Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

SEMANTIC ENRICHMENT OF SPATIO-TEMPORAL PRODUCTION DATA TO DETERMINE LEAD TIMES FOR MANUFACTURING SIMULATION

Carina Mieth

TRUMPF Werkzeugmaschinen GmbH & Co. KG
Johann-Maus-Str. 2
Ditzingen, 71254, GERMANY

ABSTRACT

Data from real-time indoor localization systems (RTILS) based on ultra-wideband (UWB) technology provide spatio-temporal information on the material flows of production orders on the shop floor. This paper investigates how historical position data can be used for the determination of lead times and respective time shares. We propose three different approaches for the enrichment of spatio-temporal trajectories with process information. Two of them are online algorithms for the automated posting of process times using either points or areas of interest. The third is an offline classification problem that minimizes the error that occurs during the assignment of measurements to processes when generating semantic trajectories. Furthermore, a sensor fusion concept is presented, which is necessary to split up the lead times of the operations in smaller time shares for simulation input modeling.

1 INTRODUCTION

High demands must be placed on the provision of trustworthy data for simulation input modeling because the success of a simulation study relies on the quality of the simulation's input data (Wenzel et al. 2007). Unfortunately, information gathering and processing still takes a lot of time in manufacturing simulation studies as necessary interviews, time studies and inspections of the production can be very time-consuming. In addition, it must always be doubted whether the collected values are representative since, for example, the subjective impression of the interviewee may have a bias or the observed workers behave differently than usual. With the increasing dissemination of cyber-physical system (CPS) in manufacturing, there will be a valuable proposition of new data sources that can be exploited for simulation input modeling. These CPS are considered as an essential element of the digital twin concept as they connect the physical and the digital world (Yang et al. 2017; Shao and Kibira 2018; Srewil and Scherer 2017) and thus are useful to keep simulation models close to reality. In this work, the considered CPS is a real-time indoor localization systems (RTILS) based on ultra-wideband (UWB) technology.

In Mieth et al. (2019), we presented a framework for the determination of simulation inputs from data of RTILS within a cyber-physical production system and showed how this contributes to an improvement of data quality in manufacturing simulation studies. For each simulation input, it was elaborated whether it can be derived from the RTILS data with or without additional context information about e.g., process plans or the shop floor layout. The focus in this work is on the determination of lead times from the spatio-temporal trajectories of the tracked production orders by enriching semantic process information to the trajectories. Semantic enrichment is the process of annotating spatio-temporal trajectories with meaningful context information (Arslan et al. 2018). These annotated trajectories are called semantic trajectories (Zheng 2015; Wu et al. 2015) and are of great use for data mining purposes. Jensen et al. (2009) emphasize that it is of interest to apply data mining techniques to indoor trajectories and Nikitopoulos et al. (2018) stress that the analysis of "spatio-temporal data has the potential to discover hidden patterns or result in non-trivial insights". For applications in manufacturing, the context information that can be enriched to the spatio-temporal trajectories are e.g., process plans, layout information like an affiliation to points and areas of interest or the part's geometry (Mieth et al. 2019).

This paper is structured as follows: In Section 2, the related work is presented. In Section 3, RTILS are shortly introduced and their data is characterized as spatio-temporal trajectories. In Section 4, we propose three approaches to determine lead times by means of semantic annotation of the spatio-temporal trajectories of the RTILS. The semantic annotation of the production orders' trajectory can be solved in real-time (see Subsections 4.1 and 4.2) or after the completion of the production order, when the whole trajectory is known (see classification problem in Subsection 4.3). We illustrate the differences of the proposed algorithms with a small example in Subsection 4.4. In Subsection 4.5, a sensor fusion concept for the calculation of smaller time shares is presented.

2 RELATED WORK

Yan et al. (2013) emphasize that techniques for higher level and semantic events inferred from raw GPS-like data should be developed (GPS = global positioning system). An example for such a semantic event in the manufacturing domain would be 'production order x from company y is ready for processing at station z'. Besides, they mention that the annotation could also include a recording of the respective activity, speed, or means of transportation of the moving object.

Zheng (2015) classified trajectory data mining applications in four major categories, that are the analysis of movements of people, transportation vehicles, animals and natural phenomena. Although the trajectories of a RTILS are similar to GPS trajectories, manufacturing was not named as an area of application, probably because the author unconsciously limited himself to outdoor applications. Richly (2018) developed a trajectory data mining framework and analyzed different concepts to store, compress, index, and process spatio-temporal data. He identified four key challenges, which are: the data volume, the high update rate, the query latency of analytical queries, and the inherent inaccuracy of the data. The latter is the most important challenge in this work since incorrect measurements must not affect the calculation of lead times. For this, the pre-processing of the data plays a major role.

Fazzinga et al. (2016) incorporated domain knowledge in form of three types of integrity constraints into the cleaning task of radio-frequency identification (RFID) indoor data. The first constraint is the "direct unreachability constraint" that uses an indoor model to check whether it is possible to go from one room directly to the next one. The second constraint is a "traveling-time constraint", that ensures that a specific travel speed is not exceeded between two consecutive position measurements. The third constraint is referred to as "latency constraint" and guarantees that a stay at a site has lasted long enough to detect at least a certain number of measured positions at that site.

There are some publications about the analysis of RFID data for the purpose of determining times, e.g., Zhong et al. (2014) mined operating times from RFID-enabled production data and Srewil and Scherer (2017) enriched RFID-data with location context for construction sites using a Point-In-Polygon test. However, UWB-based RTILS data is different to RFID data, where tagged objects are identified when they are close to a reader and more comparable to spatio-temporal data from the GPS. Rashid et al. (2017) and Arslan et al. (2018) analyzed spatio-temporal GPS-data of workers in construction sites with the aim to generate location-based safety alerts and enhance safety management. Rashid et al. (2017) divided the trajectories in equally sized parts for a K-means clustering to find similarities. Then, they trained a hidden markov model for trajectory prediction which is used to calculate a risk index for workers. The approach to split trajectories in equally sized parts is disadvantageous as meaningful patterns usually do not have the same length and can be longer than the size of the parts (Fu 2011). In our case, it is not feasible to split the trajectories in equally sized parts as process times differ widely for different operations and the length of the trajectories' part depends on the process time.

Li et al. (2008) developed an algorithm that uses human GPS motion data to determine whether a person has been at a specific location, such as an attraction or sight. Their algorithm first checks whether the distance between a measured point and its successors in a trajectory is greater than a specified threshold. If this is the case, the time span between the measured point and the last successor within the threshold is calculated. If the time span is also greater than a specified time threshold, a stay at a specific location is detected. An advantage of this approach is, that it does not need a specification of the specific location in advance, which means, that it can be used for the determination of points of interest. Nevertheless, the points determined in this way lack any semantic information. According to de Graaff et al. (2016), most points of interest are detected through approaches that are very similar to the one of Li et al. (2008). Furthermore, they say that it is less common, that the set of points of interest is given.

Alvares et al. (2010) presented an algorithm for spatio-temporal trajectories from GPS-data that is called "Intersection-Based Stops and Moves of Trajectories (IB-SMoT)". The trajectories are split up whenever the border of an area of interest was crossed. Since the areas of interest are scattered over the map, the trajectory is divided into parts that correspond either to a stay at the area or to the transition of a person from one area to another. de Graaff et al. (2016) criticizes that this algorithm is not robust against distortion of the GPS-signal and that the accuracy of the signal is ignored. Another algorithm for the semantic trajectory annotation is the "Clustering-Based Stops and Moves of Trajectories" (CB-SMoT) algortihm from Palma et al. (2008) . The algorithm has a clustering parameter that is adaptively adjusted for each trajectory via a quantile function based on the input area. Thus, trajectories of fast moving persons have larger distance thresholds than those of slow moving persons. Besides, there is the input of a minimum time threshold which must be exceeded to form a cluster in the trajectory. The identified clusters in the trajectory are then mapped to the known areas of interest (polygons). Rocha et al. (2010) considered frequent changes of the direction of a trajectory as a reference to a stop. Their algorithm is named "Direction-Based Stops and Moves of Trajectories" (DB-SMoT) and has three inputs, which are the minimum direction change, the minimum time interval for building a new cluster and a maximum number of measurements inside a cluster for which the minimum direction change is not exceeded.

It is more difficult to determine semantic trajectories of indoor rather than outdoor movements under uncertainty since the points and areas of interest in a production system are closer to each other, adjacent, or even overlapping, and not as far apart as the private house, school, and supermarket. To the best of our knowledge, there is no approach for the annotation of context information about the production process to the position data from an UWB-based RTILS available, which necessitates the development of new approaches that are suitable for the lead time determination in manufacturing. Another decisive difference to earlier work on semantic enrichment for GPS data is that explicit or implicit knowledge about the production process is available for manufacturing and this knowledge should be incorporated profitably into the analysis of spatio-temporal data from UWB-based RTILS.

3 SPATIO-TEMPORAL TRAJECTORIES FROM UWB-BASED RTILS

Real-time indoor localization systems using ultra-wideband consist of mobile devices, which are attached to objects and a sensor network, that makes up the reference system for localization (see gray parts in Figure 1). The positioning of objects is carried out via a runtime method of the radio signals to at least three satellites. UWB is a short-range radio technology that uses extremely large frequency ranges with a bandwidth of at least 500 MHz. It works with a low transmission power to avoid interference with already occupied frequency ranges. Thanks to this robustness and its high data rate, it is particularly suitable for precise indoor positioning.

In the production scenario, RTILS were developed for the precise localization of production orders on the shop floor to reduce search times and to replace the old-fashioned accompanying documents. For this purpose, a mobile device with an e-ink display is attached to each production order at the beginning of the production process and sticks to it until the production process is completed. The positions of production orders are recorded over time according to the RTILS sampling rate, but not necessarily equidistantly because if no movement is detected, the mobile devices switch to sleep mode where no positions are sent. This position data is very valuable for various applications like for example indoor navigation (Scholz and

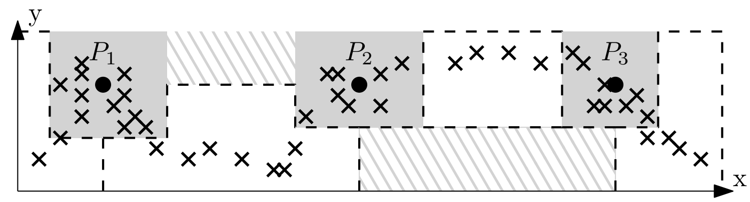

Figure 1: Exemplary trajectory of a production order within the shop floors' coordinate system (x,y).

Schabus 2014), process transparency (Uckelmann and Wendeberg 2015) and material flow optimization (Richly 2018). An overview on the potentials of data from indoor-localization systems in production can be found in (Mieth et al. 2019).

The data from the RTILS contains spatio-temporal trajectories for each production order o_j. These trajectories are two-dimensional time series $\underline{S}_j(t)$ of positions $\underline{s}_j(t) = (x_j(t), y_j(t)) \in \mathbb{R}^2$ measured at T moments in time: $\underline{S}_j(t) = (\underline{s}_j(t_1), \underline{s}_j(t_2), \ldots, \underline{s}_j(t_T))$. The elapsed time for a particular process can easily be calculated by the difference of two points in time. The only difficulty is to determine these points in time, so-called events, from the spatio-temporal trajectories.

In Figure 1, there is an exemplary trajectory of a production order. The black crosses are the measured positions $\underline{s}_j(t)$ at different times. For reasons of clarity, the measured positions were not connected with each other according to their temporal sequence. The hatched areas represent separate zones in the coordinate system (production layout), where the production order cannot be located due to architectural restrictions or similar. The gray rectangles are areas of interest A_i, where production orders are processed. Therefore, these areas of interest also contain the points of interest P_1, P_2 and P_3. The dashed rectangles are zones in which storage or transportation takes place.

4 DETERMINATION OF LEAD TIMES WITH SPATIAL REFERENCE

The terms used and their relations to each other are graphically represented in an ontology for the manufacturing domain in Figure 2. For the derivation of lead times, the spatio-temporal trajectories from the RTILS need to be divided into parts that correspond to specific places on the shop floor like points of interest (stay points) or areas of interest (geo-fences) that the production order has visited. These parts are called *segments* and are by definition non-empty, disjunctive sets of measurements from the RTILS, which can also be arranged in a chronological order.

For the sake of simplicity, it is assumed that the allocation of operations to locations is unambiguous. For this, all N areas of interest A_i are disjunctive $\cap A_i = \emptyset$, $\forall i = 1 \ldots N$. There is exactly one point of interest P_i in each area of interest A_i. As a consequence, the entire shop floor space that can be visited by the production order is the union of the individual areas of interest and no two points of interest can be at the same position. To be able to determine the segments of the trajectories, the respective point or area of interest must be defined in a way, that the operation are locally demarcated. For the algorithms

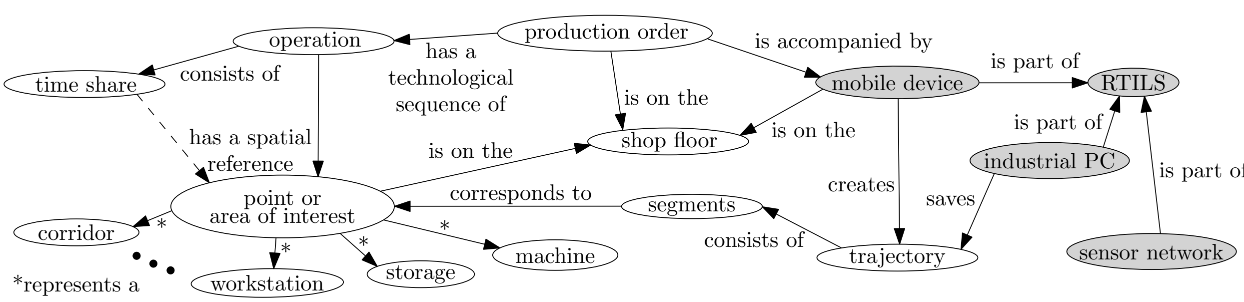

Figure 2: Manufacturing domain ontology for the semantic enrichment of trajectory from RTILS.

presented, it is assumed that the information about the boundaries of the areas of interest or the coordinates of the points of interest are given. The algorithms enrich the trajectories with semantic information about the visited points or areas of interest. The result is a semantic trajectory $\underline{S}_{a,j}$ for each production order, which can then be split into segments, whenever its spatial reference has changed. Once these segments are known, the process times can be easily calculated by forming the difference between the start and end time of the trajectory's segment. At this abstraction level, the lead time can be determined for each process that has an unambiguous spatial reference i.e., for each operations' transportation time and waiting plus operating time. For a finer subdivision of the time shares, a sensor fusion concept will be introduced later in Subsection 4.5.

The complicated step of determining lead times is the allocation of position measurements from the trajectory to the respective points or areas of interest. If all measurements are correct, a measurement could simply be either allocated to the nearest point of interest or to the area of interest containing that measurement. Following this intuitive approach, we found that the resulting semantic trajectory is not suitable for the calculation of time shares of lead times. This is due to the inaccuracy of the measurements. In the proximity of the boundaries of areas of interest and between two points of interest, the assignment to a process changes much more frequently than the underlying process would allow. Richly (2018) also stressed that a huge challenge with spatio-temporal data is the inherent inaccuracy. The trajectories are not smooth and contain outliers caused by measurement errors of the RTILS. In the following, two algorithms are presented that segment trajectories for automated posting. For this, the semantic annotation must work reliably in real-time, which is particularly challenging as faulty measurements need to be filtered online.

For a more reliable assignment of the measurements to the points or areas of interest, technological restrictions will be considered in the form of logical operation sequences. All possible operation sequences within the production system are part of a directed graph $G = (V, Arc)$. Each node V_i in the node set V represents an operation that is performed at the point or area of interest P_i or A_i, respectively. The arc set Arc contains all directed arcs (V_a, V_b) that always contain two logically sequentially executable production operations. A path $G_j \in G$ corresponds to a sequence of operations that a production order o_j passes through. Let $G^\star = (V^\star, Arc^\star)$ be an extension of G_j that results by adding edges that join each vertex of the path to itself (loops).

4.1 Algorithm for Online Semantic Annotation Based on the Distance to Points of Interest (POI)

A point of interest P_i refers to a location (x_i, y_i) on the shop floor that has a semantic meaning e.g., a machine or a workplace. An advantage of points of interest over areas of interest is, that the definition of points is easier than to define areas on the shop floor because the latter requires a clear demarcation of the borders in advance. The idea of Algorithm 1 is to use the distance to a point of interest as discriminator for the allocation of a measurement. In Figure 3, a segment of an exemplary trajectory from a production order o_j is given with two points of interest P_1 and P_2. The gray lines from each measured position in the trajectory to the points of interest is calculated via a distance metric $d_{o_j,P_i}(t)$. Here, the euclidean distance $d_{o_j,P_i}(t) = \sqrt{(x_i - x_j(t))^2 + (y_i - y_j(t))^2}$ in $\mathbb{R}^2$ was used. To avoid a division by zero, if the measured point is exactly equal to the defined point of interest and to avoid that the fraction for distances $<< 1$ goes

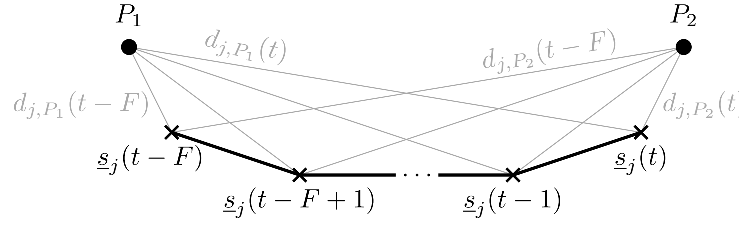

Figure 3: Points of interest P_1 and P_2 with the trajectory of a production order o_j.

```
input  : spatio-temporal trajectories S_j for each production order o_j, window size F, weighting
         factors w_1,...,w_F, points of interest P_i ∀i = 1...N
output: semantic trajectory S_{a,j}, event log

for each t = 1...T do
    check plausibility constraints for measurement s_j(t)
    for each production order o_j do
        for each point of interest P_i do
            if t ≤ F then
                R_j(t,P_i) := 1 / d_{o_j,P_i}(t) ;              // initilization of first ratings
            else
                R_j(t,P_i) := 1 / d_{o_j,P_i}(t) + Σ_{f=1}^{F} 1 / (w_f d_{o_j,P_i}(t−f)) ;   // rating function
        assign P_i with max{R_j(t,P_i)} to s_{a,j}(t) ;          // chose most likely P_i
        if P_i(t) ≠ P_i(t−1) then
            save timestamp to eventlog ;                         // location has changed
return S_{a,j}(t), event log
```

Algorithm 1: Pseudocode for the allocation of measurements to points of interest (POI).

against plus infinity, we set every distance to one if $d_{o_j,P_i}(t)$ is smaller than one. This can be envisioned as if the point of interest has an area with radius one within which all measurements are set to distance one. It is also possible to use other distance metrics than the euclidean distance, if it turns out that it is too weakly discriminatory.

All measurements that are not within the shop floor boundaries and which exceed a realistic distance (i.e., distance that could be traveled with maximum speed) to the previous measurement are deleted (plausibility constraints). Since for noise filtering of GPS data, mean or median filter, Kalman or Particle filters and heuristics-based outlier detection have been proven to be successful (Zheng 2015), our algorithm uses a weighted mean filter based upon a rating function $R_j(t,P_i)$ for production order o_j at each point of interest P_i at time t. The rating function $R_j(t,P_i)$ was designed in a way, that its value increases for shorter distances to a point of interest and that older measurements account for less than newer ones.

The weighting factors $w_1,\dots,w_F$ of the mean filter should be chosen descending $w_1 \geq w_2 \geq \dots \geq w_F \geq 1$. Note that the case, in which all weighting factors are equal to one, corresponds to an average filtering with the window size $F+1$. The parameter window size $F \in \mathbb{N}$ of the filter delays the assignment to a point of interest by $\frac{F}{2}$ for all odd F and by $\lfloor \frac{F}{2} \rfloor$ for all even F for the mean filter with equal weights. Note that the time interval between two events remains unaffected. For the determination of lead times, the window size can therefore be larger than for automatic posting since for the latter, the delay must be so small that it is not noticeable during operation.

With this algorithm, a pseudo probability for allocation to a points of interest can be calculated by relating each current rating to the sum of all ratings at the time. Note that the positions of the points of interest have an influence on the accuracy of the algorithm and that the usage of the euclidean distance is not a strong classifier if production areas around the points of interest are not equally big.

4.2 Algorithm for Online Semantic Annotation Based on Areas of Interest (AOI)

An area of interest A_i, sometimes also called geo-fence, refers to an arbitrarily shaped area on the shop floor that has a semantic meaning e.g.; the work area around a machine or a storage. The idea of Algorithm 2 is to allocate a measurement to an area of interest if the measured coordinates are inside that area. The discriminator is the point-in-area (PIA) test that checks if the measured point (x,y) is in A_i:

$$PIA((x,y),A_i) = \begin{cases} 1 & \text{inside} \\ 0 & \text{on border} \\ -1 & \text{outside} \end{cases} \tag{1}$$

For Algorithm 2, a mean filter with a window size F is used. If the sum of all values returned by the PIA-test inside the window is greater or equal zero, then a location change is detected. The window size plays an important role since it is the only design parameter of this algorithm. As with the previous algorithm, the window size must be chosen small enough to guarantee a quick response for automated posting but huge enough to filter out faulty measurements.

The blue loops in both Algorithm 1 and 2 denote the progression in time and can be neglected for the implementation for an online scenario, but we included it since it can be used to find an initial solution for the classification problem in Subsection 4.3. To improve the run-time of both Algorithm 1 and 2, the selection of points or area of interest can be restricted to a subset during the loop over all points or areas of interest, respectively. For example, all points or areas of interest that have not yet been visited can be used as a subset. Furthermore, production orders that rest at the time (in sleep mode) can also be neglected in the first loop over all production orders.

```
input : trajectories S_j for each production order o_j, window size F, disjoint areas of interest
        A_i ∀i = 1...N
output: semantic trajectory S_{a,j}, event log

for each t = 1...T do
  check plausibility constraints for measurement s_j(t)
  for each production order o_j do
    for each area of interest A_i do
      if (t > F) then
        if PIA(s_j(t), A_i) = 1 then
          if A_i(t) ≠ A_i(t−1) then
            if Σ_{f=1}^{F} PIA(s_j(t−f), A_i) ≥ 0 then
              save event at t − ⌊F/2⌋ to eventlog ;       // location has changed
              A_i to s_{a,j}(t) ;                          // assign measurement
              A_i to s_{a,j}(t−1)...s_{a,j}(t−⌊F/2⌋) ;     // update previous ones
            else
              assign previously identified area of interest A_i at (t−1) to s_{a,j}(t)
          else
            assign previously identified area of interest A_i at (t−1) to s_{a,j}(t)
        else if PIA(s_j(t), A_i) = 0 then
          assign previously identified area of interest A_i at (t−1) to s_{a,j}(t)
      else
        if PIA(s_j(t), A_i) = 1 then
          assign area of interest A_i to s_{a,j}(t);       // initilization
return S_{a,j}(t), event log
```

Algorithm 2: Pseudocode for the allocation of measurements to areas of interests (AOI).

4.3 Formulation of the Trajectory Segmentation as a Classification Problem (CP)

We describe the segmentation of spatio-temporal trajectories $\underline{S}_j$ from production orders o_j as a classification problem (CP). Each trajectory segment belongs to a time share of the orders' lead time that is of interest for simulation input modeling. Thus, the trajectory is enriched with semantic information about the process, such that the time share can be calculated by forming the difference of two events (i.e., change of semantic meaning).

For the classification problem, the spatio-temporal trajectory $\underline{S}_j$ of a production order o_j is given. The initialization of a first semantic meaning can be determined using the presented Algorithms 1 and 2 with the blue loops or simply by just using the PIA-test from (1) or by assignment to the nearest point of interest. When using the PIA-test, the resulting semantic trajectory will have zero assignment errors according to (2), but will violate the constraints from (6) and (7).

Let k_t be the decision variable that indicates for each position measurement $\underline{s}_j(t))$ to which class $V_t \in V$ it belongs. The classes are defined in a one-to-one correspondence to the points or area of interest, respectively. This results in $|V| = N$. $PIA^\star(t) \in V^\star$ is a function that returns the class affiliation in which the recent measurement $s_j(t) = (x(t), y(t))$ is contained. In the following, two cases can be distinguished: It can either be assumed that all measurements are equally trustworthy (case I), which leads to the error function

$$e_j(t) = \begin{cases} 0 & PIA^\star(t) = k_t \\ 1 & PIA^\star(t) \neq k_t \end{cases} \tag{2}$$

or that each measurement has its own probability $p(t)$ that the given assignment based on the measured coordinates (x,y) is correct (case II), which leads to the following error function

$$e_j(t) = 1 - p(t). \tag{3}$$

The error function in the first case of equal probabilities either returns zero for a correct assignment of a measurement or one for a wrong assignment of a measurement at time t. In the second case, the probability is given as an output of the RTILS system. The total assignment error is in both cases calculated as the sum of all return values of the error function over time. Further, the following transition function u is defined

$$u(k_t, k_{t+1}) = \begin{cases} 0 & \text{no change detected: } k_t = k_{t+1} \\ 1 & \text{change detected: } k_t \neq k_{t+1} \end{cases} \tag{4}$$

that checks whether a class change has taken place between k_t and k_{t+1}. Such a class change always corresponds to an event, whereby an event always takes place if the status of the order has changed (order moved to next process step). Solving the classification problem, the assignment error should be minimized

$$\underset{k_t \in V^\star}{\text{minimize}} \sum_{t=1}^{T} e_j(t) \tag{5}$$

with respect to the following two constraints

$$(k_t, k_{t+1}) \in Arc^\star, \tag{6}$$

$$\sum_{t=1}^{T-1} u(k_t, k_{t+1}) = 2 \cdot |V^\star| - 1. \tag{7}$$

The first constraint in (6) ensures that the sequence of processes is not violated. Therefore, there must always be an arc (k_t, k_{t+1}) in the arc set $Arc^\star$ for two consecutive measurements. This constraint was the reason why the loops were added to the path that represents the feasible sequence of production operations at the beginning of Section 4 because they are needed for consecutive measurements inside the segments.

The second constraint in (7) is for the segmentation of trajectories in the number of allowed segments, which is equal to the sum of the number of processes $|V^{\star}|$ plus the number of transports $|V^{\star}|-1$. A possibly good heuristic for this integer linear program is to first divide the trajectory into $|V^{\star}|$ equally sized segments and then iteratively shift the boundaries until the total assignment error is minimized.

Sometimes a production orders needs to be reworked. This can be recognized by the fact that the assignment error is higher than usual. Such trajectories should be analyzed separately because the constraint in (6) would not allow for that. For the few trajectories with rework, the number of classes in (7) must be increased if a clear distinction between time for rework and initial work time is desired.

4.4 Comparison of the Algorithms

The algorithms from Subsection 4.1 - 4.3 are now compared using as example the trajectory in Figure 4. All measurements are indicated with black crosses in the coordinate system (x,y) and can also be found in Table 1. Note that due to measurement errors, the measurements do not correspond exactly to the actual locations, where the job was located. Looking at Figure 4, it is obvious that none of the plausibility constraints is violated as all measurements lie within the shop floor's boundaries and there are no unrealistic jumps of the measurements.

On the right side of Figure 4, there are two process graphs that result, when the measurements are used within the different algorithms. The graph at the top is not cycle-free which is a major problem for further analysis as this does not represent the underlying sequential process. Graphs with cycles occur when the PIA-test is used (measurements always jump at the areas' boarders) or if the window size is chosen too small in the Algorithm 1 and 2. In general, only the solution of the classification problem yields a cycle-free graph. In the given example, the window size was chosen to be $F=2$.

Table 1: Comparison of the algorithms using the examplary trajectory from Figure 4.

time t	(x,y)	**PIA**	**POI**	$R_j(t,P_A)$	$R_j(t,P_B)$	**AOI**	$\mathbf{CP_I}$	e	p_A or p_B	$\mathbf{CP_{II}}$
1	(3,1)	A	A	0,50	0,19	A	A	0	0,9	A
2	(3,2)	A	A	1,00	0,20	A	A	0	0,95	A
3	(4,2)	A	A	1,37	0,40	A	A	0	0,8	A
4	(6,2)	B	A	1,00	0,63	A $\rightarrow$ B	B	0	0,51	A
5	(6,3)	B	B	0,73	0,80	B	B	0	0,55	A
6	(5,4)	A	A	0,719	0,715	B	B	1	0,7	A
7	(7,4)	B	B	0,58	1,03	B	B	0	0,7	B
8	(8,4)	B	B	0,47	1,46	B	B	0	0,8	B
9	(8,5)	B	B	0,37	2,74	B	B	0	0,9	B

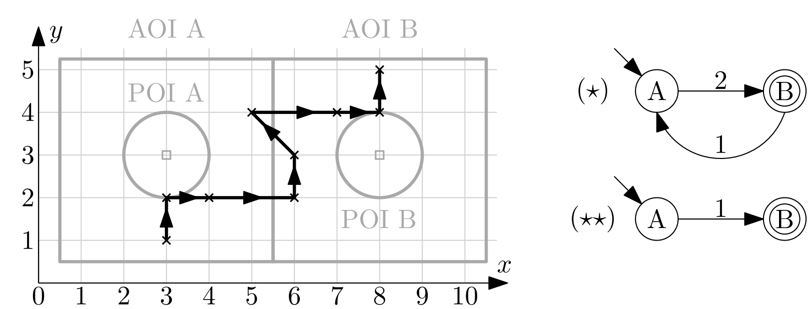

Figure 4: *on the left:* illustration of an exemplary trajectory traveling from A to B; *on the right:* $(\star)$ mined process graph with cycle using PIA-test or POI-algorithm, $(\star\star)$ mined cycle-free process graph using AOI-algorithm or classification problem (CP).

In Table 1, there are the resulting assignments for each algorithm and additional information. For the Algorithm 1 (POI) in Table 1, it can be seen that the window size was to small and that this results in a graph that is not cycle-free (see ($\star$) in Figure 4). For the Algorithm 2 (AOI) in Table 1, it can be seen that the window size was sufficiently chosen which leads to a cycle-free graph (see ($\star\star$) in Figure 4). The arrow in the table means that decision was changed at the fourth measurement. First the measurement was assigned to A, which was updated to B in the next iteration. The estimated lead times are $A = 4$ time units (TU) and $B = 5$ TU by using PIA-test, $A = 5$ TU and $B = 4$ TU by using POI-algorithm, $A = 3$ TU and $B = 6$ TU by using AOI-algorithm or solving the assignment problem with equally probabilities and $A = 6$ TU and $B = 3$ TU by solving the assignment problem with given probabilities. The sum of all assignment errors assuming equally probabilities (CP_I) is 1 or $2,21 = (0,1+0,05+0,2+0,51+0,55+0,2)+(0,3+0,2+0,1)$ for the given probabilities p_A or p_B (CP_{II}).

This example should stress the major advantages of the presented classification problem (CP) over the Algorithms 1 and 2, which are that firstly, a resulting cycle-free graph is guaranteed by design of the constraints and secondly, no parameters like the window size or the weighting factors have to be specified in advance. Another drawback of the online Algorithms 1 and 2 is the difficulty to annotate the trajectory, if the production order is moving to a point or area of interest that is not adjacent. In this case, the velocity of the moving production order should be considered additionally, so that a booking to a point or area of interest is not allowed until the production order is resting again. However, assuming that the manufacturing shop floor layout was optimized according to the direction of the material flow, successive process steps should be adjacent in most cases.

4.5 Sensor Fusion Concept

With the previously presented approaches, it is only possible to detect time shares between events (see black arrows in Figure 5), where the spatial reference has changed. Sometimes, it is also of interest to split these time shares in smaller parts. In Figure 5, the definition from Wiendahl (2014) for a production order's lead time is illustrated. For each operation, there are five time shares: waiting after processing, transportation, waiting before processing, setting-up, processing. Thus, the number of classes for a classification problem that creates a semantic trajectory should be equal to the product of five times the number of operations. The same holds for the definition of the respective points and areas of interest. Using the approaches described in the subsections before, it is possible to determine the transportation time and a cumulative time for the other time shares. But if, for example, the setup times should also be determined, a different

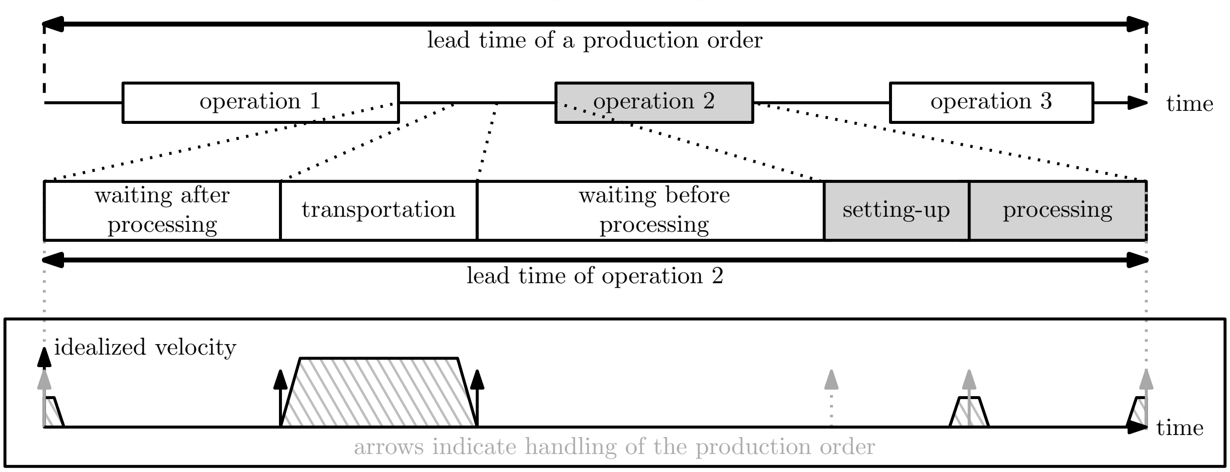

Figure 5: In the upper part, the definition of the lead time of a production order from Wiendahl (2014), (p.260) is illustrated. The lower part is an extension that shows the idealized velocity of the production order and handling events.

approach must be taken, which we describe as a sensor fusion concept. Sensor fusion denotes the process to combine knowledge from different sensor types. In this case, an inertial measurement unit inside the mobile device can be used to identify the events where a specific handling of the production order takes place. Work on activity recognition has already shown that it is beneficial to consider measurements from inertial measurement units to determine information about human movement patterns in production and logistics (Grzeszick et al. 2017). Susanti et al. (2018) for example, use accelerometer and magnetometer measurements from mobile devices for the reconstruction of indoor movements of people. The success of human activity recognition motivates to investigate the same for production orders as well with the aim to identify different types of handling events.

These handling events can separate the trajectory into suitable segments and are indicated as arrows in the lower part of Figure 5. Between setting-up an processing for example, the mobile device needs to be removed necessarily from the tracked production order, otherwise it would be destroyed during processing. Afterwards, the mobile device is reattached to the production order, before it has to wait for transportation. Both of these events (see gray arrows in Figure 5) imply a necessary handling and should be considered to further split the time shares. The gray dotted arrow that marks the event, where the set-up time starts is more difficult to detect since usually the production order does not need any handling. This can be achieved, for example, by using gesture control (e.g., shaking of the mobile device) to show the setup information on a display. This motivates the production employee to trigger the event, which can later be used for segmentation of the spatio-temporal trajectories.

5 CONCLUSION

Three approaches for the semantic enrichment of spatio-temporal trajectories from UWB-based RTILS production data were proposed. The algorithms that have been developed add semantic information about the production process to the individual measurements and thus generate semantic trajectories. Two of the algorithms also work online and can be used to automatically post production progress. Further research will focus on the quantification of classification accuracy and the integration of more contextual information into the algorithms such as velocity, wall positions and uncertainty of the measurements.

In this work, we assumed that the points and areas of interest were precisely known, but the current procedure for production managers is to mark the desired points and areas of interest in the layout plan based on intuition, which is inherently prone to error. Thus, it is of great interest to explore the potential of clustering methods to identify points and areas of interest at and around hot spots. In addition, a discussion with simulation experts is planned to determine what types of times the simulation requires to make an appropriate selection of classes for the segmentation problem.

REFERENCES

Alvares, L. O., A. Palma, G. Oliveira, and V. Bogorny. 2010. "Weka-STPM: From Trajectory Samples to Semantic Trajectories". In *Proceedings of the XI Workshop de Software Livre*. January 13th-15th, Porto Alegre, Brazil, 164-169.

Arslan, M., C. Cruz, and D. Ginhac. 2018. "Understanding Worker Mobility within the Stay Locations Using HMMs on Semantic Trajectories". In *14th International Conference on Emerging Technologies*. November 21st, Islamabad, 1-6.

de Graaff, V., R. A. de By, and M. van Keulen. 2016. "Automated Semantic Trajectory Annotation with Indoor Point-of-interest Visits in Urban Areas". In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*. April 4th-8th, Pisa, Italy, 552-559.

Fazzinga, B., S. Flesca, F. Furfaro, and F. Parisi. 2016. "Exploiting Integrity Constraints for Cleaning Trajectories of RFID-Monitored Objects". *ACM Transactions on Database Systems* 41(4):24.

Fu, T.-C. 2011. "A Review on Time Series Data Mining". *Engineering Applications of Artificial Intelligence* 24(1):164–181.

Grzeszick, R., J. M. Lenk, F. M. Rueda, G. A. Fink, S. Feldhorst, and M. ten Hompel. 2017. "Deep Neural Network Based Human Activity Recognition for the Order Picking Process". In *Proceedings of the 4th International Workshop on Sensor-based Activity Recognition and Interaction*. September 21st-22nd, Rostock, Germany, 14.

Jensen, C. S., H. Lu, and B. Yang. 2009. "Indexing the Trajectories of Moving Objects in Symbolic Indoor Space". In *Advances in Spatial and Temporal Databases*, edited by N. Mamoulis, T. Seidl, T. B. Pedersen, K. Torp, and I. Assent, Volume 5644 of *Lecture Notes in Computer Science*, 208–227. Berlin-Heidelberg, Germany: Springer.

Li, Q., Y. Zheng, X. Xie, Y. Chen, W. Liu, and W.-Y. Ma. 2008. "Mining User Similarity Based on Location History". In *Proceedings of the 16th ACM Sigspatial International Conference on Advances in Geographic Information Systems*. November 5th-7th, Irvine, California, 34.

Mieth, C., P. Humbeck, and G. Herzwurm. 2019. "A Survey on the Potentials of Indoor Localization Systems in Production". In *Interdisciplinary Conference on Production, Logistics and Traffic*. March 27th-28th, Dortmund, Germany, 42–154.

Mieth, C., A. Meyer, and M. Henke. 2019. "Framework for the Usage of Data from Real-time Indoor Localization Systems to Derive Inputs for Manufacturing Simulation". *Procedia CIRP* 81:868–873.

Nikitopoulos, P., A.-I. Paraskevopoulos, C. Doulkeridis, N. Pelekis, and Y. Theodoridis. 2018. "Hot Spot Analysis over Big Trajectory Data". In *Big Data*. December 10th-13th, Seattle, USA, 761–770.

Palma, A. T., V. Bogorny, B. Kuijpers, and L. O. Alvares. 2008. "A Clustering-based Approach for Discovering Interesting Places in Trajectories". In *Proceedings of the 2008 ACM Symposium on Applied Computing*. March 16th-20th, Fortaleza, Brazil, 863–868.

Rashid, K. M., S. Datta, and A. H. Behzadan. 2017. "Coupling Risk Attitude and Motion Data Mining in a Preemtive Construction Safety Framework". In *Proceedings of the 2017 Winter Simulation Conference*, edited by W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, 2413–2424. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Richly, K. 2018. "A Survey on Trajectory Data Management for Hybrid Transactional and Analytical Workloads". In *2018 IEEE International Conference on Big Data*. December 10th-13th, Seattle, USA, 562–569.

Rocha, J. A. M., V. C. Times, G. Oliveira, L. O. Alvares, and V. Bogorny. 2010. "DB-SMoT: A Direction-based Spatio-temporal Clustering Method". In *2010 5th IEEE International Conference Intelligent Systems*. July 7th-9th, London, UK, 114–119.

Scholz, J., and S. Schabus. 2014. "An Indoor Navigation Ontology for Production Assets in a Production Environment". In *International Conference on Geographic Information Science*. September 24th-26th, Vienna, Austria, 7204–220.

Shao, G., and D. Kibira. 2018. "Digital Manufacturing: Requirements and Challenges for Implementing Digital Surrogates". In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 1226–1237. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Srewil, Y., and R. J. Scherer. 2017. "Construction Objects Recognition in Framework of CPS". In *Proceedings of the 2017 Winter Simulation Conference*, edited by W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, 2472–2483. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers.

Susanti, R. M., K. M. Adhinugraha, S. Alamri, L. Barolli, and D. Taniar. 2018. "Indoor Trajectory Reconstruction Using Mobile Devices". In *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*. May 16th-18th, Cracow, Poland, 550–555.

Uckelmann, D., and J. Wendeberg. 2015. "Echtzeitortungssysteme für Industrie 4.0. Potentiale und Stand der Technik". In *Management Logistischer Informationsflüsse: 3. Wissenschaftlicher Industrielogistik-Dialog in Leoben (WiLD)*, edited by H. E. Zsifkovits and S. Altendorfer-Kaiser, 3–13. Munich, Germany: Rainer Hampp Verlag.

Wenzel, S., M. Weiß, S. Collisi-Böhmer, H. Pitsch, and O. Rose. 2007. *Qualitätskriterien für die Simulation in Produktion und Logistik: Planung und Durchführung von Simulationsstudien*. Berlin-Heidelberg, Germany: Springer-Verlag.

Wiendahl, H.-P. 2014. *Betriebsorganisation für Ingenieure*. Carl Hanser Verlag GmbH Co. KG.

Wu, F., H. Wang, Z. Li, W.-C. Lee, and Z. Huang. 2015. "SemMobi: A Semantic Annotation System for Mobility Data". In *Proceedings of the 24th International Conference on World Wide Web*. May 18th-22nd, Florence, Italy, 255–258.

Yan, Z., D. Chakraborty, C. Parent, S. Spaccapietra, and K. Aberer. 2013. "Semantic Trajectories: Mobility Data Computation and Annotation". *ACM Transactions on Intelligent Systems and Technology (TIST)* 4(3):49.

Yang, W., Y. Tan, K. Yoshida, and S. Takakuwa. 2017. "Digital Twin-driven Simulation for a Cyber-physical System in Industry 4.0". In *DAAAM International Scientific Book 2017*, 227–234. Vienna, Austria: DAAAM International.

Zheng, Y. 2015. "Trajectory Data Mining: An Overview". *ACM Transactions on Intelligent Systems and Technology* 6(3):29.

Zhong, R. Y., G. Q. Huang, Q. Dai, and T. Zhang. 2014. "Mining SOTs and Dispatching Rules from RFID-enabled Real-time Shopfloor Production Data". *Journal of Intelligent Manufacturing* 25(4):825–843.

AUTHOR BIOGRAPHIES

CARINA MIETH is a PhD candidate in mechanical engineering at the TU Dortmund University. She received her M.Sc. in electrical engineering from the Karlsruhe Institute of Technology in 2017. Currently, she works in the predevelopment networked systems at TRUMPF GmbH + Co. KG in Ditzingen near Stuttgart. She is an associated member in the DFG-funded research training group *adaption intelligence of factories in a dynamic and complex environment*. Her research focuses on cyber-physical production systems and simulation input modeling for manufacturing simulation. Her email address is carina.mieth@trumpf.com and her profile on researchgate is https://www.researchgate.net/profile/Carina_Mieth.