COMPUTER ASSISTED APPROACHES IN PRODUCTION ENGINEERING

Graph-based prediction of missing KPIs through optimization and random forests for KPI systems

Marvin Carl May[1] · **Zeyu Fang[1]** · **Michael B. M. Eitel[1]** · **Nicole Stricker[1]** · **Debarghya Ghoshdastidar[2]** · **Gisela Lanza[1]**

Abstract

Key performance indicators (KPIs) are widely used to monitor and control the production in industry. On an aggregated level, often represented as graphs or interrelated KPI systems, a comprehensive overview is given. However, missing or inaccurate sensor data and KPIs, as well inconsistencies in KPI based management are a major hurdle disturbing operations. To counter the impact of such missing KPIs, we propose a value optimization based approach to reconstruct the values of missing KPIs within a KPI system. While the approach shows successful reconstruction in the case study, the value optimization can be sped up through a random forest prediction of the initial optimization set. Thus, the inclusion of previous knowledge about the system behavior proves beneficial and superior to the pure optimization based approach, as validated by both randomized and simulation-based measurement data.

Keywords KPI System · Graph machine learning · KPI prediction · Optimization

1 Introduction

1.1 Motivation

Reaching and maintaining companies' strategic goals is always of highest priority to managers and can be monitored and managed through key performance indicators (KPIs). KPIs offer effective measurements to translate business models into well-defined and comparable parameters [1]. Nevertheless, the selection of suitable KPIs remains a complex problem [2], in particular regarding interrelations between certain indicators and the need to cover all levels sufficiently, from machinery [3], production systems [4] to entire companies [1]. In the field of logistics, production and operations management, a set of 34 important KPIs is formulated in form of the ISO-22400-2:2014 standard [5], was conceived. However, the interrelations within such KPI systems are hardly understood [6]. To mitigate this lack of understanding, a hierarchical approach of a KPI graph model recently became presented by Kang et al. [4] and extended by Stricker et al. [2]. This permits further understandability, but also leads to imperfect shopfloor optimization measures in the case of missing KPIs [7], due to sensor failure, manual intervention [8] or overall missing data acquisition [9]. In addition to a hierarchical approach, graphical KPI models help to address a better and precise understanding [10]. However, the challenge of accurately reflecting production system behavior, given missing or additional relevant KPIs, still prevail. Brundage et al. additionally mention the need to take special care of the KPIs' dependencies [11]. Against this background, this paper proposes a novel, value optimization based method to make use of the interrelations between KPIs in a KPI system to reconstruct incomplete KPI systems stemming from sensor failure, manual interventions or missing KPIs. This is extend by a random forest to initialize

✉ Marvin Carl May
marvin.may@kit.edu

Zeyu Fang
ugiun@student.kit.edu

Michael B. M. Eitel
michael.eitel@student.kit.edu

Nicole Stricker
Nicole.Stricker@hs-aalen.de

Debarghya Ghoshdastidar
ghoshdas@in.tum.de

Gisela Lanza
gisela.lanza@kit.edu

[1]   wbk Institute of Production Science, Karlsruhe Institute of Technology (KIT), Kaiserstr. 12, Karlsruhe 76131, Germany

[2]   Department of Informatics, Technical University Munich (TUM), Arcisstraße 21, Munich 80333, Germany

the value optimization and improve convergence, as shown in the case study.

The problem can be exemplified with Fig. 1, where a few KPIs are missing (red), due to data inconsistencies, sensor failures, incomplete KPI capturing and other reasons. KPIs that neighbor missing KPIs (blue), i.e. that are relevant for its calculation or vice versa, should remain at their known value while the red KPI values should be identified in order to fully reconstruct the incomplete KPI system.

1.2 Research questions and main contribution

In this paper the following research questions are addressed:

1. How can an incomplete KPI system be accurately completed, so that both dependencies and mutual influences are respected during the completion?
2. How can knowledge about a complex production system and its previous behavior be used to properly facilitate the KPI system completion?

2 Literature review

Defining KPIs and KPI systems is a common topic in literature, where KPIs describe a set of parameters [11] and KPI systems describe a set of KPIs with inherent relationships [2]. The latter can have a holistic structure [12] and is often used to improve the robustness of entire production systems. Other KPI systems use a hierarchical structure [4] and focus on the continuous improvement and management of smart manufacturing systems.

First, KPI systems are often used to observe and manage production, ranging from single processes [13] and machines to large-scale systems [12]. Hence, a major problem is the
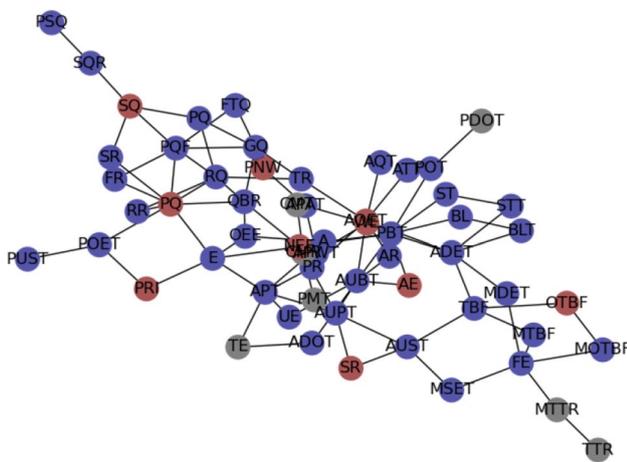

Fig. 1 Exemplary KPI set with missing (red) and relevant (blue) KPIs

selection of the KPIs such that they not only monitor low-level processes [14] but are also suitable for overall performance measurements [15]. Nevertheless, such graph-based systems are rarely regarded holistically [10].

Secondly, the derivation of KPIs and their relationships in KPI systems commonly follow a mathematical approach. Most commonly, matrices [16], pairwise relations [11] or the direct incorporation of standardized ISO 2240 [5] are used. Alternatively, systematic relations are conveyed in graph-based approaches, e.g. Stricker et al. [12] use nodes and links to express the interconnections within KPI sets. This brings along the advantage of a clear relationship definition [17].

Given the complex interrelations between KPIs [4] it is important to make the hidden mathematical interrelations between KPIs understandable by a dimensional analysis [19]. For a similar goal, but focused on sustainable manufacturing, Amrina et al. [21] use structural modeling for KPI relationship analysis. In order to incorporate the different information content of KPIs under different circumstances Stricker et al. [2] select KPIs through a linear programming approach. Approaches that focus on describing KPI system interrelations are presented in Table 1.

In spite of optimizing the production, the relevance of predicting future or unknown values of KPIs is of particular importance [25]. While the application of Bayesian networks (BN) [26] to obtain optimal KPI values and predict their development or a locally weighed regression for KPI prognosis [27] recently gained interest, the majority, i.e. Zhang et al. [28], focus on an analysis of static and dynamic relationships between process parameters and KPIs. Such approaches are enhanced by factorization [29] for fault prognosis or Modified Kernel Ridge Regression [30] for process parameter decomposition. Thus, the integration of Machine Learning approaches, such as Wu et al. [31], who apply Hidden Markov Models for machine deterioration estimation and remaining useful life prediction based on multiple KPIs, show promising results. A holistic overview is shown in Table 2.

A third body of knowledge is focused on monitoring and improving the performance of production systems by regarding its corresponding KPIs. For instance, Cao et al. [26] and Kornas et al. [22] add a feedback obtained by bottleneck analysis to overcome the production system's weakness. Furthermore, Tahir, Mahmoodpour, and Lobov [18] use an extensible visible language to allow decision makers to identify improvement potential within the defined KPI system.

The application of KPI systems to accurately steer production systems has still not been widely accepted within industry [4]. One major drawback of KPI systems, however, is their instability towards missing, tampered or inaccurate signals. Thus, a research gap in filling missing information within KPI Systems arises, and, additionally, the integration

Table 1 Comparison of different KPI relationship descriptions

Approach by		KPI Standard		KPI relationship definition				Main topic
		Self defined	ISO 22400	Matrix	Flow chart	Graph	Derivation	
Kang et al. (2016)	[4]	○	●	●	◐	◐	●	Hierarchical KPI structure
Tahir et al. (2019)	[18]	○	●	○	○	○	○	Extensive KPI description
Stricker et al. (2016)	[12]	○	●	○	●	○	○	Holistic KPI sets
Brundage et al. (2017)	[11]	○	●	●	●	●	○	Graph-based visual KPIs
Panicker et al. (2019)	[19]	●	○	○	○	●	●	KPI-Cost relationship
Joppen et al. (2019)	[17]	○	●	○	●	○	●	KPI relationship analysis
Bhadani et al. (2020)	[20]	○	●	○	○	○	●	KPI process simulation
Amrina and Vilsi (2014)	[21]	●	○	◐	○	○	○	KPIs for sustainability
Kornas et al. (2019)	[22]	●	○	○	○	○	●	KPI-based quality assurance
Stricker et al. (2017)	[23]	○	●	○	●	○	●	KPI resource interdependence
Zhu et al. (2017)	[24]	◐	●	○	○	○	●	Practical KPI set application
Zhu et al. (2018)	[16]	●	●	●	○	○	●	ISO 22400 and real needs gap
Stricker et al. (2017)	[2]	○	●	○	○	○	●	Linear selection of KPIs
Legend: ● regarded				◐ rudimentary regarded			○ not regarded	

Table 2 Comparison of different KPI prediction approaches

Approach by		KPI Standard		Solution Strategy				Main topic
		Self defined	ISO 22400	BN	PLS	LCF	Derivation	
Cao et al. (2016)	[26]	●	○	●	○	○	●	Bayesian network multi prediction
Yin et al. (2015)	[27]	●	○	○	●	○	●	Fuel Cell System Long-term prognosis
Panicker et al. (2019)	[19]	●	○	●	○	○	●	KPI & production cost interrelationship
Xie, Sun, and Cheung (2015)	[32]	●	○	○	○	○	●	Outliers for KPI prediction & diagnosis
Bhadani et al. (2020)	[20]	○	●	○	○	○	●	KPI process simulation
Krueger et al. (2015)	[29]	●	○	○	○	●	●	KPIs prediction for wastewater
Zhang et al. (2015)	[25]	●	○	○	●	○	●	KPI-based process monitoring s
Sun et al. (2017)	[30]	●	○	○	●	○	●	Measuring production process variables
Wu et al. (2018)	[31]	●	○	○	○	○	●	KPI-based machine det. estimation
Zhang et al. (2017)	[28]	●	○	○	●	●	●	KPI-based fault detection framework
Yin, Zhu, and Kaynak (2014)	[33]	●	○	○	●	●	●	KPI-based System Fault diagnosis
Yin, Wang, and Yang (2013)	[34]	●	○	○	●	●	●	Robust KPI-related prediction
Legend: ● regarded				◐ rudimentary regarded			○ not regarded	
Abbreviations: BN Bayesian Network			PLS Partial Least Squares			LCF Low Angle Fatigue		

as well as usage of previous knowledge about KPI systems is not yet regarded holistically [6].

3 Selected KPI system

The paper considers the KPI system presented by Kang et al. [4], as it encompasses all KPIs mentionend in the ISO standard [5] and additionally is the largest regarded holistic KPI set. It is widely known [2] and for reference the KPI list is introduced in Table 5 in the appendix. Nevertheless, certain corrections, with respect to the KPI's calculations as presented by Kang et al. [4] have to be made:

1. $CMR = CMT/PMT$ is corrected to
   $CMR = CMT/(CMT + PMT)$ according to the definition of CMR: "ratio of the total corrective maintenance time CMT to the sum of CMT and the preventive maintenance time PMT" [4].

2. Some KPIs such as $AUIT = ADET$, $TTF = TBF$ or $MTTF = MTBF$ have the same definition and mathematical calculation expressions in the regarded production

system. In order to keep the prediction system stable and avoid unnecessary problems by formulating the optimization problem, only *ADET*, *TBF*, *MTBF* are kept in the KPI system.

The following requirements for missing KPI sets can derived from the literature: According to ISO-222400 [5] there are 34 relevant manufacturing KPIs, divided hierarchically into 3 groups, which we distinguish: low level KPIs (directly measured from production lines), basic KPIs (calculated from low level KPIs) and comprehensive KPIs (calculated by using basic and low level KPIs). Kang et al. [4] and Stricker et al. [2] have illustrated the interrelationships between each KPI set on all levels and extended the set to 59 interrelated KPIs. KPIs can be easily calculated by means of the formal definition from ISO-22400 [5], Stricker et al. [2] and Kang et al. [4]. Thus, if only few KPI values are missing and these are mathematically independent from one another, the same calculation logic can be applied. Hence, there is no need to use the developed comprehensive combination of a random forest and subsequent value optimization under low dimension scales. However, the more complex the interaction between missing KPIs and the larger the set and relationships among these is, the more difficult it is to find the optimal solution. In particular, this counts for KPIs that result in equations with more than one unknown impede convergence. For instance, if *CMR* is known and the given relationship is: $CMR = CMT/PMT$, there are infinitely many possible solutions to satisfy this one condition. Thus, KPI sets with few missing KPIs can be solved easily and, therefore, in this case study there are only KPI sets with more than ten missing KPIs and a graph based distance larger than one between two missing KPIs considered. Otherwise, solely the less accurate random forest prediction approach can be used, which means that no subsequent value optimization is performed.

Furthermore, the KPIs to be predicted are randomly selected from the whole KPI system according to ISO-22400 [5] and the extensions from both Kang et al. [4] and Stricker et al. [2], obeying the selection rules mentioned before: mathematically independent from each other and graphically distanced at least two edges, which are easily identifiable through the corresponding adjacency matrix.

4 Methods

The ability to find the values of missing KPIs within a KPI system is targeted by using mathematical relationships between the nodes of the corresponding KPI graph. Hence, the core of the approach is twofold: (1) predicting missing KPIs based on previous observations and (2) defining an objective function that minimizes the error between

contradicting KPI definitions which is optimized to obtain the value of missing KPIs. The initial values for the KPI value optimization can be obtained either by a random initialization (referred as cold start) or by a random forest based KPI prediction (referred to as warm start). Random forests are selected as bagging and bootstrapping allow successful handling of high dimensional data with low bias and moderate variance as well their robustness towards outliers and nonlinearity [35]. The applied methods are presented in detail in the following.

4.1 Optimization

Value optimization is based on the application of mathematical search to find the convergence point of an objective function. It offers significant efficiency on convex functions and can be divided into constrained and unconstrained optimization. In this paper the performance and suitability to solve the described KPI optimization problem with the following two different optimization methods is investigated: unconstrained optimization and constrained optimization, i.e. constraining the solution space to meaningful KPI values. Due to their superior results in application the following algorithms are applied: Nelder-Mead as an unconstrained algorithm [36, 37] and L-BFGS-B (Limited-memory (L-) Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm with bound constraints (-B)) for the constrained optimization [38].

The problem of finding the missing value of a KPI that is defined by given and known KPIs is trivial by application of the describing functions. However, in a KPI system if more than one KPI is missing, a KPI value is defined twofold:

1. The numerical, known (i.e. measured) value of the KPI i (denoted as A_i), and
2. the value obtained by the interrelationships with it's defining neighboring KPIs x and their values: $v_i = v_i(x)$, where x is the vector of relevant neighboring KPIs, excluding i itself.

Thus, finding the true value of an unknown KPI i is similar to ensuring that the known or estimated value A_i equals the defined value v_i. Since several KPIs must be missing to regard non trivial cases, the following total objective function can be set up by using a quadratic derivation punishment:

$$\min f(\cdot)_i = \sum \left(\frac{A_i - v_i(x)}{A_i}\right)^2$$

where $f(\cdot)$ signifies the total objective function, which is accumulated from each optimization expression of KPIs that are relevant for finding the missing KPIs. Hence, i refers

to the index of the KPI to be optimized for and A is the value of the KPI that is either known (A remains constant) or unknown (A converges over time). The parameter $v(x)$ is the new value of the KPI in the current optimization step, which usually is used to improve A within each iteration.

Based on the newly defined objective function f, the applicable solution space is subject to the constraints g_i and h_j. Given that f, g_i and h_j are real-valued functions on X including at least one nonlinear term, a nonlinear value optimization problem is formulated as:

$$
\begin{aligned}
min \quad & f(X) \\
s.t. \quad & g_i(X) \leq 0 \; \forall i \in (1,2,3,...,m), \\
& h_j(X) = 0 \; \forall j \in (1,2,3,...,p), \\
& where \; X = \left[x_1, x_2, ..., x_n\right]^T \in R^n.
\end{aligned}
\tag{1}
$$

The variable $x_i \in R$ refers to measurable variables, i.e. KPIs, with n missing KPIs. Equality constraints (h_j) and inequality constraints (g_i) can be used to integrate further domain or system specific knowledge about the KPI system. For instance are upper bounds ($A_i \leq 100\%$) for an individual KPI i implemented through the inequality constraint $g_i(x) = v_i(x) \leq 100\%$. In application the problem in general appears not only once but whenever the KPI system was recorded, so that each individual KPI system solution according to the above describe value optimization can be regarded as a sample of the overall problem.

The optimization problem, when considering known bounds of the missing values, can efficiently be solved by the application of L-BFGS-B, which operates on an approximated Hessian matrix and belongs to the field of Newton methods. Thus, in cases with vastly differing boundaries, the optimization can stop immediately as the second order partial derivatives might be, falsely, considered to be equal to zero. Without additional boundaries, the optimization is solved through a more stable Nelder-Mead solver, implemented in Python by [39].

4.2 Optimization initialization

Solving the above described optimization problem with constrained or unconstrained value optimization requires feasible initial values for the missing KPIs. Better initialization values decrease the number of value iterations and, hence, can significantly improve the speed of convergence. To that end, random initialization values for unknown KPIs can be used, i.e. standard implementations, or alternatively knowledge acquired by observation of previous KPI values can be used. The former is regarded as the base case, while the latter can be approached by taking the known or optimized KPI values of KPI system at time t as the initialization value of the succesor at time $t + 1$. Due to the sensitivity of minor derivations between the values of each KPI at time t

and $t + 1$, this simple, time invariant approach approach is rejected. However, the dynamics of KPI systems and the individual KPI values could be estimated based on a large observed time frame, to predict accurate, well-performing initialization values.

4.3 Random forest for initialization improvement

A random forest [35] can be used as a method for classification and regression problems. Random forests do bagging on decision trees [40], i.e. by creating many random regression trees. These types of decision trees, with leaves representing numeric output values, search for a function $H(X) : X_i \longrightarrow y_i$, that maps from a given KPI system X_i excluding the values y_i to the missing values y_i. (X_i, y_i) is referred to as a sample and denotes the incomplete KPI system at a certain time. Within the bagging process, random forests introduce two sources of randomness for each regression tree: First, a bootstrap sample (i.e. sampling with replacement from the training data set) is used to construct each individual regression tree. Second, during splitting a node when constructing the tree, the best split is randomly drawn from all input feature (here X_i) or a random subset. The size of the random subset can be controlled, yet, in the regarded case equal to 90% of the input feature size.

Random forests use bagging and, as outlined, is a simple machine learning algorithm with few parameters, (partial) interpretability, good generalizability, non-linearity [35] that is well established [40]. For the subsequent validation, the random forest implementation of sci-kit learn [41] is used. Note, that in contrast to original work, which averages on each classifier vote or regression [35], their average probabilistic prediction, or regression, is used [41].

In this paper a random forest is applied as a warm start tool for the optimization based prediction of certain missing KPIs delivering start values in a very likely region. In contrast to simple cold start optimization, where the initial values are randomly selected or based on previous values the idea is to boost the optimization by selecting more likely well performing initial optimization values. Random Forest algorithms have many advantageous properties: Training speed is quick, even with high dimensional attributes, results are acceptable concerning stability and, thus, it is not required to frequently retrain from scratch, while storing is also easy to perform. To compare the performance, a cold start refers to a random initial value generation. Thus, the searching space in the cold start is larger assuming a sufficiently well-performing random forest prediction. Using the warm start method, the optimizations obtains certain benefits:

- The value overflow errors exhibited by Nelder-Mead can be widely reduced.

- The value optimization process can be significantly shortened due to a faster convergence resulting in a prediction system with quicker responses.
- Contribution to the stability by high dimension scale KPI prediction model.
- Partially restrain the local minimum problem.

5 Results

In order to compare the proposed methods, four control test are performed. The performance and the number of iteration steps of the value optimization process are compared regarding the following ceteris paribus tests:

Test 1: Warm Start vs. Cold Start with the same number of prediction KPIs
Test 2: Warm Start vs. Cold Start with the same number of samples
Test 3: Nelder-Mead vs. L-BFGS-B with the same number of prediction KPIs
Test 4: Nelder-Mead vs. L-BFGS-B with the same number of samples

Each iteration number in the case study refers to the average result from 20 identical tests with random missing KPI set selections, where the number of KPIs to be predicted, i.e. prediction KPIs, is controlled. In addition, a performance check of the objective function under the influences of the number of prediction KPIs and samples is carried out. This targets the performance of the optimization methods under a high dimension scale and with plenty of data samples. For all of these investigations large KPI data sets based on random realizations of various production

systems are used. At last, the performance of the prediction methods is validated on simulation-based data. For this purpose, simulated KPI sets of a real production system are provided by running an ontology based discrete event simulation [42] initialized with a real-world production system data set. Given the incompleteness, random noise and time variant KPI tracking and calculation in reality, applying the OntologySim is beneficial as the KPI sets introduced by both [5] and [4] are tracked [42].

5.1 Initialization with warm start (Tests 1 & 2)

In this section the influence of the initialization methods (i.e. warm start and cold start) on the performance of KPI predicting system, with respect to the number of samples and prediction KPIs, is discussed.

Figure 2a illustrates, that the parameters iteration steps and number of samples show a nearly linear relationship using both cold start and warm start. Note, that the number of iteration steps is used to compare performance, as time measurements depend greatly on the used hardware. Clearly, the warm start offers a significant advantage in terms of average required iteration steps compared to the cold start. It is obvious that with an increasing number of missing KPIs, both optimization procedures require more iteration steps. Nevertheless, the number of needed iterations till convergence increases slower using the warm start. However, this speed up in optimization only becomes significant dealing with more than 12 missing KPIs, as shown in Fig. 2b.

Overall, the warm start delivers a better performance than the cold start on both, iteration steps and convergence time respectively.

Fig. 2 Comparison of the performance between Warm Start and Cold Start in iteration steps for different numbers of missing KPIs and samples, showing the improvements in runtime achieved through warm start
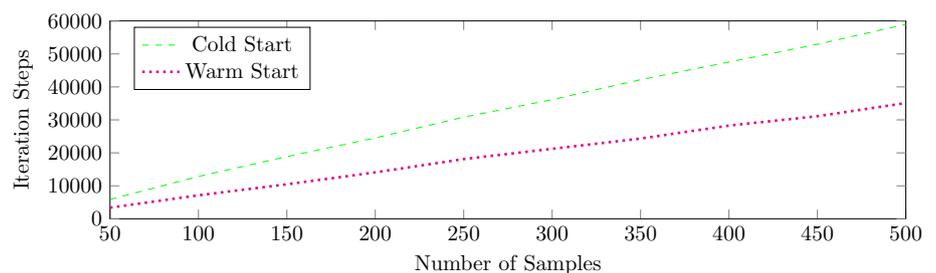
(a) The influence from the number of samples on iteration steps
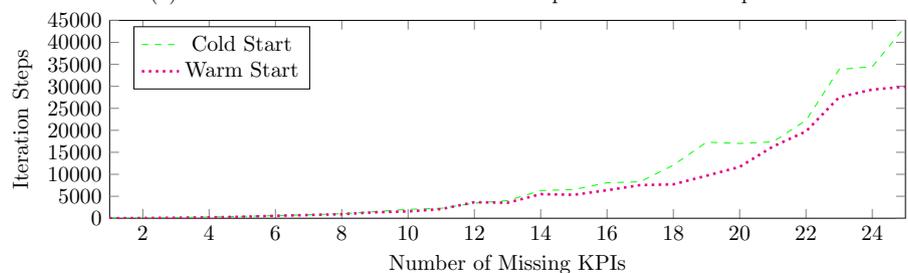
5.2 Comparison of the value optimization algorithms (Tests 3 & 4)

In this section, the performance of the two value optimization algorithms (Nelder-Mead and L-BFGS-B) is compared. Figure 3a shows again a nearly linear relationship between the required iteration steps and the number of samples. However, the L-BFGS-B needs significantly less iteration step to converge towards a solution. Hence, the quasi newton method L-BFGS-B provides a more efficient optimization algorithm for this application. Additionally, this accentuates the worth of including domain knowledge, i.e. reasonable bounds for the unknown KPIs.

Regarding the number of predicted KPIs, the L-BFGS-B still delivers the better results compared to Nelder-Mead, as demonstrated in Fig. 3b. This performance gap widens for higher numbers of prediction KPIs. However, Nelder-Mead has a relative stable performance when the number of prediction KPIs increases using the improved high dimension adaptive function mentioned in [37].

In addition to the two above visible effects, the iteration steps also vary due to the settings of the objective function tolerance: the smaller the tolerance of the iteration process is, the more calculation steps are needed.

With respect to the value optimization process we recall that each number of missing KPI trials consists of 20 tests with different KPI sets. Figure 4 illustrates, how many times the value optimization method successfully reaches the convergence point.

Nelder-Mead shows a very stable optimization process, when the dimension scale of the objective function, i.e. the number of missing KPIs, is under 13. Beyond that the optimization process easily becomes unstable, and in some cases Nelder-Mead even failed to find the optimal solution before the process reached the termination criterion. On the other hand L-BFGS-B less often provides a stable result even if the dimension scale of the objective function is at a low level. The reason might be an invalid Hessian matrix during the value optimization process, which terminates the process improperly. Another typical optimization problem occurs in cases of (partly) non-convex objective functions.

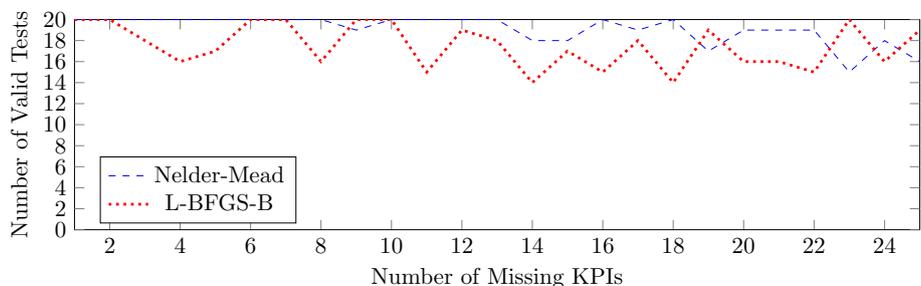**Fig. 3** Comparison of the performance between Nelder-Mead and L-BFGS-B
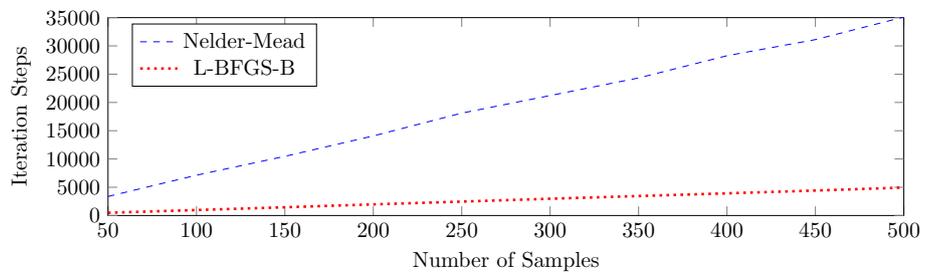
(a) The influence from the number of samples on iteration steps
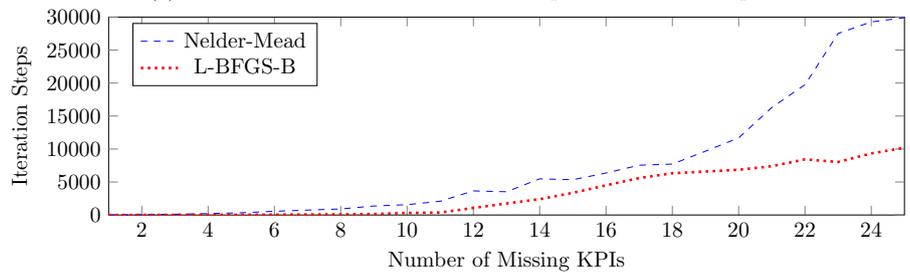
(b) The influence from the number of prediction KPIs on iteration steps

Fig. 4 The number of valid value optimization tests

5.3 Performance regarding the objective function value

In this section the comprehensive performance of the KPI prediction system is analyzed, considering the influence of the number of samples and KPIs that are predicted. In Fig. 5, which show the normalized value of the objective function for an exemplary study of the influence of sample size and missing KPI number, there are two trends clearly visible:

1. The value of the objective function is nearly independent from the number of samples. Once the number of prediction KPIs is set, the value of objective function keeps almost the same level despite some small oscillations, as the number of samples is (almost) irrelevant.

2. The number of prediction KPIs affects the accuracy of the KPI prediction system. If the number of prediction KPIs increases, the value of the objective function rises significantly. However, the value oscillation is not dependent on the number of missing KPIs. Otherwise, the number of samples has a relative small contribution to the oscillation on the value of objective function or the accuracy of the KPI prediction system.

## 5.4 Individual KPI performance comparison

In the following, a comparison of the predictability, i.e. the deviation from the original value for a missing KPI, is presented. For this experiment, 500 randomly selected sets of missing KPIs per missing number, each consisting out of 20 samples, are optimized. The number of missing KPIs

is varied to identify the effects of larger numbers of missing effects on the prediction quality. The resulting deviation is reported in a log_{10} scale, which is illustrated in Fig. 6. Apparently, there are certain KPIs that are prone to deviate largely, especially the earlier mentioned problematic KPIs, as well as ratios. These KPIs can easily result in sub-optimal points satisfying a reasonably low optimization function. With a higher number of missing KPIs, in general, the error remains similarly small or increases, as expected. The overall performance is consistently high, even for larger numbers of missing KPIs. Thus, the proposed method can be effectively applied to complement missing KPIs in KPI systems that are standardized and widely applied in industry.

5.5 Validation on industrial use-case simulation data

Since the combination of random forest initialization and subsequent optimization shows significantly superior results, the validation focuses on the warm start method comparing the two different optimization algorithms. The KPI data sets are generated using a multi-machine simulation of a real production system implemented with OntologySim [42]. The comparison itself targets the prediction accuracy and the convergence speed per iteration. Concerning the optimization result, the mean squared relative error (MSRE) is regarded, where every single KPI of the presented KPI system is included in 100 missing KPI sets varying the prediction number. Tables 3 and 4 show the results of the iteration speed comparison and the accuracy respectively.

Fig. 5 Influence of sample number and missing KPI number on function value
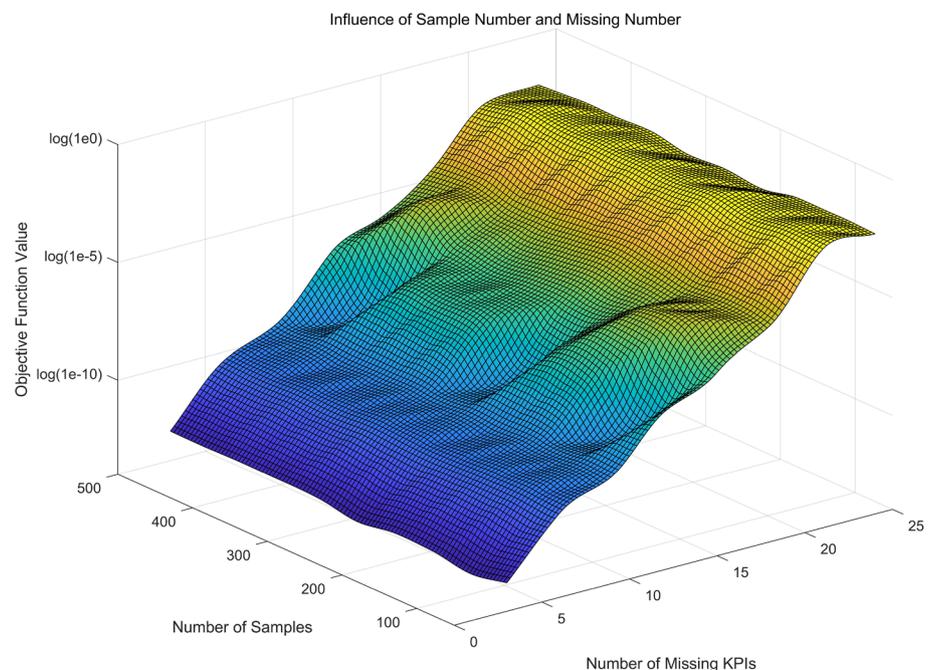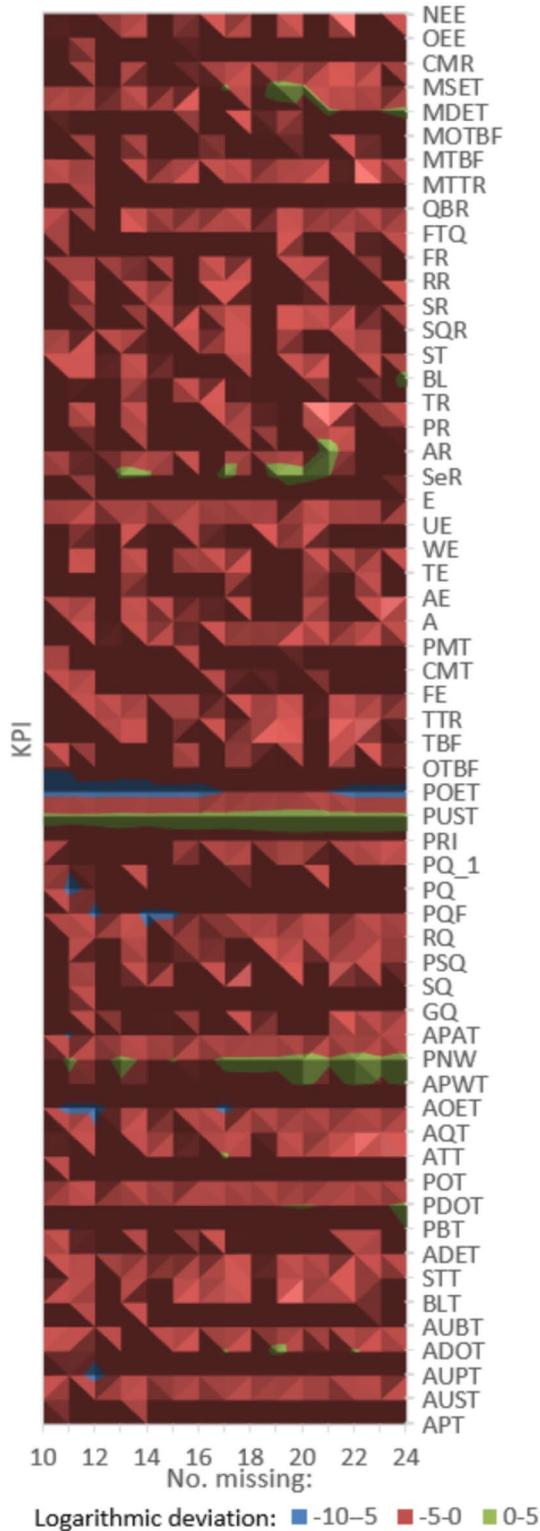
Fig. 6 Deviation between optimized and true values in log_{10} per individual KPI depending on number of missing KPIs

Table 3 Speed comparison of both optimization methods w.r.t. the number of missing KPIs (n)

n	Nelder-mead	L-BFGS-B
10	10,03 s	14,25 s
15	30,47 s	127,02 s
20	95,25 s	415,99 s

Table 4 Accuracy comparison of both optimization algorithms w.r.t. the number of missing KPIs (n)

n	$MSRE_{max}$	KPIs with MSRE < 0.01 (of 59)
Nelder-mead		
10	$2 \cdot 10^{-4}$	59
15	$5 \cdot 10^{-4}$	59
20	4,66	54
L-BFGS-B		
10	11,85	47
15	40,97	36
20	73,32	29

In both categories the Nelder-Mead algorithm reveals its strengths compared to the L-BFGS-B. Despite a significantly larger number of iteration steps, the Nelder-Mead predicts the KPI-system faster. For 20 missing KPIs it even converges more than four times faster than the L-BFGS-B. L-BFGS-B may have got the more efficient algorithm but also is way more complex to calculate, resulting in a slower optimization convergence. The comparison of the accuracy demonstrates the second advantage of using Nelder-Mead. The maximum MSRE stays close to zero even up to 15 missing KPIs. Considering high scaled problems with 20 missing KPIs, the Nelder-Mead still predicts 54 KPIs with a MSRE below one percent. The L-BFGS-B on the other hand cannot compete with this performance. Moreover, the stability of Nelder-Mead is higher, yet it lacks the ability of L-BFGS-B to incorporate boundary values for KPIs.

Besides the two presented comparisons, the effects pointed out in previous sections are also visible during the validation with real simulation data. This especially includes the stability and the needed amount of iterations steps per algorithm. Table ?? summarizes the validation of both optimization strategies. Overall, the presented prediction technique allows to reconstruct almost half of KPI system, which is a remarkable performance.

6 Conclusion and outlook

6.1 Conclusion

This paper presents a KPI system prediction model that can be used with KPI systems that are calculated based on a graphical approach as introduced by Kang et al. [4] for missing data in a manufacturing environment. Given the mathematical relationships of KPIs, a random forest can be trained to initialize the value optimization, dubbed warm start. A subsequent optimization predicts the missing KPI values, handling different sizes of missing KPI sets. The proposed method performs suitably well and allows the identification of KPI values that are only indirectly observed in the form of its neighboring KPIs in the KPI system. The method is robust when increasing the sample size and applying different optimizers, however, best results are obtained in the warm start scenario with Nelder-Mead. The presented approach is validated based on simulation data of a real production system and is able to accurately predict almost half of the comprehensive KPI structure.

6.2 Outlook

In future work, it is possible to extend the method by integrating a very small $\epsilon > 0$ in the prediction formula for KPIs in order to improve the accuracy of the prediction and in rare cases avoid an invalid Hessian matrix. For dealing with the stochastical influences in real-world applications, where sensors or gathered KPIs might not perfectly fit to one another, this can alleviate the shortcoming of hardly being able to reduce the actual prediction error to zero. Furthermore, the integration of knowledge about the previous behavior of the KPI system can be extended, as the training model of the random forest can become more comprehensive and abundant to provide the value initialization with a better start point. The internal interdependence within the KPI system follows the patterns effectively used by the random forest. In future, the interdependence within a KPI system should be further evaluated. Additionally, beyond random forests alternative predictors can be evaluated, however, the core of the approach lies not in the predictor application. Also, it can be considered to effectively use L-BFGS-B, as this should avoid large value ranges of missing KPIs, that could help in calculating the Hessian matrix significantly better and faster. However, a more comprehensive approach requires the transition from this simulation based-data gathering approach towards a real world application. Further research shall also focus on the actual effects of implementing a more detailed KPI

system that can be completed in the case of missing KPIs. Managerial implications and value adding tasks have to be rigorously analyzed. Nonetheless, this research can enable a suitable comparison of different products according to their, then completed, KPI representation. Including more product-based KPIs for multi-product production systems can amplify the methods real-world applicability.

Appendix

An overview of the KPIs and abbreviations used is introduced in Table 5.

Table 5 Abbreviations for KPIs according to [2, 4]

Abbreviation	KPI
NEE	Net equipment effectiveness
OEE	Overall equipment effectiveness
CMR	Corrective maintenance ratio
MSET	Mean setup time
MDET	Mean delay time
MOTBF	Mean operating time between failures
MTBF	Mean time between failure
MTTR	Mean time to repair
QBR	Quality buy rate
FTQ	First time quality
FR	Fall off ratio
RR	Rework ratio
SR	Scrap ratio
SQR	Actual to planned scrap ratio
ST	Starvation ratio
BL	Blockage ratio
TR	Throughput rate
PR	Production process ratio
AR	Allocation ratio
SeR	Setup ratio
E	Effectiveness
UE	Utilization efficiency
WE	Worker efficiency
TE	Technical efficiency
AE	Allocation efficiency
A	Availability
PMT	Preventive maintenance time
CMT	Corrective maintenance time
FE	Failure events
TTR	Time to repair
TTF	Time to failure
OTBF	Operating time between failures
POET	Planned order execution time
PUST	Planned unit setup time
PRI	Planned run time per item

Table 5 (continued)

Abbreviation	KPI
PQ	Part quantity
PQ_1	Produced quantity
PQF	Produced quantity in first operation
RQ	Rework quantity
PSQ	Planned scrap quantity
SQ	Scrap quantity
GQ	Good quantity
APAT	Actual personnel attendance time
PNW	Personal not working
APWT	Actual personnel work time
AOET	Actual order execution time
AQT	Actual queueing time
ATT	Actual transportation time
POT	Planned operation time
PDOT	Planned down time
PBT	Planned busy time
ADET	Actual unit delay time
STT	Starving time
BLT	Blocking time
AUBT	Actual unit busy time
ADOT	Actual unit down time
AUPT	Actual unit processing time
AUST	Actual unit setup time
APT	Actual production time

References

1. Maté A, Trujillo J, Mylopoulos J (2012) Conceptualizing and specifying key performance indicators in business strategy models. In: International Conference on Conceptual Modeling, vol. 7532. Springer, pp. 282–291
2. Stricker N, Echsler F, Lanza G (2017) Selecting key performance indicators for production with a linear programming approach. Int J Prod Res 55(19):5537–5549
3. Kapp V, May MC, Lanza G, Wuest T (2020) Pattern recognition in multivariate time series: towards an automated event detection method for smart manufacturing systems. J Manuf Mater Process 4(3):88
4. Kang N, Zhao C, Li J, Horst JA (2016) A hierarchical structure of key performance indicators for operation management and continuous improvement in production systems. Int J Prod Res 54(21):6333–6350
5. Automation systems and integration—key performance indicators (kpis) for manufacturing operations management—part 2: Definitions and descriptions. Standard, International Organization for Standardization, Geneva, CH (January 2014)
6. Kechaou F, Addouche S-A, Zolghadri M (2022) A comparative study of overall equipment effectiveness measurement systems. Prod Plan Control, 1–20
7. Khan MAR, Bilal A (2019) Literature survey about elements of manufacturing shop floor operation key performance indicators. In: 2019 5th International Conference on Control, Automation and Robotics (ICCAR), pp. 586–592. IEEE
8. May MC, Maucher S, Holzer A, Kuhnle A, Lanza G (2021) Data analytics for time constraint adherence prediction in a semiconductor manufacturing use-case. Procedia CIRP 100:49–54
9. May MC, Behnen L, Holzer A, Kuhnle A, Lanza G (2021) Multivariate time-series for time constraint adherence prediction in complex job shops. Procedia CIRP 103:55–60
10. Hesse S, Spehr M, Gumhold S, Groh R (2014) Visualizing time-dependent key performance indicator in a graph-based analysis. In: Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA), pp. 1–7. IEEE
11. Brundage MP, Bernstein WZ, Morris KC, Horst JA (2017) Using graph-based visualizations to explore key performance indicator relationships for manufacturing production systems. Procedia CIRP 61:451–456
12. Stricker N, Pfeiffer A, Moser E, Kádár B, Lanza G (2016) Performance measurement in flow lines-key to performance improvement. CIRP Ann 65(1):463–466
13. Zhang K, Shardt YA, Chen Z, Yang X, Ding SX, Peng K (2017) A kpi-based process monitoring and fault detection framework for large-scale processes. ISA Trans 68:276–286
14. Abe M, Jeng J-J, Li Y (2007) A tool framework for kpi application development. In: IEEE International Conference on E-Business Engineering (ICEBE'07), pp. 22–29. IEEE
15. Ishak Z, Fong SL, Shin SC (2019) Smart kpi management system framework. In: 2019 IEEE 9th International Conference on System Engineering and Technology (ICSET), pp. 172–177. IEEE
16. Zhu L, Johnsson C, Varisco M, Schiraldi MM (2018) Key performance indicators for manufacturing operations management-gap analysis between process industrial needs and iso 22400 standard. Procedia Manuf 25:82–88
17. Joppen R, von Enzberg S, Gundlach J, Kühn A, Dumitrescu R (2019) Key performance indicators in the production of the future. Procedia CIRP 81:759–764
18. Tahir MA, Mahmoodpour M, Lobov A (2019) Kpi-ml based integration of industrial information systems. In: 2019 IEEE 17th International Conference on Industrial Informatics (INDIN), vol. 1, pp. 93–99. IEEE
19. Panicker S, Nagarajan HP, Mokhtarian H, Hamedi A, Chakraborti A, Coatanéa E, Haapala KR, Koskinen K (2019) Tracing the inter-relationship between key performance indicators and production cost using Bayesian networks. Procedia CIRP 81:500–505

20. Bhadani K, Asbjörnsson G, Hulthén E, Evertsson M (2020) Development and implementation of key performance indicators for aggregate production using dynamic simulation. Miner Eng 145:106065

21. Amrina E, Vilsi AL (2014) Interpretive structural model of key performance indicators for sustainable manufacturing evaluation in cement industry. In: 2014 IEEE International Conference on Industrial Engineering and Engineering Management, pp. 1111–1115. IEEE

22. Kornas T, Knak E, Daub R, Bührer U, Lienemann C, Heimes H, Kampker A, Thiede S, Herrmann C (2019) A multivariate kpi-based method for quality assurance in lithium-ion-battery production. Procedia CIRP 81:75–80

23. Stricker N, Micali M, Dornfeld D, Lanza G (2017) Considering interdependencies of kpis-possible resource efficiency and effectiveness improvements. Procedia Manuf 8:300–307

24. Zhu L, Johnsson C, Mejvik J, Varisco M, Schiraldi M (2017) Key performance indicators for manufacturing operations management in the process industry. In: 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), pp. 969–973. IEEE

25. Zhang K, Hao H, Chen Z, Ding SX, Peng K (2015) A comparison and evaluation of key performance indicator-based multivariate statistics process monitoring approaches. J Process Control 33:112–126

26. Cao Z, Liu X, Hao J, Liu M (2016) Simultaneous prediction for multiple key performance indicators in semiconductor wafer fabrication. Chin J Electron 25(6):1159–1165

27. Yin S, Xie X, Lam J, Cheung KC, Gao H (2015) An improved incremental learning approach for kpi prognosis of dynamic fuel cell system. IEEE Trans Cybern 46(12):3135–3144

28. Zhang K, Shardt YA, Chen Z, Yang X, Ding SX, Peng K (2017) A kpi-based process monitoring and fault detection framework for large-scale processes. ISA Trans 68:276–286

29. Krueger M, Luo H, Ding S, Dominic S, Yin S (2015) Data-driven approach of kpi monitoring and prediction with application to wastewater treatment process. IFAC-PapersOnLine 48(21):627–632

30. Sun W, Wang G, Yin S, Jiao J, Guo P, Sun C (2017) Key performance indicator related fault detection based on modified krr algorithm. In: 2017 36th Chinese Control Conference (CCC), pp. 7015–7020. IEEE

31. Wu Z, Luo H, Yang Y, Lv P, Zhu X, Ji Y, Wu B (2018) K-pdm: Kpi-oriented machinery deterioration estimation framework for predictive maintenance using cluster-based hidden markov model. IEEE Access 6:41676–41687

32. Xie X, Sun W, Cheung KC (2015) An advanced pls approach for key performance indicator-related prediction and diagnosis in case of outliers. IEEE Trans Ind Electron 63(4):2587–2594

33. Yin S, Zhu X, Kaynak O (2014) Improved pls focused on key-performance-indicator-related fault diagnosis. IEEE Trans Ind Electron 62(3):1651–1658

34. Yin S, Wang G, Yang X (2014) Robust pls approach for kpi-related prediction and diagnosis against outliers and missing data. Int J Syst Sci 45(7):1375–1382

35. Breiman L (2001) Random forests. Mach Learn 45(1):5–32

36. Lagarias JC, Reeds JA, Wright MH, Wright PE (1998) Convergence properties of the Nelder-mead simplex method in low dimensions. SIAM J Optim 9(1):112–147

37. Gao F, Han L (2012) Implementing the Nelder-mead simplex algorithm with adaptive parameters. Comput Optim Appl 51(1):259–277

38. Byrd RH, Lu P, Nocedal J, Zhu C (1995) A limited memory algorithm for bound constrained optimization. SIAM J Sci Comput 16(5):1190–1208

39. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, van der Walt SJ, Brett M, Wilson J, Jarrod Millman K, Mayorov N, Nelson ARJ, Jones E, Kern R, Larson E, Carey C, Polat L, Feng Y, Moore EW, Vand erPlas J, Laxalde D, Perktold J, Cimrman R, Henriksen I, Quintero EA, Harris CR, Archibald AM, Ribeiro AH, Pedregosa F, van Mulbregt P, Contributors S (2020) Scipy 1.0: Fundamental algorithms for scientific computing in python. Nat Methods

40. Liaw A, Wiener M et al (2002) Classification and regression by random forest. R News 2(3):18–22

41. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E (2011) Scikit-learn: machine learning in Python. J Mach Learn Res 12:2825–2830

42. May MC, Kiefer L, Kuhnle A, Lanza G (2022) Ontology-based production simulation with ontologysim. Appl Sci 12(3):1608