



# Empowering Digital Twin for Industry 4.0 using metaheuristic optimization algorithms: case study PCB drilling optimization

David Balderas<sup>1</sup> · Alexandro Ortiz<sup>1</sup> · Efraín Méndez<sup>1</sup> · Pedro Ponce<sup>1</sup> · Arturo Molina<sup>1</sup>

Received: 24 September 2020 / Accepted: 15 January 2021 / Published online: 3 February 2021  
© The Author(s), under exclusive licence to Springer-Verlag London Ltd. part of Springer Nature 2021

## Abstract

The fourth industrial revolution, Industry 4.0, has been characterized by novel concepts introduction in manufacturing systems that enable smart factories with vertically and horizontally communication to improve their performance. Many virtual systems allow to predict foul conditions, save energy, study special cases, and so on, yet they need to implement new digital tools that allow developing manufacturing process in a better manner. As a result, Digital-Twin platforms are a good alternative since they are virtual models that could receive online and offline data. Thus, programmed algorithms can be evaluated to know the performance of the manufacturing process. These virtualizations and interconnections between elements of the manufacturing process become important components with an increasing role in dealing with supply, production times, and delivery chains as they run in parallel and find optimal performance before implementing these conditions into the real system. This study focuses on the use of a Digital-Twin that integrates a metaheuristic optimization and a direct Simulink model for printed circuit boards (PCB) design and processing focused on the drilling process. The results show that metaheuristic optimization can be integrated into the Digital-Twin concept as part of the production system into the drilling process. In the first part, it shows that depending on the penalization the optimization focuses on the lower path and forgets on changing the tools, yet as the penalization raises it focuses on finishing drilling with one tool before changing. Second, it is important where on the PCB it starts the drilling, with less time depending on each plaque. Third, it can be observed that using optimization can triple the amount of PCBs that can be manufactured. Finally, on an 8-hr run the Digital-Twin that didn't use optimization can only work with three different designs, differently with optimization it can have 7-8 changes in the PCB design.

**Keywords** Digital Twin · Industry 4.0 · Optimization · Simulation

## 1 Introduction

The marketplace is so demanding that companies must change their procedures and approaches to stand on top. This comes from the fact that new products are being demanded by users. Furthermore, competitors are delivering products with new features that only some of them are appealing to the users. Hence, new product development (NPD) requires quickening its process to stay on top, and in turn, be competitive.

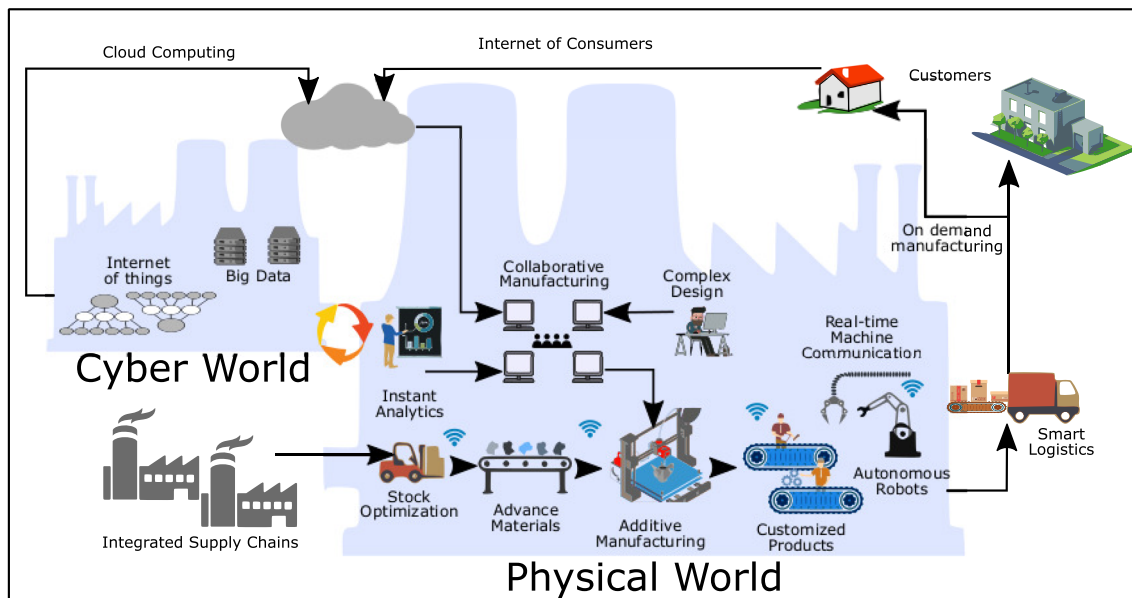
Technological advances have caused growth in factory productivity since the industrial revolution. It happens

in the fourth stage: the first, at the end of the 18th century, with the steam engine powering factories. The second, at the beginning of 20th century, with electricity responsible for mass production. It had iconic names such as Henry Ford and Frederick Taylor. The third revolution started in the 1970s with digital automation entering industries and benefited from the power of electronics and information technologies [1]. Currently, the fourth stage, or Industry 4.0, takes advantage of technologies with relationships between all stages of product development. Further, Industry 4.0, involve the physical world and the cyber world to create a Smart Factory (refer to Fig. 1). This factory needs components for monitoring, control, and likely interconnection to the cloud. It would result in a decentralized and optimizable factory [2].

Standard processes in system development are modeling and simulation (e.g., verify properties or reinforce decisions). Normally, simulation solutions optimize operations

✉ David Balderas  
dc.balderassilva@tec.mx

<sup>1</sup> Tecnológico de Monterrey National Department of Research, Puente 222, Del. Tlalpan, Mexico City, 14380, Mexico



**Fig. 1** Illustration of components and connections for a Industry 4.0 Smart Factory

and predict failure [3, 4]. Yet, they are commonly performed before the process and only updated after some fault appears which might result in flaws.

Further, as data is growing from the production process, an important feature of Industry 4.0 is that it requires to analyze what data is useful to improve the manufacturing process [5]. For example, [5] shows that it is important to analyze which data is useful to improve the manufacturing process. Additionally, in Industry 4.0 to improve the quality it requires viewing the system as a multi-tiered system that requires optimizing its data. For example, having machine data, inner sensors, external sensors, and the human component requires to detect which of the data is relevant and from this data develop a quality monitoring system determining whether a part is within specifications or out of specifications [6].

As an alternative, Industry 4.0 uses Digital-Twins for prediction and prevention [7, 8]. Manufacturing systems couple with its digital equivalence to predict errors with a small delay between data acquisition and response. Additionally, the use of a Digital-Twin can simulate various scenarios, exploiting synchronizations with the sensors and provide a virtual representation of systems [8, 9]. Digital Twin is thought of as the next stream of modeling, simulation, and optimization technology [10]. A definition is given by Glassgen and Stargel [11]: “digital twin is an integrated multi-physics, multi-scale, probabilistic simulation of a complex product and uses the best available physical models, sensor updates, etc., to mirror the life of its corresponding twin”.

Even though much literature on Digital-Twin, a broad concept and an agreement over its features and range has not been reached. For instance, whether the Digital Twin data flows with the physical in both directions (physical-to-virtual and vice versa) [8, 12]. Or if the Digital-Twin would act on the control system of its physical counterpart [12], or if the control must adapt immediately the parameters or only after an event has occur [13]. Further, how much information the Digital-Twin pass, the communication speed, or if it is only an offline simulation approach [14].

A smart factory process that can benefit from a Digital-Twin (using simulations and optimization) is the Computer numerical control (CNC) machine tool (CNCMT) [15]. According to Luo et al. [15] CNCMT must use precise simulations using design parameter and actual working conditions; must be self-sensing of its conditions; should self-adjust to produce in less time, less waste and better quality; should self-predict faults timely before any serious fault; and should self-assess its status, optimize working parameters, and make decisions based on machine learning. Moreover, CNCMT is autonomous robots, and using simulations can find better paths and optimize their process.

All those requirements are quite time and resource consuming. For this reason, the use of a Digital-Twin can be beneficial by exchanging information back and forth with the CNCMT about the process (e.g., position, movements, sensors, and information of energy consumption) and update the control of the CNCMT immediately or pass the data for the machining of the next board. In other words, the Digital-Twin can simulate the process several times and

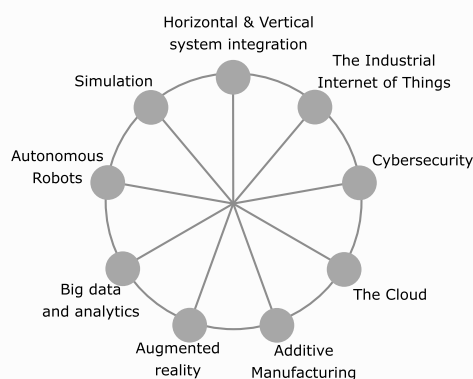
find the optimal parameters and send them back to the CNCMT (while was performing other activities) and correct its behavior. Moreover, the Digital-Twin creates a path for the cyber-physical integration in manufacturing, which is important for smart manufacturing [16].

This work presents how a metaheuristic optimization algorithm helps printed circuit board (PCB) manufacturing, a Simulink implementation that uses scheduling and workers that can work simultaneously for a Digital-Twin. Furthermore, this work presents a case of smart manufacturing using CNCMT with the use of synchronized simulation with optimization of the drilling process of three-phase inverter PCB. The drilling process besides having many holes in different positions it also has different diameters that need a change of drilling tool. This causes three optimization requirements: first, there is a need to optimize the path taken and travel the shortest distance. Second, there are cases it is useful to change many times the tool and focus on traveling the least distance (changing tools is almost automatically), yet, there are situations that it is useful to stay with the same tool for a longer period since changing it would be time and energy-consuming. And third, that the machine could change to new designs for products and optimize for the new conditions.

## 2 Industry 4.0

Industry 4.0 is driven by nine foundations or interconnected technology advances [17] (see Fig. 2): (1) autonomous robots, (2) simulation, (3) horizontal and vertical system integration, (4) the Industrial Internet of things, (5) cybersecurity, (6) the cloud, (7) additive manufacturing, (8) augmented reality, and (9) big data and analytics.

These 9 technological advances can work together to help to develop and speed up the process. An example of interconnectivity is a machining tool that uses autonomous



**Fig. 2** The nine foundations of Industry 4.0

robots and additive manufacturing for some of its parts, moreover, it has its sensors connected to the network to verify the programmed trajectories. This information is also used for planning times and movements using simulations, optimization, the cloud, and big data analytics. Then, information is sent back to correct actuators' movements reducing energy consumption. Finally, all the different data transmissions need cybersecurity to guarantee safe communication.

### 2.1 Digital twin

The term Digital-Twin concerns a digital duplicate of physical entities that virtualize physical conditions. They can model, simulate, and optimize technology [10, 18] using the direct connection between the physical and virtual model, simulation can be done in real time. The information must send seamlessly to allow virtual and physical entities to exist together.

Digital-Twins integrate all 9 technological advances to create a digital simulation model that updates and changes as it receives information from their physical analog. Furthermore, Digital-Twins can analyze theoretical values of big data and real values to optimize, simulate, monitor, and verify system operations [19]. Additionally, if the Digital-Twin is correctly implemented, it could have direct interaction with the supply chain and smart logistics. For example in Fig. 3 the physical component and the Digital-Twin share information from the sensors, from which the Digital-Twin can make simulations and perform optimization to improve the production. Further, the Digital-Twin can be improved using information from the cloud, supply chain, management, and smart logistics.

Some of the popular uses for Digital Twins in manufacturing include:

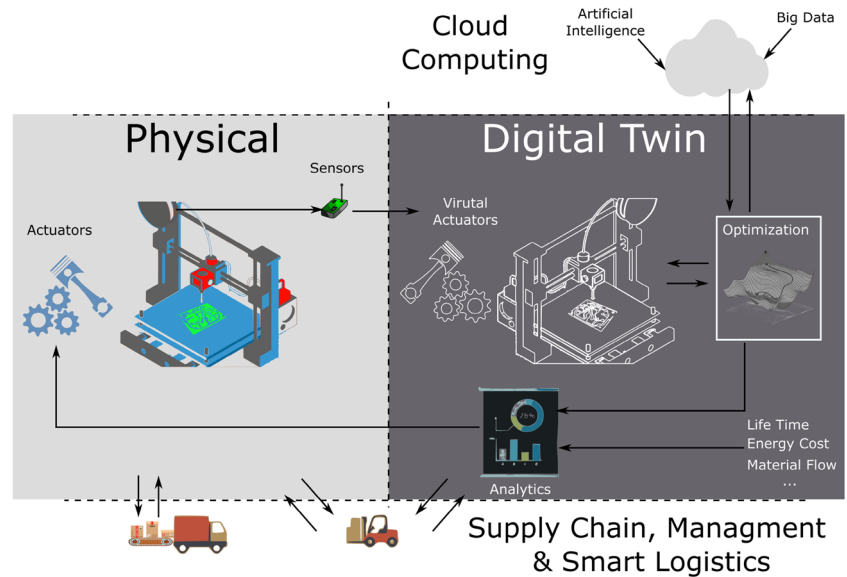
- **Quality management**

Continuous check of product data has clear benefits over random inspection in quality management. Using a Digital-Twin can track and model all the production process to determine where a quality problem might happen [20]. Also, analyzing the product materials to check whether there are better materials and/or production process can be used [21].

- **System planning/virtual start-up**

Historical analysis of similar systems allows the prediction of the system's performance that has not been built. Digital-Twins would use historical information to model various scenarios resembling the desired one and determine in what sections to enhance a factory [22]. Further, using cloud computing a data bank can be created with images of old machining information (e.g., images of past PCBs designs). Then,

**Fig. 3** The structure proposed for a Digital-Twin. In the proposal a physical device shares information from its sensors with the Digital-Twin, which uses it to make simulations and optimize the production



using these images and a classification algorithm (i.e., convolution neural network) can detect components and type of design, and suggest improvements [19].

#### – Logistics planning

The supply chain can be optimized with the help of a Digital-Twin by providing a clearer view of how the materials are being used and automatizing goods supply. For example, if the plant is working with lean manufacturing Digital-Twin can increase its efficiency [23].

#### – Product development

Digital-Twins can help developing new products using virtual simulations permitting to mix production information with other real-world information (e.g., customer experience) [19, 24].

#### – Product redesign

Adaptation in manufacturing to different products can run first in a Digital-Twin allowing the model to observe how much production will be affected and check how to adjust the process to the new design. This can be done using simulation on how the new product interacts with the existing equipment and optimizing its production time [25, 26].

As it can be seen in the previous list, the use of optimization and simulation is a high part of using Digital-Twin. The selected algorithm was Ant Colony Optimization, yet, this study does not focus on how one optimization algorithm is more beneficial than another, it focuses on how optimization is beneficial for a Digital-Twin. Hence, the algorithm is not important but is explained to show it requires some time to finish obtaining the optimal values.

Additionally, this work compares to other works such as [27] which uses the simulation of a beam to prove

the concept of interaction of Digital-Twin and its physical counterpart. Also, using the idea of [28] the physical-Twin can be observed, and using the Digital-Twin prediction can be made using simulation. Differently, this work besides using simulation and optimization to get the best performance it also integrates schedule, workers, and different design of PCB into the Digital-Twin.

## 3 Implementation

Most problems in real-life are complex and often are not solved easily, at least not analytically. For instance, a Traveling salesman problem (TSP), would not be simple since it is an NP-complete problem and they are hard to solve [29]. For example, for 52 drilling holes it would have  $51!/2 = 7.7556 \times 10^{65}$  different possible combinations of drilling holes and finding the best solution will be time-consuming. Thus, finding the best solution for a PCB that has hundreds of holes will be close to unfeasible.

Even so, often you need to get an operation value that, although it might not be the best one (global best), it is good enough for the problem (local best). Then, optimization algorithms try solving such problems finding “good-enough” solutions from a set of alternatives available.

### 3.1 Optimization

Moreover, optimization algorithms find minimums or maximums using heuristics to quicken exploration of the search space. As previously explained, TSP are a specific case of optimization problems where the main goal is to find the route with the shortest distance to visit a set of cities.

To solve the TSP problem one of the most used algorithms is the Ant Colony Optimization (ACO). ACO mimics how ants try to find the shortest path to food putting a trail of pheromones from the nest to the food source. The trail that is more explored would be the route with the shortest distance.

ACO algorithm could simply be described as follows (a more elaborate form can be found in [30])

- Initialize an ant colony
- Initialize pheromone trails and random attraction levels
- Repeat until a termination criterion
  - Choose for each ant a path with a probability  $P$
  - Advance to the next chosen state
  - Update the traces of pheromones of ants
  - Update pheromone attraction levels
- Return the best pheromone trail

Mathematically, all the ants will move from node  $i$  to node  $j$  as:

$$p_{i,j} = \frac{(\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)}{\sum (\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)} \tag{1}$$

with  $\alpha$  and  $\beta$  as the parameters to control the influence of  $\tau_{i,j}$  and  $\eta_{i,j}$  which are the amount of pheromone on edge  $i$  and  $j$  and how desirable is the edge  $i$  and  $j$ . The parameter to update the amount of pheromone is updated according to:

$$\tau_{i,j} = (1 - \rho)\tau_{i,j} + \Delta\tau_{i,j} \tag{2}$$

where  $\rho$  and  $\Delta\tau_{i,j}$  as the rate of pheromone evaporation and the amount of pheromone deposited, given by:

$$\Delta\tau_{i,j} = Q/L_k \quad \text{if ant } k \text{ travels on edge } i, j \\ = 0 \quad \text{otherwise} \tag{3}$$

where  $Q$  and  $L_k$  are the pheromones deposited constant and the cost of the  $k$ th ant’s tour, which is normally the distance. The pseudocode can be found in Fig. 4.

```

1: Set Function  $f(x)$ ,  $x = [x_1, \dots, x_d]$ .
2: Initialize tours  $\tau$ 
3: Define  $\alpha$ ,  $\beta$ ,  $\rho$  and  $Q$ 
4: while  $t < \text{Max iterations}$  do
5:   for each ant in the colony do
6:     Set all nodes as unvisited
7:     while Number of un-visited nodes  $> 0$  do
8:       Compute probability of going to un-visited
       nodes using 1
9:       Select next node accordingly
10:    end while
11:   end for
12:   Update pheromone level 2 and 3
13: end while
14: Return Best solution with the shortest ant tour  $L$ 
```

Fig. 4 Ant colony pseudocode

### 3.2 Cost equation

As seen before, this problem involves finding the shortest distance between each perforation, with the addition that each tool change adds extra time. Hence, this problem would be a mixture of TSP with some extra conditions for tool change.

As an initial solution, a first cost function includes a penalization for tool change with extra distance traveled. These are on the next following cost equation:

$$\mathcal{L}_{ij} = \frac{d_{ij} + P}{v} \tag{4}$$

where  $d_{ij}$  is the distance from node  $i$  to node  $j$ ,  $P$  is the penalization constant and  $v$  is the horizontal speed of the drilling tool. The distance was calculated as

$$d_{ij} = \sqrt{(X_j - X_i)^2 + (y_j - y_i)^2} \tag{5}$$

As a secondary test, the cost function was further developed and was implemented so it uses a changing point (or home) as a reference for the optimization. The resulting cost equation was

$$\mathcal{L}_{ij} = \begin{cases} \frac{d_{ij}}{v}, & \text{if no tool change is required} \\ T_{Tc}, & \text{if tool change is required} \end{cases} \tag{6}$$

where  $T_{Tc}$  as the time for tool changing defined as

$$T_{Tc} = \frac{d_{io}}{v} + \frac{d_{oj}}{v} + tc \tag{7}$$

where  $d_{io}$  and  $d_{oj}$  as the distance from node  $i$  to the tool changing point and the distance from the tool changing point to the node  $j$ , and  $tc$  as a time constant to change the tool and the energy consumption according to the transitory response of position into actuators.

Since controllers in actuators can achieve almost every position, usually the energy consumption is linked with the controller effort. This means, if the position controller requires to reach the reference position in a short period of time, the amount of energy required by the controller increases. So, there is a trade-off between the time to reach the position and the amount of energy spent; sometimes if the energy is high extra cooling must be included and the price of manufacturing the PCB increases. Hence, the value of  $tc$  must consider the time for changing tools as well as the time for reaching the reference position with the amount of energy demanded by the controller. To deal with it another optimization algorithm could run to find its optimal value. In this paper, the value of  $tc$  is fixed according to the conventional requirements without running an optimization algorithm. As a result,  $tc$  is defined as

$$tc = tct + trp \tag{8}$$

where  $tct$  is the time for changing the tool and  $trp$  is the time for reaching the position reference that equals the ratio



between the energy spent by the controller and the number of products required per day. And time for changing the tool, which is considered as a priority when there are more than 2 tools to change in a short period of time.

Once the problem has been adapted to just distance to optimize the perforations the ACO algorithm was implemented. ACO ran over 250 iterations with a population of 30 ants and learning parameters rate of evaporation  $\rho = 0.5$ , control of influence  $\alpha = 2$ , control parameter  $\beta = 6$ , and pheromone deposited constant  $Q = 1$ .

### 3.3 Simulink

In addition, to further elaborate on the Digital-Twin, the model was set in Simulink (see Fig. 5). The model had workers allocation, a part generator, a buffer to store not milled PCBs, a milling machine, and a conveyor belt to take the piece out.

To further elaborate, the milling machine had a sub-model consisting of: a system that gets the card, allocates a worker, waits for loading the PCB, release the worker, the milling machine, another worker allocation, waiting to unload the PCB, release the worker and finally it sends the finished PCB out. More in detail, the sub-model receives information about the holes' distribution to start making the plaques (Fig. 5b). This sub-model starts the work on the first PCB with a random drill sequence. At the same time, the sub-model starts the optimization process, which returns a new optimized drilling sequence before the first PCB finish that could be used for the rest of that batch. Further, to have information on the time a worker takes to load and unload a PCB, the sub-model uses worker allocation and deallocation with 40 s to finish each task (this time could be adjusted to the real loading time). Lastly, the system adds a random time of 5% of the time to finish the plaque to emulate any eventuality.

In this case, the model work used a predefined schedule that sends a new batch of similar PCBs to the drilling machine once it has finish processing all the previous batch plaques. Each PCB batch has random sizes, random amount, and position of holes. It is worth mentioning, that a random

**Table 1** Inverter's design parameters

Maximum power	1600 (W)
DC bus voltage	12–100 (V)
Logic voltage	1.2–6 (V)
Switching frequency	120 (kHz)
Turn-on propagation delay	125 (ns)
Turn-off propagation delay	105 (ns)

schedule generator was used for next-day planning that could be replaced with a real plant that processes the orders as they arrive.

### 3.4 Case study parameters

For this case study, the main goal of the designed inverted was to serve as a driver that allows controlling BLDC motors. Table 1 shows the main parameters for the implemented circuit.

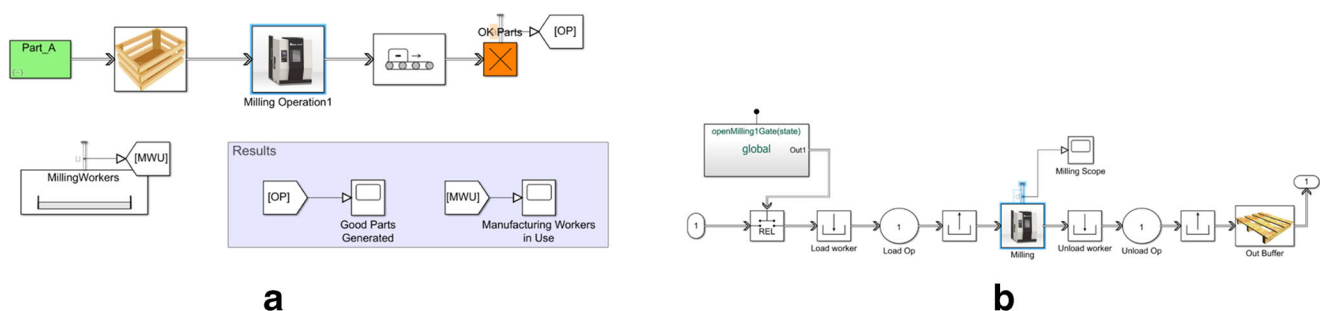
As early stage on the design was implemented in a surface mounting circuit. The components are dual in-line packages (DIP), which had to either be through-hole mounted or inserted in a socket in a Printed Circuit Board (PCB).

The PCB was manufactured with DIP components that need different perforation diameters. For automation, the PCB goes through a CNC machine that must change the tools for every diameter.

The main problem is to go through all the points in conjunction with all the time lost in tool changes, it often compromises the manufacturing time, and in some cases, it might as well compromise the tool's structural integrity. Additionally, it is normal that simulation software gives a non-optimal solution of the sequence for drilling. Hence, it needs to have an optimization of paths and the tool change, which as a result it will tend to better usage of the machine time.

The holes are distributed as follows:

- The 0.8128 (mm) is required by the sockets for the components with DIP packages, such as ATmega16



**Fig. 5** Developed model and sub-model of the drilling process. **a** Main model. **b** Sub-model

microcontroller, the MOSFET gate drivers IR2112, the serial interface for the MAX232 microcontroller, among others.

- The 3.302 (mm) is used for the through-hole mounting components like resistances, capacitors, and crystal oscillator. As it might be clear from Fig. 6, these are the most common hole required for the implementation.
- The 1.2 (mm) is used by the screw terminals, which are used for the digital power connections, the motor connections, and the voltage of the power electronics stage.
- The 1.1 (mm) is required by the 5 (W)–10 ( $\Omega$ ) resistance selected for the dynamic brake stage.
- 1.016 (mm) is for the terminals of the DV-9 female terminals (USB-serial communication), the headers connectors, the lineal voltage regulator 7805, and the IRF3710 MOSFETs.
- The 3.302 (mm) is also for the DV-9 female terminal screw terminals to ensure a good connection with the USB Serial cable.
- The 0.9144 (mm) is necessary for the 104k polyester capacitors, connected in parallel to the power supply.

Table 2 summarizes all the components needed for the design. It shows the quantity of each component, its name, the drill radius, and the number of holes per component.

The number of drills by each tool that is required by the final design can be seen in Table 3.

Figure 7 show the simulated diameters and trajectories of the design without the optimization sent by the PCB

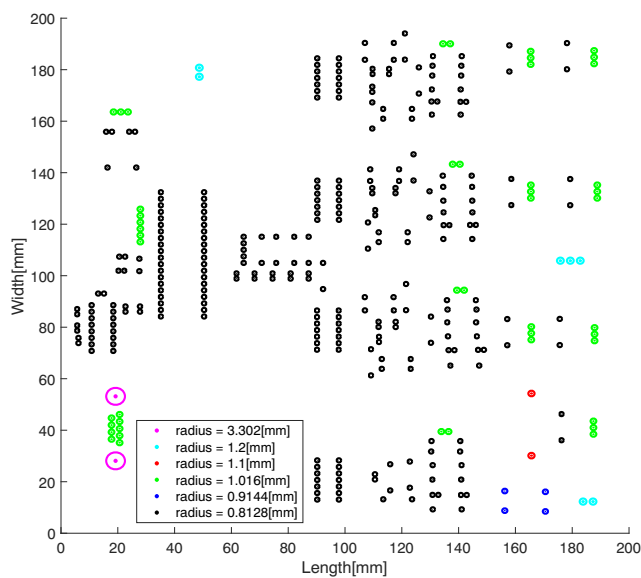


Fig. 6 Holes’ distribution with different colors for each diameter

Table 2 List of the components for the PCB

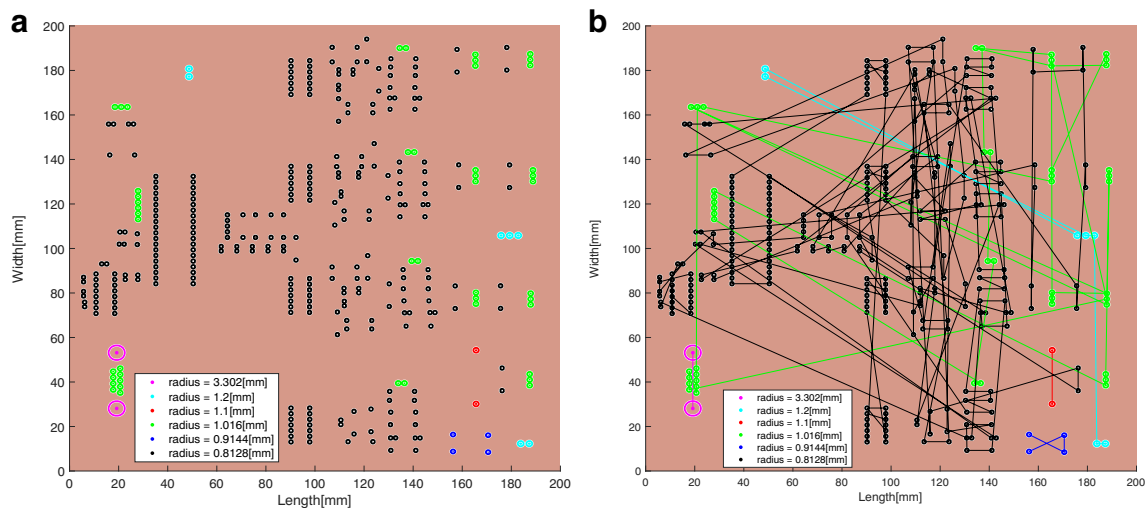
Qty.	Component	Drill radius (mm)	Required drills
1	MAX232	0.8128	16
1	ATmega16	0.8128	40
4	IR2112 gate drivers	0.8128	14
7	IRF3710 MOSFETs	1.016	3
1	Crystal oscillator	0.8128	14
9	Electrolytic capacitors	0.8128	2
23	Ceramic capacitors	0.8128	2
2	Polyester capacitors	0.9144	2
18	1N4148 diodes	0.8128	2
1	7805	1.016	3
1	5 (W)–10 ( $\Omega$ ) resistor	1.1	2
26	Resistors	0.8128	2
1	DV-9 female connector	1.016	9
		3.302	2
1	3 pin screw terminals	1.2	3
2	2 pin screw terminals	1.2	2
19	Male headers	1.2	1

software used for the designing task (EAGLE). For the manufacturing process, the copper plate to be used was 20 (mm) wide by 20 (mm) long.

Figure 7 left and Table 3 show that the total number of drilling holes is 361 and that they have six different diameters. Further, observing the calculated final trajectory (Fig. 7 right), and considering a distance of 200 every time the tool needs to be changed, it would result in a total distance traveled of 51,668 mm. Additionally, considering a velocity of 50 mm/s it would result in  $\sim 17$  min. The result is far from optimal since only  $\sim 3$  of these PCBs can be manufactured in 1 h, which would be higher if it had more holes. Hence,

Table 3 Required drill radius with their corresponding number of drills

Drill radius (mm)	Number of drills
3.302	2
1.2	7
1.1	2
1.016	47
0.9144	4
0.8128	299
Total	361



**Fig. 7** Simulated drilling of the PCB. Left: Simulated scaled positions and diameters of the drills in the copper plate. Right: Simulated non-optimized coordinate sequence sent by the design software. **a** Positions and diameters. **b** Drilling

the optimization must solve for all these holes positions and the required tool changes.

## 4 Methodology

For industrial purposes, the algorithm was implemented in MATLAB and Simulink to find the drilling process's final path. Both software were selected since they have dedicated hardware, which allows them to connect in real-time simulation and real systems. In MATLAB the optimization function was implemented reads all the coordinates ( $X$  and  $Y$ ) and the tool required for each drilling. Then, a model is created that would reduce the distance between the coordinates. For all conditions, a velocity of 50 mm/s was considered.

As a start, to measure the cost equation two situations were evaluated: first, if there was a tool change a penalization of 50 mm, 100 mm and 200 mm was added. Second, to test the second cost equation four different conditions for the tool changing point:  $(x, y) = (0, 0)$ ,  $(x, y) = (0, 200)$ ,  $(x, y) = (200, 0)$  and  $(x, y) = (200, 200)$  covering all four corners.

The second part consisted of testing the Simulink Digital-Twin model. The model worked using a predefined schedule and ran for 8 h. The schedule sends a new batch of PCBs to the drilling machine after it has finished working on the previous batch. Each PCB batch has random sizes with a distribution of 1 to 10, a random number of holes with a

distribution of 200 to 400, and a random position of holes in an area of  $200 \times 200$  mm.

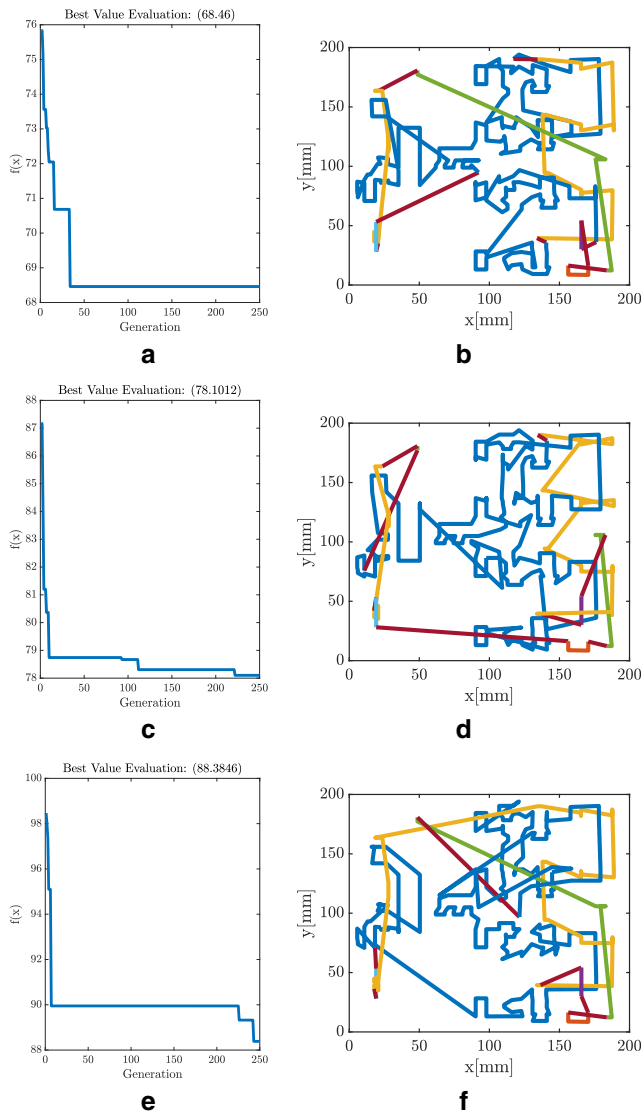
The process operates using no optimization, and optimization with the first and second cost equations. It is important to remember that the optimized route is calculated while the first PCB of each batch is being drilled, hence, it will run as if it does not have the optimization for that first PCB. But, the consequent PCBs with the same configuration will use the optimized route. Additionally, to represent a real change of PCB a random extra time between 0 and 5% is added to each route.

Lastly, the system runs for the PCB case study and shows the result with the holes and components installed.

## 5 Results

First, Fig. 8 shows the evolution of the ACO algorithm using a penalization of 50 mm, 100 mm, and 200 mm for tool change. This results in the process requiring a change of tools 8, 7, and 6 times, respectively. It shows on the left the evolution of the ACO algorithm with a final cost of 68.46s, 78.10s, and 88.38s; on the right, it shows the final drilling path. Each color represents a tool and the red line stands in for a change of tool in the drilling process. It is worth noticing that using a low penalization primarily focuses on closer drilling holes, without considering that changing the tool. But if the penalization gets higher it starts focusing on the first on the holes with the same diameter and later

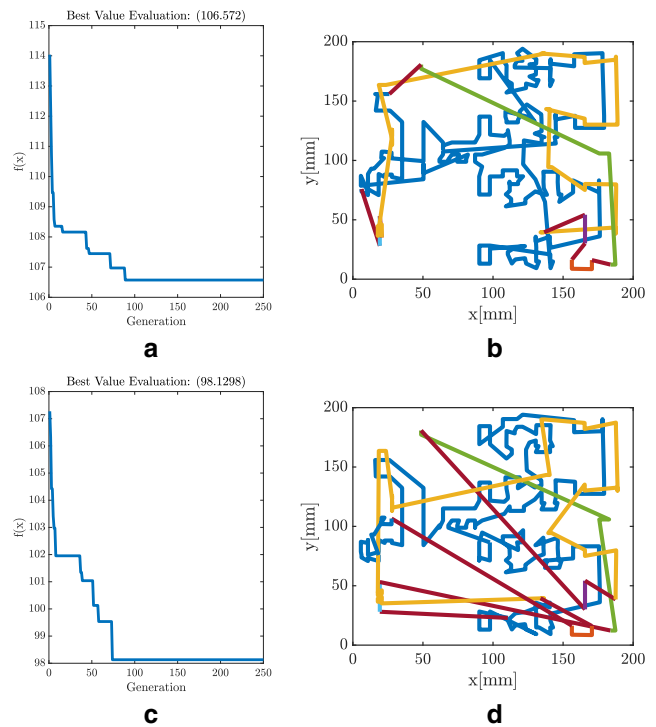




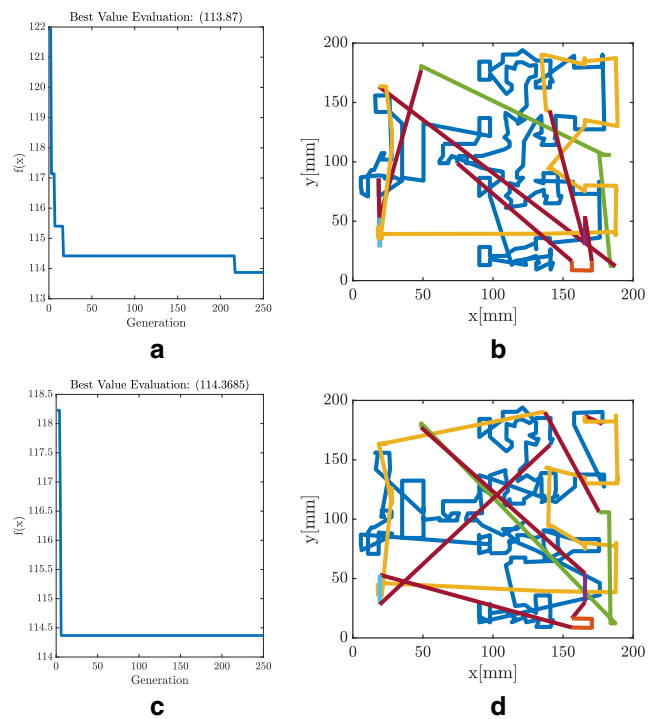
**Fig. 8** ACO using a penalization of 50 mm, 100 mm and 200 mm. On the left evolution of the optimization and on the right each color line represents the path taken for each of the different hole diameters, the red represents a change of tool between the paths. The first line represents 50mm, the second 100mm and the third 200mm

with the tool change. This implies that the optimization would avoid unnecessary tool changes if they are not strictly necessary.

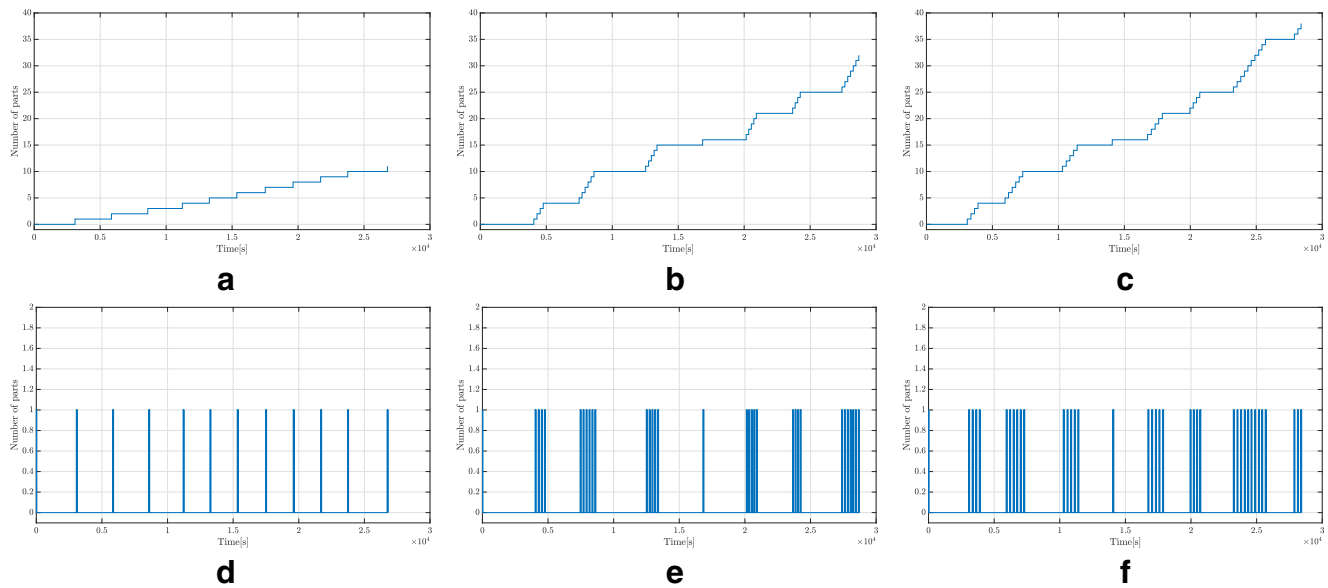
Second, the experiment runs using the second cost equation with different tool changing points. Figures 9 and 10 shows these changes. It is worth noticing that using this cost only changes 6 times. Also, that lower right corner  $(x, y) = (200, 0)$  has the smallest cost 98.12s. This can be explained as there are more changes at that corner and it would be better to start there. Hence, it would be important to check initially each corner which could be done in the Digital-Twin before starting, which would be implemented in a future version.



**Fig. 9** ACO starting at lower left in the top row and lower right corners on the bottom row. On the left evolution of the optimization and on the right each color line represents the path taken for each of the different hole diameters, the red represents a change of tool between the paths



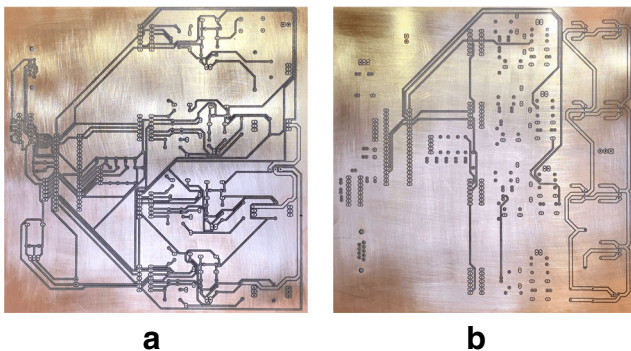
**Fig. 10** ACO starting at upper left in the top row and upper right corners on the bottom row. On the left evolution of the optimization and on the right each color line represents the path taken for each of the different hole diameters, the red represents a change of tool between the paths



**Fig. 11** The number of plaques made and workers' activation time for an 8-h simulation. In the first row, each step represents a new PCB, being drilled, the second row shows every time a worker downloads

Third, Fig. 11 in the first row shows the number of parts produced each run (8 h), the second row shows each time a worker was allocated in the run. Each one of the columns represents, from left to right, the runs with no optimization, with optimization using the first and second cost equation. It can be seen that without optimization only 11 plaques can be made, on the other hand with optimization 32 and 38 plaques with each of the equations. Further, without optimization it can only work with 2 or 3 changes, differently with optimization it can have 7 to 8 different PCB designs. Thus, the use of the Digital-Twin can highly improve the production of plaques.

Lastly, as evidence, Fig. 12 shows the final PCB drilling, where it shows the bottom and top view after the drilling.



**Fig. 12** Final PCB with its corresponding drilling holes. **a** Bottom view of the manufactured PCB. **b** Top view of the manufactured PCB

and uploads a new PCB. First column without optimization, and a second and third column with first and second cost equation

## 6 Conclusions

Our aim was to test whether or not the use of a Digital-Twin can be highly beneficial in product redesign, by simulating how new products will affect the production line. It is used for smart manufacturing of a PCB and how it will affect the time consumption. The results of this research suggest that:

- Depending on the variable analysis, cost function, and the use of optimization the design of how the drilling holes can be done before construction.
- Using a speed of 50 mm/s the time without optimization was  $t \approx 17$  m, conversely, using the optimization for most cases is less than half  $\sim 2$  min = 120 s.
- It took less time to run the optimization than drilling one PCB, hence, it can run in parallel for the first PCB and use the results for the next PCBs in the batch.
- Using the Simulink model with optimization increases the number of manufactured PCBs.
- With optimization the workers would have to replace the PCBs more often.

All these would reduce the time enormously, especially for large scale production, which is helpful for a great demand for new PCB orders and reconfiguration is required.

## 7 Future work

The presented work has the following limitations and considerations:

1. It does not include IoT communication in real-time for updating the information from the complete process
2. The Digital-Twin is only a local representation that does not integrate the complete supply chain
3. The optimization algorithm has to be deployed into an embedded digital system that could provide information in order to predict the performance of the process
4. The data regarding the failures is not stored and modeled
5. It is not included an economic study that shows the main advantages of using this optimization algorithm
6. This research does not assess all the metaheuristic optimization methods
7. Only one type of material is evaluated in designing the PCB
8. The manufacturing time could also change according to with new degrees of freedom of each tool so an optimization about the number of degrees of freedom
9. This research does not study the components optimization placement

As a part of future work, as in [5, 6] it would be important to check what variables are relevant and from this data develop a quality monitoring system determining whether a part is within specifications or out of specifications. Also, check the use of different materials, and the optimization to use more of the PCB. Finally, it would require checking further models of PCB with a higher number of holes and different specifications.

## Declarations

**Conflict of interest** The authors declare that they have no conflict of interest.

## References

1. Zhou K, Liu T, Zhou L (2015) Industry 4.0: Towards future industrial opportunities and challenges. In: 2015 12th International conference on fuzzy systems and knowledge discovery (FSKD). IEEE, pp 2147–2152
2. Stark R, Kind S, Neumeyer S (2017) Innovations in digital modelling for next generation manufacturing system design. *CIRP Ann* 66(1):169–172
3. Boschert S, Rosen R (2016) Digital twin—the simulation aspect. In: *Mechatronic futures*. Springer, pp 59–74
4. Moreno A, Velez G, Ardanza A, Barandiaran I, de Infante ÁR, Chopitea R (2017) Virtualisation process of a sheet metal punching machine within the industry 4.0 vision. *Int J Interact Des Manuf (IJIDeM)* 11(2):365–373
5. Farahani S, Brown N, Loftis J, Krick C, Pichl F, Vaculik R, Pilla S (2019) Evaluation of in-mold sensors and machine data towards enhancing product quality and process monitoring via industry 4.0. *Int J Adv Manuf Technol* 105(1-4):1371–1389
6. Farahani S, Loftis J, Xua B, Pilla S (2020) Towards multi-tiered quality control in manufacturing of plastics and composites using industry 4.0, San Antonio, TX
7. Uhlemann TH-J, Lehmann C, Steinhilper R (2017) The digital twin: Realizing the cyber-physical production system for industry 4.0. *Procedia Cirp* 61:335–340
8. Kritzingner W, Karner M, Traar G, Henjes J, Sihn W (2018) Digital twin in manufacturing: a categorical literature review and classification. *IFAC-PapersOnLine* 51(11):1016–1022
9. Negri E, Fumagalli L, Macchi M (2017) A review of the roles of digital twin in cps-based production systems. *Procedia Manuf* 11:939–948
10. Rosen R, Wichert GV, Lo G, Bettenhausen KD (2015) About the importance of autonomy and digital twins for the future of manufacturing. *IFAC-PapersOnLine* 48(3):567–572
11. Glaessgen E, Stargel D (2012) The digital twin paradigm for future nasa and us air force vehicles. In: 53rd AIAA/ASME/ASCE/AHS/ASC Structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA, p 1818
12. Cimino C, Negri E, Fumagalli L (2019) Review of digital twin applications in manufacturing. *Comput Ind* 113:103130
13. He R, Chen G, Dong C, Sun S, Shen X (2019) Data-driven digital twin technology for optimized control in process systems. *ISA Trans* 95:221–234
14. Wang Y, Wang S, Bo Y, Zhu L, Liu F (2020) Big data driven hierarchical digital twin predictive remanufacturing paradigm: architecture, control mechanism, application scenario and benefits. *J Clean Prod* 248:119299
15. Luo W, Hu T, Zhang C, Wei Y (2019) Digital twin for cnc machine tool: modeling and using strategy. *J Ambient Intell Humaniz Comput* 10(3):1129–1140
16. Qi Q, Tao F (2018) Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison. *IEEE Access* 6:3585–3593
17. Rüßmann M, Lorenz M, Gerbert P, Waldner M, Justus J, Engel P, Harnisch M (2015) Industry 4.0: the future of productivity and growth in manufacturing industries. Boston Consulting Group 9(1):54–89
18. El Saddik A (2018) Digital twins: the convergence of multimedia technologies. *IEEE MultiMed* 25(2):87–92
19. Tao F, Cheng J, Qi Q, Zhang M, He Z, Sui F (2018) Digital twin-driven product design, manufacturing and service with big data. *Int J Adv Manuf Technol* 94(9-12):3563–3576
20. Söderberg R, Wärmeffjord K, Madrid J, Lorin S, Forslund A, Lindkvist L (2018) An information and simulation framework for increased quality in welded components. *CIRP Ann* 67(1):165–168
21. Xiang F, Zhang Z, Zuo Y, Tao F (2019) Digital twin driven green material optimal-selection towards sustainable manufacturing. *Procedia CIRP* 81:1290–1294
22. Lopes MR, Costigliola A, Pinto R, Vieira S, Sousa JMC (2019) Pharmaceutical quality control laboratory digital twin—a novel governance model for resource planning and scheduling. *Int J Prod Res* 1–15
23. Korth B, Schwede C, Zajac M (2018) Simulation-ready digital twin for realtime management of logistics systems. In: *IEEE International conference on big data (Big Data)*. IEEE, p 2018
24. Liu Y, Jin J, Ji P, Harding JA, Fung RYK (2013) Identifying helpful online reviews: a product designer’s perspective. *Comput Aided Des* 45(2):180–194
25. Stark R, Damerau T, Lindow K (2018) Industrie 4.0—digital redesign of product creation and production in berlin as an industrial location. In: *The internet of things*. Springer, pp 171–186

26. Ma X, Tao F, Zhang M, Wang T, Zuo Y (2019) Digital twin enhanced human-machine interaction in product lifecycle. *Procedia CIRP* 83:789–793
27. Haag S, Anderl R (2018) Digital twin–proof of concept. *Manuf Lett* 15:64–66
28. Schleich B, Anwer N, Mathieu L, Wartzack S (2017) Shaping the digital twin for design and production engineering. *CIRP Ann* 66(1):141–144
29. Hahsler M, Hornik K (2007) Tsp-infrastructure for the traveling salesperson problem. *J Stat Softw* 23(2):1–21
30. Yang J, Shi X, Marchese M, Liang Y (2008) An ant colony optimization method for generalized tsp problem. *Prog Nat Sci* 18(11):1417–1422

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.