applied sciences

MDPI

Article

Enhancing Digital Twins of Semi-Automatic Production Lines by Digitizing Operator Skills

Angela Lago Alvarez *, Wael M. Mohammed, Tuan Vu, Seyedamir Ahmadi and Jose Luis Martinez Lastra

FAST-Lab, Faculty of Engineering and Natural Sciences, Tampere University, 33720 Tampere, Finland
* Correspondence: angela.lagoalvarez@tuni.fi

Abstract: In recent years, Industry 4.0 has provided many tools to replicate, monitor, and control physical systems. The purpose is to connect production assets to build cyber-physical systems that ensure the safety, quality, and efficiency of production processes. Particularly, the concept of digital twins has been introduced to create the virtual representation of physical systems where both elements are connected to exchange information. This general definition encompasses a series of major challenges for the developers of those functionalities. Among them is how to introduce the human perspective into the virtual replica. Therefore, this paper presents an approach for incorporating human factors in digital twins. This approach introduces a methodology to offer suggestions about employee rotations based on their previous performance during a shift. Afterward, this method is integrated into a digital twin to perform human performance assessments to manage workers' jobs. Furthermore, the presented approach is mainly comprised of a human skills modelling engine and a human scheduling engine. Finally, for demonstrating the approach, a simulated serial single-product manufacturing assembly line has been introduced.

Keywords: Industry 4.0; digital twin; human factors; data-driven modeling; industrial application

Citation: Lago Alvarez, A.; Mohammed, W.M.; Vu, T.; Ahmadi, S.; Martinez Lastra, J.L. Enhancing Digital Twins of Semi-Automatic Production Lines by Digitizing Operator Skills. *Appl. Sci.* **2023**, *13*, 1637. https://doi.org/10.3390/app13031637

Academic Editor: Vittorio Solina

Received: 20 December 2022
Revised: 24 January 2023
Accepted: 26 January 2023
Published: 27 January 2023

1. Introduction

The industrial revolution in recent decades has mainly affected the level of autonomy in production systems, among other aspects. As a result of the advances in computer and information technologies, factory automation systems became smarter, more flexible, and more human-independent [1]. This evolution in the production systems has been directly affecting the employment of human workers at factories. This effect has been studied intensively in recent years. In fact, a human is considered the most valuable asset in factories [2]. Thus, more research activities have been conducted to assess and analyze the human factors in the manufacturing sector. The main goal is to keep the human-in-the-loop of production while providing safe, trustful, and comfortable working environments, especially when the working space is shared between humans and machines (i.e., robots) [3]. To address these challenges, several technologies have emerged to help in optimizing and allocating human workers properly at the factory shopfloor level. These technologies may include smart human sensoring and digitization, processes simulation and planning, and safety simulation and monitoring. As an example, digital twins, one of these emerging technologies, have been recently employed for modeling, simulating, and optimizing manufacturing processes as presented in [4].

Digital twin technology is one of the fast-spreading and trending concepts in the academic and industrial worlds during the past decade. The term mainly refers to continuous interaction between two entities: a physical entity and a digital entity. The original definition by Michael Grieves in [5] declares the digital twin as a physical system that is replicated by a digital system and these two systems are continuously exchanging information. Digital twins are mainly used for simulating and monitoring systems to allow access to inaccessible features of the physical system. For instance, simulating the effect of the aerodynamics in the new surface paint of a car, as the current technology cannot visualize the effect. According to [6], several technology vendors have developed digital twins for commercial use such as General Electric, PTC, IBM, and Siemens. Furthermore, these solutions are categorized based on their usage and deployment. The majority of these available solutions do not address the human factor as this requires a very specific development. This lack of available solutions drives researchers to develop such applications.

To tackle these problems, the EU Commission granted funding to several projects to address the human at work. One of these projects is the Smart Human Oriented Platform for Connected Factories (SHOP4CF) project (https://www.shop4cf.eu/ accessed on 26 January 2023). This project aims at providing a reconfigurable and flexible application-based solution to support humans in factories. In this regard, the need to have a human-centered digital twin proof is crucial for achieving the goals of the project. Therefore, this paper aims at proposing a solution for including human factors in digital twins. In more detail, the objectives of this paper include:

- Conducting research for analyzing the technologies that relate digital twins to human factors.
- Build a digital twin that does a human performance assessment to manage the worker's job.
- Create a methodology to suggest workers' rotations based on their previous performance during a scheduled shift.
- Expose an implementation case to apply the proposed methodology.

The structure of this article is composed of five sections. Section 1 presents the main aspects and ideas that are behind this paper. Section 2 presents a literature review on related topics. Specifically, to introduce the concept of digital twins in the context of Industry 4.0 and the future steps of its integration into Industry 5.0. In addition, a brief review is centered on human scheduling and job rotations. Section 3 describes the chosen approach and Section 4 introduces an implementation case. Finally, Section 5 presents the conclusion and future work.

2. Literature Review

Introducing human factors in digital twins requires research on the concepts and methods in the domain. Such groundwork is presented as follows: Section 2.1 shows the status of digital twins in manufacturing systems, and Section 2.2. introduces the scheduling of shifts and the concept of job rotations.

2.1. Digital Twins in Manufacturing

Digital twin (DT) is one of the main technologies born with the development of Industry 4.0. The definition of a DT is complex and has been investigated thoroughly due to an incomplete understanding of this concept. Articles such as Kritzinger et al. [7] and Negri et al. [8] reflect extensively on the notions of digital twins. In terms of manufacturing, both articles include the definition offered by Garetti et al. [9], recognizing a DT as a virtual representation of a physical production system. Both systems are synchronized to exchange information (e. g., sensed data, real-time data, and results of mathematical models). The aim of the DTs is to forecast and optimize in real-time the behavior of manufacturing systems in each phase of their life cycle.

In Industry 4.0, DTs represent the bidirectional union of the physical world with the virtual one, interconnecting the physical elements with their digital counterpart [8]. If combined with other tools, such as Big Data, IoT data, Artificial Intelligence (AI) technologies, and high-level business data (e.g., MES, ERP), it enables complex interactions between cyber-physical systems [10]. Within this concept, digital twins help to develop smarter manufacturing systems with high efficiency and reliability. Durão, et al. [11] introduced the vision that exists both in the industry and academy of DTs under Industry

4.0. It relates the requirements that are considered necessary for the development of a DT and its challenges, such as real-time data management, physical-virtual integration, and fidelity of the model.

Although there is exhaustive research on digital twins [7,8,12,13] and their applications [14–16], most of the existing perspectives are asset specific. Human factors are not deeply considered, due to their complexity in the modeling and, in general, the relationship between production assets and production resources is not very detailed. Al-Yacoub et al. in [17] introduced the human perspective in a DT to improve the system maintainability and debug failures. In manufacturing systems, DTs are centered on the modeling of the manufacturing process itself. Workers are described as resources to allocate or as a part of human-machine interactions (HMI) [18,19]. Berti et al. [20] and Lui et al. [21] made an extensive review of DT and how this virtual representation introduces human factors. Both conclude that more research is needed on how DT could improve ergonomics and balance workloads and consider the operator to promote their safety conditions in a way that the DT could give real-time feedback about the state of the operator. These comments follow the new industrial revolution approach called Industry 5.0., which places the well-being of the workers and sustainability at the center of the production process [22]. Consequently, digital twins need to be developed in the same way.

2.2. Scheduling and Balancing Production Systems

Many manufacturing facilities generate production schedules, which refer to the allocation of resources to produce valuables. Workers are considered part of the resources of the factory. In fact, effective human scheduling is needed since the performance of the manufacturing system is heavily affected by individual human performance [23]. The relationship between good job performance and an increase in productivity has been extensively studied and analyzed [24]. Job performance is affected by several factors, such as the cognitive state of the worker, qualifications, or the workplace [25,26]. Diamantidis and Chatzoglou [27] propose an empirical model that defines employee performance and analyze the interrelation between environment-related, job-related, and employee-related factors and their impact on their performance.

Therefore, good performance has two main consequences: job satisfaction, since it improves performance itself, and it increases worker productivity [25]. In this way, to keep high productivity, it is important to stimulate the employee. One way to perform this is by introducing worker rotation methods, which are the practices of moving employees between workstations or tasks. Triggs et al. [28] list many of the advantages of applying this method. Regarding the employee perspective, workers' rotations (WRs) reduce the monotony of the work and boredom. They help to relieve the stress of the tasks, increase motivation and reduce labor absenteeism. From the employer's perspective, applying this distribution helps to increase production while obtaining a cross-trained workforce. Additionally, the introduction of WRs in the shifts helps to reduce the probability of suffering work-related musculoskeletal disorders (WRMD) [29,30]. WRMD are lesions that affect the different parts of the body associated with the movement: hands, wrists, elbows, neck, shoulders limbs, and back. They are related to working activities that require a high frequency of repetitions, for example, in assembly lines. As an example, Moussavi et al. in [31] conducted a study to smooth the daily workload of the workforce in an automotive assembly line. Job rotation proved to be an effective way of protecting against WRMDs in repetitive jobs in a long-term scope.

There are many articles explaining different methods to design rotations, from human observation analysis to mathematical procedures optimizing different aspects [32], such as ergonomics constraints [33] or safety [34]. However, there is little information available on the generation of several rotations during working times based on sensed data from the production line. This limitation is key in the development of this research.

2.3. Manufacturing Execution Systems

As manufacturing systems become more demanding and sophisticated, more complex tools are needed to keep track of them. These tools help to make decisions, not only in the business field or in the process but between them. The manufacturing execution systems (MES) fulfill this role and bridge the gap between the planning system and the controlling system that defines a production plant. In [35], MES is defined as "an on-line extension of the planning system with an emphasis on execution or carrying out the plan." Manufacturing Execution Systems Association (MESA) [36], is an important global non-profit organization that assigns the same role to MES. Additionally, Verein Deutsche Ingenieure (VDI) [37] coincides with [35] and [36] mentioned above, both in definition and in the functions MES performs. An extensive review of these functions are included in the article proposed by S. Iarovyi et al. [38].

Among these functionalities, all of the previous experts cited include a personnel management/resources scheduling category. Due to this, the human scheduling can be considered a part of the MES definition. In order to seek available commercial solutions regarding this, Table 1 offers a summary of some MES solutions that include human scheduling.

Table 1. Commercial solutions in MES providers.

Commercial Solutions	Main Functionalities	Does It Include Human Scheduling?	Does It Consider Human Well-Being in the Process?
Plex Execution System [39]	Production finite scheduling Closed-loop quality management Inventory and production management	Yes	No, it considers humans as a resource.
Siemens Opcenter Execution Process [40]	Resources allocation and control Inventory and production management Data collection and acquisition Maintenance operations management	Yes	Yes, it includes performance analysis.
Proficy Smart Factory MES by GE Digital [41]	OEE and quality with analytics Production scheduling Manufacturing data management Enterprise scale predictive analytics	Not in MES software, but GE Digital have a separate software to perform it.	N/A
SAP Manufacturing Execution [42]	Simplified integration with SAP ERP Inventory and production management Performance analysis Data collection and acquisition Production scheduling	Yes, but as external module.	No, it considers humans as a resource.

As Table 1 shows, not many MES providers include human scheduling in their software, and only the Siemens solution includes human performance analysis. Since there are not many available MES solutions that include human scheduling, other software/applications that are more specific could provide this information. In order to perform that, some support shopfloor level human resources (HR) management solutions can be seen in Table 2.

Table 2. Shopfloor supporter HR management solutions.

Solutions	How Human Scheduling Is Included?	Does It Consider Human Well-Being in the Process?	Pricing
ATOSS workforce scheduling [43]	Duty and shift planning with cost control.	Not specified	€
Infor Workforce Management [44]	Demand-driven scheduling and absence management tools. Scheduling rules with payment management tools.	No	€
Papershift [45]	Auto assign AI based on availability of the worker, absence, and rest periods.	Yes	€
ABC Roster [46]	Auto assign workers based on availability with predefined constrains (i.e., max. number work hours per scheduled).	No	Free
When2Work [47]	Autofill feature based on constraints (i.e., approved time off, employee working time preferences, minimum time between shifts, maximum shifts set per employee).	No	€

These tools are specific for human scheduling and shift planning. All applications offer a different way to perform the human scheduling, considering different attributes but only one of the solutions includes the well-being of the worker.

Even the commercial solutions shown in Tables 1 and 2 were considered; however, they do not provide any technical insight of the methodologies used since the information available is what is shown in the marketing material of these applications.

3. Approach

The objective of this research is to incorporate human factors into a DT so that performance is considered when balancing an operator´s workload within a production line. Among the benefits of applying this method is the possibility of increasing product quality while reducing the number of defects [26]. Additionally, changing activities regularly reduces work monotony, fatigue, absenteeism, and the risk of a long-term muscular disorder. [48,49]. Thus, a DT that provides a human scheduling capability at the shopfloor can help eliminate bad effects of repetitive manual work. This section presents a methodology to assess human performance based on live data. Section 3.1 provides an overview of the system built. Section 3.2 explains how performance is considered. Section 3.3 focuses on the rotation suggestion logic. Finally, Section 3.4 reports how this logic is included in the DT.

3.1. Solution Overview

As was introduced in the literature review, one of the main challenges for the DTs is data management and communication between systems. In this approach, a software that enables active communication between the shopfloor and the DT is used. The proposed system is presented in the diagram shown in Figure 1.

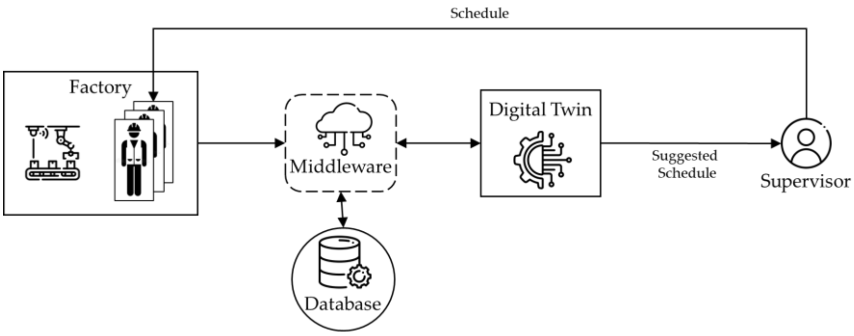

Figure 1. Shopfloor—human scheduling DT relations diagram.

The factory input data (e.g., ERP data, IoT system data, etc.) is sent to a middleware, connecting the shopfloor with the DT. This component sends bidirectional updates in data to both components and contains the database that stores information about the workers and the production line. Figure 2 presents the relationships between the entities.

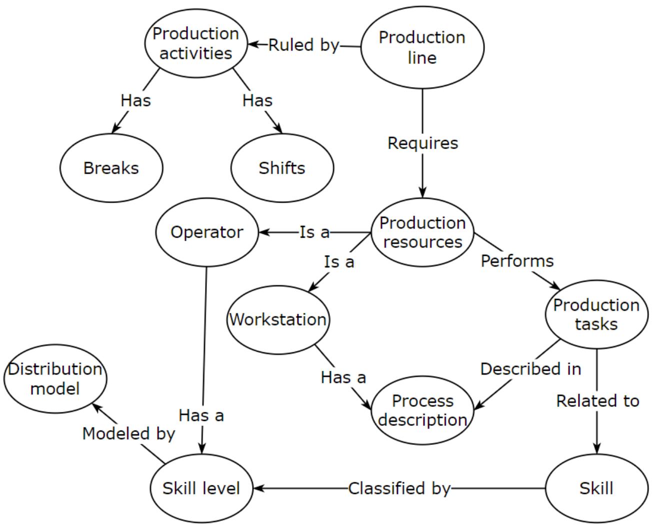

Figure 2. Knowledge-based representation of the described case.

It shows that the production line is ruled by production activities. Those are shifts, known as the period during which production is operational, and breaks, known as the time lapses during which production is not operational and the workers can rest. The line is composed of production resources which can be workstations, places where production tasks are performed (i.e., machines or assembly tables), or operators, that possess skill levels for each skill.

The DT is regularly updated with new input data and processes it to inform the user about the general state of the factory and suggest the operator's rotations based on their performance. To perform that, some assumptions are made regarding the process:

- The rotation suggestion is obtained every time it is triggered. The way it is triggered may vary depending on the specific types of signals received from the production line.
- The production line or process is made up of several workstations and many operators can work on them. Each station has its process description composed of several tasks and each task is related to a skill defined in the skill matrix. The tasks within the workstations are ordered based on an importance index defined by an expert on the production line. Since each task corresponds to a skill, the skills share the same importance index as the tasks. No skill has the same level of importance within the same workstation.

- Operators cannot repeat workstations on the same day. That is, if an operator has already worked at one of the assigned workstations in a day, the rotation logic cannot make them repeat or let him/her continue working in the same workstation on that shift. This restriction is established to reduce the risk of suffering muscular disorders (WMSDs) [48,50] and balance the workloads.

The architecture of the DT regarding this functionality is presented in Figure 3.

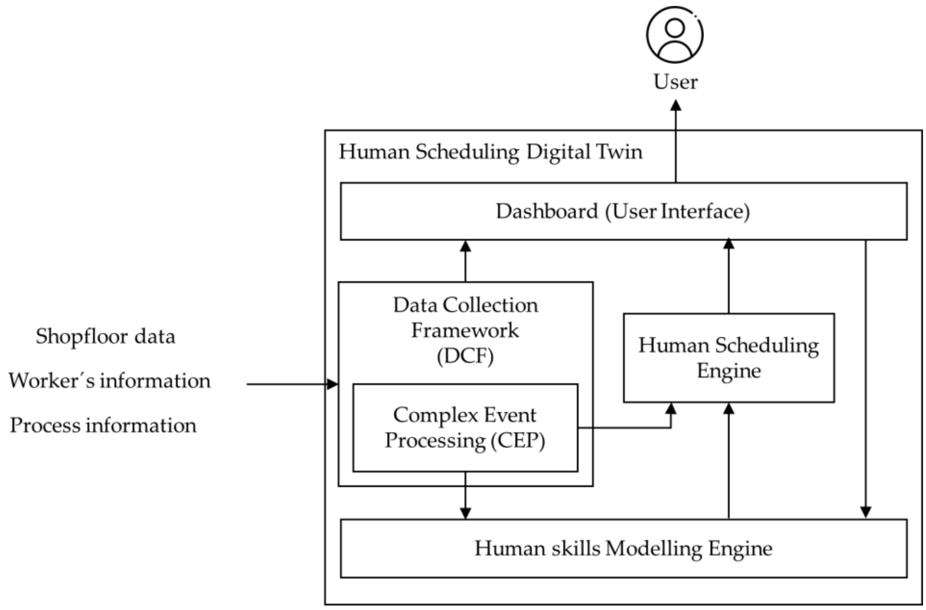

Figure 3. Human scheduling digital twin block diagram.

The data updates are received from the middleware (Figure 1) by the Data Collection Framework (DCF). The DCF is part of the backend and is the infrastructure that collects, manages, and shares the data with the other components of the digital twin. It has two main functions: send these updates to the user interface to inform about the status of the production line and convert the shopfloor data into skill measures. To perform this, this component retrieves the workers, product, and process information from the database. After that, the DCF sends the skill measurement to the human skills modeling engine to classify and structure the input data, in a way that evaluates if the performance of the worker for that concrete skill in that product, is up to the execution capacity of the operator. This process is carried out during the time that the line is producing. Once a rotation is triggered, this information is sent to the human scheduling engine to suggest a new rotation of the workers to the user, based on the data structure.

Finally, the suggestion reaches the user through the dashboard, and he/she can accept it or modify it.

3.2. Human Skills Modeling Engine (Data-Driven)

The human skills modeling engine is a mechanism that, using the skills measures provided by the DCF, creates a model to classify and structure data. The configuration is presented in Figure 4.

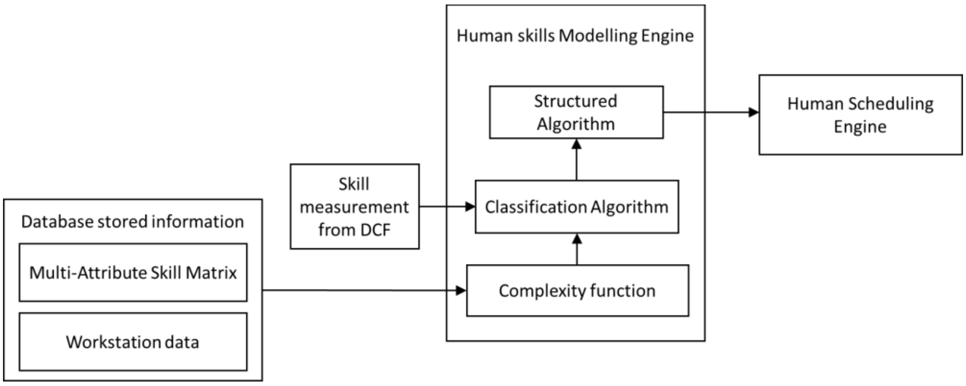

Figure 4. Human skills modeling engine.

The system retrieves information about the workstations and the multi-attribute skill matrix, which presents the data skill levels regarding the skills that are needed to perform the work, from the database. Additionally, skill measurements are sent by the DCF. The human skills modeling engine is divided into the following parts: a complexity function, that models different gaussian distributions for each skill defined in the skill matrix with recorded data from the line, and a microsystem that classifies the data received and a micro service that structures it.

3.2.1. Multi-Attribute Skill Matrix

In manufacturing systems, skill matrices are common to plan and manage skills for jobs, teams, projects, or departments. It is a tool used to track and rank employees′ skills. This framework helps to select the right people for a job and to identify missing competencies or gaps between the staff. This procedure allows the employer to keep track of their employee's development. The conventional skill matrix of operators in production systems is generally made in a single table with the workstations on one side as rows and the workers in the columns. These stations have some operations that describe the process, and the skill level of the worker represents the ability that this person must have to carry out the work in the workstation [51].

The skill matrix proposed in this paper implies adding one more dimension to the conventional skill matrix. The skill levels do not refer to the workstations per worker, but rather to the skills. This matrix is composed of different sheets regarding the skills to evaluate (e.g., grasping, assembling, visual inspection, etc.). Thus, the worker will have a skill level for each skill per workstation. This way, the performance is evaluated in more detail. This matrix definition is explained in Figure 5.

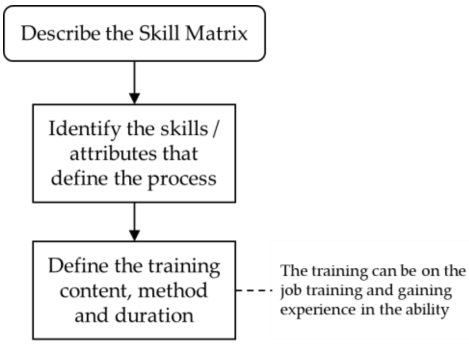

Figure 5. Skill matrix definition. [Source: inspired by the chapter "Skill management" (pages 113–119) in reference [12]].

To develop every sheet of the skill matrix, the first step is to identify the skills or attributes that define the production process. Then, create specific training content (material, type, method, and duration) for each skill to be able to train the new workers or improve the skill levels of the current workforce. As previously mentioned, the skill matrix workstations are related to the skills by a process description (list of tasks). Figure 6 shows how to establish this relationship and the skill matrix flow.

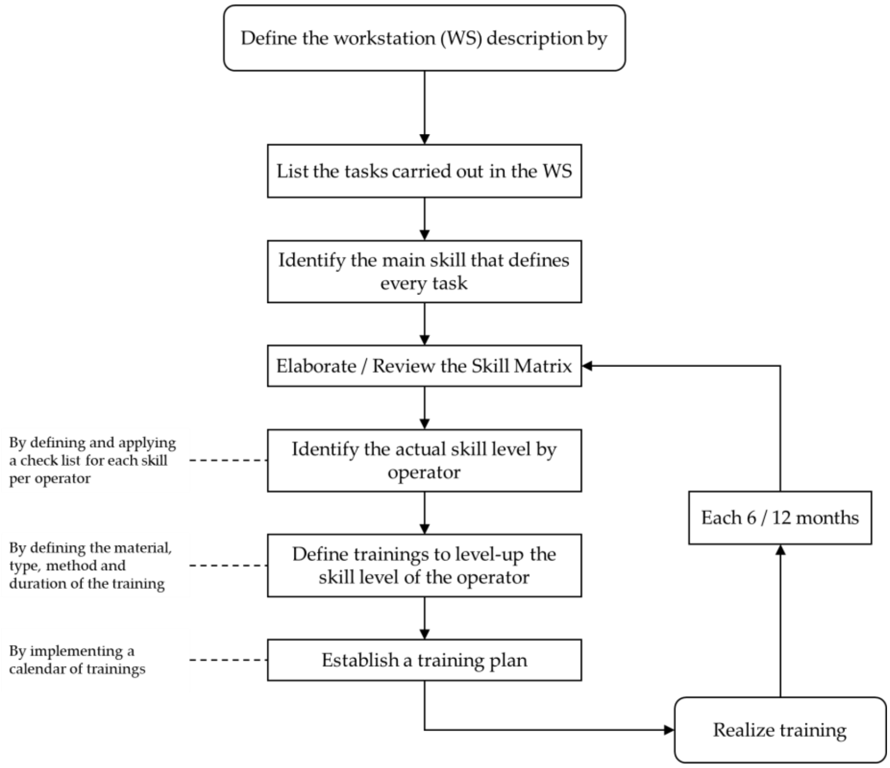

Figure 6. Skill matrix flow. [Source: inspired by the chapter "Skill management" (pages 113–119) in reference [12]].

In this way, each workstation has a process description made up of simple and individual operations, which will be included in one of the skills defined in the skill matrix definition. For example, there is a workstation A in the assembly line of the main part called C. The skills defined in the skill matrix definition are grasping, gluing, screwing, assembling, welding, and visual inspection. Workstation A is composed of the following tasks: grasping object B, assembling B into C, and screwing top left object B. These individual operations are included within the skills: grasping, assembling and screwing.

Following the flow presented in Figure 6, the next step is to elaborate or review the skill matrix by identifying the skill level of the worker for each skill, and the training plan (monthly, annual, etc.). After that, the operator should perform the training and the supervisor should review the multi-attribute skill matrix every six/twelve months to keep it up to date. Figure 7 presents the method to identify the current skill level of the worker.

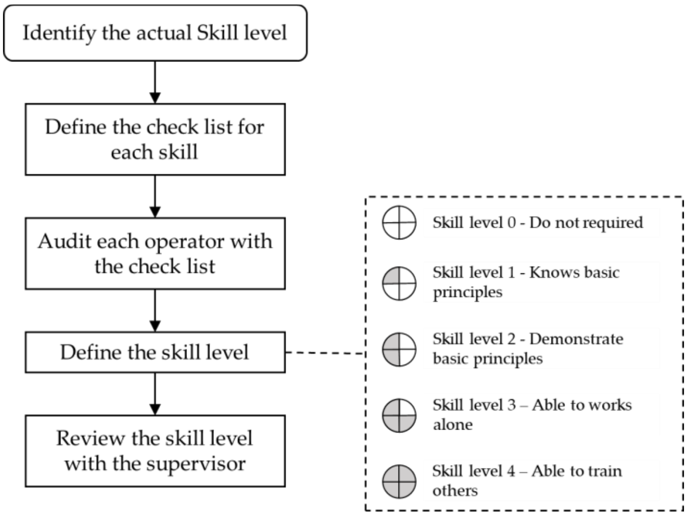

Figure 7. Skill level identification [Source: inspired by the chapter "Skill management" (pages 113–119) in reference [12]].

Workers should be audited following a checklist created per skill and the levels are ranked:

- Skill level 0. The worker is not required to learn this skill.
- Skill level 1. The worker has gone through the training and knows the basic principles of the skill.
- Skill level 2. The worker has gone through the training and has demonstrated the basic principles learned but needs occasional support.
- Skill level 3. The worker has gone through the training and has demonstrated the skill of working alone without supervision.
- Skill level 4. The worker has demonstrated the skill daily and is able to train others.

After the skill level is defined, it should be reviewed with the supervisor. The multi-attribute skill matrix is stored in the database. Isolated, it does not trigger any change in the human skills modeling engine. However, as the system collects the data from the database through the middleware and performs classifications based on the information available, the skill matrix should be built correctly, be accessible, and kept up to date for the microservice to work properly.

3.2.2. Complexity Function

As was introduced in Section 3.2.1., the multi-attribute skill matrix is based on the skills that define the production process. For each skill, the complexity function is going to create a Gaussian distribution with the recorded data that relates the task that is associated with the skill measures, and with the skill level of the worker. The Gaussian distribution is chosen as the statistical model since it correctly adjusts to many natural phenomena (i. e., age, measurements, etc.) and applies to all data sets with finite variance. The recorded data is stored in the database and contains a batch of different measurements of different skill level workers. The competency measurements refer to the parameter that controls the skill. For example, if the skill is drilling, the skill measure can be in millimeters of material removed.

Figure 8 presents the Gaussian distribution of the generic skill recorded data.

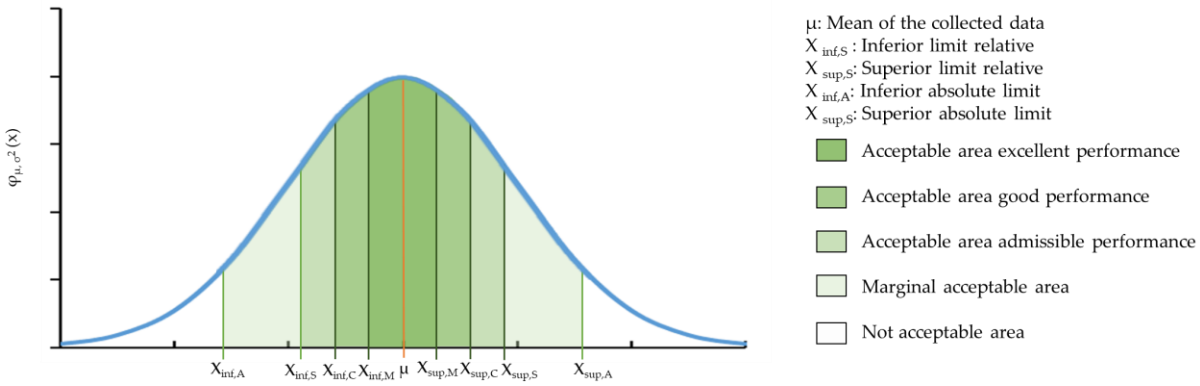

Figure 8. Gaussian distribution probability density function (PFD) with thresholds for each skill level.

Apart from the shape of the distribution model, Figure 8 has the following parameters that are important in the next steps of the human assessment:

- Mean (μ): Measure of the central tendency of a data set. Specifically, the sum of the values divided by the number of values. As the Gaussian distribution is a model representation of the real data, the shape of it is centered in the average of the values.
- Nominal measured value (NMV): Theoretical measure in which the tasks related to the skill defined in the multi-attribute skill matrix should be carried out. Normally, this parameter is obtained from manufacturing optimization software and is specific to each skill on the production line. If the process is modeled precisely and is faithful to the reality of what happens on the production line, this value should be close or equal to the mean of the data.
- Superior absolute limit ($X_{sup,A}$): Maximum measured value in which an operator can make a product. This value is defined by an expert in the production line.
- Inferior absolute limit ($X_{inf,A}$): Minimum measured value that an operator must make a product. This value is defined by an expert in the production line.
- Superior relative limit ($X_{sup,S}$): Recommended maximum measured value for an operator to make a product according to his skill level, reflected in the multi-attribute skill matrix. This parameter is obtained by adding a certain percentage to the NMV, as shown in Equation (1). This percentage called threshold (Th) is unique for each skill level number and calculated based on the experience of an expert in the production line.

$$X_{sup,S} = \text{NMV} \times \left(1 + \frac{\text{Th}}{100} \times \left(X_{sup,A} - 1\right)\right), \quad (1)$$

- Inferior relative limit ($X_{inf,S}$): Recommended minimum measured value for an operator to make a product according to his skill level reflected in the skill matrix. This parameter is obtained by deducting a certain percentage from the NMV, as shown in Equation (2). This percentage called threshold (Th) is unique for each skill level number and calculated based on the experience of an expert on the production line.

$$X_{inf,S} = \text{NMV} \times \left(1 - \frac{\text{Th}}{100} \times \left(X_{inf,A} - 1\right)\right), \quad (2)$$

These parameters define the colored areas under the Gaussian distribution.

- Acceptable area (blue color): Includes data between the limits $X_{inf,S}$ and $X_{sup,S}$ and its area defines the expected performance of the operator for its skill level. This acceptable area is also divided into three equal parts to differentiate the area closest to the NMV that is excellent performance, the closest to the marginal acceptable area that is admissible, and the one in the middle corresponds to a good performance. The limits of these intervals are shown in Figure 8. These parameters are defined by Equations (3)–(6).

$$X_{inf,C} = \mathrm{NMV} - \frac{2 \times (\mathrm{NMV} - X_{inf,S})}{3}, \quad (3)$$

$$X_{sup,C} = \mathrm{NMV} + \frac{2 \times (\mathrm{NMV} - X_{inf,S})}{3}, \quad (4)$$

$$X_{inf,M} = \mathrm{NMV} - \frac{(\mathrm{NMV} - X_{inf,S})}{3}, \quad (5)$$

$$X_{sup,M} = \mathrm{NMV} + \frac{(\mathrm{NMV} - X_{inf,S})}{3}, \quad (6)$$

- Marginal acceptable area (green color): Includes data between $X_{inf,A}$–$X_{inf,S}$ and $X_{sup,S}$–$X_{sup,A}$ limits. These two areas collect data that, although they are not negative results for the process, are not considered normal or expected for the skill level associated with it.
- Not acceptable area (white color): Includes data below $X_{inf,A}$ and above $X_{sup,A}$. These values are not typical of the behavior of the operator due to his/her level of skill and are not acceptable for the process itself.

These areas are specific to skill levels and the shape of the Gaussian is defined by the data collected from the skill. Figure 9 presents how this general Gaussian would look for different skill levels. In this case, it is considered that the process was perfectly designed so that the mean is equal to the NMV.

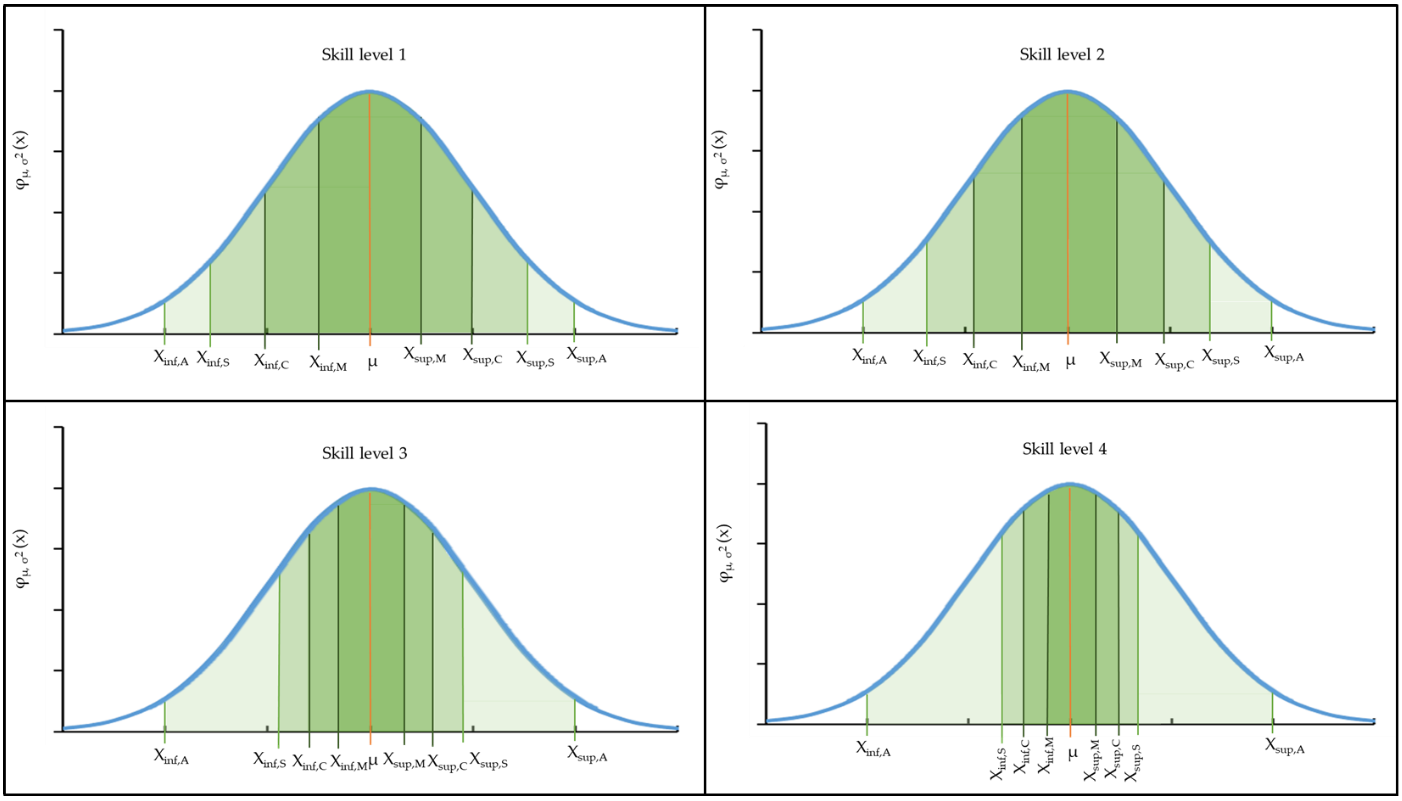

Figure 9. Gaussian distribution PDF with thresholds for different skill levels.

An operator with skill level one is expected to have a larger acceptable area than an operator with skill level four since he/she is not as experienced in the target skill.

3.2.3. Classification and Structured Algorithms

As previously mentioned, one workstation process description is composed of many tasks and every task is related to a skill defined in the multi-attribute skill matrix. Therefore, for every skill, the code presented in Algorithm 1 is run to classify the skill measured related to that skill.

Algorithm 1. Function to classify the input data.

```
procedure CLASSIFYDATA
    delta_value                                 // input data
    nmv_ value                                  // NMV
    max_abs_ value, min_abs_ value              // X_sup,A, X_inf,A
    max_skill_value, min_skill_value            // X_sup,S, X_inf,S
    max_close, min_close                        // X_sup,C, X_inf,C
    max_medium, min_medium                      // X_sup,M, X_inf,M
    if min_medium < delta_value < max_medium then
        acceptable_excell += 1
    else if min_close <= delta_value <= min_medium or max_medium <= delta_value <= max_close then
        acceptable_good += 1
    else if min_skill_value <= delta_value < min_close or max_close < delta_value <= max_skill_value then
        acceptable_admis += 1
    else if min_abs_value <= delta_value < min_skill_value or max_skill_value < delta_value <= max_abs_value then
        marginal_acceptable += 1
    else if delta_value < min_abs_value or delta_value > max_abs_value then
        not_acceptable += 1
    end if
```

Following the procedure shown in Algorithm 1, the data is saved by skill for each WS. To be able to suggest how to rotate the workers when it is triggered, the data is restructured following the code presented in Algorithm 2.

Algorithm 2. Structured data.

```
procedure STRUCTUREDATA
    data_dic = { "name": ws,                              // name of the WS
            "Rotation_number": rot_num,                   // Identifier of the rotation
            "Actual_operator": user,                      // Operator identifier. Data extracted from DB
            "Skill_involved": skill                       // Data extracted from DB
            "Skill_level": skill_level,                   // Data extracted from DB
            "Importance_index": importance_index,         // Data extracted from DB
            "Nominal_value": NMV,                         // Data extracted from DB
            "Average_value": mean,                        // Measurement
            "Minimum_value_skill": min_skill_value,       // Data extracted from DB
            "Maximum_value_skill": max_skill_value,       // Data extracted from DB
            "Minimum_value_abs": min_abs_value,           // Data extracted from DB
            "Maximum_value_abs": max_abs_value,           // Data extracted from DB
            "Total_Acceptable_excell": acceptable_excell,         // Learned data
            "Total_Acceptable_good": acceptable_good,             // Learned data
            "Total_Acceptable_admissible": acceptable_admis,      // Learned data
            "Total_Marginal_acceptable": marginal_acceptable,     // Learned data
            "Total_Not_acceptable": not_acceptable                // Learned data
            }
    return data_dic
```

3.3. Human Scheduling Engine

Once the classification and restructuration of the data are complete, it is used to generate rule-based human scheduling. This system collects the data from the human skills modeling engine and compares each worker´s results, in order to assign them to another workstation. The assignment criteria are based on the complexity of the workstations, to be able to balance the workloads and the availability of the workers (assumption 3). Therefore, it is necessary to establish a comparison between workstations to know which are the most demanding. The list of activities carried out in each station may vary, so there is a high difficulty in the assessment of the complexity level. This way, the complexity of the workstation is evaluated following the model presented by Zeltzer et al. [52] and developed by Mattsson et al. [53]. After that, the WS is ranked based on the evaluation. The rules used to assign the batch of operators are presented in Algorithm 3.

Algorithm 3. The suggestion of rotation function.

```
procedure SUGGESTROTATION
    all_ws_data_current_rot // processed and stored data of all WS during the previous turn
    acceptable_optimal_dic, acceptable_medium_dic, acceptable_close_dic, not_acceptable_skill_dic,
    not_acceptable_abs_dic ← dic ()
    suggest_rot_list ← list ()
    for workstation in all_ws_data_current_rot
        assign in the dic () created the classified data from data_dic
    criteria_list = [not_acceptable_dic,
                     marginal_acceptable_dic,
                     acceptable_admis_dic,
                     acceptable_good_dic,
                     acceptable_excell_dic]
    historic_worker_previous_ws ← list ()
    current_total_ws ← list () // ordered from lowest workload to highest workload
    for worker in all_ws_data_current_rot
        get worker performance with the highest "Importance_index"
        assign to first_round_selection ← list ()
    for worker in first_round_selection
        for value in criteria_list
            get worker with highest criteria_list[value]
            if two workers same value then
                get worker performance with 2nd highest "Importance_index"
            end if
        select worker idx
        for ws in current_total_ws
                if worker idx do not exist in historic_worker_previous_ws(current_total_ws[ws]) then
                    add worker idx to suggest_rot_list
                    delete ws from current_total_ws
    return suggest_rot_list
```

The algorithm suggests assigning workers based on the operator's performance (stored data). The assignment starts selecting the worker with the worst performance, so the operator with the worst results, according to the criteria list, in the skill with the highest importance index, is selected. This person should be assigned to the less complex WS since he/she could have some problems with the current one. These problems can include many factors, from the fact that the operator is not in the best cognitive or physiological state, so his performance is not optimal according to his abilities, to the possibility that the worker´s skill levels are not correctly defined. This assignment will allow the operator to recover on the next turn or ease the workload. If the worker has been working at the suggested station during the shift, he/she cannot work at it again (assumption 3) and the next station that meets the availability and task complexity requirements is selected. If two workers have the same results for the highest importance index skill, the second highest importance index skill results are considered. This loop is repeated until all operators are assigned.

3.4. Digital Twin Interactions

The DT proposed at the beginning of this chapter takes the shopfloor data, the worker´s information, and process description from the database to inform the user about the status of the production line and to give rotation suggestions based on previous recorded live data. Based on the premises mentioned before, the following flow of events is proposed to send a rotation suggestion (Figure 3):

- The complex event processing (CEP) component of the DT detects when a rotation of the workers has been triggered. The CEP is a method for tracking and analyzing information flows about things that happen and extracting a conclusion from them. In this case, it analyzes the process information flow received by the DCF. The frequency at which the DCF receives information from the shopfloor depends on how the information is updated in the system, by pulling the data or listening to events. This configuration may vary according to how the middleware and DCF are programmed in the specific system. Then, the CEP in every instance checks the status.
- Regular information flow: the process workflow is normal, and they are producing; therefore, the DT must be recording data.
- Interrupted information flow: digital twin has stopped receiving signals from the production line. This can be due to two circumstances: the shopfloor had a problem that caused it to stop production. In that case, DT informs the user through different notifications in the dashboard. The other scenario contemplates that a break has occurred, so a suggestion of rotation is needed.
- In this last case, a rotation request is sent. This request contains the ID of the rotation requested as well as the current position of the workers in the workstations. It also includes the historical record of which operator has been working in which station during the entire shift (assumption 3).
- The human scheduling engine sends a proposed rotation based on the data processed by the human skills modeling engine before it was triggered.
- The digital twin displays the proposed rotation to the user through the UI. It can be accepted or modified since it is a proposition, and the user has the right to choose. However, the modification of the suggested rotation is subject to the following conditions.
- The system does not allow modifying the rotation of workers to keep the worker in the same workstation continuously during the shift.
- The worker proposed for a job can only be changed in the case of real need (e.g., illness) and can only be replaced by another from the general list of workers and not by one of the workers proposed for another station.
- Once the position of one worker is modified, the user needs to fill a report indicating the causes of the change.
- Then, the final rotation is confirmed to the DCF and CEP as a rotation confirmed.
- Finally, the data processing with the new workers positioned at their new workstations begins. This cycle is repeated throughout the work shift.

Figure 10 shows the proposed workflow.

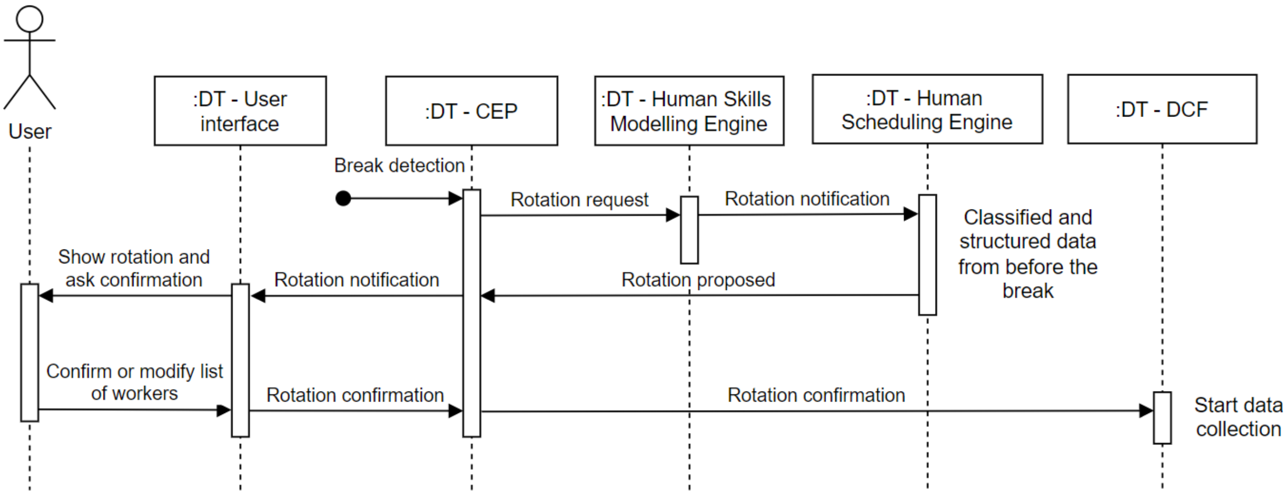

Figure 10. Sequence diagram of a rotation request.

4. Implementation

To test the approach presented in the previous section, a use case is designed. The scenario to be exemplified aims to show that human factors can be introduced into a DT applied to manufacturing systems. Furthermore, it seeks to use the available data from the production line to balance the workload between operators. To achieve this, a methodology to offer rotation suggestions in a work shift was developed.

To describe the use case, the following structure is used: first, Section 4.1 introduces the use case description and Section 4.2 reports the results obtained.

4.1. Use Case Description

The case in point is based on an assembly line since it is the most used method in mass production. It focuses on a serial single product within multiple workstations. The distribution of the stations is linear and the existence of buffers between WS is not considered. For the sake of simplifying the use case, instead of applying it to a complete assembly line, the target was reduced to a selection of some workstations. Figure 11 presents the use case description detailed with the data needed to run the experiment. All the information is saved and stored in the database of the system shown in Figure 1.

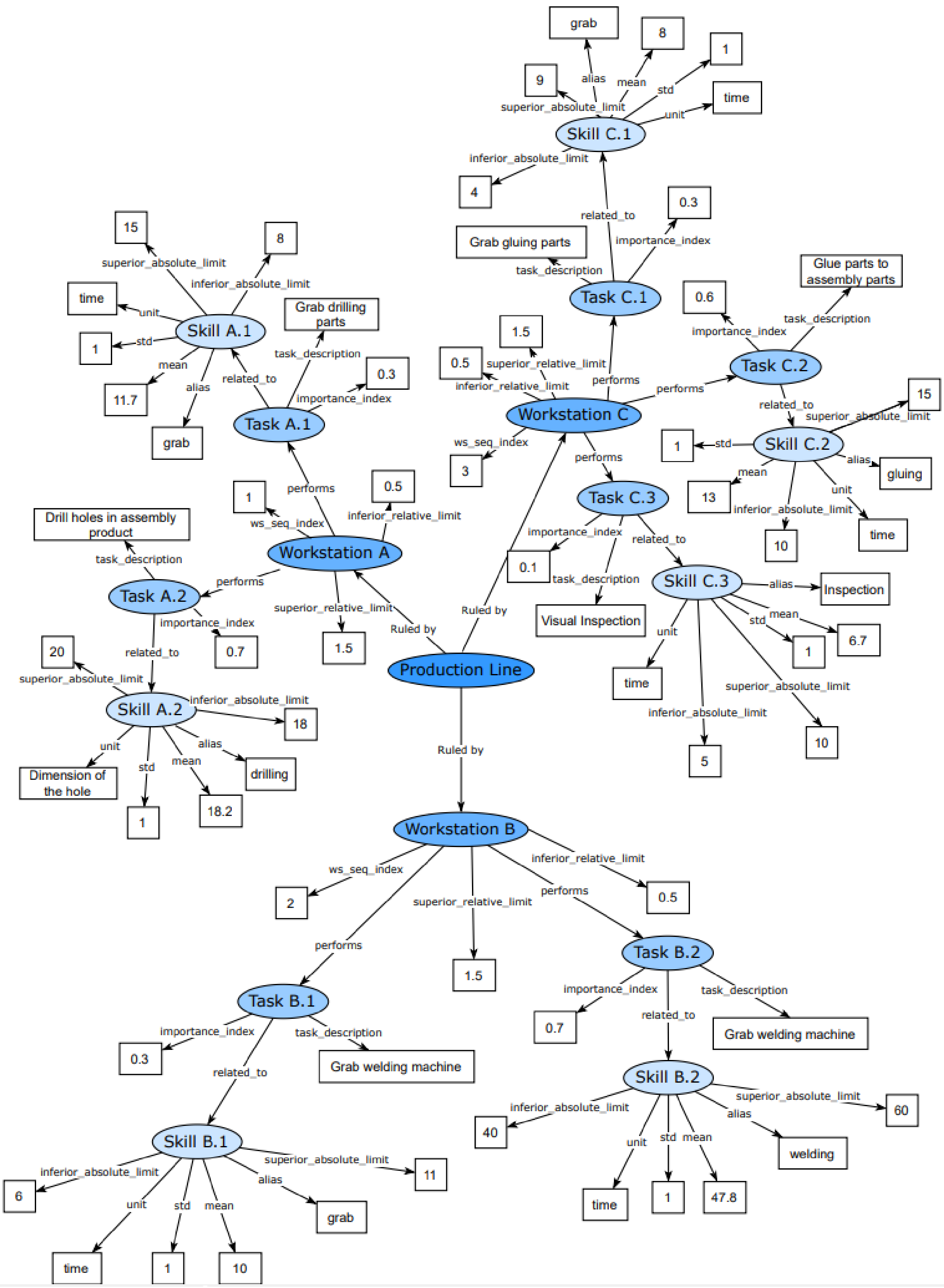

Figure 11. Implementation case description.

Regarding the manufacturing schedule, the production line works in three daily shifts of eight hours, and each work period is composed of smaller timeframes separated from each other by breaks. There are three pauses that last 15 to 30 min. For this implementation, data from the shopfloor is retrieved in an event driven process. The CEP detects the breaks during a shift and sends the rotation requests. This means that the workers rotate between stations after the breaks. In addition, three workstations are considered. Each of them includes the following information in the database: a *sequence_indicator*, specifying the order of the WS, a *process_description,* which can be summarized as a list of tasks to carry out, and the relative limits, that express the interval defined for an operator to make a product according to his skill level reflected in the multi-attribute skill matrix. Table 3 shows these restrictions for each skill level. As previously mentioned, the relative limits are selected by an expert on the production line.

Table 3. Example of superior and inferior relative limits.

Skill Level	Superior Relative Limit (%)	Inferior Relative Limit (%)
0	Not required/Not applicable	Not required/Not applicable
1	60	3
2	35	7
3	26	10
4	15	15

Tasks are related to skills, one by one. For each skill, the information provided is the mean and the standard deviation of the recorded data, the absolute limits, the end values that an operator must make a product, so the production rate is steady, and the units of the measurements.

Regarding human resources, Table 4 summarizes the multi-attribute skill matrix generated for this demonstration. Operators are identified by IDs.

Table 4. Use case multi-attribute skill matrix.

Skill: Grasp			
Operator ID	**Workstation A**	**Workstation B**	**Workstation C**
100	2	3	3
101	1	3	3
102	4	4	4
103	2	2	2
Skill: Drilling			
100	1	Not required	Not required
101	1	Not required	Not required
102	4	Not required	Not required
103	3	Not required	Not required
Skill: Welding			
100	Not required	2	Not required
101	Not required	2	Not required
102	Not required	4	Not required
103	Not required	3	Not required
Skill: Glue			
100	Not required	Not required	2
101	Not required	Not required	2
102	Not required	Not required	4
103	Not required	Not required	3
Skill: Visual inspection			
100	Not required	Not required	3
101	Not required	Not required	3
102	Not required	Not required	4
103	Not required	Not required	2

4.2. Results

For the implementation case explained in Section 4.1, synthetic data is created to be able to apply the methodology proposed. The datasets regarding each skill are created randomly over 1000 data points and they are shown in Figures 12 and 13. The absolute limits for each skill are shown in Figure 11.

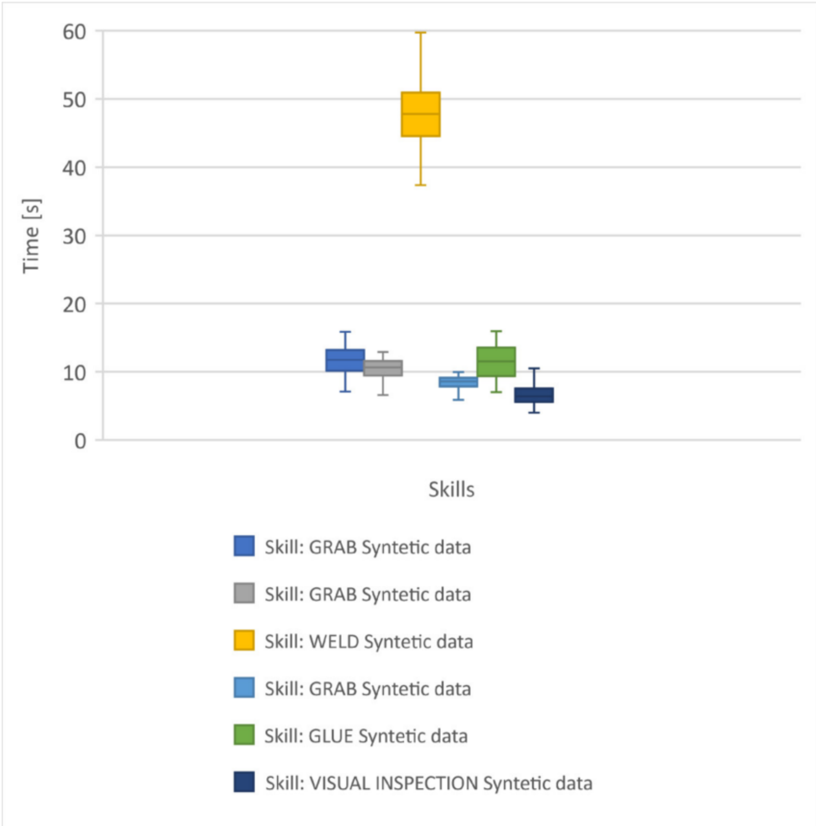

Figure 12. Time-based datasets used as synthetic data for the implementation case.

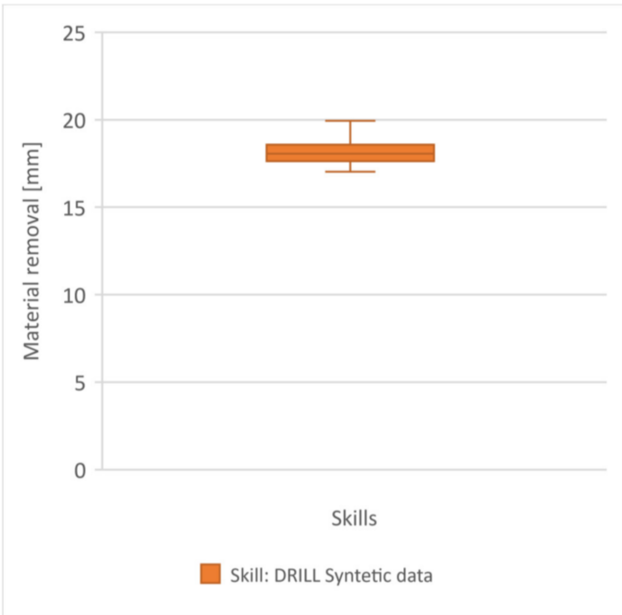

Figure 13. Material removal-based datasets used as synthetic data for the implementation case.

Initially, operator 100 is working at WS A, worker 102 is performing in workstation B and 103 at station C. For workstation A, the highest importance index activity is the drilling. At station B, welding is the most important skill, and in WS C, gluing. This importance index is defined to rank the skills to consider them as shown in Algorithm 3.

Starting with the complexity function, the following seven figures present the Gaussian distribution of the synthetic data for each skill per skill level. The colored areas have the same meaning as in Figure 6, but all acceptable areas are marked in the same blue tone. The vertical orange line displays the mean for that skill, and the yellow one identifies the NMV, and the green and blue marked ones show the absolute and relative limits.

Figure 14 shows the skill of grasping tools and materials for workstation A. The NMV and mean are slightly different. It is common for this to happen since the first one models the theoretical steps and the second expresses the average value for all workers. The closer these values are, the more it indicates that the model is well done according to the needs of the operators. In the graph corresponding to skill level 4, the mean coincides with the inferior relative limit.

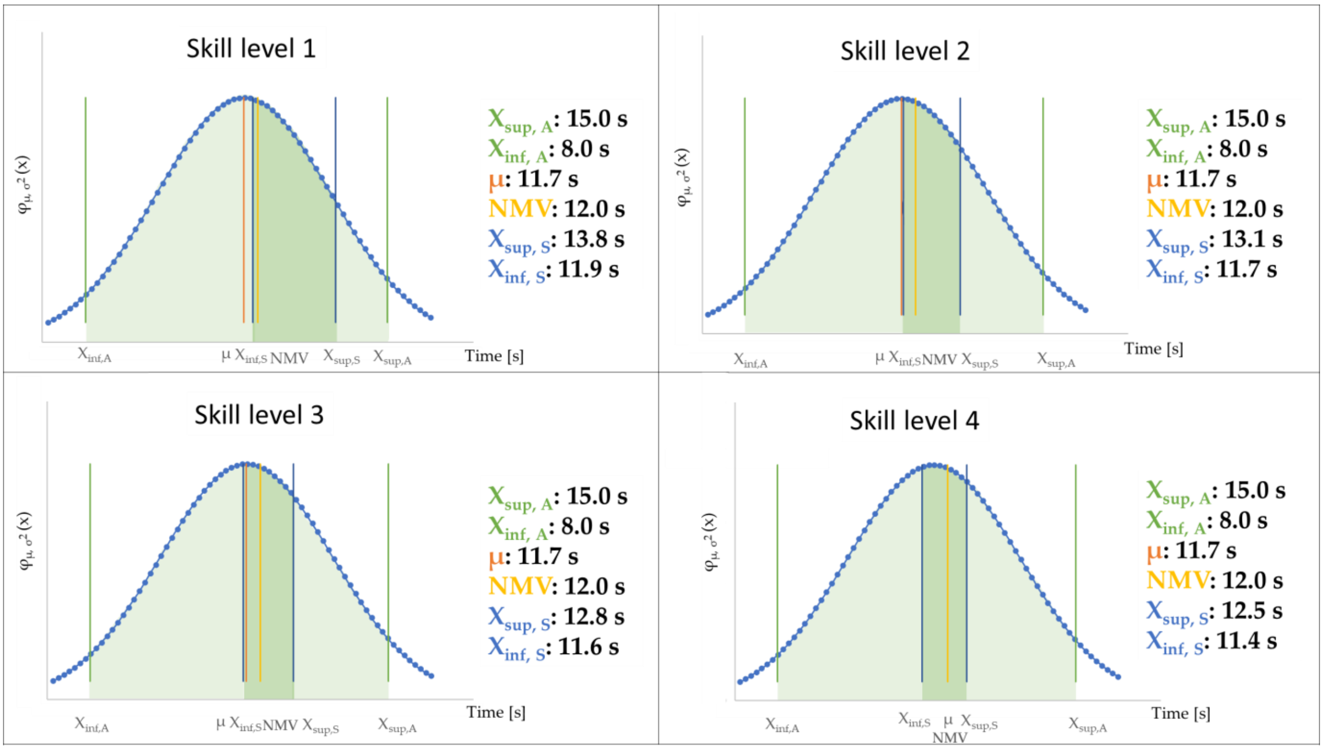

Figure 14. Gaussian distribution of the grasping in workstation A.

Figure 15 represents the drilling skill, where the mean coincides with the NMV.

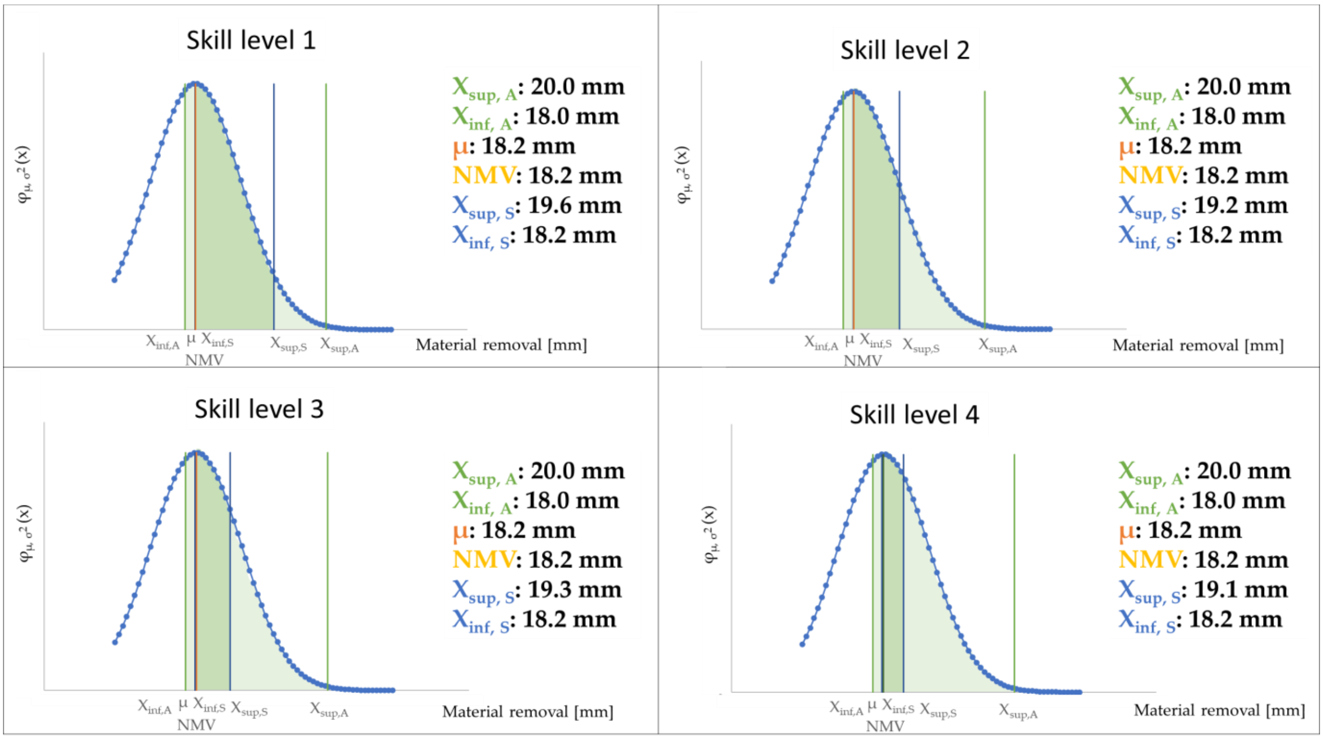

Figure 15. Gaussian distribution of drilling.

Figure 16 shows the grasping abilities in WS B. Mean and MNV values fall apart, which can be translated as either the NMV is incorrectly calculated or there is a reason that it needs to be analyzed by the experts in the line, which would explain why the workers cannot fulfill the requirements.

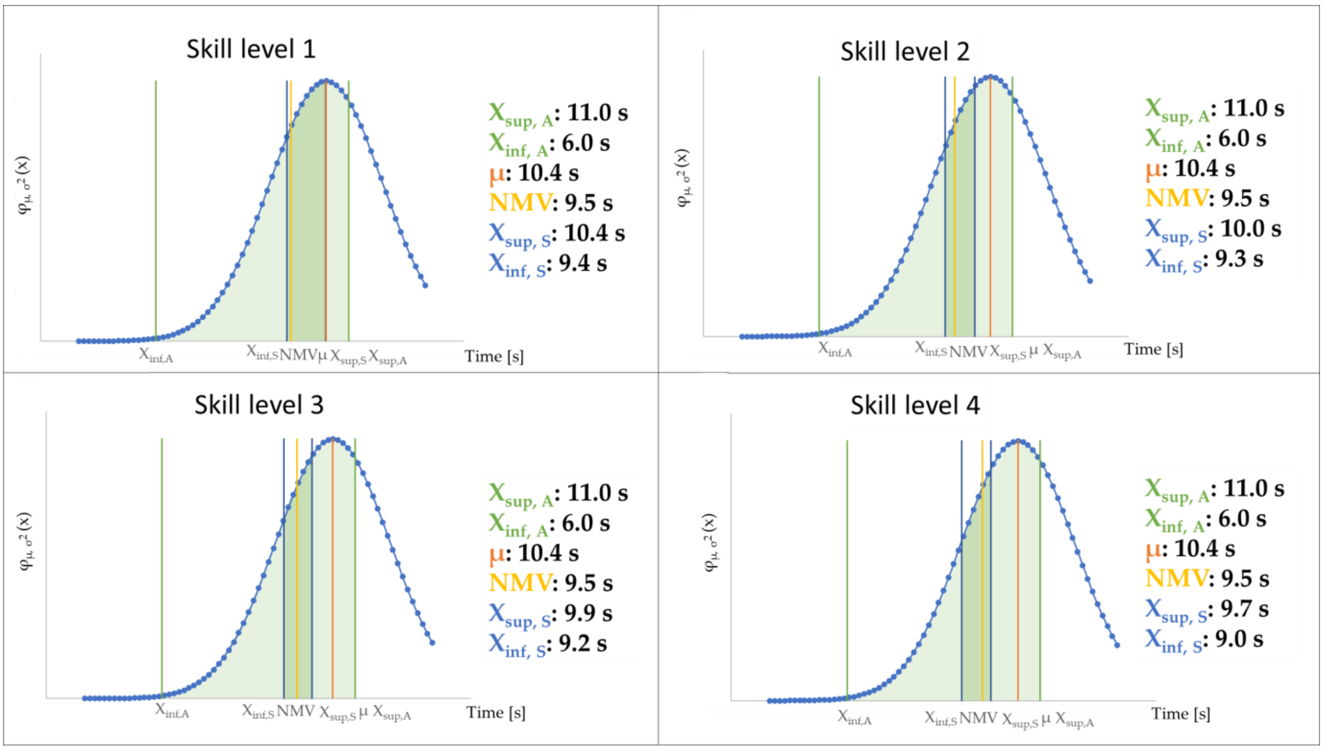

Figure 16. Gaussian distribution of the grasping in workstation B.

Figure 17 introduces the welding skill graphs that fall in the same case as shown in Figure 15.

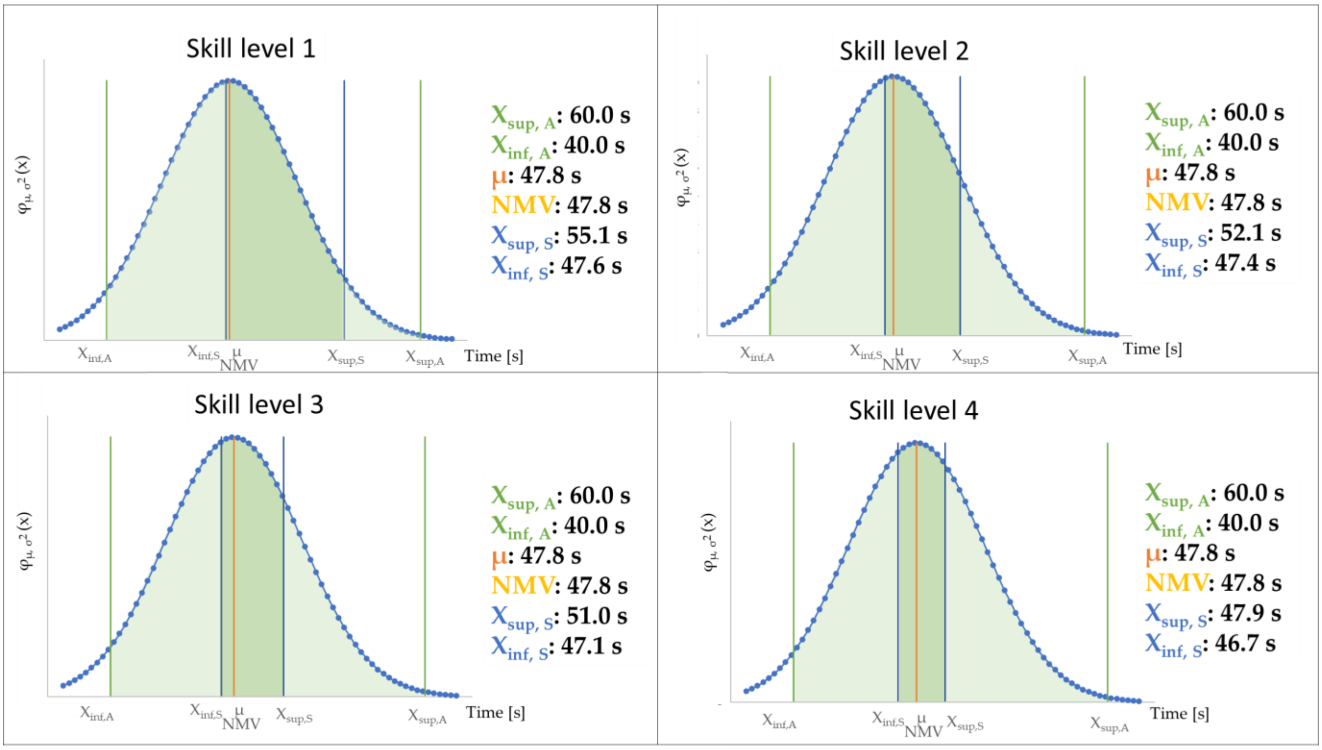

Figure 17. Gaussian distribution of welding.

Figure 18 shows the grasping abilities at station C, for which the scenario coincides with the one shown in Figure 16.

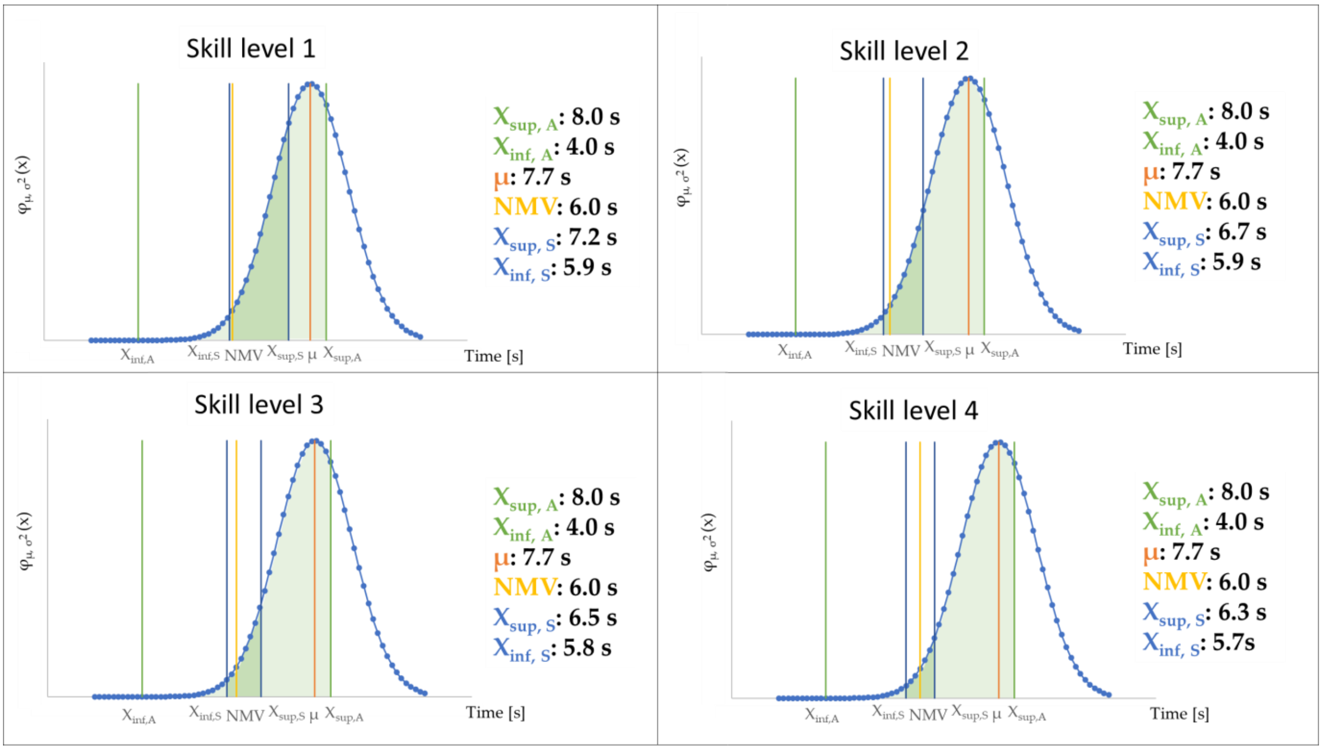

Figure 18. Gaussian distribution of the grasping in workstation C.

Figure 19 presents the gluing skills of the operators, for the same scenario as seen in Figure 14.

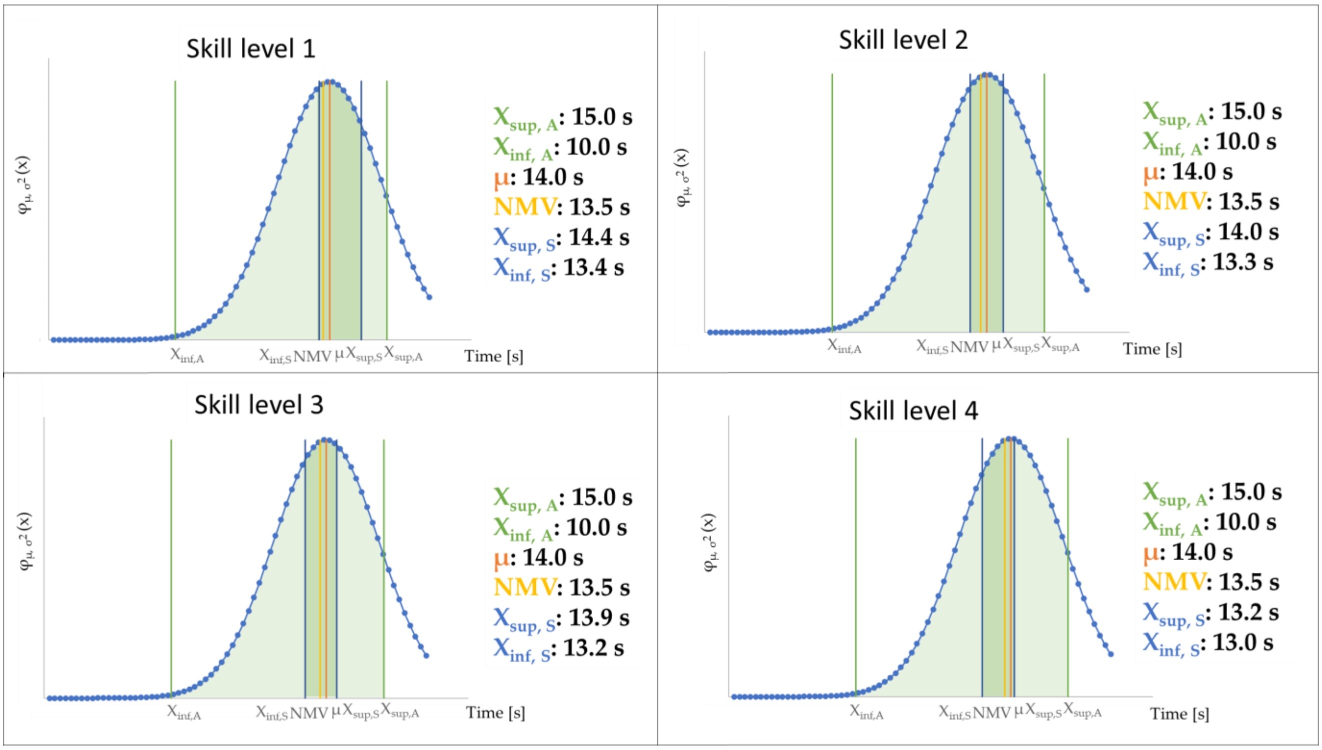

Figure 19. Gaussian distribution of gluing.

Last, Figure 20 presents the visual inspection skills of the operators, which coincides with the type of case as in Figure 14. All values for each skill and skill level are summarized in Table 5.

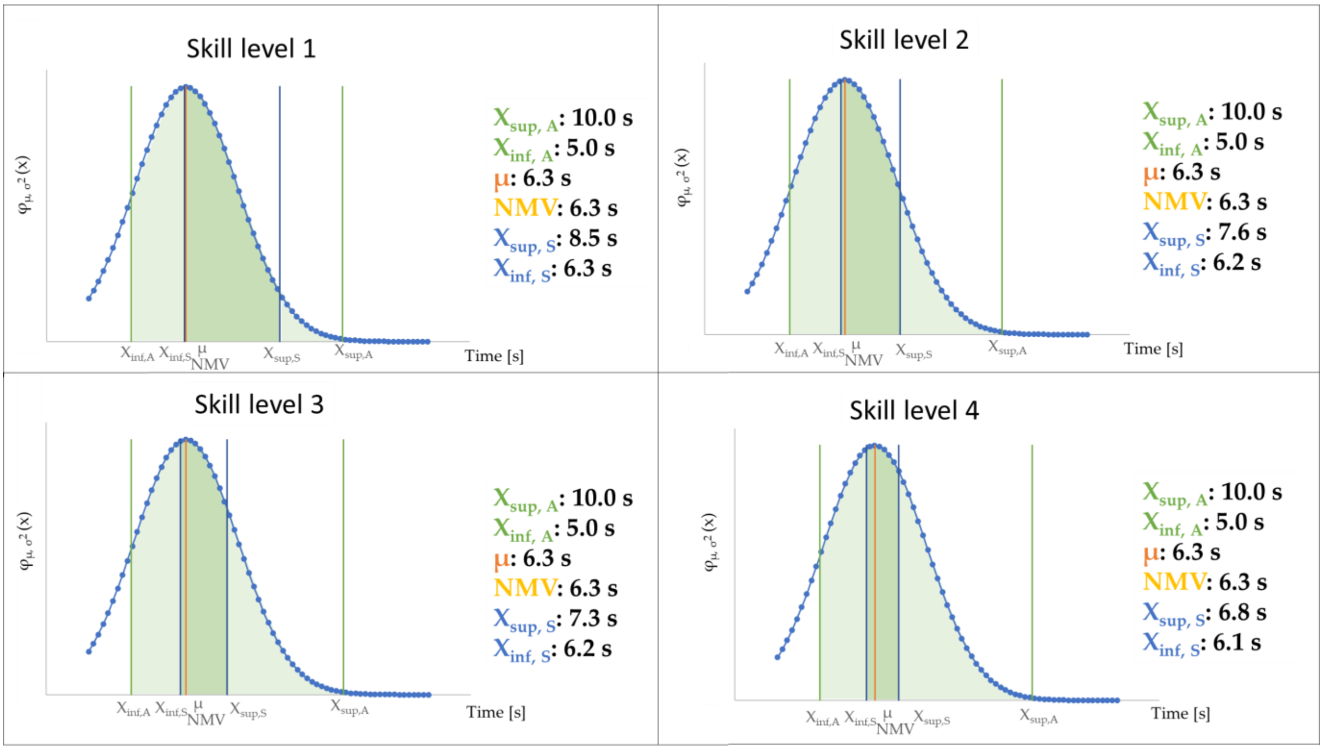

Figure 20. Gaussian distribution of the visual inspection.

Table 5. Summary if the intervals of the graphs considered in Figures 13–19.

Skill: Grasp in WS 1						
Skill Level	**$X_{sup,A}$ [s]**	**$X_{inf,A}$ [s]**	**$X_{sup,S}$ [s]**	**$X_{inf,S}$ [s]**	**Mean [s]**	**NMV [s]**
1	15.0	8.0	13.8	11.9	11.7	12.0
2	15.0	8.0	13.1	11.7	11.7	12.0
3	15.0	8.0	12.8	11.6	11.7	12.0
4	15.0	8.0	12.5	11.4	11.7	12.0
Skill: Drill in WS 1						
Skill Level	**$X_{sup,A}$ [mm]**	**$X_{inf,A}$ [mm]**	**$X_{sup,S}$ [mm]**	**$X_{inf,S}$ [mm]**	**Mean [mm]**	**NMV [mm]**
1	20.0	18.0	19.6	18.2	18.2	18.2
2	20.0	18.0	19.2	18.2	18.2	18.2
3	20.0	18.0	19.3	18.2	18.2	18.2
4	20.0	18.0	19.1	18.2	18.2	18.2
Skill: Grasp in WS 2						
Skill Level	**$X_{sup,A}$ [s]**	**$X_{inf,A}$ [s]**	**$X_{sup,S}$ [s]**	**$X_{inf,S}$ [s]**	**Mean [s]**	**NMV [s]**
1	11	6	10.4	9.4	10.4	9.5
2	11	6	10.0	9.3	10.4	9.5
3	11	6	9.9	9.2	10.4	9.5
4	11	6	9.7	9.0	10.4	9.5
Skill: Weld in WS 2						
Skill Level	**$X_{sup,A}$ [s]**	**$X_{inf,A}$ [s]**	**$X_{sup,S}$ [s]**	**$X_{inf,S}$ [s]**	**Mean [s]**	**NMV [s]**
1	60.0	40.0	55.1	47.6	47.8	47.8
2	60.0	40.0	52.1	47.4	47.8	47.8
3	60.0	40.0	51.0	47.1	47.8	47.8
4	60.0	40.0	47.9	46.7	47.8	47.8
Skill: Grasp in WS 3						
Skill Level	**$X_{sup,A}$ [s]**	**$X_{inf,A}$ [s]**	**$X_{sup,S}$ [s]**	**$X_{inf,S}$ [s]**	**Mean [s]**	**NMV [s]**
1	8.0	4.0	7.2	5.9	7.7	6.0
2	8.0	4.0	6.7	5.9	7.7	6.0
3	8.0	4.0	6.5	5.8	7.7	6.0
4	8.0	4.0	6.3	5.7	7.7	6.0
Skill: Glue in WS 3						
Skill Level	**$X_{sup,A}$ [s]**	**$X_{inf,A}$ [s]**	**$X_{sup,S}$ [s]**	**$X_{inf,S}$ [s]**	**Mean [s]**	**NMV [s]**
1	15.0	10.0	14.4	13.4	14.0	13.5
2	15.0	10.0	14.0	13.3	14.0	13.5
3	15.0	10.0	13.9	13.2	14.0	13.5
4	15.0	10.0	13.2	13.0	14.0	13.5
Skill: Visual inspection in WS 3						
Skill Level	**$X_{sup,A}$ [s]**	**$X_{inf,A}$ [s]**	**$X_{sup,S}$ [s]**	**$X_{inf,S}$ [s]**	**Mean [s]**	**NMV [s]**
1	10.0	5.0	8.5	6.3	6.3	6.3

2	10.0	5.0	7.6	6.2	6.3	6.3
3	10.0	5.0	7.3	6.2	6.3	6.3
4	10.0	5.0	6.8	6.1	6.3	6.3

In Section 4.1., it was commented that workstations should be ranked in terms of complexity. After applying the method explained in [52], the most complex workstation is C, followed by B, and finally A. Then, data is classified and structured constantly following the format of Algorithms 2 and 3 until the CEP detects the break. In order to detect the breaks in this use case, a specific logic is developed. In a high level, it has stored the schedule of the factory daily breaks. When the time of the break is close, the machines are expected to stop. This way, an interval of time is defined to all machines to stop working. As soon as this event happens, the algorithm recognizes it as a break. In this experiment, the duration of the data processed was 1 h and 50 min., meaning that the performance of the workers is classified and structured during this time before the break to offer a suggestion of rotation after it. The data assignment is presented in Table 6.

Table 6. Implementation data assignment which shows the count of tested skills during the specified duration.

Workstation	Skill	Importance Index	Total Acceptable Excellent	Total Acceptable Good	Total Acceptable Admissible	Total Marginal Acceptable	Total Not Acceptable
A	Grasp	0.3	0	23	32	32	21
A	Drill	0.7	0	0	15	41	52
B	Grasp	0.3	45	22	19	20	2
B	Weld	0.7	67	18	13	7	3
C	Grasp	0.3	3	9	59	30	7
C	Glue	0.6	22	29	20	34	3
C	Visual Inspection	0.1	13	11	27	45	12

The numbers shown in the table represent how many times, during the data processing time, the operator displayed the skill in which rank. Following the procedure shown in Algorithm 3, the assignment of the workers is carried out. The activities with the highest importance per station index are drilling, welding, and gluing, so the first round of the algorithm is going to be based on those skills. In case different workers present the same performance for the most significant skill in their respective workstations, the second skill with the highest importance index of the remaining is checked. Emulating the criteria, worker 100 should work in WS A for the next turn, but as this operator has already performed in that station, is assigned to workstation B. Then, operator 103 is placed in workstation A and worker 102 will continue at station C.

The research shown in this article's literature review proves that human representation and scheduling is becoming a more important topic to address in the manufacturing digital era. Not only to manage the company's personnel as available resources, but also to consider their environment to facilitate their work well-being and optimize their performance. In this way, the examples offered provide solutions to manage the company operators and rotate them between workdays, respecting different criteria. The approach presented in this article proposes to rotate workers taking into account their performance. This methodology helps the production line supervisor to manage worker rotations efficiently. Its advantage with respect to commercial solutions is that it proposes rotations within the daily work shift itself to optimize to the level of the working day.

5. Conclusions

The presented approach introduced a way to include human factors in digital twins. To keep the human-in-the-loop, the proposed approach considered the worker's performance assessments based on both their previous work performance and their fit to perform the needed tasks. To perform that, a multi-attribute skill matrix is developed to measure the level of the worker´s abilities. The process descriptions of the workstations are related to measurable skills; therefore, work data can be analyzed and processed in real time. Then, a ruled-based algorithm is applied to make the assessment. The conducted literature review revealed that digital twins do not often include human factors, since they require complex modeling and DTs focused on assets-oriented refinement. Additionally, a good balancing and sequencing of the tasks can easily improve workers' performance, facilitating their ergonomics and well-being. The combination of these two research directions is shown in an implementation case, which proves that workers can be reassigned workstations based on the mentioned criteria. Regarding future work, the approach can be enhanced by deleting the assumption that one task is related to one skill and exploring the multi-skilling task definition to increase the depth of the quality of the data in the human skills modeling engine. Additionally, new model distributions could be tested to adjust the data stored.

Author Contributions: Conceptualization, A.L.A., W.M.M., S.A. and J.L.M.L.; methodology, A.L.A., W.M.M. and S.A.; software, A.L.A. and T.V.; validation, A.L.A. and T.V.; formal analysis, A.L.A. and W.M.M.; investigation, A.L.A., W.M.M., S.A., T.V. and J.L.M.L.; resources, A.L.A.; data curation, A.L.A. and T.V.; writing—original draft preparation, A.L.A. and W.M.M.; writing—review and editing, A.L.A., W.M.M. and J.L.M.L.; visualization, A.L.A.; supervision, W.M.M. and J.L.M.L.; project administration, S.A. and J.L.M.L.; funding acquisition, J.L.M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 873087. The results obtained in this work reflect only the authors' views and not those of the European Commission. The Commission is not responsible for any use that may be made of the information they contain.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mohammed, W.M.; Ramis Ferrer, B.; Iarovyi, S.; Negri, E.; Fumagalli, L.; Lobov, A.; Martinez Lastra, J.L. Generic Platform for Manufacturing Execution System Functions in Knowledge-Driven Manufacturing Systems. *Int. J. Comput. Integr. Manuf.* **2018**, *31*, 262–274. https://doi.org/10.1080/0951192X.2017.1407874.
2. Cortés-Leal, A.; Cárdenas, C.; Del-Valle-Soto, C. Maintenance 5.0: Towards a Worker-in-the-Loop Framework for Resilient Smart Manufacturing. *Appl. Sci.* **2022**, *12*, 11330. https://doi.org/10.3390/app122211330.
3. Toichoa Eyam, A.; Mohammed, W.M.; Martinez Lastra, J.L. Emotion-Driven Analysis and Control of Human-Robot Interactions in Collaborative Applications. *Sensors* **2021**, *21*, 4626. https://doi.org/10.3390/s21144626.
4. Longo, F.; Mirabelli, G.; Solina, V.; Alberto, U.; De Paola, G.; Giordano, L.; Ziparo, M. A Simulation-Based Framework for Manufacturing Design and Resilience Assessment: A Case Study in the Wood Sector. *Appl. Sci.* **2022**, *12*, 7614. https://doi.org/10.3390/app12157614.
5. Grieves, M. *Origins of the Digital Twin Concept*; Florida Institute of Technology: Melbourne, FL, USA, 2016.
6. Mohammed, W.M.; Haber, R.E.; Martinez Lastra, J.L. Ontology-Driven Guidelines for Architecting Digital Twins in Factory Automation Applications. *Machines* **2022**, *10*, 861. https://doi.org/10.3390/machines10100861.
7. Kritzinger, W.; Karner, M.; Traar, G.; Henjes, J.; Sihn, W. Digital Twin in Manufacturing: A Categorical Literature Review and Classification. *IFAC-Pap.* **2018**, *51*, 1016–1022. https://doi.org/10.1016/j.ifacol.2018.08.474.
8. Negri, E.; Fumagalli, L.; Macchi, M. A Review of the Roles of Digital Twin in CPS-Based Production Systems. *Procedia Manuf.* **2017**, *11*, 939–948. https://doi.org/10.1016/j.promfg.2017.07.198.
9. Garetti, M.; Rosa, P.; Terzi, S. Life Cycle Simulation for the Design of Product–Service Systems. *Comput. Ind.* **2012**, *63*, 361–369. https://doi.org/10.1016/j.compind.2012.02.007.

10. Koulamas, C.; Kalogeras, A. Cyber-Physical Systems and Digital Twins in the Industrial Internet of Things [Cyber-Physical Systems]. *Computer* **2018**, *51*, 95–98. https://doi.org/10.1109/MC.2018.2876181.
11. Durão, L.F.C.S.; Haag, S.; Anderl, R.; Schützer, K.; Zancul, E. Digital Twin Requirements in the Context of Industry 4.0. In *Proceedings of the Product Lifecycle Management to Support Industry 4.0*; Chiabert, P., Bouras, A., Noël, F., Ríos, J., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 204–214.
12. Deshpande, A.K. *Making Apparel Manufacturing Lean*; Apparel Resources Pvt. Ltd.: New Delhi, India, 2021.
13. Tao, F.; Zhang, H.; Liu, A.; Nee, A.Y.C. Digital Twin in Industry: State-of-the-Art. *IEEE Trans. Ind. Inform.* **2019**, *15*, 2405–2415. https://doi.org/10.1109/TII.2018.2873186.
14. Opoku, D.-G.J.; Perera, S.; Osei-Kyei, R.; Rashidi, M. Digital Twin Application in the Construction Industry: A Literature Review. *J. Build. Eng.* **2021**, *40*, 102726. https://doi.org/10.1016/j.jobe.2021.102726.
15. Jiang, Y.; Yin, S.; Li, K.; Luo, H.; Kaynak, O. Industrial Applications of Digital Twins. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2021**, *379*, 20200360. https://doi.org/10.1098/rsta.2020.0360.
16. Cimino, C.; Negri, E.; Fumagalli, L. Review of Digital Twin Applications in Manufacturing. *Comput. Ind.* **2019**, *113*, 103130. https://doi.org/10.1016/j.compind.2019.103130.
17. Al-Yacoub, A.; Eaton, W.; Zimmer, M.; Buerkle, A.; Ariansyah, D.; Erkoyuncu, J.A.; Lohse, N. Investigating the Impact of Human In-the-Loop Digital Twin in an Industrial Maintenance Context. In Proceedings of the TESConf 2020—9th International Conference on Through-Life Engineering Services, Bedford, UK, 3–4 November 2020.
18. Bilberg, A.; Malik, A.A. Digital Twin Driven Human–Robot Collaborative Assembly. *CIRP Ann.* **2019**, *68*, 499–502. https://doi.org/10.1016/j.cirp.2019.04.011.
19. Michalos, G.; Kousi, N.; Karagiannis, P.; Gkournelos, C.; Dimoulas, K.; Koukas, S.; Mparis, K.; Papavasileiou, A.; Makris, S. Seamless Human Robot Collaborative Assembly—An Automotive Case Study. *Mechatronics* **2018**, *55*, 194–211. https://doi.org/10.1016/j.mechatronics.2018.08.006.
20. Berti, N.; Finco, S. Digital Twin and Human Factors in Manufacturing and Logistics Systems: State of the Art and Future Research Directions. *IFAC Pap.* **2022**, *55*, 1893–1898. https://doi.org/10.1016/j.ifacol.2022.09.675.
21. Liu, M.; Fang, S.; Dong, H.; Xu, C. Review of Digital Twin about Concepts, Technologies, and Industrial Applications. *J. Manuf. Syst.* **2021**, *58*, 346–361. https://doi.org/10.1016/j.jmsy.2020.06.017.
22. Xu, X.; Lu, Y.; Vogel-Heuser, B.; Wang, L. Industry 4.0 and Industry 5.0—Inception, Conception and Perception. *J. Manuf. Syst.* **2021**, *61*, 530–535. https://doi.org/10.1016/j.jmsy.2021.10.006.
23. Shahrokhi, M.; Bernard, A. A Framework to Develop an Analysis Agent for Evaluating Human Performance in Manufacturing Systems. *CIRP J. Manuf. Sci. Technol.* **2009**, *2*, 55–60.
24. Murphy, K.R. Job Performance and Productivity. In *Psychology in Organizations*; Psychology Press: London, UK, 1990; ISBN 9780203771891.
25. Akbari, A.; Maniei, R. The Effect of Job Rotation on Employee Performance. *Res. J. Manag. Rev.* **2017**, *6*, 21–26.
26. Michalos, G.; Makris, S.; Chryssolouris, G. The Effect of Job Rotation during Assembly on the Quality of Final Product. *CIRP J. Manuf. Sci. Technol.* **2013**, *6*, 187–197. https://doi.org/10.1016/j.cirpj.2013.03.001.
27. Diamantidis, A.D.; Chatzoglou, P. Factors Affecting Employee Performance: An Empirical Approach. *Int. J. Product. Perform. Manag.* **2018**, *68*, 171–193. https://doi.org/10.1108/IJPPM-01-2018-0012.
28. Triggs, D.D.; King, P.M. Job Rotation. *Prof. Saf.* **2000**, *45*, 32–34.
29. Simoneau, S.; St-Vincent, M.; Chicoine, D. *Work-Related Musculoskeletal Disorders (WMSDs)*; IRSST: Montreal, QC, USA, 1996.
30. Colombini, D. *Risk Assessment and Management of Repetitive Movements and Exertions of Upper Limbs: Job Analysis, Ocra Risk Indicies, Prevention Strategies and Design Principles*; Elsevier: Amsterdam, The Netherlands, 2002; ISBN 9780080540603.
31. Moussavi, S.E.; Zare, M.; Mahdjoub, M.; Grunder, O. Balancing High Operator's Workload through a New Job Rotation Approach: Application to an Automotive Assembly Line. *Int. J. Ind. Ergon.* **2019**, *71*, 136–144. https://doi.org/10.1016/j.ergon.2019.03.003.
32. Tharmmaphornphilas, W.; Norman, B.A. A Methodology to Create Robust Job Rotation Schedules. *Ann. Oper. Res.* **2007**, *155*, 339–360. https://doi.org/10.1007/s10479-007-0219-8.
33. Carnahan, B.J.; Redfern, M.S.; Norman, B.A. A Genetic Algorithm for Designing Job Rotation Schedules Considering Ergonomic Constraints. In Proceedings of the Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406), Washington, DC, USA, 6–9 July 1999; Vol. 2, pp. 1093–1098.
34. Carnahan, B.J.; Redfern, M.S.; Norman, B. Designing Safe Job Rotation Schedules Using Optimization and Heuristic Search. *Ergonomics* **2000**, *43*, 543–560. https://doi.org/10.1080/001401300184404.
35. McClellan, M. *Applying Manufacturing Execution Systems*; CRC Press: Boca Raton, FL, USA, 1997; ISBN 9781574441352.
36. Home. Available online: https://mesa.org/ (accessed on 20 January 2023).
37. VDI.de. Available online: https://www.vdi.de/ (accessed on 20 January 2023).
38. Iarovyi, S.; Mohammed, W.M.; Lobov, A.; Ferrer, B.R.; Lastra, J.L.M. Cyber–Physical Systems for Open-Knowledge-Driven Manufacturing Execution Systems. *Proc. IEEE* **2016**, *104*, 1142–1154. https://doi.org/10.1109/JPROC.2015.2509498.
39. Plex. Production Planning, Scheduling, and Management. Available online: https://www.plex.com/products/manufacturing-execution-system/production-planning-scheduling-and-management-software (accessed on 23 January 2023).
40. Siemens Software. Opcenter Execution Process—Manufacturing Execution System (MES). Available online: https://www.plm.automation.siemens.com/global/en/products/opcenter/execution-process.html (accessed on 23 January 2023).

41. GE Digital. Smart Factory MES. Manufacturing Execution Systems. Available online: https://www.ge.com/digital/applications/manufacturing-execution-systems (accessed on 23 January 2023).
42. SAP Manufacturing Execution Features and Benefits. MES Solutions & Production Planning. Available online: https://www.sap.com/products/scm/execution-mes/features.html (accessed on 23 January 2023).
43. Customer | ATOSS AG Available online: https://www.atoss.com/en/customers (accessed on 23 January 2023).
44. Optimize Labor Productivity and Compliance. Available online: https://www.infor.com/solutions/people/wfm (accessed on 23 January 2023).
45. Online Rota Planning Made Easy for Shift Planners. Available online: https://www.papershift.com/en (accessed on 23 January 2023).
46. ABC Roster—A Free Software Application for Employee Shift Scheduling. Available online: https://www.abc-roster.com/ (accessed on 23 January 2023).
47. When2Work.Com—WhenToWork Status. Available online: https://when2work.com (accessed on 23 January 2023).
48. Isometric-Based Approach for Detecting Localized Muscular Fatigue during Complex Dynamic Manufacturing Operations. IEEE Conference Publication. IEEE Xplore. Available online: https://ieeexplore.ieee.org/abstract/document/9551478 (accessed on 10 November 2022).
49. Tharmmaphornphilas, W.; Norman, B.A. A Quantitative Method for Determining Proper Job Rotation Intervals. *Ann. Oper. Res.* **2004**, *128*, 251–266. https://doi.org/10.1023/B:ANOR.0000019108.15750.ae.
50. Thun, J.-H.; Lehr, C.B.; Bierwirth, M. Feel Free to Feel Comfortable—An Empirical Analysis of Ergonomics in the German Automotive Industry. *Int. J. Prod. Econ.* **2011**, *133*, 551–561. https://doi.org/10.1016/j.ijpe.2010.12.017.
51. Khalil, H.; Byrne, A.; Ristevski, E. The Development and Implementation of a Clinical Skills Matrix to Plan and Monitor Palliative Care Nurses' Skills. *Collegian* **2019**, *26*, 634–639. https://doi.org/10.1016/j.colegn.2019.05.002.
52. Zeltzer, L.; Limère, V.; Aghezzaf, E.-H.; Landeghem, H.V. Measuring the Objective Complexity of Assembly Workstations. In Proceedings of the In Proceedings of the Measuring the Objective Complexity ICCGI 2012, The Seventh International Multi-Conference on Computing in the Global Information Technology, Venice, Italy, 24–29 June 2012; p. 6.
53. Mattsson, S.; Tarrar, M.; Gullander, P.; Van Landeghem, H.; Gabriel, L.; Limère, V.; Aghezzaf, E.-H.; Fasth, F.-B., Åsa; Stahre, J. Comparing Quantifiable Methods to Measure Complexity in Assembly. *Int. J. Manuf. Res.* **2013**, *9*, 112–130. https://doi.org/10.1504/IJMR.2014.059602.