Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 93 (2020) 1164–1169

www.elsevier.com/locate/procedia

53rd CIRP Conference on Manufacturing Systems

A stochastic approach to calculate assembly cycle times based on spatial shop-floor data stream

Júlia Bergmann[a,*], Dávid Gyulai[a], Dávid Morassi[a], József Váncza[a,b]

[a] *EPIC Centre of Excellence in Production Informatics and Control, Institute for Computer Science and Control (SZTAKI), 13-17 Kende, Budapest 1111, Hungary*
[b] *Department of Manufacturing Science and Engineering, Budapest University of Technology and Economics, Müegyetem rkp. 3, Budapest, 1111, Hungary*

* Corresponding author. Tel.: +36-1-279-6166. *E-mail address:* bergmann.julia@sztaki.hu

Abstract

Indoor positioning systems (IPS) allow assets on the shop-floor to be tracked with a relatively high accuracy. In order to obtain the useful, underlying production information, smart and fast processing algorithms are needed, as IPSs produce an immense amount of data in a very short period. In the paper, a novel approach is presented that offers the near real-time calculation of assembly times, based on the dynamically streamed spatial data stream of assets. The approach relies on probabilistic analytic models, respecting the needs of manufacturing and operations management. The efficiency of the results is presented through an industry-related application case.

Keywords: Type your keywords here, separated by semicolons ;

1. Introduction

Just as a costumer wishes to track the status of her ordered items, a real-time tracing of shop-floor assets embrace fruitful knowledge for plant management. With the spread of digital technologies, the opportunity of collecting spatial data in industrial environments is not a troublesome question anymore, but rather the efficient use of these process-related data in enterprise level decision making processes. Considering the managerial objectives, the key requirements related to the digital technologies are the real business value that they are able to bring, and the associated return on investments. Many new technologies in the prototype and introduction stages have uncertain business-related benefits, as the high-level performance indicators and cost factors depend on the environment in which they are applied. Therefore, the importance of the so-called *proof-of-concept* projects is crucial in the digitization era, as many new solutions are available and each company seeks for those that best fit in their value chains.

Among these new applications, indoor positioning systems (IPS) have also received higher attention from the manufacturing industry, as they provide the opportunity of tracking and tracing assets in shop-floor environment more efficiently than ever. IPSs can be used for locating almost any kind of physical asset in a production environment; typical examples are the tracing of products, tools and fixtures. The relevance of accurate positioning might be even higher in production logistics, as transportation resources' routes are usually more complicated to follow than those of the products that can be located by e.g., Radio Frequency IDentification (RFID), where receivers are installed on predefined places. In contrast, tugger trains, automated guided vehicles (AGV), industrial drones or forklifts can move almost freely on the shop-floor, increasing the complexity to locate them, and optimize their utilization based on their historical paths'.

In the paper, a novel statistical solutions is presented that enables the utilization of IPS data in production management related decision, e.g., to balance assembly lines, predict lead times or optimize the utilization of certain resources. As IPSs usually provide the data in raw or semi-processed formats, therefore advanced analytics methods are often required to obtain the information that is useful for decision makers in the aforementioned processes.

The paper is structured as it follows. First, a literature review is provided, focusing on the introduction of recently applied IPSs and their utilization in production management and control (Section 2). In Section 3, the problem in question is specified, with the description of the production environment, the nature of the collected data and the results expected. Section 4 provides data analytics techniques that are applied to obtain information to support decision in production management. In order to demonstrate the applicability of IPSs in such decision making processes, numerical experimental results are presented in Section 5. The summation and future views are provided in Section 6.

2212-8271 © 2020 The Authors. Published by Elsevier B.V.

10.1016/j.procir.2020.03.052

2. Literature Review

In the era of the Internet-of-Things (IoT), smart devices are gaining more attention from the industry, with the aim of increasing the digitization rate of shop-floor applications [11]. A typical IoT application is the indoor positioning, as it can be applied at nearly every domain of manufacturing industry, and can be also installed in already operating systems. Several technology providers offer accurate IPS solutions, however the applied machinery ranges from visual sensors [21], through ultra-wideband (UWB) technology – that enables to achieve up to 2-5 cm accuracy, depending on the environment [19] –, to radar-based tracking [2]. Utilizing the fast wireless communication and the accurate asset tracking, IPSs enable to implement scalable and reliable real-time location systems (RTLS) used in warehouse management, fleet management of shop-floor management [4]. As for the physical architecture, a typical IPS is built up of a central data management server that implements the storage and processing of the data, received from the field devices. The latter is a set of tags that are emitting a signal in certain periods, and a set of fix anchors that are capable of receiving the tags' signals, and calculating the positions by using triangulating and/or trilateration functions [9]. The tags are usually equipped with a battery that—depending on the usage—can last up to months with a single charge. Thanks to the small size of an average tag, they can be attached to even small-size products, tools or machines.

As a result of decreasing prices of smart devices, the hardware-related costs of an industrial IPS application are relatively low [15], and the real strength of these systems relies in their scalability and flexibility in terms of use [1]. They enable the digitization of production systems besides relatively low IT investments, while useful data can be obtained about the product, processes and resources in near real time. However, the continuously generated data stream requires special care to be taken to ensure that the compressed summary faithfully captures the overall information that the data hold [10]. In industrial applications, the target shop-floor area is usually subdivided in zones [13], and the IPS system can determine the zone in which a given tag was in an active state, based on its x and y (and relatively rarely z) coordinates. Although a typical IPS employs advanced signal processing and noise filtering algorithms to assign tags to zones [6,24], some further post-processing algorithms [25] are often necessary to derive the target metrics, indirectly from the raw coordinates. Typical data and signal processing techniques—among others—rely on Kalman-filters [3,16], Monte Carlo [7,8] and machine learning approaches [12,17].

The aforementioned metrics are typically utilized in a higher level of the decision making hierarchy, e.g., to derive production control logic, scheduling policies or to improve processes based on actual parameters that reflect the real system behaviour. In production management and especially in control, data-driven decisions that consider the actual state of the system at any given point of time are called situation-aware ones. They usually utilize the fusion of a model-based system representation, and the real parameters obtained from the system, so as implementing the digital twin of it. In this way, one can make decisions about the system operation with a foresight on possible outcomes of certain scenarios, without disturbing the operation of the real system. In the paper, the IPS data is processed with the aim of obtaining the real values of some process-related metrics, enabling the later implementation of a situation-aware production control.

3. Problem statement

In the paper, two data analytics problems are investigated, namely, how spatial data provided by an IPS could be processed to gain profitable information and how it can be utilized efficiently in production management. The positioning system provides raw data about the asset locations over time, and the overall goal is to mine out such performance metrics that characterize the dynamics of the system, considering cycle times, utilization rates and workloads.

3.1. Description of the Production Environment

First, the production environment is introduced where the IPS is operated, and collects data about the products' locations. In the experiments of the paper, a discrete-event simulation (DES) model was used as a test environment, however, a real industrial use case with the corresponding infrastructure is the main motivation for this implementation of the study. Although the original use-case is from the automotive sector, the presented approaches and the applied analytics architecture are not limited to this industrial domain, but also applicable in any discrete manufacturing environment where asset location with IPS can be solved. The simulation model is a realistic testbed of the system in a sense that it provides information about the tracked assets' locations in near-real-time, reflecting the operation of an industrial IPS system. Replacing both the physical production environment and the IT infrastructure of the IPS, the simulation model implements both functions in a single model, and capable of streaming location data towards any application in real time.

The layout of simulation model of the manufacturing environment with seventeen workstations (WS_1, ..., WS_{17}) and buffers (B_1, ..., B_{17}), one rework area (WS_r and B_r) and a quality checkpoint is shown in Figure 1. The prefixed routing path of the products is also marked. In this production environment the elements are moved from one station to the other by operators of the shop-floor, and every assembly operation is performed by operators also. In a highly operator-based environment like this one, IPS might be the best solution for tracking assets, as – besides attachment at the beginning and detachment at the end of the line – it does not require any further attention from the operators (as opposed to RFID systems). On the assembly line, one main product type is assembled, but the method can be easily adopted into a multi-type production environments. The headcount of operators ranges between seven to seventeen, therefore, output rate and lead times strongly depend on the amount of available manual workforce. In order to avoid blocked processes and smooth the material flow, part buffers are placed between each consecutive workstations. After the assembly process at WS_{14}, a functional test is performed by a robot, and the rejected parts are transferred to a dedicated rework station to be corrected by a specially skilled operator. From the data processing perspective, it might be important that the shape of the line shows some typical patterns (e.g. U-shape).

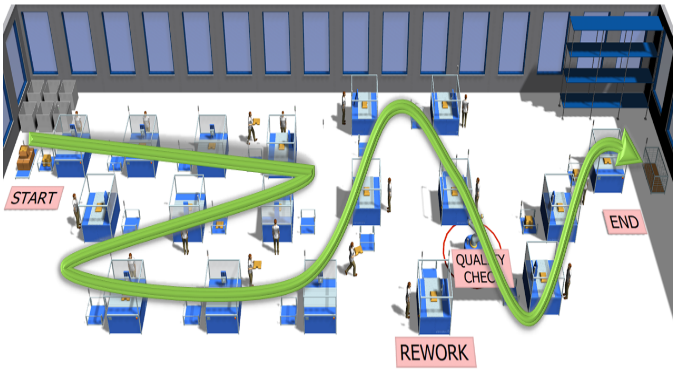

Fig. 1. Production layout

This DES model is independent from any specific industrial domain, therefore these workstations can be used to symbolize any arbitrary assembly operations. Since the product is also a general one, the complete model is ready to be applied at any shop-floor with an installed IPS. As for the processes under study, the DES model of an assembly system was implemented in *Siemens Tecnomatix Plant Simulation*.

3.2. Structure of position logs

As already mentioned before, the simulation model does not only represent the physical production environment, but also replaces the real IPS by streaming the parts' location data in real time. In the name of IPS installation, a data streaming interface (representing the IoT assets) and also a data collection platform are implemented. The data streaming is performed by the DES model itself, which is able to log the location of the tracked assets in every 8-10 seconds (relative to simulation, can be changed arbitrarily) in mySQL [23] database, including the ID of the tracked tag, its raw (unfiltered) x and y coordinates and the corresponding timestamp. Following the architecture of a real positioning system, depending on the amount of work-in-progress (WIP), the system can generate hundreds or even thousands of logs under a minute of operation. This leads to a massive amount of data over days and weeks of operation, asking for an efficient way of capturing, storing and filtering it.

As for the nature of the data, raw position logs are typically noisy, mostly because of the dynamic operating environment. In order to simulate this phenomena, a random noise was added to the position log stream, based on experiences from the original use case. The analyzed assembly area is cca. 25x50 metres, and the workstations have a cca. 2x2 metres size. The IPS system has an accuracy of cca. 10 cm, reflected by a normally distributing random noise on the position data. Following a realistic case, there are some outlier values in the data, resulted by environmental changes and issues. These outliers are simulated by a larger noise on the same position data, i.e., with a combination of geometrical and normal distributions. Accordingly, a normally distributed position error is added with 0 cm mean and 80 cm variance to some data points determined with a geometrical distribution, where the probability of a value 0 is set to be $p = 0.5$. Accordingly, this "larger" noise is added to cca. every second data sample of the stream.

3.3. Purpose of the Analysis and Questions to be Addressed

The paper is aimed at obtaining production management related metrics from the above characterized noisy IPS logs. Applying efficient approaches to filter the noise from a large amount of streamed data, the overall objective is to calculate such metrics from the positions that can be utilized in production control and process improvement decisions. The task is to calculate assembly cycle times, production lead times and stations' workloads by using the IPS data. The cycle times are considered to be the effective amount of human labor put in performing a certain assembly operation, as the products are only staying at a workstation when they are assembled, otherwise they stay in a buffer. In the know of the actual cycle times, engineers can refine the assembly line balances and the production schedule if needed. The workloads, more specifically the utilization rates of the workstations are indirectly calculated from the cycle times, supporting production managers to derive Overall Equipment Effectiveness (OEE) related metrics.

4. Data processing

Every IPS system has its weaknesses and usually it manifests in disposition, which may lead to calculating highly incorrect statistics, resulting in corrupted data to analyze. Some papers (see e.g., [14]) provide an overview of the existing wireless indoor positioning solutions and attempt to classify different techniques and systems. This section focuses on solving the problem of disposition by using a novel method based on noise reduction and the theory of *Markov chains*.

4.1. Noise reduction

The first step of spatial data cleansing is the reduction (or filtration) of additional noise. Several effective filtering methods exist, however, selecting the right one always depends on the problem in question [20]. A Savitzky-Golay filter [22] [18] is a digital filter that can be applied to a set of data points for the purpose of smoothing, that is, to increase the precision of the data without distorting the signal tendency. This is achieved—in a process known as convolution—by fitting successive subsets of adjacent data points with a low-degree polynomial by the method of linear least squares. When the data points are equally spaced, an analytical solution to the least-squares equations can be found, in the form of a single set of "convolution coefficients" that can be applied to all subsets of data, to give estimates of the smoothed signal, (or derivatives of the smoothed signal) at the central point of each subset. The process of S–G filtering is presented in Algorithm 1.

Algorithm 1 Savitzky–Golay filter

input: $(\tau_t, x_t)_{t=1}^{T} \in \mathbb{R} \times \mathbb{R}$ where τ_t is the tth timestamp
Set parameters $p, n \in \mathbb{N}$ where n must be odd
for $t = \frac{n-1}{2} : \left(T - \frac{n-1}{2}\right)$ **do**
  $\hat{x}_t = \sum_{s=\frac{1-n}{2}}^{\frac{n-1}{2}} C_s x_{s+t}$, where the convolution coefficients C_t depend on parameter p (discussed in details in [18])
end for

One of the main advantages of the S–G process is the fact that new data can be added easily and incrementally. The latter attribute enables the user to implement easily the concept even on extremely large and constantly increasing data sets. By all means, numerous variations of noise reduction exist, e.g. spline fitting [5].

4.2. Stochastic Rezoning

Matching the observed spatial data with a predefined routing consists of two parts: first, the smoothed data must be dragged onto the route, then a probability-based correction is applied. Formally, the prefixed process routing is described by a directed graph G which consists of N vertices ($v_i \in V$) and directed edges ($e_{ij} \in E$). The vertices of this graph are called zones, as they represent distinct workstations on the shop-floor. The exact spatial coordinates of every zone are assumed to be known. For each product k, the filtered spatial data of movements are available: $\left(\left(\tau_t^k, \mathbf{x}_t^k\right)_{t=1}^{T_k}\right)_{k=1}^{K}$ where $\mathbf{x}_t^k = \left(x_t^k, y_t^k, z_t^k\right) \in \mathbb{R}^3$ is a multidimensional (2-D or 3-D) vector. The elements of this sequence are dragged onto graph G, simply by finding the closest (by any arbitrary distance metric, e.g., Euclidean or Manhattan) vertex a_i^k, i. e. finding the closest zone. In this way, another sequence $\lambda^k = \left(a_1^k, a_2^k, \cdots, a_{T_k}^k\right)$ is born whose elements are the vertices of G where $a_t^k \in V$. Let us also define $\Lambda^k = \left(\left(a_1^k, a_2^k\right), \left(a_2^k, a_3^k\right), \cdots, \left(a_{T_k-1}^k, a_{T_k}^k\right)\right)$ sequence of state pairs which will be referred to as steps from one zone to another.

The steps defined as above are assigned into two categories: *true* and *false* steps. If the step $\left(a_t^k, a_{t+1}^k\right)$ has the same start and end points (i.e., $a_t^k = a_{t+1}^k$), then the step is considered to be true. Otherwise, a certain step must complete two conditions to be a true step. First, it has to be enabled by the prefixed routing line, i.e. the step $\left(a_t^k, a_{t+1}^k\right)$ can be a true step if there is a directed edge in the graph G from a_i^k to a_{t+1}^k. Secondly, there must not be coming backs later, i.e. for all $r > t+1 : a_t^k \neq a_{t+1}^k \Rightarrow a_r^k \neq a_t^k$ stands. If any of these statements are not completed for the observed step, then it is considered to be a false step. Even after the noise filtration, several false steps might emerge in Λ^k due to the inaccuracy of IPS. This phenomena requires some further correction.

To accomplish the probability-based correction on Λ^k, for each edge e_{ij} from v_i to v_j of graph G, we assign a p_{ij} probability based on the frequency of good steps. The p_{ij} probabilities can be mathematically formulated as

$$p_{ij} \triangleq \frac{\sum_k \#\tilde{S}_{ij}^k}{\sum_k \#S_{ij}^k}, \tag{1}$$

where # denotes the cardinality of the sets. The set S_{ij}^k contains all steps from v_i to v_j zones (vertices of G graph), i.e $S_{ij}^k = \left\{(\alpha,\beta) \in \Lambda^k : (\alpha,\beta) = \left(v_i, v_j\right)\right\}$. The set $\tilde{S}_{ij}^k$ consists of only the true steps of Λ^k from v_i to v_j, formally, $\tilde{S}_{ij}^k = \left\{(\alpha,\beta) \in \Lambda^k : \forall r > \mathtt{ind}(\beta) : a_r^k \neq v_i\right\}$, where $\mathtt{ind}(\beta)$ means the lower index of element $\beta \in \lambda^k$. If the routing is a simple path, i.e. from every station only one other station is reachable (graph G is a directed acyclic graph), then the denominator of Equation 1 is exactly the number of finished products.

By using the above-defined p_{ij} probabilities, sequences λ^k are updated w.r.t. the predefined routing line. By running through λ^k, whenever a false step is found, a Bernoulli trial with probability $1 - p_{a_{kt}a_{kt+1}}$ is performed. If the trial is successful, then all later occurrences of the starting zone must be removed from λ^k, therefore the false step is purified into a true step. This process can be imagined as tossing a special coin. This coin says *'stay'* with probability $1 - p_{a_{kt}a_{kt+1}}$, or *'move'* with probability $p_{a_i^k a_{t+1}^k}$. When the result says *'move'* then the jump is accepted and all later occurrences of a_{kt} are removed i.e. going back becomes impossible. However if it says *'stay'* then a_{kt+1} is set to a_{kt} so the state is not changed. With this method, a well defined sequence of movements is obtained. Algorithm 2 summarizes the calculation steps discussed above.

Algorithm 2 Routing path integration

1: **input:** $\left(\left(\tau_t^k, \mathbf{x}_t^k\right)_t^{T_k}\right)_{k=1}^{K}$ coordinates, prefixed routing path
2: Noise filtration (e.g. S–G filter) : $\left(\tau_t^k, \tilde{\mathbf{x}}_t^k\right)_{t=1}^{T_k}$
3: Match $\left(\tilde{\mathbf{x}}_t^k\right)_{t=1}^{T_k}$ points to the nearest workstation
4: λ^k and Λ^k as above
5: p_{ij} probabilities as in (1)
6: **for** k in Products **do**
7:   **for** $t = 1 : (T_k - 1)$ **do**
8:     **if** $a_t^k \neq a_{t+1}^k$ **then**
9:       **if** $\exists r > t : a_r^k = a_t^k$ **then**
10:         Delete all following occurrences of a_t^k with probability of $p_{a_t^k a_{t+1}^k}$
11:         Rewrite $a_{t+1}^k = a_t^k$ with probability of $1 - p_{a_t^k a_{t+1}^k}$
12:       **end if**
13:     **end if**
14:   **end for**
15: **end for**

Note that, in real life cases it often happens that not so many false steps occurs after noise filtration. In those cases, it might be time-saving to consider simply removing those false steps instead, if the removal does not induce a significant amount of data loss.

4.3. Periodic refinement

The core idea behind repeatedly performing the above rezoning method is based on the stochastic nature of the system. As the algorithm highly depends on estimation of probabilities, the reestimation is essential to possess the reliable parameters. Another – maybe even grater – question is how to handle IPS's continuously flowing data stream, and in what way could it be possible to process and use the latest arriving set of coordinates without losing the information that was already gained from previous calculations. Furthermore, the environment of a real-life production system can change overtime which may cause uncertainties in the precision of IPS, therefore a regular refinements are necessary.

As the purpose of periodic refinement is to learn about the recent behaviour of the system without losing previous results, a measure of goodness of rezoning must be defined. This measure is a function (σ) of a chosen KPI, e.g. the cycle time (CT) of assembly stations. The primary requirement for this function is to be able to surely compare the outcome of two rezoning models. If the chosen KPI is known to be almost stable (i.e. the full sample is derived from the same distribution), then the variance might be an effective measuring function. Let us note here, that the σ must be defined individually for every production system, since each of them has its own specifications and conditions.

Algorithm 3 Periodic refinement of probabilities

1: **input:**
2: For the first $K \geq 1$ product: $p_{ij}^0 \triangleq p_{ij}$ from Algorithm 2
3: Let σ_i^0 be the variance of CTs at WS_i, and $\sigma^0 \triangleq \frac{\sum_i \sigma_i^0}{K}$
4: $l \triangleq 1$
5: **while** σ^{l-1} is not sufficiently small **do**
6: For the next K products: $\tilde{p}_{ij}^l \triangleq p_{ij}$ from Algorithm 2, and $\tilde{\sigma}^l \triangleq \frac{\sum_i \sigma_i^0}{K}$ as in step 3
7:
$$p_{ij}^l \triangleq \frac{\tilde{\sigma}^l}{\tilde{\sigma}^l + \sigma^{l-1}} p_{ij}^{l-1} + \frac{\sigma^{l-1}}{\tilde{\sigma}^l + \sigma^{l-1}} \tilde{p}_{ij}^l \qquad (2)$$
8: $l+ = 1$
9: Rezone on this set of products as in Algorithm 2 using the new p_{ij}^l probability values
10: Similarly to the step 1 and step 2: $\sigma^l \triangleq \frac{\sum_i \sigma_i^l}{K}$
11: **end while**

In the next section, an experiment of the overall process is presented.

5. Numerical experiments and results

In order to assess the effectiveness of the IPS data processing method described above, here we perform an experiment by using the DES model of the assembly system, which was introduced in the previous section (Figure 1). All calculations were performed by C++. The training data set was obtained by simulating the production within one working shift, which resulted in cca. 18 K data points in the IPS log, stored in the mySQL database. For the sake of comparability, the true cycle times were also exported from the simulation experiments, and by nature, the idle times spent in mid-process buffers are disregarded. During the simulation run, 1000 products were assembled in the target area. The first step of data cleansing is the filtration of the random noise for each and every product. Figure 2 shows the effect of applying the Savitzky-Golay filter (Algorithm1) with parameters $n = 9$ and $p = 2$, and the result of spline fitting. It can be easily observed that without the noise filtration, the collected data might lead to corrupted cycle time calculations.

Then, the smoothed data of the first 100 products was fitted to a predefined routing by following the steps of Algorithm 2. In our case, the process routing is the following: Buffer $\rightarrow$ WS_1 $\rightarrow$ WS_2 $\rightarrow \cdots \rightarrow$ WS_{17} $\rightarrow$ OutBuffer. The Rework zone is only visited in certain cases, and it is located between WS_{14} and WS_{15}. Then, for the rest of the IPS data set, Algorithm3 is run with the period of $K = 100$ products.

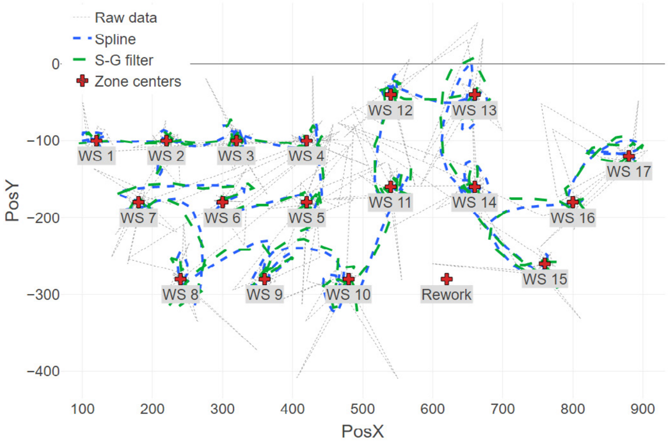

Fig. 2. Different filtering methods

A fine enough approximation of the cycle times at the workstations is of crucial importance in the scope of leadtime prediction models. To analyze the accuracy of our method, we estimated the cycle times from the raw, the once processed and the 7-times processed data as well. Considering the mean absolute error (MAE) as the measure of comparison, the quartiles of multi-cleaned data's MAE were closer to zero than those of the raw data's, almost everywhere. This phenomenon corresponds to our vision, according to which repeated data cleansing develops more precise approximation of cycle time (Figure 3). At every workstation, the approximation based on processed data produces lower MSE than that based on raw data, except for one stations, WS_1. This anomaly can be explained by the behaviour of the DES software: even before an item is logged into the first station, it has already appeared in the system as a floating object.

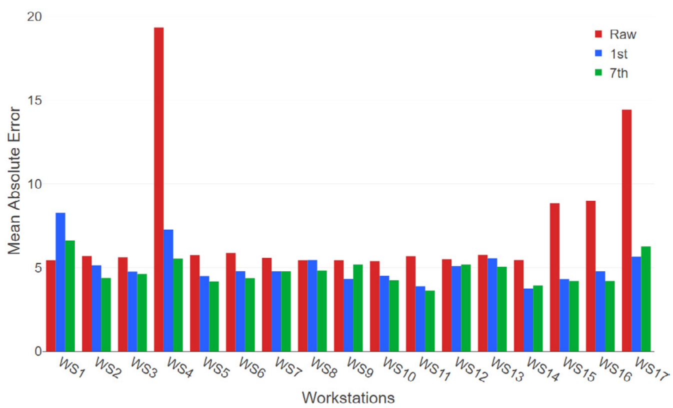

Fig. 3. Mean Absolute Error of each WS's CT estimation, without any data processing (red), after rezoning once (blue) and after refining 7 times (green)

6. Conclusions and future work

Performing the numerical comparison of the IPS calculations based on raw, filtered and repeatedly processed data, let us summarize the main benefits of the above described algorithms in production management.

In industrial environments, viability of advanced IoT applications is determined by the business value that they can bring. Similarly to any IoT data analytics application, the garbage-in-garbage-out law holds, namely, the right conclusions cannot be drawn of an analytics project, in case of unrealistic or unreliable input data is applied. In case production engineers aim at improving the processes based on the pre-calculated utilization rates and cycle times, only the realistic ones of those will provide a good starting point for the improvement. Similarly, if line balancing or scheduling problems are solved based on a set of parameters provided by an IPS analysis, the structure of the optima solution (e.g., a line balance) can really much depend on the accuracy of the considered cycle times. Conclusively, it is worth to implement and apply advanced analytical methods in IPS calculations, as they provide more reliable process parameters, than those calculated from the raw location data.

As for the future work, the authors plan to further enhance the applied methods to increase the overall accuracy of the analytics. Furthermore, a more comprehensive benchmark of filtering and smoothing algorithms is planned to be performed, with the aim of assessing their accuracy in production environments, considering various different assembly and machining shop-floor configurations.

A major part of the future work relates to the predictive analytics domain, including the prediction of manufacturing lead times, makespans of various production sequences or resource allocation rules. In this way, predictive analytics results could be integrated directly in the decision making processes, so as making a step towards the so-called prescriptive production management.

Acknowledgement

This research has been supported by the European H2020 EPIC grant under No. 739592 and the GINOP-2.3.2-15-2016-00002 grant of Hungary. The authors would like to thank the MTA Cloud (`https://cloud.mta.hu`) for providing the cloud infrastructure of the project that was used to implement the analytics applications and perform the computational experiments.

References

[1] Alarifi, A., Al-Salman, A., Alsaleh, M., Alnafessah, A., Al-Hadhrami, S., Al-Ammar, M., Al-Khalifa, H., 2016. Ultra wideband indoor positioning technologies: Analysis and recent advances. Sensors 16 (5), 707.

[2] Antonucci, A., Corrà, M., Ferrari, A., Fontanelli, D., Fusari, E., Macii, D., Palopoli, L., 2019. Performance analysis of a 60-ghz radar for indoor positioning and tracking. In: 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN). IEEE, pp. 1–7.

[3] Benkouider, Y. K., Keche, M., Abed-Meraim, K., 2013. Divided difference kalman filter for indoor mobile localization. In: International Conference on Indoor Positioning and Indoor Navigation. IEEE, pp. 1–8.

[4] Boulekrouche, B., Haddad, H., Jabeur, N., Alimazighi, Z., 2015. Integrating multiple geometric representations within spatial data warehouse structures for enhanced collaborative decision-making processes. In: FNC/MobiSPC. pp. 566–572.

[5] Chambers, J. M., 1991. Statistical models in S. CRC Press, Inc.

[6] Daras, G., Agard, B., Penz, B., 2018. A spatial data pre-processing tool to improve the quality of the analysis and to reduce preparation duration. Computers & Industrial Engineering 119, 219–232.

[7] Doucet, A., Godsill, S., Andrieu, C., 2000. On sequential monte carlo sampling methods for bayesian filtering. Statistics and computing 10 (3), 197–208.

[8] Fetzer, T., Ebner, F., Deinzer, F., Köping, L., Grzegorzek, M., 2016. On monte carlo smoothing in multi sensor indoor localisation. In: 2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN). IEEE, pp. 1–8.

[9] Galov, A., Moschevikin, A., 2013. Bayesian filters for tof and rss measurements for indoor positioning of a mobile object. In: International Conference on Indoor Positioning and Indoor Navigation. IEEE, pp. 1–8.

[10] Hershberger, J., Shrivastava, N., Suri, S., 2006. Cluster hull: A technique for summarizing spatial data streams. In: 22nd International Conference on Data Engineering (ICDE'06). IEEE, pp. 138–138.

[11] Huang, S., Guo, Y., Zha, S., Wang, F., Fang, W., 2017. A real-time location system based on rfid and uwb for digital manufacturing workshop. Procedia Cirp 63, 132–137.

[12] Jedari, E., Wu, Z., Rashidzadeh, R., Saif, M., 2015. Wi-fi based indoor location positioning employing random forest classifier. In: 2015 international conference on indoor positioning and indoor navigation (IPIN). IEEE, pp. 1–5.

[13] Li, K.-J., Zlatanova, S., Torres-Sospedra, J., Perez-Navarro, A., Laoudias, C., Moreira, A., 2019. Survey on indoor map standards and formats. In: 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN). IEEE, pp. 1–8.

[14] Liu, H., Darabi, H., Banerjee, P., Liu, J., 2007. Survey of wireless indoor positioning techniques and systems. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 37 (6), 1067–1080.

[15] Lu, S., Xu, C., Zhong, R. Y., Wang, L., 2017. A rfid-enabled positioning system in automated guided vehicle for smart factories. Journal of Manufacturing Systems 44, 179–190.

[16] Nurminen, H., Ristimäki, A., Ali-Löytty, S., Piché, R., 2013. Particle filter and smoother for indoor localization. In: International Conference on Indoor Positioning and Indoor Navigation. IEEE, pp. 1–10.

[17] Salamah, A. H., Tamazin, M., Sharkas, M. A., Khedr, M., 2016. An enhanced wifi indoor localization system based on machine learning. In: 2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN). IEEE, pp. 1–8.

[18] Savitzky, A., Golay, M. J., 1964. Smoothing and differentiation of data by simplified least squares procedures. Analytical chemistry 36 (8), 1627–1639.

[19] Stephan, P., Heck, I., Krau, P., Frey, G., 2009. Evaluation of indoor positioning technologies under industrial application conditions in the smartfactorykl based on en iso 9283. IFAC Proceedings Volumes 42 (4), 870–875.

[20] Su, X., Khoshgoftaar, T. M., 2009. A survey of collaborative filtering techniques. Advances in artificial intelligence 2009.

[21] Wen, H., Clark, R., Wang, S., Lu, X., Du, B., Hu, W., Trigoni, N., 2018. Efficient indoor positioning with visual experiences via lifelong learning. IEEE Transactions on Mobile Computing 18 (4), 814–829.

[22] Whittaker, E. T., Robinson, G., 1940. Calculus of observations: Treatise on Numerical Mathematics 3rd Edition. Blackie and Sons Limited, London.

[23] Widenius, M., Axmark, D., Arno, K., 2002. MySQL reference manual: documentation from the source. " O'Reilly Media, Inc.".

[24] Yang, H., Zhang, Y., Huang, Y., Fu, H., Wang, Z., 2019. Wknn indoor location algorithm based on zone partition by spatial features and restriction of former location. Pervasive and Mobile Computing 60, 101085.

[25] Zoubert-Ousseni, K., Villien, C., Le Gland, F., 2016. Comparison of post-processing algorithms for indoor navigation trajectories. In: 2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN). IEEE, pp. 1–6.