the elements of Industry 4.0 are relevant for most businesses.

The current usage of the term Industry 4.0 has been criticized as essentially meaningless. The 4.0 points to the forth industrial revolution under a premise that digitalization is the really new thing. But why digitalization and not Nano technology? Further, the content of Industry 4.0 also seems to vary from industry to industry, and from author to author. From a scientific point of view it might therefore be better to avoid a precise definition but rather focus on Industry 4.0 *elements*.

The aim of this paper is to shed light on Industry 4.0 elements that are relevant for the interaction between production and maintenance. Here production has a very broad meaning, it could cover manufacturing, logistics, transportation systems, hospitals, power supply and so on.

The *Internet of Things* (IoT) is the network of physical devices, production facilities, cars, air-planes and in general items embedded with electronics, software, sensors, actuators, and network connectivity which enable these objects to connect and exchange data. Each "thing" is able to inter-operate within the existing Internet infrastructure.

The IoT allows objects to be sensed or controlled remotely across existing network infrastructure, creating opportunities for more direct integration of the physical world into computer-based systems. When IoT is augmented with sensors and actuators, the technology becomes an instance of the more general class of *Cyber-Physical Systems* (CPS).

Cyber-physical systems (CPS) refers to smart systems that include engineered interacting networks of physical and computational components.

Cloud computing is an information technology paradigm that enables access to shared pools of configurable system resources. The companies can focus on their core businesses instead of expending resources on computer infrastructure and maintenance. Downsides of such a strategy could be unexpected operating expenses if administrators are not familiarized with cloud-pricing models and vulnerabilities and security issues. In some presentations the term Internet of Services (IoS) is used rather than cloud computing.

The term *digital twin* refers to a digital replica of physical assets, processes and systems that can be used in real-time for control and decision purposes. The digital twin representation is seen as a prerequisite for effective synchronization of operation and maintenance within the manufacturing industry as well as in other industries. The relation between production plans and activities and actual production can to some extent be described by deterministic models. The relations between maintenance plans and activities and the production system availability on the other and require probabilistic representations.

A *stochastic digital twin* is a computerized model of the stochastic behaviour of a system where the model is updated in real time based on sensor information and other information accessed via the internet and the use of cloud computing resources.

The digital twin concept has several implications in the development of methods for synchronization of operation and maintenance. A detailed digital representation of the physical asset, from the single device to the complex interaction of the components in the value chain, is a first basic requirement for the digital twin. The second requirement is a real-time two-way communication from the physical asset to the digital twin, enabled by the industrial IoT where sensors continuously upload data related to the current state of operation into the cloud, paired with the capability of remote plant control. Finally, the digital twin and the real-time two-way communication between the twin and the plant is expected to achieve the maximum potential benefit from the digital transformation, if also machine-learning methodologies are implemented in this framework in order to exploit the information in historical data and the current state, together with predictive simulation capabilities.

To be useful a digital twin needs "what-if" capabilities. This means that the decision makers, i.e., humans or computers, shall be able to "ask" the digital twin what will be the consequences of various decisions. For a stochastic digital twin this means that the "answer" is given as a set of probability statements.

A *real-time model* is a model where it is possible to obtain values of system performance and system states in real-time. With real-time we mean that data referring to a system is analysed and updated at the rate at which it is received. As for the digital twin a real-time model typically connects to the "real world" via the IoT, although other means of communication is also possible. A real-time model is also referred to as an on-line model.

A *test model* is a mathematical model describing relations between future and current values of the variables of interest, but where we are not able to monitor system performance and system sates in real-time. Such a model is often referred to as an off-line model. A test model is still valid in order to establish decision rules to be used in real-time.

Most methods and models used in production planning and optimization as well as in maintenance planning and optimization are off-line models. These models can be used for establishing optimal strategies, but they can not give real-time decision support. A real-time model is often used to describe a limited part of a system, whereas a digital twin aims at giving a complete digitalized representation of the system and decision processes.

A *real-time decision support systems* is a system where relevant data is collected and processed into relevant information in real time. This means that the raw data stream is automatically collected and processed into information. Information is further interpreted in such a way that it gives meaningful decision support.

A *real-time execution system* is a system which implement algorithms to determine optimal decisions at time t, and then execute these decisions. An example of a real-time execution system is an automated replenishment program (ARP). The aim is to provide automated replenishment of products based on real-time demand information to the production, warehouses and distribution processes in the supply chain. This corresponds to real-time control in control theory. Similarly for maintenance a real-time execution system will automatically issue a work order with task descriptions and due date.

Predictive maintenance builds on the idea to utilize the condition of a component and the future expected loads in order to judge the correct time for "hard" maintenance such as overhaul, replacement of worn parts, calibration and so on. Sensor technology is usually used to capture the condition of components or a system, and the term 'condition monitoring' is often used to describe the collection and analysis of state data relevant for predictive maintenance. It should be noted that manual inspection and use of "human sensors" to capture noise, smell, vibration could also be treated as condition monitoring.

3 THE DIGITAL TWINS

This section presents principal elements of the digital twins for maintenance and production.

3.1 *Maintenance*

To a large extent the computerized maintenance management system (CMMS) could be seen as a digital twin for maintenance. Principal content found in the CMMS are the asset register covering all components, the preventive maintenance (PM) program covering the type of maintenance and the plan for maintenance. The CMMS will also contain required spare parts, resources and tools for conducting maintenance and so on. But there is relevant information not found in the CMMS which is essential for the stochastic digital twin to be developed. First of all the CMMS has no inherent mathematical models to be used for degradation development and time to failure. Further information regarding component condition is often not part of the CMMS, and needs to be obtained from stand alone systems operated in parallel to the CMMS. Further the CMMS is not connected to the supervisory control and data acquisition (SCADA) system and other systems giving information regarding process parameters and future loads from production and the environment.

It is beyond the scope of this presentation to write out the details regarding the content of a stochastic digital twin for maintenance. For illustrative purposes and for use in the case study presented later a very simple digital twin is presented in the following. Although we in many situations can do much better, the classical failure rate function is used as a basis. The situation relates to so-called delay time models (Christer 1987), often referred to as PF-interval models. The situation is as follows: A component is put into service at time $t = 0$. Then after a random time T_{P} the component enters a degraded state. This state is often referred to as a potential failure. It is assumed that a condition monitoring activity can reveal such a potential failure with some detection probability, say $1 - q$. If no action is taken the component will fail after another random time T_{PF}. A Cox proportional hazard rate function (Cox 1972) is used as a basis for formulating the failure rate function, $z(t) = f(t)/R(t)$ for T_{PF} (t is running time after the potential failure has occurred). T_{PF} is often referred to as the PF-interval, and the corresponding failure rate function is:

$$z(t|\mathbf{y}, \overline{\mathbf{x}(t)}) = z_0(t)e^{\boldsymbol{\beta}_1\mathbf{y}}e^{\boldsymbol{\beta}_2\overline{\mathbf{x}(t)}} \tag{1}$$

where $z_0(t)$ is a baseline failure rate function, typically on the form $z_0(t) = \alpha\lambda^\alpha t^{\alpha-1}$ in the Weibull case. $\mathbf{y}$ is a vector of state variables at the point of time of the potential failure is observed, and $\overline{\mathbf{x}(t)}$ is the average load profile t time units ahead. $\boldsymbol{\beta}_y$ and $\boldsymbol{\beta}_x$ are regression coefficient vectors established by for example statistical analysis of data.

The failure rate function in eq. (1) is a classical model and it could be questioned whether this model represent at digital twin. A prerequisite for being at least a part of a digital twin is that $\mathbf{y}$ could be accessed from sensor readings and communicated via the IoT. Further $\overline{\mathbf{x}(t)}$ needs to be accessed in real time from enterprise resource planning (ERP) systems and other system for future production plans.

If eq. (1) is part of the stochastic digital twin we may now "ask" for the probability of failure if we wait for example t time units before the potential failure is fixed:

$$F(t|\mathbf{y}, \overline{\mathbf{x}(t)}) = 1 - e^{-\int_0^t z(u|\mathbf{y}, \overline{\mathbf{x}(u)})\, du} \tag{2}$$

Only a few aspect of the "maintenance twin" are elaborated here. For real applications it will be an enormous amount of work to structure the raw data, information, knowledge, models and so on to have a digitalized stochastic maintenance twin.

3.2 *Production*

There are so many aspects to deal with when it comes to production and logistic optimization that we will not even make an attempt to cover these in this presentation. However, with respect to maintenance there are some important aspects that we will emphasize when setting up the digital twin for production.

3.2.1 *Objective function*

Operations research (OR) is the systematic approach to optimize production under various constraints (Phillips, Ravindran, & Solberg 1976). The objective function, Z, is typically the quantity to maximize or minimize with respect to some vector of decision variables, say $\mathbf{x}$, i.e., $Z = Z(\mathbf{x})$.

3.3 *Constraints and conditions*

Usually there are constraints to take into account in the optimization, for example a set of functions, say $g_j(\mathbf{x})$ should all be positive. In addition to these constraints we also have to optimize $Z = Z(\mathbf{x})$ subject to $\mathbf{S}$, where $\mathbf{S} = [s_1, s_2, \ldots, s_n]$ is the state vector of the components in the system. For example $s_i = 1$ could represent that component i is functioning, and $s_i = 1$ represents a fault state.

It should be emphasized that both the objective function and the constraints and conditions are changing all the time. It is therefore required to have real-time access via the IoT to the "physical" plant, existing orders, inventory levels and so on.

3.4 *Maintenance interaction*

The digital twin for production will also be a *stochastic* twin due to the probabilistic nature of production optimization. From classical OR examples variability in supply and demand are the main sources for uncertainty. However, we will focus on the relation to maintenance. Important aspects that need to be structured as part of the digital twin for production are:

- Slots for maintenance, i.e., possible opportunities for doing maintenance
- Specifications of how utilizing possible slots will affect the objective function $Z = Z(\mathbf{x})$ and the constraints $g_j(\mathbf{x})$
- Specification of possible "relaxes" in production, for example avoid running a component with full load if a "potential failure" has been revealed
- Specifications of how such "relaxes" will affect the objective function $Z = Z(\mathbf{x})$ and the constraints $g_j(\mathbf{x})$.

Note that the objective function $Z = Z(\mathbf{x})$ in traditional OR does not include maintenance. Since the objective of this paper is to investigate synchronization and coordination of activities in the production and maintenance departments, the objective function should cover both departments.

4 CASE STUDY

4.1 *Introduction*

A railway example is used to demonstrate challenges in synchronization and coordination of activities in the production and maintenance departments. Only few aspects are dealt with, and issues related to really establish the stochastic digital twins and have them to play together is not addressed in this presentation. One aspect of "digitalization" within maintenance is related to increased used of predictive maintenance. Turnouts (switches) are important components in the railway infrastructure, and failure of a turnout will usually give large problems with the circulation, and delays are expected. Although various condition monitoring techniques exist for turnouts they have not been implemented on the Norwegian railway network due to high cost. In Norway Bane NOR is a state-owned company responsible for the Norwegian national railway infrastructure. In recent years Bane NOR has been running a test project on a simplified predictive maintenance strategy for turnouts based on measuring only power as function of time when the traction motor is activated to change the position of the turnout. The time required for changing the position of the turnout varies from one to up to 20 seconds. The idea is that the power as function of time for each individual turnout is a "signature" for that turnout, and deviation from that signature could be seen as a potential failure as discussed in Section 3.1. The main advantage of this system is that the information is available more or less "free of charge". The challenge is to use it in an efficient way.

The system has been piloted over a period of almost 3 years. As part of the pilot project data have been analysed for one of the turnouts. In a follow up project it is planned to conduct more comprehensive analyses. During the test period 11 failures were observed. For 3 of these failures a potential failure was not observed at all. Thus the reliability of the condition monitoring system is only some 70%. The analysis was conducted by visual analysis of the power/time curve for all movements of the turnout for a period of 10 days prior to the failure. More comprehensive analysis could obvious give a higher reliability.

4.2 *The PF-interval model*

The average PF-interval, i.e., the estimate of $\mathrm{E}(T_{\mathrm{PF}})$ were found to be 80 hours. However, 2 failures had an observed PF-interval of less than 3 hours. To estimate the parameters in the failure rate function in eq. (1) assuming a Weibull distribution and ignoring covariates $\mathbf{y}$ and $\overline{\mathbf{x}(t)}$ we may use the following procedure:

1. Let x be such that $F_T(x) = p_x$ where both x and p_x are known. It can be shown that the following iterative scheme may be used to estimate α: $\alpha_{i+1} = \frac{\ln(-\ln(1-p_x))}{\ln(x\Gamma(1+1/\alpha_i)/\mathrm{E}(T_{\mathrm{PF}}))}$
2. For the location parameter we use set: $\lambda = \Gamma(1 + 1/\alpha)/\mathrm{E}(T_{\mathrm{PF}})$.

In the example we had $p_x = 2/(11-3)$ and $x = 3$ hours, and $\mathrm{E}(T_{\mathrm{PF}}) = 80$ hours. Applying the procedure this will give $\hat{\alpha} \approx 0.49$ and $\hat{\lambda} \approx 0.026$. It should be noted that $\alpha < 1$ means that the PF-interval is not very consistent. The reason for the low value of α is that we are mixing several failure mechanisms. There are three main failure mechanisms with quite different characteristics here, i.e., snow and ice with short

PF-interval, lack of lubrication with a medium PF-interval, and mechanical failure with a rather long PF-interval. This means that we need to apply the procedure above for each separate failure mechanisms. From the failure statistics obtained from the pilot project we do not have sufficient number of observations to apply the procedure above. For the case study we will proceed with assuming that the failure mechanism is related to lack of lubrication and without any statistical support we set $\hat{\alpha} = 2$ and $\hat{\lambda} = 0.0246$ corresponding to $\mathrm{E}(T_{\mathrm{PF}}) = 36$ hours.

4.3 *The initial cost model*

The operational hindrance cost of executing a "hard maintenance" task, and the cost of a failure depends on the position of the turnout, the time of the day, the traffic and so on. Therefore an example situation is presented in the following.

The location of the turnout is assumed to be on a part of the line where access only can be made by means of a work train. We assume a single track line where access by the work train will disturb the circulation. Investigating the time table for today four opportunity windows have been identified. They are shown in column 1 in Table 1 where t is the beginning of the time slots. The first three of these time slots will, however, cause delays in circulation. The expected delay minutes for each window is shown in column 2 in Table 1. In average there are 150 passengers per train and a minute delay cost per passenger of 3 NOKs is used by Bane NOR. In addition to the delay cost there is a fixed cost of NOK 5 000 for ordering the work train and associated personnel cost. If the failure can not be "caught" in due time, the expected total delay is 3 hours. The cost equation to minimize is:

$$C(t) = c_{\mathrm{PM}}(t) + c_{\mathrm{U}}F(t) \tag{3}$$

where $c_{\mathrm{U}} = 3 \cdot 150 \cdot 60 \cdot 3 = 81\ 000$ NOKs, $F(t)$ is given by eq. (2). Table 1 shows that in this situation one should utilize the last maintenance window since the circulation is not affected, and the probability of failure is still rather low. It can be shown that if

Table 1: Optimization results

t (hours)	Delay (min)	c_{PM}	c_{F}	c_{Tot}
3	30	18 500	441	18 941
5	15	11 750	1 218	12 968
7	10	9 500	2 370	11 870
9	0	5 000	3 880	8 880

the failure mechanism is ice and snow, and assuming $\mathrm{E}(T_{\mathrm{PF}}) = 10$ hours and $\hat{\alpha} = 2$ the risk is much higher, and one should rather use the first opportunity. Note that the optimization here is seen from the maintenance department, i.e., the only "production" related cost is the increased c_{PM}-cost by rushing the maintenance.

4.4 *The refined cost model*

So far the covariates $\mathbf{y}$ and $\overline{\mathbf{x}(t)}$ have been ignored. We now introduce two variables, y which is a measure of degradation at the point of detection of the potential failure, and x as the number of times per hour the turnout will be operated. The proposed Cox proportional hazard model reads:

$$z(t|y, x(t)) = z_0(t)e^{\beta_y y + \beta_x x t} \tag{4}$$

For simplicity we have assumed that the number of train passages per hour is constant over the day. From the case study we do not have sufficient data to estimate β_y and β_x. We will therefore proceed with illustrative values for these parameters. That is, for the example we proceed with $\beta_y = \ln 2 \approx 0.69$ and $\beta_x = 0.1 \ln 2 = 0.069$

Now, assume that at the time of the potential failure we assess $y = 0.15$ by analysing the power vs time curve from the condition monitoring system, and further from the time table we wind $x = 2$. Table 2 shows the result when the covariates are taken into account. Compared to the original situation we have to advance the point of time for doing hard maintenance, i.e., lubrication and required adjustments. The number of times per hour we operate the turnout, x, is a decision variable seen from operation. Since the failure rate function is increasing with increasing value of x, we should investigate whether it pays off to reduce x. Now, assume that we can completely remove the need for operating the turnout by changing the station where trains are crossing. This corresponds to set $x = 0$ in the model. Rerunning the model shows that the optimal value of t is $t = 9$. The cost has been reduced from 15 172 to 12 249, i.e., a total saving of $\approx$ NOK 3 000. However, if this causes total delays of more than 7 minutes the delay cost will be higher than the savings. For the railway case it seams unrealistic that changing the crossings for the actual station for an entire day will not cause more than 7 minutes of total delay.

Table 2: Optimization results - with covariates

t (hours)	Delay (min)	c_{PM}	c_{F}	c_{Tot}
3	30	18 500	822	19 322
5	15	11 750	3 422	15 172
7	10	9 500	9 812	19 312
9	0	5 000	22 562	27 562

A first attempt to formalize such a "relax" strategy is to add an extra cost term in the objective function, $c_{\mathrm{R}}(x)$:

$$C(t, x) = c_{\mathrm{PM}}(t) + c_{\mathrm{U}}F(t|y, x) + c_{\mathrm{R}}(x) \tag{5}$$

The joint optimization of t and x is not pursued further in this presentation.

5 DISCUSSION

The objective of this paper has been to investigate "Industry 4.0 solutions" to facilitate synchronization and coordination of operation and maintenance. By a "paper exercise" it is rather easy to demonstrate how this can be done, and potential savings. This section discusses challenges when such ideas are to be implemented for real systems.

5.1 *Slots for maintenance and consequences for the production model*

In order to synchronize and coordinate production and maintenance it is essential that the digital twin on request can provide time slots for maintenance and evaluate the production consequences for each possible slots. In the example we assumed that "some" could establish the time slots at 3,5,7 and 9 hours. Here, "some" could be a train manager at the train control centre (TCC). But this is not really a part of the "digital twin" for production. To develop a digital twin all production plans, cost optimization functions etc. need to be implemented in a computerized system supported with algorithms to both find possible slots, and do calculations to evaluate the consequences. For the railway example we are far from realizing such systems. To the author's knowledge the situation is the same in most Norwegian industries.

The way forward is therefore to develop simplified production models. For example in Norway a simplified circulation model for use by the TCC-personnel upon traffic deviations to assist planning has been developed. The model acts like a "what-if" tool that can simulate the consequences if crossing is moved to station A rather than on the scheduled station B. It is hard to spot significant achievements here unless modelling competence within the companies is significantly increased. A vision behind "cloud computing" is that ready to use models could just be plugged in whenever needed. But still this is a vision.

5.2 *Specification of possible "relaxes" in production*

In the example, and in many real case situations a mitigating measure upon a component degradation is to reduce the load on that component to increase residual life. An even more realistic example than the railway example is maintenance and operations of wind farms. A wind farm can be difficult to access in periods of the year due to harsh weather conditions. Upon a potential failure of for example the main bearing of the turbine it may be better to close down the turbine in situations with high wind loads. Although this will reduce the power produced for some hours, it might prevent a failure which would have made the turbine unavailable for weeks and even months.

The digital twin for production therefore need to respond upon request on what are the possible "relaxes" that could be made in production that will have a positive impact on residual life of a component. In addition to respond on *what* can be done, the digital twin also needs to specify the consequences, for example by quantifying the reduced production.

5.3 *Maintenance models*

The literature in the field of maintenance optimisation produces every year a huge number of models. Very few of these models are used in practice. One reason for this is that it is hard to get access to statistical data for estimation of model parameters. In our example we need to estimate α, λ, β_{Y} and β_{X}. Further this have to be done for all failure mechanisms. We can easily imagine an enormous workload. Next, comes the question whether the Cox-proportional hazard model is the appropriate model to use. It is rather simple, but it does not really take into account the physical aspects of the phenomenon causing a failure.

The prospects for the maintenance twin is therefore also not that promising. Again, starting with a set of rather simplified models seems a natural first step.

5.4 *Machine-learning*

Machine learning is a field of computer science that gives computers the ability to learn without being explicitly programmed. Machine-learning is quite different from the model based approach advocated here. A strength of machine-learning is it's efficiency to produce huge amount of results without the explicit need to do all the "hard work". From a model based approach perspective most of us are reluctant to just "let the computer work out the answers". However, for sub-problems like establishing a failure model, looking into machine-learning approaches are more acceptable.

5.5 *Real-time execution models*

An objective of Industry 4.0 solutions is to have automated decision processes. For simple situations such as replenishment in retail we see automated replenishment policies. However for mixed problems as discussed here it is a long way to go to get trust in real-time execution models.

6 CONCLUSIONS

This paper has discussed steps in synchronization of operation and maintenance. An example was provided to illustrate some of the challenges and opportunities this will give. With idealized examples and simplified assumptions we are able to carry out relevant modelling. Still, these models are test (off-line) models and integration into real-time (on-line) models require significant effort. To succeed it is recommanded to start with a relative small set of standardized models for critical processes in the value chain of the company.

REFERENCES

Cheng, G. Q., B. H. Zhou, & L. Li (2017). Joint optimization of lot sizing and condition-based maintenance for multi-component production systems. *Computers & Industrial Engineering 110*, 538 – 549.

Christer, A. (1987). Delay-time model of reliability of equipment subject to inspection monitoring. *Journal of the Operational Research Society 38 (4)*, 329–334.

Cox, D. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society, Series B. 34 (2)*, 187–220.

Hehenberger, P., B. Vogel-Heuser, D. Bradley, B. Eynard, T. Tomiyama, & S. Achiche (2016). Design, modelling, simulation and integration of cyber physical systems: Methods and applications. *Computers in Industry 82*, 273 – 289.

Kacem, M. A. H., Z. Simeu-Abazi, E. Gascard, G. Lemasson, & J. Maisonnasse (2017). Application of a modeling approach on a cyber-physical system 'robair'. *IFAC-PapersOnLine 50*(1), 14230 – 14235. 20th IFAC World Congress.

Kim, S. & S. Park (2017). Cps(cyber physical system) based manufacturing system optimization. *Procedia Computer Science 122*, 518 – 524. 5th International Conference on Information Technology and Quantitative Management, ITQM 2017.

Phillips, D., A. Ravindran, & J. Solberg (1976). *Operations research: Principles and practice*. New York: John Wiley&Sons.

Rojas, R., R. Rauch, R. Vidoni, & T. D. Mat (2017). Enabling connectivity of cyber-physical production systems: A conceptual framework. *Procedia Manufacturing 11*, 822–829.

Upasani, K., M. Bakshi, V. Pandhare, & B. K. Lad (2017). Distributed maintenance planning in manufacturing industries. *Computers & Industrial Engineering 108*, 1 – 14.