Journal of Manufacturing Systems 68 (2023) 354–367

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Towards smart layout design for a reconfigurable manufacturing system

Halldor Arnarson [a,*], Hao Yu [a], Morten Monland Olavsbråten [a], Bernt Arild Bremdal [b,c], Bjørn Solvang [a]

[a] Department of Industrial Engineering, UiT The Arctic University of Norway, Lodve Langesgate 2, Narvik, 8514, Nordland, Norway
[b] Smart Innovation Norway, Hakon Melbergs vei 16, Halden, 1783, Nordland, Norway
[c] Department of Computer Science and Computational Engineering, UiT The Arctic University of Norway, Lodve Langesgate 2, Narvik, 8514, Nordland, Norway

ARTICLE INFO

Keywords:
Reconfigurable manufacturing system (RMS)
Layout problem for RMS
Digital twin simulation
Industry 4.0

ABSTRACT

Global competition and increased variety in products have created challenges for manufacturing companies. One solution to handle the variety in production is to use reconfigurable manufacturing systems (RMS). These are modular systems where machines can be rearranged depending on what is being manufactured. However, implementing a rearrangeable system drastically increases complexity, among which one challenge with RMS is how to design a new layout for a customized product in a highly autonomous and responsive fashion, known as the layout design problem. In this paper, we combine several Industry 4.0 technologies, i.e., IIoT, digital twin, simulation, advanced robotics, and artificial intelligence (AI), together with optimization to create a smart layout design system for RMS. The system automates the layout design process of RMS and removes the need for humans to design a new layout of the system.

1. Introduction

With a global market and interconnected supply chains, the competition between manufacturing companies has risen substantially. In addition, the product life cycle has become shorter and the manufacturing industry is moving from mass production towards mass customization and mass personalization. This means that manufacturing systems need to be changed so that they can better adapt to the changes in the market and capture new business opportunities. Therefore, there is a need for a manufacturing system that can be easily changed and scaled up or down depending on the various demands of consumers.

To solve these problems, Koren et al. [1] proposed the idea of a reconfigurable manufacturing system (RMS). An RMS can be described as a manufacturing system that can be changed and adjusted by rearranging and changing the components. They are designed for the reconfiguration of both hardware and software components in the system [2].

However, having a system that can be rapidly reconfigured adds new challenges and complexity to the system [3]. One of the challenges with RMS is the layout problem. The layout problem focuses on how to design/rearrange the RMS, when considering both the capacity and operational performance of the system [4]. To be able to reconfigure the manufacturing system quickly, it would be beneficial to give the exact placement of the machines to minimize the reconfiguration time of the RMS. In addition, when a new customized order comes, planning and designing a new product-based layout for an RMS is a time-consuming job that requires a significant amount of human labor and input.

There is, however, a lack of research on the layout problem for RMS. Sabioni et al. [5] reveal that most papers that work on the layout problem for RMS, focus on cost minimization, and there are few papers that focus on the design optimization problem. Thus, there is a need for a model that can support the redesign of the layouts [6]. One method to solve the layout problem can be to implement other tools/technologies that can help in the design. Maganha et al. [6], note that there are few investigations on supportive tools for RMS design.

Industry 4.0 is the next technological revolution and brings several cutting-edge technologies such as big data, industrial internet of things (IIoT), simulation, cloud computing and cyber–physical systems. These technologies are important for the success of RMS [7] and can be used to further automate the systems. However, Brotolini et al. [8] indicate that there is a lack of research on implementing and using Industry 4.0 technologies in RMS.

Applying digital twins and simulation enables a faster method that allows for testing, optimization, development, and deployment of new layouts for the RMS [9]. Maganha et al. [6] note that there is a need to investigate the use of simulation to design manufacturing facilities since simulation tools can be used to test the performance of the system

* Corresponding author.
E-mail address: halldor.arnarson@uit.no (H. Arnarson).

https://doi.org/10.1016/j.jmsy.2023.03.012
Received 25 December 2022; Received in revised form 23 March 2023; Accepted 27 March 2023
Available online 2 May 2023

in a more realistic way. In addition, industry 4.0 technologies can be used to achieve smart layout design of the RMS [6]. In this paper, we will define smart layout design, as combining multiple Industry 4.0 technologies to solve the layout problem in an automatic manner.

Arnarson et al. [10] propose an RMS that uses a mobile robot to reconfigure the system automatically without any human intervention. In the paper, they showed that different placement of the platforms in an RMS gives different manufacturing times. It is therefore important to minimize the distance the manufactured part has to move in the system. This paper also reveals that designing and rearranging an RMS can be extremely time-consuming and usually requires a large amount of human labor and input, so there is a need for a method to design the layout of the manufacturing system automatically.

In the literature, there are few papers on combining multiple industry 4.0 technologies together to solve the layout design problem for RMS. To fill this gap, in this paper, we implement several industry 4.0 technologies such as IIoT, digital twin/model, simulation, advanced robotics, and artificial intelligence (AI) with optimization to develop a smart layout design system for RMS. Furthermore, we use evolutionary computations, known as a subfield of AI, where a population-based algorithm produces a population of candidates that evolves toward an optimal or near-optimal solution [11]

More specifically, we formulate a mathematical model for the platform-based RMS proposed by Arnarson [10] and use optimization to find a layout automatically. From the optimization, a digital model is generated, which can be tested with simulation for further validation of the system. Finally, the system is tested on a physical RMS to verify and validate if the layout optimization with a digital model simulation can work effectively and correctly in the real-world system.

The main contributions of the work are as follows:

- Investigate how Industry 4.0 technologies such as IIoT, digital model, simulation, and advanced robotics can be combined with optimization to create smart layout design for RMS.
- Develop a mathematical model which gives the exact position/coordinates of a platform-based RMS.
- Use AI and evolutionary computations to search/optimize for a layout configuration for the platform-based RMS.
- Generate a digital model automatically from the solution of the optimization.
- Connect the optimization program together with the digital model simulation software for further testing and validation in a digital environment.
- Use IIoT technology to connect the optimization, digital model simulation, and a physical RMS together for communication.

The rest of the paper is structured as follows: Section 2 reviews previous studies on the layout design problem for RMS. Section 3 develops the mathematical model of the system, and Section 4 looks at the implementation and the results from the system. Finally, we discuss the results in Section 5 and conclude the paper in Section 6.

2. Prevous studies

2.1. Facility layout problem

In more broad research, the layout design problem for manufacturing systems in general is referred to as the facility layout problem [12]. Besbes et al. [13] looked at the layout facility problem, where they arranged facilities on a planar site and considered geometric constraints for the facilities. They tested the system using the proposed algorithm to optimize eight facilities on the plan floor. Lim et al. [14] evaluated hybrid algorithms, where they used the algorithms for layout optimization of multi-cellular manufacturing systems.

Guo et al. [15] used a digital twin to optimize the manufacturing workshop. A digital twin was used to optimize different parts of the workshop and the distribution routes. The method was tested in a physical welding workshop, which resulted in an increased production capacity of 29.4%. This shows the potential of implementing digital twins when doing optimizations of the layout. The authors also mention that there is a lack of research on using digital twins with layout optimization, and for further research, more methods should be developed for layout optimization using digital twins.

Moreover, in a literature review on the facility layout problem [16] reveals that most researchers did not include simulation and safety drivers with the facility layout design problem. They also noted that there was less focus on industry 4.0 technologies such as IIoT and digital twin. Zubaidi et al. [16] note that implementing elements of industry 4.0 can help in creating a more reliable, comprehensive, and sustainable layout design. It is also important to note that the facility layout problem is often considered a static problem. In contrast, the layout problem for RMS is a dynamic problem since the RMS layout is made to be changed. Since it is a dynamic problem, it requires powerful and flexible simulation tools.

2.2. Layout design of RMS

Layout design for RMS encompasses many elements, including process planning [17,18], scheduling [19], scalability planning [20], and cost optimization [21–24]. There are, however, fewer papers that look at the placement of the machines.

Koren et al. [2] proposed a method on how to design an RMS. Their method requires planning, and if the RMS has many processes and machines, the problem will become more complex. They also mention that each new product that is manufactured should include a new design of the RMS. Guan et al. [25] investigated the layout design for RMS where they considered automated guided vehicles for material handling instead of using conveyors. In the study, precedence graphs are used to show the flow and positions of the workstation.

Haddou Benderbal et al. [26] studied the machine layout problem for RMS, where they developed a system that could propose the best placement for the machines. In addition, Haddou Benderbal et al. [27] also developed a decision-support approach for switching between products in the same product family. However, in both cases, the machines could only be placed in predefined positions.

Another paper from Besbes et al. [28] investigated the facility layout problem for RMS. In the study, the goal was to minimize the material handling cost. The layout was generated with a genetic algorithm, and then an A* search algorithm was used to find the shortest distance between manufacturing cells. Nevertheless, the authors mention that the method is tested offline and for further work, the system should be tested on a physical RMS system. In addition, they mention that the model should be expanded toward a multi-objective problem that considers the shape and orientation of the manufacturing cells.

There are few examples of systems that can generate a layout for the RMS. Abdelkrim et al. [12] note that there were few researchers working on solving the layout design problem for RMS. From a literature review, Sabioni et al. [5] reveal that most papers working on optimizing of RMS configurations looked at cost minimization. The study did not find any relevant researches that combined both the layout design and machine configuration problem at the same time. It is also noted that it is difficult to find industries or laboratories that have implemented an RMS.

2.3. Simulation for layout design

A few attempts have been made to implement industry 4.0 tools, such as simulation and digital twin, to solve the layout design problem. Yamada [29] used 3D simulation to do analysis and design evaluation for the reconfiguration of an RMS. In the study, he looked at a manufacturing system with transport robots, input stations, output stations, movable manufacturing cells and processes, where he tried to minimize the manufacturing time using particle swarm optimization. The simulation is rather simple, where the manufacturing cells and other stations are modeled in the simulation as circles and squares. Zheng et al. [30] proposed a simulation framework for the layout, cost, and performance of the system. They used the simulation tool "Plant Simulation", which is a discrete-event simulator, to analyze the behavior of a system. Petroodi et al. [31] used a discrete event simulation tool (Simul8) together with optimization to solve the resource allocation and production planning problem. These studies show the potential of combining simulation and optimization together. However, these examples of using simulations are simple and are not validated with a real RMS.

Work has also been done on using 3D manufacturing simulations and digital twins to support the layout design process. Santos et al. [32] used a simulation-based approach to support the design and operational management of the system. The simulation allowed the planner to test different configurations and layouts virtually. Touckla et al. [33] proposed a framework with a digital twin design and simulation model for RMS. These studies do not use optimization to create the layout and require human operators to design the system.

There is also research on using digital twins for planning in RMS. Leng et al. [34] proposed a digital twin for fast reconfiguration of RMS, which was used as a tool to shorten the time of production changeover. Kurniadi et al. [35] investigated the use of digital twin simulation for reconfiguration planning. They used both discrete-event simulation (DES) and visual simulation to show that digital twins can help effectively integrate RMS into a production system. The RMS digital twin framework proposed by Hajjem et al. [36] suggested that using digital twins with RMS provides improved functionalities, e.g., simulation and intelligent sensors, which can improve the system's intelligence and efficiency.

2.4. Summary

All the papers investigating the layout design problem for RMS have not tested their system or method on a physical RMS to validate if the system works. Rosio et al. [3] did also find limited examples of industrial examples of RMS, and there is a lack of knowledge on how to design an RMS.

In addition, there is a lack in the literature on exploiting the benefits of using Industry 4.0 technologies to solve the layout design problem of RMS. These existing studies have clearly shown the potential of using simulation and digital models for the layout design problem, but there is a need for more investigation, for instance, by combining both optimization and simulation. In addition, there are a few examples showing how Industry 4.0 technologies such as digital twins and simulation can be implemented in a physical RMS. Integrating various industry 4.0 technologies can lead to a smart layout design system for RMS which can automate the layout design process.

3. Mathematical model

In this project, a mathematical model is formulated based on the concept of a modular platform based RMS described in [10]. This type of system has multiple modular platforms that can easily be added or removed depending on the demand or what is being manufactured. The goal is to develop a general mathematical model which can be used to automatically generate layouts for a platform based RMS.

3.1. Assumptions

To develop the mathematical model, the following assumptions are made:

1. The mathematical model is a 2D plane, and the 3D dimension is not considered.

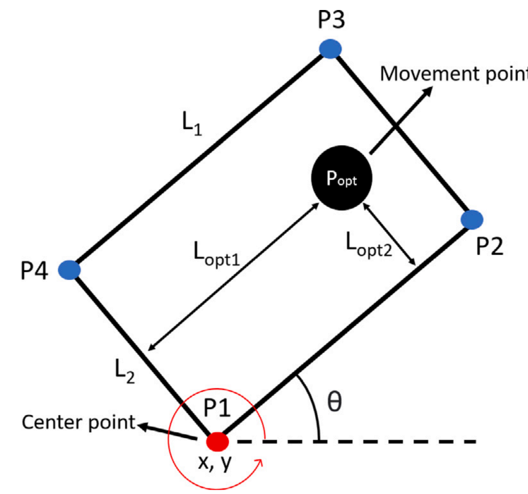

Fig. 1. The center point and movement point of the platforms. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2. All platforms are modeled as rectangles
3. The platforms can be placed in a space of 10 × 10 m
4. All manufactured parts move from a singular point on the platforms.
5. The amount of platforms is given, and the model can use all platforms to design the layout. There is a risk of the layout becoming chaotic if too many platforms of the same type are used.
6. All platforms are made to be the same- or similar height.

3.2. Describe the platforms

Manufacturing systems usually contain different machines depending on the tasks. In the system, each platform can contain a 3D printer, a CNC machine, a conveyor or a robot arm. To categorize these platforms and be able to generalize the system, we divide the platforms into four categories:

- Input platform: A platform that gives material to the system, or an input part of the system
- Movement platform: A platform that is used to move parts between platforms (can be robot arms or humans).
- Work platform: A platform used to do a process, such as quality control, machining process, and assembly station.
- Output platform: A platform that moves the parts out of the system (can be conveyors).

Each of the platforms has three variables used in the optimization, x and y for the position and theta for the rotation of the platform. The platforms do also have size variables and the position of the movement points.

The point of rotation (center point) is highlighted with the red circle as shown in Fig. 1. In this project, we test two types of rotations for theta. The first type sets a fixed 0, 90, 180, or 270 degrees rotation for the platforms and the second type uses a number between 0–360 for theta.

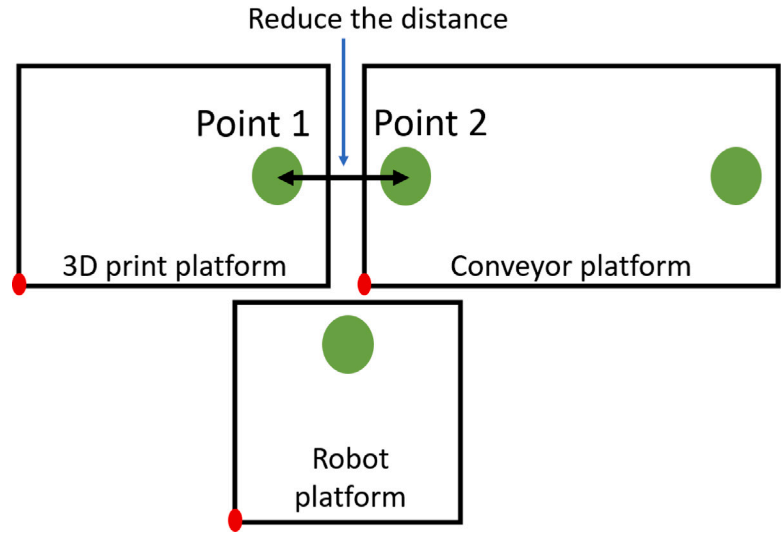

Fig. 2. Demonstration of a simplified RMS with three platforms.

To calculate the x and y positions of the corners (P2-P4) the following formulas are used when theta:

$$
\begin{aligned}
P1_x &= x \\
P1_y &= y \\
P2_x &= L_1 * cos(\theta) + x \\
P2_y &= L_1 * sin(\theta) + y \\
P3_x &= L_1 * cos(\theta) - L_2 * sin(\theta) + x \\
P3_y &= L_1 * sin(\theta) + L_2 * cos(\theta) + y \\
P4_x &= -L_2 * cos(\theta) + x \\
P4_y &= L_2 * sin(\theta) + y \\
Popt_x &= L_{opt1} * cos(\theta) - L_{opt2} * sin(\theta) + x \\
Popt_y &= L_{opt1} * sin(\theta) + L_{opt2} * cos(\theta) + y
\end{aligned} \tag{1}
$$

3.3. Optimization problem

In this paper, we investigate a platform-based RMS, and the optimization goal is to improve efficiency by minimizing the total movement distance of the workpiece throughout the system. Since the movement distance on each working platform is fixed, the problem becomes thus the minimization of the distance between different platforms. Fig. 2 illustrates a simple case with a 3D printer, a robot platform, and a conveyor. In this example, we try to minimize the distance between the 3D printer and the conveyor, say, the distance between point A and point B. At this stage, the distance between the robot platform (movement platform) and the other platforms is not considered, because the robot platform only moves the parts from one working platform to another. The only requirement for the robot platform is that it can reach the required points on the respective working platforms.

Thus, the objective of the optimization model is to minimize the Euclidean distance for moving the workpiece between point 1 and point 2, as shown in Eq. (2):

$$Minimize\ OBJ = \sqrt{(x_{point1} - x_{point2})^2 + (y_{point1} - y_{point2})^2} \tag{2}$$

$$Minimize\ Obj1 = \sum_{i \in M} \sum_{j \in M} d_{ij} c_{ij} wtij \tag{3}$$

Moreover, the Euclidean distances are calculated from a given order/sequence of the platforms in the system. We generalize the mathematical optimization model in Eq. (3), which minimizes the total movement distance (c_{ij}) of the workpiece throughout the whole RMS. The set of working platforms is defined by $M = \{1, 2, \ldots, m\}$, and calculate the movement distance between two working platforms, where $i, j \in M$ and $i \neq j$.

In an RMS for mass and/or individualized customization, the manufacturing procedures need to be formulated based on the requirements of specific products or product families. In this regard, c_{ij} is a binary parameter establishing the linkage and precedence between two working platforms in the RMS, which is determined based on a specific product. If the system uses multiple input and output platforms, the optimization model considers all combinations of how the part can move in the system. For instance, Fig. 3 shows an RMS system that has two 3D print platforms, working platforms, and conveyor platforms, where five linkages are established by setting c_{13}, c_{23}, c_{34}, c_{45}, and c_{46} equal to 1.

In this model, wt_{ij} is the weight of each linkage, which may help to adjust the movement distance (c_{ij}) with, for example, the flow of workpieces between two working platforms. Besides, it can also be used to solve the challenges related to a multi-platform RMS system. As shown in Fig. 4, a manufacturing system can be divided into multiple platforms. In this example, the system is divided into three platforms, where two conveyor platforms are used to connect these platforms. One challenge of having a conveyor between two platforms is that, in the optimization process, the two platforms are likely to fight for the same conveyor. Moving the conveyor in either direction may yield the same optimal result, and the conveyor may be placed in between the two platforms, which are far away from each other, as can be seen in Fig. 5. There are several ways to solve this problem. One method is to add a larger weight to the conveyor's output and input, which can help to reduce the distance between the two platforms connected by the same conveyor. This method has little impact on the rest of the system.

Next, we consider the optimal positions of movement platforms. For this system, we model the movement platforms as robot arms and will therefore need to take into consideration the reach of the robot arms in the mathematical model. As mentioned in Section 3.1, the mathematical model is based on a 2D plane. However, the robot arms have a circular reach in all axis. This means that if the platforms are of different heights, the robot arms might not be able to reach the platforms while being within the radius of the 2D plane. In this system, we assume that all platforms are at the same or similar height, and we will therefore model the reach of the robot arm as a circular radius, as shown in Fig. 6. It should be noted that the robot might still not be able to reach certain points with a particular orientation (yaw, pitch, and roll) of the tool center point. As a result, a simulation is used for further verification if the robot arm is capable of picking the item (Section 4.2).

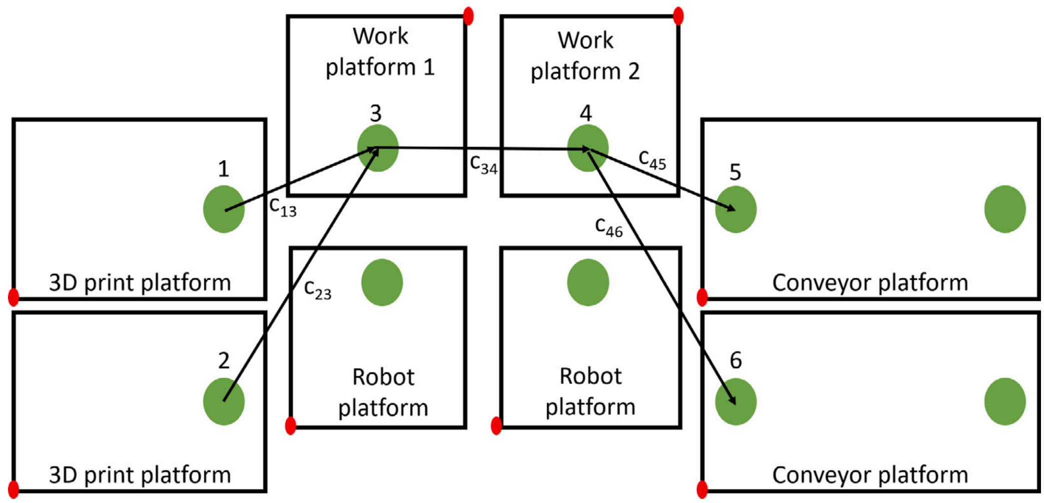

Fig. 3. A RMS with two input platforms, two working platforms, two output platforms, and two movement platforms.

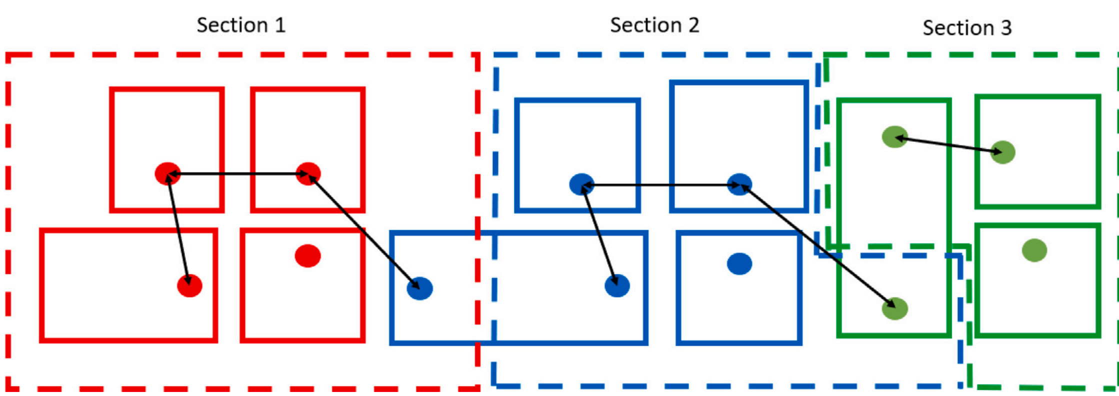

Fig. 4. A multi-platform RMS system.

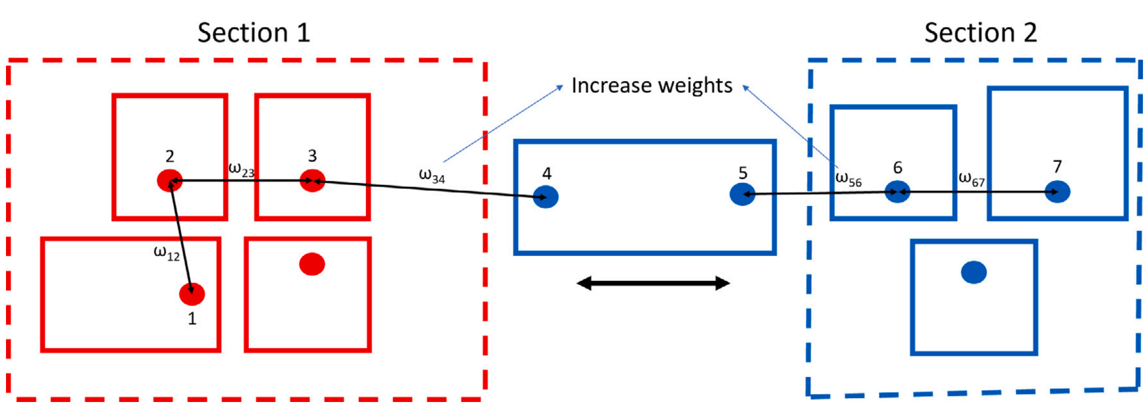

Fig. 5. Illustration of the optimization challenge related to a multi-platform RMS with shared conveyors connecting different platforms.

The total movement distance of the robot arms needs to be minimized, while at the same time, all the working platforms need to be assigned to a robot arm within its maximum reachable radius. Mathematically, the following constraint (4) needs to be held. Herein, the set of movement platforms is given by $N = 1, 2, \ldots, n$, and r_n is the maximum reachable radius of each movement platform. Moreover, a_{nm} is a binary variable that determines if a working platform is assigned to a movement platform, and p_{nm} is the movement distance (c_{ij}) between them.

$$p_{nm} \leq r_n a_{nm}, \forall\, n \in N, m \in M \tag{4}$$

Besides, each working platform must be served by a robot arm, as shown in Eq. (5):

$$\sum_{n \in N} a_{nm} = 1, \forall\, m \in M \tag{5}$$

However, the use of this non-linear hard constraint drastically increases the computational efforts needed to solve the optimization model. Thus, in this paper, it is converted to a soft constraint to improve the computational efficiency to find near-optimal solutions. These solutions will be further validated in the simulation stage, which helps to effectively eliminate all the infeasible solutions. To implement the soft constraint, we introduce a piecewise function in Fig. 7 to

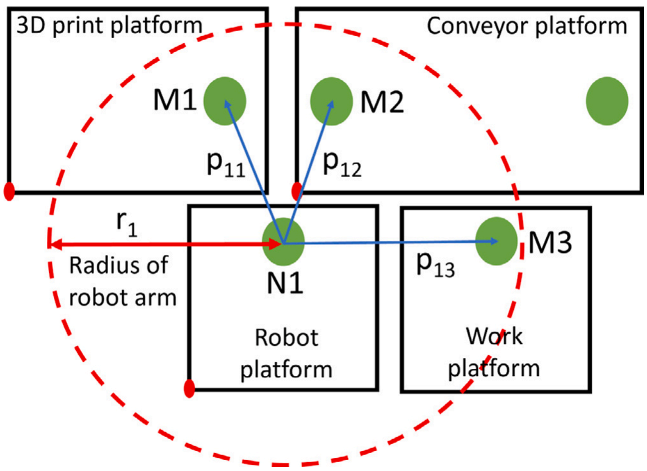

Fig. 6. Illustration of the maximum reachable radius of the robot platform.

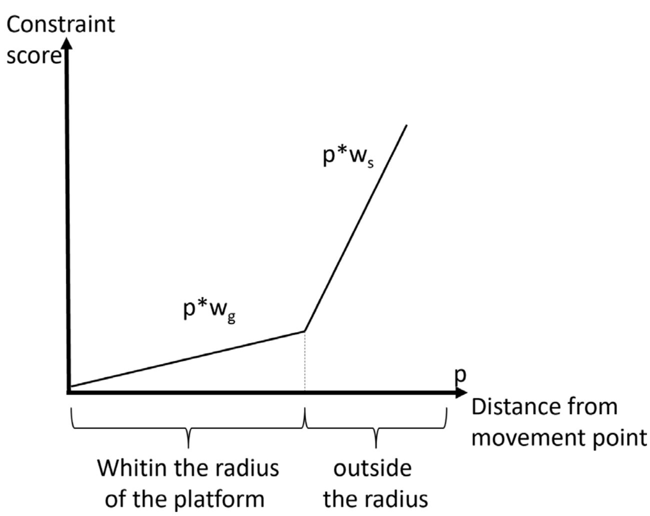

Fig. 7. Piecewise function for weight calculation.

calculate the weight of the movement distance between robot arms and working platforms. As shown, if the robot arm can reach the required points, the weight on the respective distance is very small. However, if the robot arm cannot reach the required point, a higher weight will be given as a penalty for the respective linkage between the robot arm and the working platform, which will, in most cases, lead to $a_{nm} = 0$. An illustration of how the weights can be seen in Fig. 8. The general form of the second objective as well as the respective constraint is given in Eq. (6).

$$Minimize\ Obj2 = \sum_{m\in M}\sum_{n\in N} p_{nm}a_{nm}wp_{nm} \tag{6}$$

Subject to:

$$\sum_{n\in N} a_{nm} = 1, \forall\ m \in M$$
$$wp_{nm} = \begin{cases} p_{nm}w_g, & \text{if } P_{nm} \leq r_n \\ p_{nm}w_s, & \text{if } P_{nm} \geq r_n \end{cases} \tag{7}$$

There is also a need to consider the rotation and reachable area of different types of robot arms. For instance, Universal Robots has a reach of ±360 degrees, while a Nachi MZ07 has a reach of ±170 degrees. As shown in Fig. 9, the unreachable area of the robot arm can be drawn as a triangle. A check is thus added to see if any required points on the working platforms are in the unreachable area of the robot arms. First, all the points in the triangle are calculated with the following formulas:

$$\begin{aligned} c_n^1 &= (x_{2(n)} - x_{1(n)}) \times (y_{p(m)} - y_{1(n)}) - (y_{2(n)} - y_{1(n)}) \times (x_{p(m)} - x_{1(n)}) \\ c_n^2 &= (x_{3(n)} - x_{2(n)}) \times (y_{p(m)} - y_{2(n)}) - (y_{3(n)} - y_{2(n)}) \times (x_{p(m)} - x_{2(n)}) \\ c_n^3 &= (x_{1(n)} - x_{3(n)}) \times (y_{p(m)} - y_{3(n)}) - (y_{1(n)} - y_{3(n)}) \times (x_{p(m)} - x_{3(n)}) \end{aligned} \tag{8}$$

A constant k_n is added to increase the length of the triangle to ensure the whole area is checked. For the Nachi MZ07 robot arm, the constant is 1.2. In addition, all robot arms are 90 degrees rotated on the platforms, and we therefore add 90 degrees. Using these points, we check with Eq. (8) if the point p on the working platform m is inside the triangle with these conditions when $a_{nm} = 1$:

If the point is inside the triangle, a higher penalty should be applied. In addition, in some cases, one movement platform is able to reach all the required platforms, and then there would be no need for another movement platform that is not assigned to any working platforms, as shown in Fig. 10. The redundant movement platform needs thus to be eliminated from the system.

The general form of the mathematical optimization model is then given in Eq. (9):

$$\begin{aligned} &Minimize\ Obj1 = \sum_{i\in M}\sum_{j\in M} d_{ij}c_{ij}wtij \\ &Minimize\ Obj2 = \sum_{m\in M}\sum_{n\in N} p_{nm}a_{nm}wp_{nm} \end{aligned} \tag{9}$$

Subject to Eq. (10):

$$\sum_{n\in N} a_{nm} = 1, \forall\ m \in M$$

$$wp_{nm} = \begin{cases} p_{nm}w_s, if\, wp_{nm} = 1 \text{ and if}\{p_{nm} \geq r_n\} \text{ or} \\ \quad \left\{ P_{nm} \leq r_n \text{ and } \begin{cases} C_n^1 \geq 0 \text{ and } C_n^2 \geq 0 \text{ and } C_n^3 \geq 0 \\ \text{or } C_n^1 \leq 0 \text{ and } C_n^2 \leq 0 \text{ and } C_n^3 \leq 0 \end{cases} \right. \\ p_{nm}w_g, \text{otherwise} \end{cases} \tag{10}$$

Finally, another hard constraint needs to be added to ensure the model is not to have any overlap between different platforms. One method to formulate this constraint is to use the separating axis theorem (SAT). The SAT can be used with any convex shapes to check if there is any overlap. For each of the solutions generated, the SAT is tested. If there is an overlap between the platforms, the solution is eliminated, and only the solutions without overlap are considered.

4. Implementation

4.1. Solve mathematical model

One of the challenges with RMS is the complexity of such systems. Increasing the number of platforms in the system also increases the number of possible layouts for the system. One method to find a layout for the RMS is to use evolutionary computation, which is a sub-field of AI. Evolutionary computation uses population based algorithms, where a population is maintained and evolves towards a good/optimal solution [11].

For this project, we used non-dominated sorting genetic algorithm 2 (NSGA2) [37] for the optimization since it is a powerful multi-objective algorithm [38], which has been widely used to solve process planning problems [38] for RMS design. Due to its reliability and speed, the NSGA2 has been used to solve workshop-related problems [39], allocation problems, scheduling problems, traveling salesman problems, and vehicle routing problems [40]. The NSGA2 is a multiobjective evolutionary algorithm that can find multiple Pareto-optimal solutions.

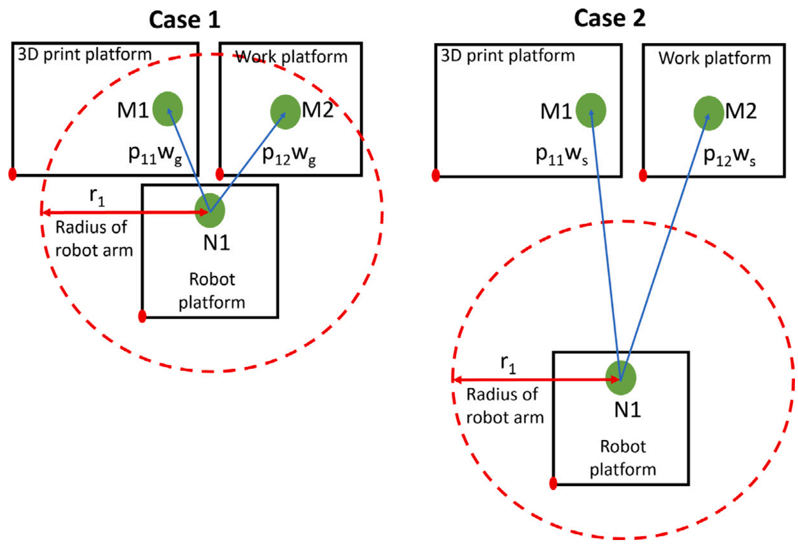

Fig. 8. Example of the Piecewise function for weight works. In case 1 the robot platform can reach the platforms and the w_g weight is used, while in case 2 the robot platform cant reach the points and the w_s is used.

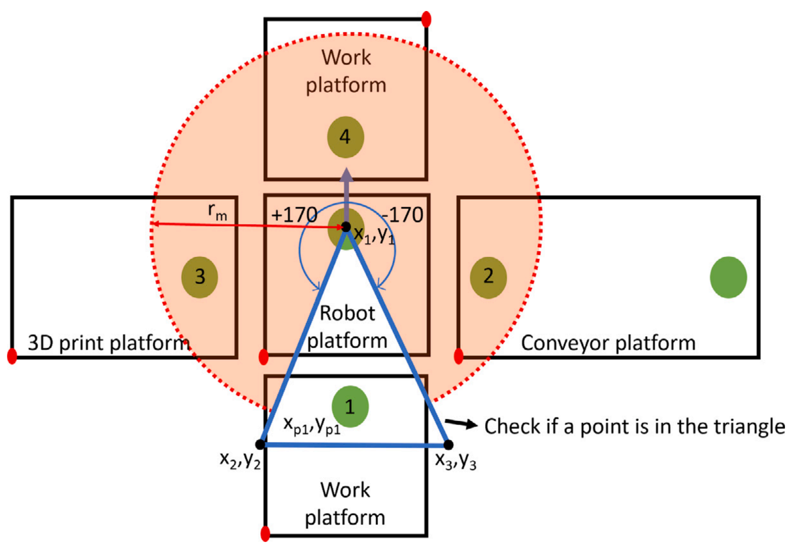

Fig. 9. Illustration of the reachable area for a robot arm, where the blind zone of the robot arm is shown.

It is an improvement on NSGA in terms of computational complexity, the need to specify sharing parameters and the lack of elitism [37].

The NSGA2 algorithm is implemented in python using the library Pymoo [41] and a flowchart of the algorithm can be seen in Fig. 11.

The input to the system is a list with all platforms in each section. For example, if a section contains *"3D printer, Robot platform, Work Table - 1, Conveyor out"*, then a part will move from the 3D printing platform to the work table - 1 to do a process and move out of the section with the conveyor platform. The robot platform is used to move the parts between the platforms. Fig. 12 shows the input to the system and the resulting layout.

The list is used to determine how the manufactured parts move through the system and the size of each platform. Then, the mathematical model is used for optimization with NSGA2 to find a layout.

In this project Pymoo 0.5.0 is used, and the optimization is executed on an AMD Ryzen 9 3950X processor.

A video example of when the layout optimization is running can be seen at https://youtu.be/UNsugBOi4cs. The video shows the best solution for each generation.

4.2. Digital model, simulation, and IIoT

It is difficult to describe and include all restrictions in a mathematical model. Making the model too complex can also make the problem unsolvable. It can therefore be beneficial to have a simpler mathematical model and connect the solution generated from the mathematical model with simulation tools, as a second layer to validate/test the solution. For this purpose, Visual Components Premium 4.4 [42] is used. Visual Components is a visual simulation software used to design and optimize manufacturing systems. It is possible to use Visual Components both for developing a visual digital model of the system, as well as for running manufacturing simulations. Hence, Visual Components is used to generate a digital model from the optimization, and then the digital model is used to run the simulation.

There is also a need for communication between all parts of the system. Since the system is made to be flexible, where the platforms can be moved to any position in the manufacturing environment. One method to allow for communication in a system is to use IIoT. IIoT is an extension of IoT in industrial applications and has a strong focus on machine-to-machine communication [43]. It is therefore used

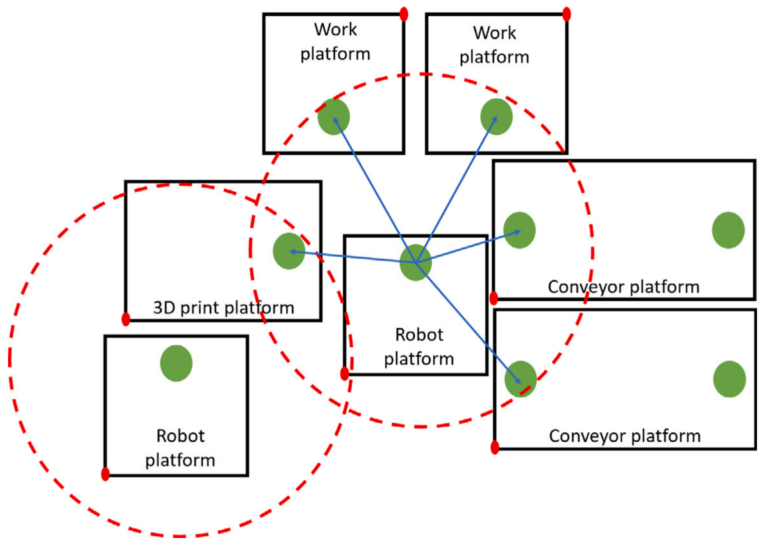

Fig. 10. Illustration of one robot arm that can move parts between all platforms, while the last robot arm cant perform any tasks.

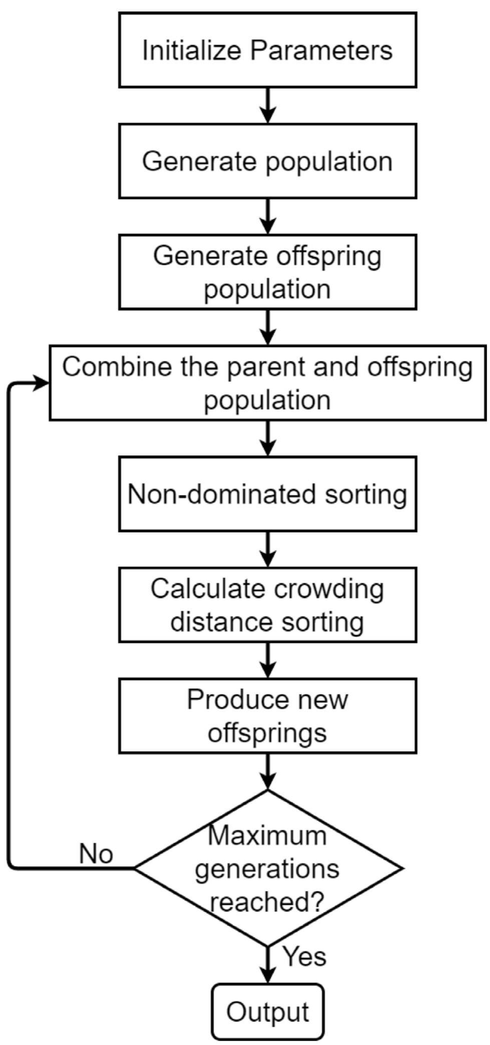

Fig. 11. A flowchart of the NSGA2 algorithm.

for communication with the platforms in the system and control the mobile robots. IIoT can also be used to transfer the layout to the mobile robot for automatic configuration of the system. In this system, the Open Platform Communications Unified Architecture (OPC UA) is used for IIoT. The OPC UA is an IEC 62541 standard, often used for communication between industrial equipment [44]. In addition, Visual Components support connectivity functions such as OPC UA and can therefore be used to connect the optimization simulation and physical system together.

The layout program in python is therefore connected to an OPC UA server, where the solution from the optimization is directly sent over to Visual Components. An illustration of how the system is connected and setup can be seen in Fig. 13.

From the layout program, the task order of the machines and the positions of all the platforms are sent over to Visual Components. When the data has been transferred, the layout is generated, the simulation is programmed automatically, and the simulation is then executed. If there is a problem when running the simulation, it will be stopped and an error message will be returned. As mentioned in Section 3.3, the robot arm might not be capable of picking up an item at certain angles of the tool center point. Therefore, the simulation serves as a verification tool to determine if the robot arm can pick the item.

In addition, the simulation can be used to:

- Validate if the RMS looks reasonable.
- Check if there is any collision between the platforms.
- Check if there is any collision when the robot arm is working.

If one of the tests fails, the simulation sends a message back to the layout program that the solution is not satisfactory. Then, the layout program will send the second-best solution and the simulation is again tested. A flowchart showing how the system work can be seen in Fig. 14.

4.3. Configuration testing

To showcase the layout generation in python, four different manufacturing layouts were tested. The layouts are tested for both optimization with rotation between 0 to 360 degrees and for fixed 0, 90, 180, and 270 degrees rotation. For the generated layouts, 3D printers, work platforms, and conveyors are used. The work platform is modeled as simple tables in the digital model. However, they are meant to represent manufacturing processes such as CNC machining, coordinate-measuring machine, assembly or other manufacturing processes.

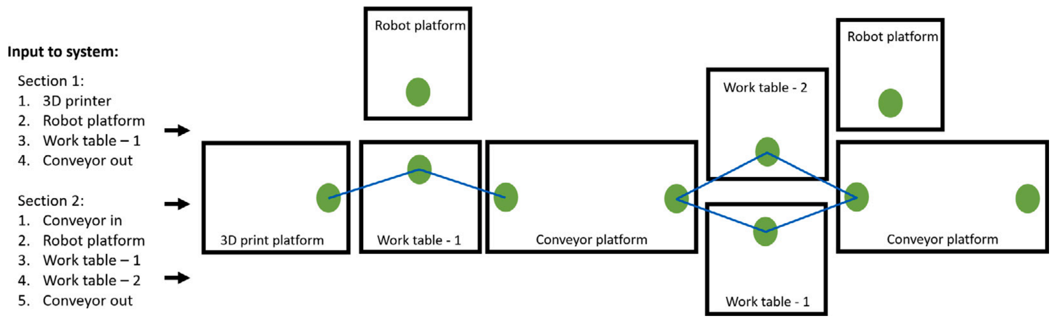

Fig. 12. The input to the optimization model and the system.

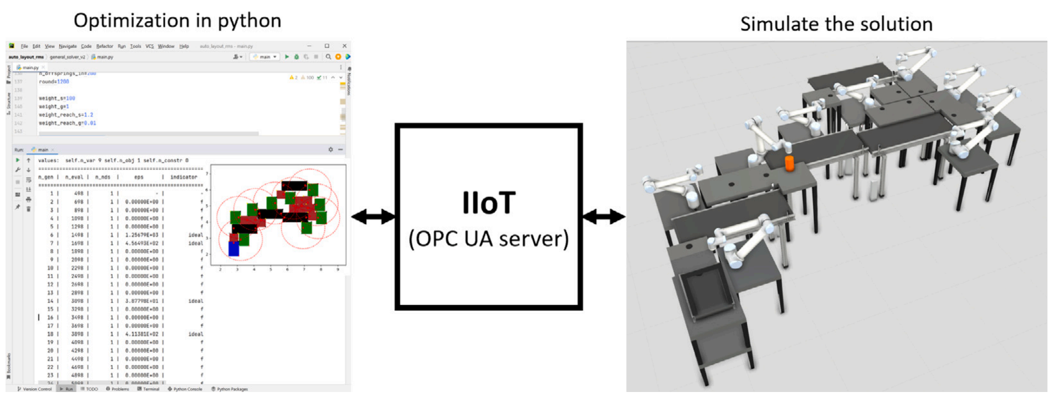

Fig. 13. How the optimization program in python is connected to the simulation software Visual Components.

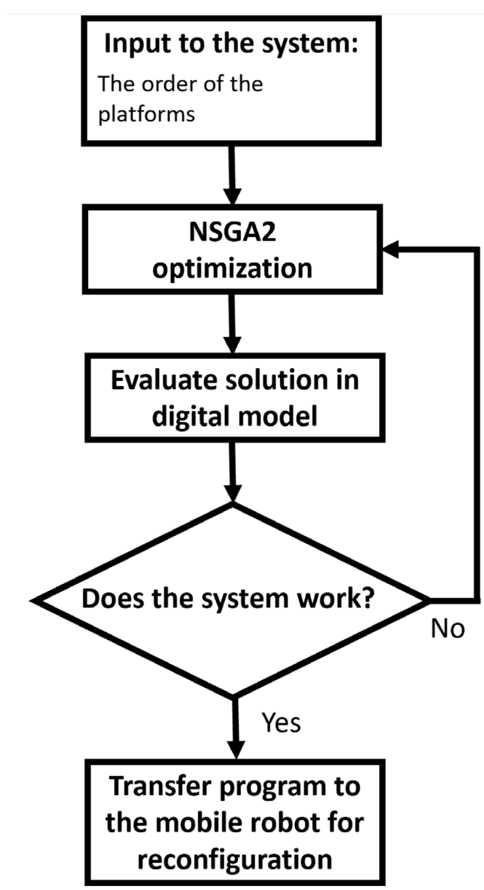

Fig. 14. Flowchart of how the smart layout design system works.

4.3.1. Layout 1 (three platforms, simplest form)

The first layout is the simplest form of the system. It includes a 3D printer platform, a robot platform and a conveyor. The results from the optimization can be seen in Fig. 15. A video of the simulation can be found at https://youtu.be/YVbpl2U_L8I.

4.3.2. Layout 2 (seven platforms in one line)

The second layout has one section with two 3D printers as input, two work platforms in parallel, two robot platforms and a conveyor. The result from the optimization can be seen in Fig. 16 and a video of the simulation in https://youtu.be/MTCSDvy0Qag.

4.3.3. Layout 3 (two sections)

There are two sections for the third layout. In this case, the conveyor is used as a bridge between the two sections. The idea of this layout is to showcase how parallel systems can be connected to create larger manufacturing layouts. The results are shown in Fig. 17, and a video demonstration can be found at https://youtu.be/gZxg1X57g3Y.

4.3.4. Layout 4 (big system)

The last layout consists of four sections with different amounts of platforms in each section. This is to test the optimization on a large system and see how much time it takes to solve the problem. Fig. 18 shows the results from the optimization and a video can be found at https://youtu.be/GFiIdPl_0_E.

Table 1 provides details on the optimization time and the number of generations necessary to produce the generated layouts.

4.4. Test on a physical system

To test and validate the layout, the system is tested on a physical RMS. The RMS consists of five platforms:

- Robot arm 1 (Nachi MZ07)
- Robot arm 2 (Scara Adept 604)

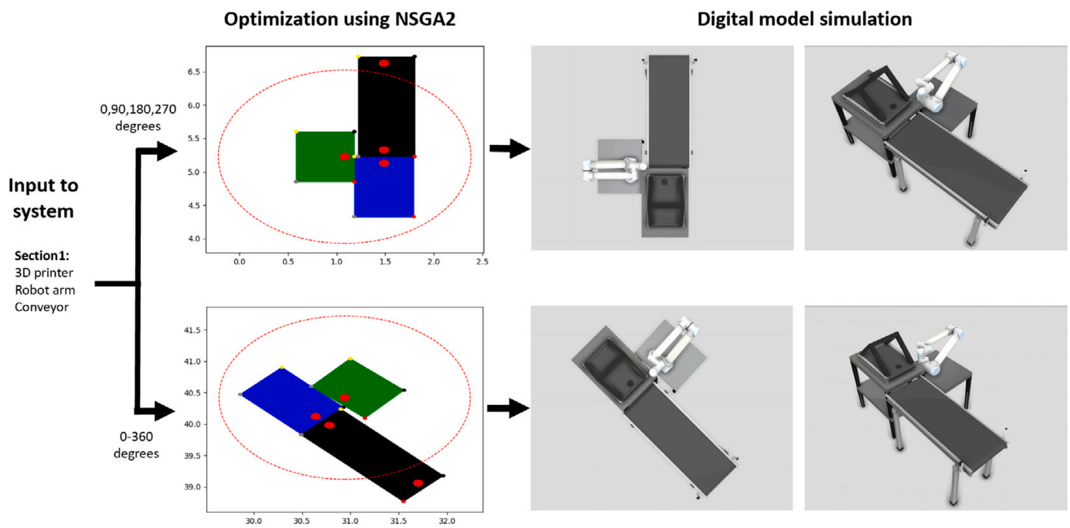

Fig. 15. The result after running optimization for layout one. On the left side is the input to the Pymoo optimizer, in the middle is the result from the optimization, and on the right side is the simulation.

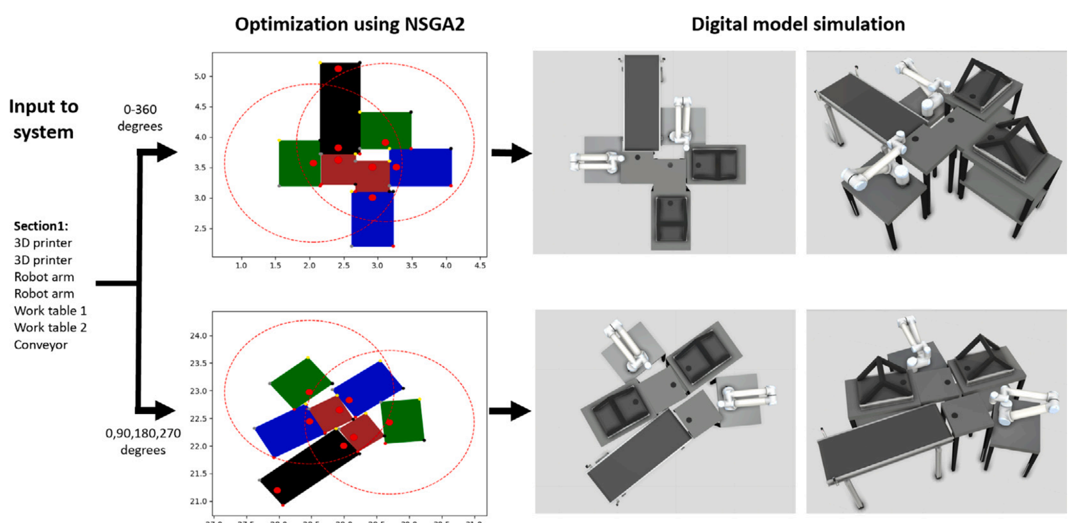

Fig. 16. The result after running optimization for layout two. On the left side is the input to the Pymoo optimizer, in the middle is the result from the optimization, and on the right side is the simulation.

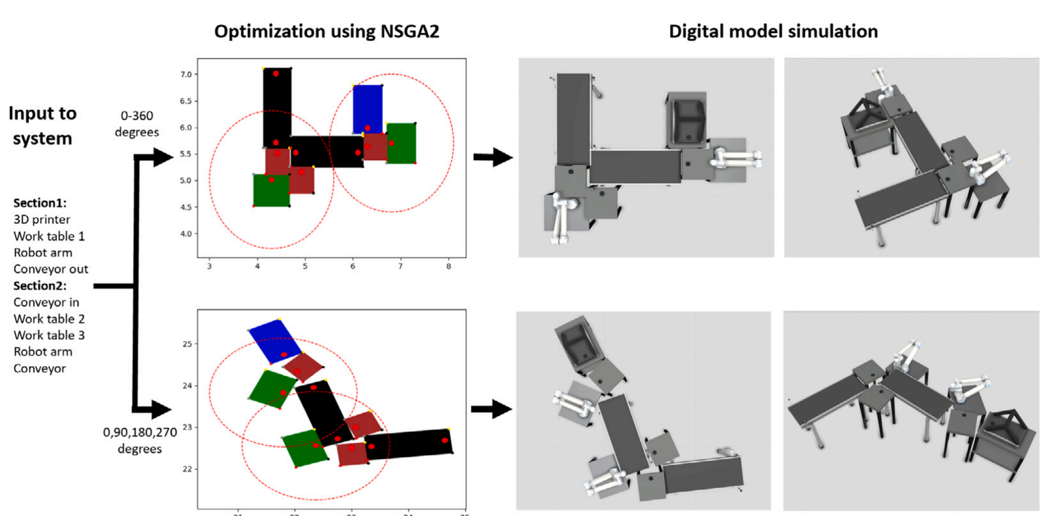

Fig. 17. The result after running optimization for layout three. On the left side is the input to the Pymoo optimizer, in the middle is the result from the optimization, and on the right side is the simulation.

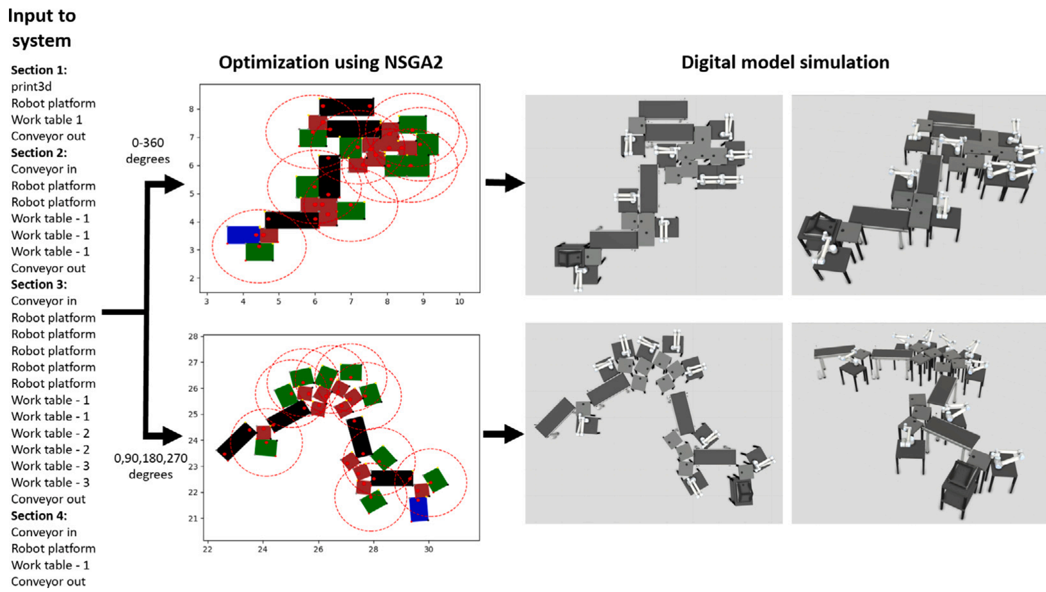

Fig. 18. The result after running optimization for layout four. On the left side is the input to the Pymoo optimizer, in the middle is the result from the optimization, and on the right side is the simulation.

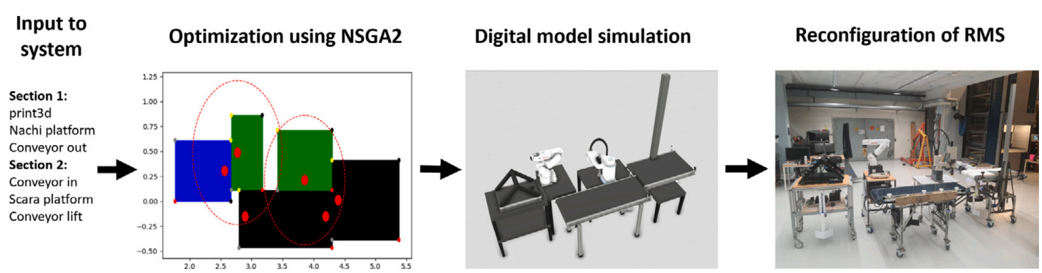

Fig. 19. Illustration of how the system works. On the left side, the input to the optimization, then the resulting layout from optimization, which is transferred over to the simulation/digital model, and on the right side the layout on the physical RMS.

Table 1
The table shows the optimization time, for the layouts tested.

	Rotation	Optimization time (min)	Generations	Number of platforms
Layout 1:	0–360	0.88	393	3
Layout 1:	0, 90, 180, 270	0.72	325	3
Layout 2:	0–360	25.66	1400	7
Layout 2:	0, 90, 180, 270	12.72	750	7
Layout 3:	0–360	77.75	3985	8
Layout 3:	0, 90, 180, 270	17.99	800	8
Layout 4:	0–360	1838.49	8380	25
Layout 4:	0, 90, 180, 270	898.95	2685	25

- 3D print platform
- Conveyor platform
- Conveyor with lifting platform

These platforms can be moved and rearranged automatically by the use of a mobile robot. The mobile robot is equipped with a docking module on top, which allows it to fasten itself to the platform, and can pull the platform. The system is controlled through the OPC UA standard and it is therefore possible to connect the optimization and digital model simulation in Visual Components directly to the physical system. An illustration of the connection can be seen in Fig. 19.

When testing the layout of the physical system, it was shown that it is not feasible to have the platforms too close to each other. This is due to the mobile robots' low accuracy when reconfiguring the platform. To solve this issue, all platforms receive a safety distance between each other, which equals 200 mm.

A video demonstrating the system can be found at https://youtu.be/TqimTSBvpTs. In the video, a layout is generated with NSGA2 optimization, tested with the digital model simulation, and then sent to the physical system for automatic reconfiguration with the mobile robot.

5. Discussion

The idea of RMS is to have a manufacturing system that can rapidly be changed depending on what is being manufactured. However, designing and reconfiguring such a system is both time-consuming and costly due to the requirement of excessive human labor. In this paper, we propose a new approach to automize the reconfiguration process of RMS. We combine optimization with industry 4.0 technologies, i.e., IIoT, digital model, simulation, and advanced robotics to create a smart layout design system for RMS.

We first formulate a mathematical model for a platform-based RMS proposed by Arnarson [10]. The mathematical model for the system is used to yield a score for the system, where penalties are added to the score if certain criteria are not met. The main goal of the system is to reduce the distance between the points of the platforms while all movement platforms can reach the points. This model is then used with an NSGA2 optimizer to find a near-optimal layout. The model can be used for manufacturing platforms of different shapes, and different constraints can be added depending on the requirements of the platforms. Constraints and platforms can easily be changed, and the system can consider other optimization requirements or constraints.

Only using the mathematical model for optimization can be limited, and it can be time-consuming to model all constraints. Therefore, adding a digital model and simulation helps test the manufacturing system. It can be used as a verification tool to validate if the solution from the optimization work in a simulation environment. In addition, connecting the optimization model to the simulation can allow for bi-directional communication between both systems. As a result, the simulation software can provide a quality check and safeguard on the optimization program's solution.

In this project, we tested different rotations for the platforms. One with fixed 0, 90, 180, and 270 degrees and one which is between 0 to 360 degrees. As can be seen from the four layouts (Figs. 15–18), when 0–360 degrees is used, the optimization function will not converge and will therefore not give an optimal layout. By limiting the rotation to 0, 90, 180, and 270 degrees, an improved solution is obtained compared to using 0 to 360 degrees. Having the rotation between 0–360 degrees adds more complexity and possibilities to the system and from the optimization, it looks like the NSGA2 gets stuck. This may be due to the hyper-parameters for the NSGA2 are not exploratory enough.

We have demonstrated four different cases of how the system works and tested the layout optimization on a physical system. In the physical test, we connected the optimization, digital model/simulation and the physical system by using an IIoT (OPC UA) server. Being able to connect the optimization model and digital model directly to a physical RMS allows for increased automation. On the other hand, manually designing the same system would require a human operator with expertise in manufacturing to design the layout. Simulating the system would also require programming, which is time-consuming. The proposed system automates the optimization of the layout, virtually test the layout with simulation, and reconfigure a system with a mobile robot allowing for full reconfiguration without any human intervention. However, the system may not be able to provide the shortest moving path for the workpieces and the most effective use of the RMS modules. Therefore, this system can work as a support tool to help the human operator quickly design and adjust the RMS layout for customized orders, which forms the foundation of the future human–machine interaction in a collaborative manufacturing environment. This system is well-suited for manufacturing systems that undergo frequent process reconfiguration, such as companies operating within industries characterized by high product variety and short product lifecycles, e.g., electronics manufacturing or manufacturing of customized products. Implementing a smart layout design system can greatly benefit manufacturing companies specializing in mass customization or mass personalization by streamlining and reducing the time required for planning and executing new production runs.

As shown in 18, the layout is chaotic and can be considered as not acceptable from a safety and industrial standards perspective. This dilemma is most likely caused by the unrestricted use of platforms to minimize the total movement distance while simultaneously ensuring the reach to all points. A possible solution would be to let the optimization system determine how many platforms are needed, thereby removing unnecessary platforms. Besides, another objective function may also be added to minimize the use of platforms so that the resource requirement could be reduced. Moreover, safety rules and industrial standards can be added to the mathematical model to get a more realistic system.

6. Conclusion

In this paper, we proposed a novel method on how to solve the layout design problem for RMS. We used optimization together with the industry 4.0 technologies, i.e., IIoT, digital model, simulation, and advanced robotics to create a smart layout design for RMS. First, we propose a new mathematical model for the layout design of a platform-based RMS. The object of the mathematical model is to find a layout that minimizes the distance the product has to move while considering the constraints of the system. Then, the NSGA2 algorithm is used to search for an optimal or near optimal layout for the system. The layout is transferred to a digital model simulation software for testing and verification of the system in a virtual space. To showcase how the system works, four different demonstrations were created. The results showed that the mathematical model works and using NSGA2 for optimization can generate a layout automatically and be tested in the digital model. In addition, we also connect the optimization and digital model to a physical RMS to validate the proposed system.

6.1. Future works

6.1.1. Solve the optimization with 0–360 degrees

As mentioned in the discussion, when 0–360 degrees rotation is used, the system will not converge into a good layout. For further work, there should be done an investigation on how to make the system converge. In addition, when the system includes a lot of platforms, it can take a few days for the system to solve the problem. There should also be an investigation into how to improve the computational efficiency of the optimization problem.

6.1.2. Combine optimizations

There has been a lot of work on optimization for process planning, in what order the machines should be in, how many machines are required, how often the system should be reconfigured, and what is the best approach to reconfiguring the system. For further work, these optimization models should be combined together in one system to better model a close-to real-world manufacturing system. For example, when multiple RMSs are set up for different products, some platforms may need to be shared by different RMSs, so not only the positions of the platforms but also the timing for their use needs to be optimized.

6.1.3. Add more objectives and constraints to the system

More and different objectives and constraints can be added to the mathematical model in order to create a more realistic solution. For instance, in this paper, we assume that all parts move from one single point on the platforms. Therefore, adding a constraint that considers an area where parts can be placed would be more realistic and should be added to the optimization. Furthermore, adding another objective to minimize the use of platforms while simultaneously ensuring an acceptable level of reach to all points may help to solve the problem shown in Fig. 18. Moreover, the model can be developed in a 3D space and also take into consideration the limitations in the orientation of the robot arms' tool center point (yaw, pitch, and roll).

6.1.4. General manufacturing systems

Use the same methods in this project to find the optimal layout of a general manufacturing system can be created. As manufacturing systems are usually divided into cells, the position of the machines, walking areas, where the robot should be placed and the different stations can be used to create the most optimal layout depending on the criteria of the model.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 825196.

References

[1] Koren Y, Heisel U, Jovane F, Moriwaki T, Pritschow G, Ulsoy G, Van Brussel H. Reconfigurable manufacturing systems. CIRP Ann 1999;48(2):527–40. http://dx.doi.org/10.1016/S0007-8506(07)63232-6, URL https://www.sciencedirect.com/science/article/pii/S0007850607632326.

[2] Koren Y, Shpitalni M. Design of reconfigurable manufacturing systems. J Manuf Syst 2010;29(4):130–41. http://dx.doi.org/10.1016/j.jmsy.2011.01.001, URL https://www.sciencedirect.com/science/article/pii/S0278612511000021.

[3] Rösiö C, Bruch J. Exploring the design process of reconfigurable industrial production systems: Activities, challenges, and tactics. J Manuf Technol Manage 2017;29(1):85–103. http://dx.doi.org/10.1108/JMTM-06-2016-0090.

[4] Maganha I, Silva C. A theoretical background for the reconfigurable layout problem. Procedia Manuf 2017;11:2025–33. http://dx.doi.org/10.1016/j.promfg.2017.07.354, URL https://www.sciencedirect.com/science/article/pii/S2351978917305620. 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy.

[5] Sabioni R, Daaboul J, Le Duigou J. Optimization of reconfigurable manufacturing systems configuration: a literature review. 2020.

[6] Maganha I, Silva C, Ferreira LMDF. The layout design in reconfigurable manufacturing systems: a literature review. Int J Adv Manuf Technol 2019;105. http://dx.doi.org/10.1007/s00170-019-04190-3.

[7] Singh A, Gupta P, Asjad M. Reconfigurable manufacturing system (RMS): Accelerate towards industries 4.0. SSRN Electron J 2019. http://dx.doi.org/10.2139/ssrn.3354485.

[8] Bortolini M, Galizia FG, Mora C. Reconfigurable manufacturing systems: Literature review and research trend. J Manuf Syst 2018;49:93–106. http://dx.doi.org/10.1016/j.jmsy.2018.09.005, URL https://www.sciencedirect.com/science/article/pii/S0278612518303650.

[9] Morgan J, Halton M, Qiao Y, Breslin JG. Industry 4.0 smart reconfigurable manufacturing machines. J Manuf Syst 2021;59:481–506. http://dx.doi.org/10.1016/j.jmsy.2021.03.001, URL https://www.sciencedirect.com/science/article/pii/S027861252100056X.

[10] Arnarson H, Mahdi H, Solvang B, Bremdal BA. Towards automatic configuration and programming of a manufacturing cell. J Manuf Syst 2022;64:225–35. http://dx.doi.org/10.1016/j.jmsy.2022.06.005, URL https://www.sciencedirect.com/science/article/pii/S0278612522001005.

[11] Al-Sahaf H, Bi Y, Chen Q, Lensen A, Mei Y, Sun Y, Tran B, Xue B, Zhang M. A survey on evolutionary machine learning. J R Soc New Zealand 2019;49(2):205–28. http://dx.doi.org/10.1080/03036758.2019.1609052, arXiv:https://doi.org/10.1080/03036758.2019.1609052.

[12] Yelles-Chaouche AR, Gurevsky E, Brahimi N, Dolgui A. Reconfigurable manufacturing systems from an optimisation perspective: a focused review of literature. Int J Prod Res 2021;59(21):6400–18. http://dx.doi.org/10.1080/00207543.2020.1813913, arXiv:https://doi.org/10.1080/00207543.2020.1813913.

[13] Mariem B, Marc Z, AFFONSO R, Masmoudi F, Haddar M. A methodology for solving facility layout problem considering barriers – genetic algorithm coupled with A* search. J Intell Manuf 2020;31. http://dx.doi.org/10.1007/s10845-019-01468-x.

[14] Lim ZY, Ponnambalam S, Izui K. Multi-objective hybrid algorithms for layout optimization in multi-robot cellular manufacturing systems. Knowl-Based Syst 2017;120:87–98. http://dx.doi.org/10.1016/j.knosys.2016.12.026, URL https://www.sciencedirect.com/science/article/pii/S0950705116305317.

[15] Guo H, Zhu Y, Zhang Y, Ren Y, Chen M, Zhang R. A digital twin-based layout optimization method for discrete manufacturing workshop. Int J Adv Manuf Technol 2021;112(5–6):1307–18. http://dx.doi.org/10.1007/s00170-020-06568-0.

[16] Al-Zubaidi SQD, Fantoni G, Failli F. Analysis of drivers for solving facility layout problems: A literature review. J Ind Inf Integr 2021;21:100187. http://dx.doi.org/10.1016/j.jii.2020.100187, URL https://www.sciencedirect.com/science/article/pii/S2452414X20300625.

[17] Azab A, ElMaraghy H. Mathematical modeling for reconfigurable process planning. CIRP Ann 2007;56(1):467–72. http://dx.doi.org/10.1016/j.cirp.2007.05.112, URL https://www.sciencedirect.com/science/article/pii/S0007850607001138.

[18] Chaube A, Benyoucef L, Tiwari M. An adapted NSGA-2 algorithm based dynamic process plan generation for a reconfigurable manufacturing system. J Intell Manuf 2010;23:1–15. http://dx.doi.org/10.1007/s10845-010-0453-9.

[19] Abderrahmane B, Dahane M, Benyoucef L. A new heuristic for integrated process planning and scheduling in reconfigurable manufacturing systems. Int J Prod Res 2014;52:3583–94. http://dx.doi.org/10.1080/00207543.2013.878056.

[20] Wang W, Koren Y. Scalability planning for reconfigurable manufacturing systems. J Manuf Syst 2012;31(2):83–91. http://dx.doi.org/10.1016/j.jmsy.2011.11.001, URL https://www.sciencedirect.com/science/article/pii/S0278612511000999.

[21] Saxena LK, Jain PK. A model and optimisation approach for reconfigurable manufacturing system configuration design. Int J Prod Res 2012;50(12):3359–81. http://dx.doi.org/10.1080/00207543.2011.578161, arXiv:https://doi.org/10.1080/00207543.2011.578161.

[22] Bensmaine A, Dahane M, Benyoucef L. A non-dominated sorting genetic algorithm based approach for optimal machines selection in reconfigurable manufacturing environment. Comput Ind Eng 2013;66(3):519–24. http://dx.doi.org/10.1016/j.cie.2012.09.008, URL https://www.sciencedirect.com/science/article/pii/S0360835212002240. Special Issue: The International Conferences on Computers and Industrial Engineering (ICC&IEs) - series 41.

[23] Moghaddam SK, Houshmand M, Valilai OF. Configuration design in scalable reconfigurable manufacturing systems (RMS); a case of single-product flow line (SPFL). Int J Prod Res 2018;56(11):3932–54. http://dx.doi.org/10.1080/00207543.2017.1412531, arXiv:https://doi.org/10.1080/00207543.2017.1412531.

[24] Moghaddam SK, Houshmand M, Saitou K, Valilai OF. Configuration design of scalable reconfigurable manufacturing systems for part family. Int J Prod Res 2020;58(10):2974–96. http://dx.doi.org/10.1080/00207543.2019.1620365, arXiv:https://doi.org/10.1080/00207543.2019.1620365.

[25] Guan X, Dai X, Qiu B, Li J. A revised electromagnetism-like mechanism for layout design of reconfigurable manufacturing system. Comput Ind Eng 2012;63(1):98–108. http://dx.doi.org/10.1016/j.cie.2012.01.016, URL https://www.sciencedirect.com/science/article/pii/S0360835212000289.

[26] Benderbal HH, Dahane M, Benyoucef L. Exhaustive search based heuristic for solving machine layout problem in reconfigurable manufacturing system design. IFAC-PapersOnLine 2018;51(11):78–83. http://dx.doi.org/10.1016/j.ifacol.2018.08.238, URL https://www.sciencedirect.com/science/article/pii/S2405896318313624. 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018.

[27] Haddou Benderbal H, Benyoucef L. Machine layout design problem under product family evolution in reconfigurable manufacturing environment: A two-phase based-AMOSA approach. Int J Adv Manuf Technol 2019;104. http://dx.doi.org/10.1007/s00170-019-03865-1.

[28] Besbes M, Mahjoub YI, Bonte T, Berger T, Sallez Y, Zolghadri M. Solving facility layout problem with safety consideration of reconfigurable manufacturing and assembly systems. Procedia CIRP 2021;104:1942–7. http://dx.doi.org/10.1016/j.procir.2021.11.328, URL https://www.sciencedirect.com/science/article/pii/S2212827121012269. 54th CIRP CMS 2021 - Towards Digitalized Manufacturing 4.0.

[29] Yamada Y. Dynamic reconfiguration of reconfigurable manufacturing systems using particle swarm optimization. In: Proceedings 2006 IEEE international conference on robotics and automation, 2006. ICRA 2006. 2006, p. 1444–9. http://dx.doi.org/10.1109/ROBOT.2006.1641912.

[30] Zheng L, Zhu L, Wang B, Bai L. A simulation analysis of facility layout problems in reconfigurable manufacturing systems. In: 2013 international conference on computer sciences and applications. 2013, p. 423–7. http://dx.doi.org/10.1109/CSA.2013.106.

[31] Petroodi SEH, Eynaud ABD, Klement N, Tavakkoli-Moghaddam R. Simulation-based optimization approach with scenario-based product sequence in a reconfigurable manufacturing system (RMS): A case study. IFAC-PapersOnLine 2019;52(13):2638–43. http://dx.doi.org/10.1016/j.ifacol.2019.11.605, URL https://www.sciencedirect.com/science/article/pii/S2405896319315927. 9th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2019.

[32] Santos R, Toscano C, de Sousa JP. A new simulation-based approach in the design of manufacturing systems and real-time decision making. IFAC-PapersOnLine 2021;54(1):282–7. http://dx.doi.org/10.1016/j.ifacol.2021.08.033, URL https://www.sciencedirect.com/science/article/pii/S2405896321007424. 17th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2021.

[33] Touckia JK, Hamani N, Kermad L. Digital twin framework for reconfigurable manufacturing systems (RMSs): design and simulation. Int J Adv Manuf Technol 2022;120(7–8):5431–50. http://dx.doi.org/10.1007/s00170-022-09118-y.

[34] Leng J, Liu Q, Ye S, Jing J, Wang Y, Zhang C, Zhang D, Chen X. Digital twin-driven rapid reconfiguration of the automated manufacturing system via an open architecture model. Robot Comput-Integr Manuf 2020;63:101895. http://dx.doi.org/10.1016/j.rcim.2019.101895, URL https://www.sciencedirect.com/science/article/pii/S073658451930167X.

[35] Kurniadi KA, Lee S, Ryu K. Digital twin approach for solving reconfiguration planning problems in RMS. In: Moon I, Lee GM, Park J, Kiritsis D, von Cieminski G, editors. Advances in production management systems. Smart manufacturing for industry 4.0. Cham: Springer International Publishing; 2018, p. 327–34.

[36] Hajjem E, Benderbal HH, Hamani N, Dolgui A. Digital twin framework for reconfigurable manufacturing systems: Challenges and requirements. In: Dolgui A, Bernard A, Lemoine D, VonCieminski G, Romero D, editors. Advances in production management systems: artificial intelligence for sustainable and resilient production systems, APMS 2021, PT II. IFIP advances in information and communication technology, vol. 631, IFIP Working Grp 5 7 Advances Prod Management Syst; IMT Atlantique, Campus Nantes; Centrale Nantes; Univ Nantes; Rennes Business Sch; Audecia Business Sch; 2021, p. 553–62. http://dx.doi.org/10.1007/978-3-030-85902-2_59, IFIP WG 5.7 International Conference on Advances in Production Management Systems (APMS), Nantes, FRANCE, SEP 05-09, 2021.

[37] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 2002;6(2):182–97. http://dx.doi.org/10.1109/4235.996017.

[38] Bensmaine A, Dahane M, Benyoucef L. Simulation-based NSGA-II approach for multi-unit process plans generation in reconfigurable manufacturing system. In: ETFA2011. 2011, p. 1–8. http://dx.doi.org/10.1109/ETFA.2011.6059047.

[39] Verma S, Pant M, Snasel V. A comprehensive review on NSGA-II for multi-objective combinatorial optimization problems. IEEE Access 2021;9:57757–91. http://dx.doi.org/10.1109/ACCESS.2021.3070634.

[40] Yusoff Y, Ngadiman MS, Zain AM. Overview of NSGA-II for optimizing machining process parameters. Procedia Eng 2011;15:3978–83. http://dx.doi.org/10.1016/j.proeng.2011.08.745, URL https://www.sciencedirect.com/science/article/pii/S1877705811022466. CEIS 2011.

[41] Blank J, Deb K. pymoo: Multi-objective optimization in python. IEEE Access 2020;8:89497–509.

[42] Visual Components. 2022, URL https://www.visualcomponents.com. [Online; accessed 10. Sep. 2022].

[43] Munirathinam S. Chapter six - industry 4.0: Industrial internet of things (IIOT). In: Raj P, Evangeline P, editors. The digital twin paradigm for smarter systems and environments: the industry use cases. Advances in computers, vol. 117, Elsevier; 2020, p. 129–64. http://dx.doi.org/10.1016/bs.adcom.2019.10.010, URL https://www.sciencedirect.com/science/article/pii/S0065245819300634.

[44] Foundation O. Unified architecture. 2022, URL https://opcfoundation.org/about/opc-technologies/opc-ua/. [Online; accessed 10. Sep. 2022].