Journal of Intelligent Manufacturing (2024) 35:793–806
https://doi.org/10.1007/s10845-023-02073-9

Task allocation model for human-robot collaboration with variable cobot speed

Maurizio Faccio[1] · **Irene Granata**[1] · **Riccardo Minto**[2]

Received: 19 May 2022 / Accepted: 1 January 2023 / Published online: 20 January 2023

Abstract

New technologies, such as collaborative robots, are an option to improve productivity and flexibility in assembly systems. Task allocation is fundamental to properly assign the available resources. However, safety is usually not considered in the task allocation for assembly systems, even if it is fundamental to ensure the safety of human operator when he/she is working with the cobot. Hence, a model that considers safety as a constraint is here presented, with the aim to both maximize the productivity in a collaborative workcell and to promote a secure human robot collaboration. Indexes that consider both process and product characteristics are considered to evaluate the quality of the proposed model, which is also compared with one without the safety constraint. The results confirm the validity and necessity of the newly proposed method, which ensures the safety of the operator while improving the performance of the system.

Keywords Collaborative systems · Cobot · Task allocation, Safety

Introduction

The current trend asks the industrial production for mass customization, which enhances the variety of products and decreases the batch volume to individual products (Da Silveira et al., 2001), thus requiring assembly systems to be more performing (Azzi et al., 2012).

To achieve that, new technologies are introduced, e.g. collaborative robots (cobots), whose demand is increasing (Tan et al., 2010). Collaborative assembly systems (CAS) (Faccio et al., 2019) can guarantee a production increase and the flexibility requested by the market (El Zaatari et al., 2019); however, safety is still considered one of the main issues in this type of cell (Surdilovic et al., 2010).

The trade-off between productivity, flexibility, and safety, Gerbers et al. (2018), leads to the disequilibrium of assembly systems toward one of them. To correctly maximize all of them, it is necessary to develop a proper solution (Bautista & Pereira, 2007), i.e., it is necessary to introduce a safety constraint in the task allocation problem. This latter is a very studied topic for assembly systems and different authors presented their solutions.

Previous strategies, reported in Johannsmeier and Haddadin (2016), Krüger et al. (2009), Müller et al. (2017) presented frameworks for human robot collaboration (HRC) with the introduction of safety systems in highly productive systems. However, the solutions presented so far focus more on the distribution of tasks and they did not include safety as a constraint.

The proposed work aims to introduce a new mathematical method for collaborative systems, introducing a safety constraint in the model. Hence, the newly proposed method defines a task allocation strategy characterized by high productivity, as it minimizes the makespan, but also by a minimum interference between the resources, leading to a safe work space. This work considers typical characteristics of HRC systems, e.g., different tasks time for the resources, technological constraints, and the parallelization of the tasks between the resources. This can have a strong impact in the industrial field because it allows having a performing and safe system, without the introduction of other devices, which would increase the cost while lowering the flexibility. Moreover, cobots' speed is usually set much lower than

✉ Maurizio Faccio
maurizio.faccio@unipd.it

[1] Department of Management and Engineering, University of Padova, Stradella San Nicola,3, Vicenza, Italy

[2] Department of Industrial Engineering, University of Padova, Via Venezia, 1, Padova, Italy

the effective capability of the cobot, due to safety reasons; indeed, *speed and separation monitoring* (SSM) methods may move the robot unnecessarily slow when close to the operator, whereas *power and force limiting* (PFL) methods may reduce the productivity when the operator is far from the cobot (Palleschi et al., 2021).

However, a proper task allocation may effectively improve the performance of the system by temporarily increasing the cobot speed when the distance between the resources is sufficient, in accordance with the standards and the cobot characteristics. Indeed, if the task allocation approach does not consider the distance between the resources, it forces the robot to maintain the speed to a minimum value due to safety reasons; on the other hand, by introducing a distance-related constraint it is possible to decrease the cobot speed by a factor, f_s, properly evaluated, only when the distance between the resources is not sufficient, thus effectively improving both the system performance and safety. This new model, in fact, maximizes the time in which the two resources are above the safety distance, allowing the cobot to keep a higher speed for more time, and this results in an increase of the efficiency but also in an improvement of the safety.

The paper is organized as follows: in Section 'Literature review' the state of the art regarding safety task allocation problem is presented; in Section 'Proposed task allocation' the hypothesis and the model are explained, while Section 'Model application and comparison' is for the application of the new model, with the results, and the comparison with the one without safety constraint. Lastly, Section 'Conclusion' concludes the work.

Literature review

In recent years, the global market requires increasingly customized products, i.e., small batches of products with a great variety (Faccio et al., 2020); hence, more flexible systems are required to achieve this goal (Browne et al., 1984; Boschetti et al., 2021a).

A first solution could be achieved by implementing a manual assembly system (MAS) (Faccio et al., 2019), which simply provides to assign to the human operator all the tasks to be performed, thus achieving the maximum degree of flexibility; however, manual systems are influenced by the performance of operators, their productivity, work rate, safety (Edmondson & Redford, 2002). Moreover, MASs are characterized by the high costs of the workforce.

However, to correctly move from the mass production paradigm to the mass customization one, it is necessary for a system to be competitive while ensuring the flexibility required. For these reasons, automatic systems such as collaborative robots (cobots) (Klumpp et al., 2019) are preferred, since they can guarantee the repeatability of robots with the flexibility of human operators (Simões et al., 2020). Indeed, cobots are designed to work in the same workspace of the operator without physical barriers, differently from traditional robots which are enclosed by safety fences due to safety reasons. Therefore, safety becomes a fundamental aspect due to the risk of contact between the robot and the person.

As stated by Villani et al. (2018), the main fields of research for human-robot collaborations (HRC) studies are:

- *Safety*, which concerns the safety standards introduced by international regulations [3-5], and how collaboration can take place.
- *Design methods*, which include the study of control laws, the choice of sensors, and tasks to be performed to ensure that the operator works safely side by side with the robot, actively sharing the workspace and tasks to be completed.

As safety is fundamental to accomplish a proper HRC, these two fields are strictly related. In order to ensure the safety of the operator, the cobot speed shall be reduced (Vicentini, 2020), affecting productivity.

In Faccio et al. (2019), an extensive analysis of the convenience of collaborative assembly systems has been carried out, showing the influence of product and process characteristics, the number of tasks, the percentage of assembly task allocation, and the degree of collaboration, on the system performance.

To achieve the maximum performance from these systems, it is important to properly address these characteristics, which is possible by correctly assigning the tasks between the resources, i.e., by solving an assembly line balancing (ALB) problem.

Previous works, which are presented in Table 1, have typically solved the task allocation problem by adopting solutions based on frameworks. The Table has been ordered according to the relevance of the studies, where relevance is understood as the number of citations per year (citations obtained from Google Scholar, 22 July 2022, 2022 = 1 year, 2021 = 2 years, etc).

Johannsmeier and Haddadin (2016), propose a framework divided into three levels, where the assembly process starts from the evaluation of information about tasks, products, and resources. They introduce the planning of cobot trajectories with sensors and vision systems. The authors address safety issues with the introduction of different systems, but it is not considered with the other optimization parameters.

Another framework-based solution was proposed by Tsarouchi et al. (2017). The authors present a decision algorithm that allows them to find a series of solutions, basing their evaluation on criteria presented in the literature, i.e., the capacity of the resource, if it is already working, and the resources' task times.

Table 1 State of the art on the task allocation problem

Authors	Task allocation			Input data				Security system		Method			Result
	Optimization of productivity and cycle time	Design workspace w.r.t. scheduling	Dynamic allocation and flexibility	Task data, graph, resources	CAD Data	Product Characteristic	Other/None	Speed reduction	Software or hardware systems	Framework	Genetic algorithm	Heuristic	
Liu and Wang (2018)	x								x				Introduction of vision systems to guarantee a safety HRI (Human-Robot Interaction) and high productivity
Krüger et al. (2009)	x			x					x	x			Division of tasks to have high productivity with safety systems like laser scanner, safety sensor, collision avoidance
Tsarouchi et al. (2017)	x			x						x			Developed a planning system that minimizes makespan and average resource utilization
Ranz et al. (2017)			x	x		x				x			Developed a method for task allocation based on resources capability
Johannsmeier and Haddadin (2016)			x	x					x	x			Developed a planning system that results dynamic and flexible
Malik and Bilberg (2019)	x					x				x			Developed a method to assign tasks to robot or operator based on the complexity of tasks themselves
Pearce et al. (2018)	x			x							x		Reduce makespan considering physical strain
Byner et al. (2019)	x							x	x				Introduction of SSM to have production consistency
Faccio et al. (2019)	x			x			x			x			An extensive economic analysis highlighted which factors influence the convenience of cobots
Michalos et al. (2018b)		x			x	x				x			Developed a planning system that considers layout
Chen et al. (2013)	x			x							x		Developed a mathematical model for task allocation and resolved with GA

Table 1 continued

Authors	Task allocation			Input data				Security system		Method			Result
	Optimization of productivity and cycle time	Design workspace w.r.t. scheduling	Dynamic allocation and flexibility	Task data, graph, resources	CAD Data	Product Characteristic	Other/ None	Speed reduction	Software or hardware systems	Framework	Genetic algorithm	Heuristic	
Nikolakis et al. (2018)			x	x						x			Reduces time spent on a specific job
Bettoni et al. (2020)	x			x					x			x	Adaptive human-robot collaboration based on a decision maker that changes the support level offered by a collaborative robot.
Bogner et al. (2018)	x			x								x	Developed a mathematical model for task allocation and resolved with heuristic
Takata and Hirano (2011)			x				x			x			Developed a method to find the best task allocation in the manufacturing process with multiple scenarios
Gualtieri et al. (2020)	x								x	x			Efficiency is ensured thanks to automatic vision systems and avoiding re-orientations of the parts
Fechter et al. (2018)		x		x	x							x	Developed a planning system that considers layout
Galin et al. (2020)			x						x				Intelligent manufacturing process based on the division of task between the two resources, taking into account safety measures
Müller et al. (2017)			x			x						x	Through the use of different methodologies a safe task allocation was realized
Tan et al. (2010)			x	x			x			x			Shorter assembly time using collaboration

On the other hand, (Michalos et al., 2018b; Fechter et al., 2018) suggest solving the task allocation problem with the development of an appropriate layout. Their method is based on an analysis of several criteria, comparing the different solutions in terms of productivity, ergonomics, and process quality with the possible layouts that allow efficient use of the work plan.

Ranz et al. (2017) try to determine optimal scheduling by taking into account the ability of the resources, thus, placing greater emphasis on the final quality of the product. In this way, they aim to provide an objective approach based on empirical observations for the allocation of those actions that cannot be assigned a priori to one resource or to another.

Differently, Chen et al. (2013) propose a mathematical model that describes the task balancing problem and whose resolution leads to a task scheduling that minimizes the total assembly time of the product. The authors assume that the entire process of collaborative assembly can be divided into two parts: parallel or sequential assembly, i.e., tasks that can be executed in parallel or that have a sequential relationship. Thus, their optimization algorithm is divided into a number of sub-problems equal to the number of the previously identified groups, leading to significant simplification.

Another solution for the task allocation problem is proposed by Boschetti et al. (2021b), where a model for collaborative assembly line balancing (C-ALB) is developed, including paralleling tasks and collaboration.

Task assignment is also considered by Michalos et al. (2018b), where a framework that considers the design of the layout for HRC is presented. Through a multi-criteria approach, different solutions are analyzed in terms of productivity, quality, and efficiency.

A robotic system implementation that combines the capabilities of industrial robots with those of human operators is presented in Michalos et al. (2018a). The authors here propose a case study where each task is assigned to the resource that best fits the request. Through manual guiding and wearable devices, they establish a safe collaboration without fences, which leads to the conclusion that the resources are oriented toward collaboration more than the competition.

From the presented literature review it is possible to observe how the task allocation problem in HRC systems is an important one, Inkulu et al. (2021), with a great effort by the literature on methods focused on finding the optimal task allocation to improve productivity.

On the other hand, there are different solutions to improve safety but also guarantee productivity, e.g., the introduction of sensors, gesture recognition systems (Liu & Wang, 2018), and collision avoidance systems (Boschetti et al., 2022).

The regulations themselves, (ISO, 2016; UNI, 2011), introduce safety paradigms:

- *Safety-rated monitored stops (SRMS)*: if humans come too closer to the cobot, it will stop.
- *Hand guiding (HG)*: operator guides the cobot, which has to maintain the object in position.
- *Speed Separation Monitoring (SSM)*: based on the distance from the operator, the cobot can change its speed.
- *Power and Force Limiting (PFL)*: cobot's arm will stop if it receives an impact with a preset force value.

Starting from these, an example of how it is possible to improve productivity with SSM is offered by Byner et al. (2019), where a laser scanner is used to monitor the closeness between the operator and the cobot, i.e., if the cobot is close enough to the operator, its speed is reduced to a fixed limit. This guarantees the consistency of the production because the robot does not stop if it is too close to the operator.

Gualtieri et al. (2020) present a framework to consider safety, ergonomics, and efficiency in the design of a collaborative work cell. Safety and ergonomics are guaranteed with the introduction of a motion planning approach that implements smooth trajectories, considering also the posture of the operator to avoid overload and flexion or extension of the human body. Efficiency is ensured by a correct design of the workspace, which is provided by automatic vision systems and the avoidance of manual parts reorientations.

Similarly, (Krüger et al., 2009) propose a survey on how to properly achieve a shared workspace, with remote and haptic interfaces to reach an efficient human-robot collaboration and to realize interactive learning. They also suggest the use of laser scanners, safety sensors, pre- and post-collision systems, and intelligent grippers to satisfy safety requirements. The authors recommend dividing assembly tasks into four steps: separation, transfer, orientation, and positioning of the parts to have an easier and more flexible feeding and so to maintain high productivity.

Galin et al. (2020), state that productivity is directly influenced by safety and by the collaboration between the human operator and the cobot. The latter can be achieved with an intelligent manufacturing process based on the division of tasks between the two resources, obviously taking into account safety measures that can be both hardware or software.

Lastly, (Bettoni et al., 2020) propose an adaptive human-robot collaboration based on a decision-maker implemented by machine learning techniques. Their aim, thanks also to a physiological monitoring system, is to provide relief from both mental and physical workload, which is achieved by changing the support level offered by a collaborative robot. In this way, the operator is less stressed and the total productivity increases by 16%, while the quality issues decrease by 95%.

From the presented literature review, it is possible to observe, to the authors' knowledge, how the task allocation and safety issues have not been merged in the literature. This is because in task allocation problems the cobot is considered intrinsically safe thus, focusing mainly on improving productivity through optimal scheduling and considering issues such as task parallelization. On the other hand, frameworks that improve the safety in HRC systems are mainly focused on safety devices, which reduce the flexibility of the work cell, hence, task allocation is not used to improve the safety in the workspace.

The novelty this article introduces is a task allocation model with a new constraint based on the distance between the operator and the cobot, which considers how the cobot speed changes when the distance between the resources is lower than a safety value. Moreover, this new constraint aims to maximize the time in which the two resources are sufficiently far from each other, making it possible to improve safety since the danger of a collision decreases and also to raise the efficiency since the makespan decreases thanks to the fact that the cobot can keep a higher speed for more time.

Proposed task allocation

Nomenclature

Input variables and parameters

J	Number of tasks
j	Task indexes $j = 1, \dots, J$
K	Number of resources
k	Resource index $k = 1, \dots, K$
t_{jk}	Task time j for resource k [min]
D_{ij}	Pairwise distance matrix between the tasks i, j $[J \text{x} J]$
d_s	Minimum safety distance between the resources [cm]
f_s	Speed decrement/Time increment

Output variables Optimization variables:

x_{jkt}	Assembly line balance decision variable [binary]

Objective function:

ms	Makespan [min]
t^*_{jk}	Task time j for resource k increased by safety constraint [min] (we considered only $k = 2$, i.e., the cobot)

Indexes adopted to evaluate the obtained task allocation

$c_{\%}$	Collaboration index *Other variables used in this work*
t	Temporal instant [s]
T	Temporal horizon [s]
U_k	Set of unfeasible tasks for resource k
C	Intrusion distance, defined by [1] [m]
T_{coll}	Collaboration time [min]
S_h	Minimum separation distance due to operator's reaction time [m]
S_r	Minimum separation distance due to cobot reaction time [m]
S_s	Distance for cobot stop [m]
Z_d	Uncertainty about operator's position [m]
Z_r	Uncertainty about cobot position [m]
T_r	Cobot reaction time [s]
T_s	Cobot stop time [s]
Ts_{max}	Cobot stop time at maximum speed [s]
v_h	Operator speed [m/s]
v_r	Rated cobot speed [m/s]
v_c	Cobot speed [m/s]
v_{rel}	Relative speed [m/s]
v_{max}	Maximum cobot speed [m/s]
ν	Safety coefficient
E	Transferred energy during a contact [J]
β	Reduced mass [kg]
Δ_{ms}	Difference between the makespan without and with the safety constraint [min]
d^*_s	Optimal safety distance [cm]

Problem statement and assumptions

A set of tasks J need to be assigned to K collaborative resources with different characteristics, i.e., collaborative robot and human operator. In particular, we are focusing on an assembly process where both the resources share the workspace at the same time. A possible scenario can be seen in Faccio et al. (2019), where a collaborative assembly between a cobot and a human operator is presented. In this scenario, the two resources need to pick up and assemble five screws; a similar example is advantageous since it allows us to avoid considering the effects of precedence between the tasks on the assembly process. Some of the hypotheses that characterize the model are retained in the proposed work, which are:

- mass production of one homogeneous product by performing J operations of a given product process: single-model line hypothesis. The single-model representation is a Virtual Average Model (VAM), which characteristics are calculated from the different product variants, so, eventually, it can be easily extended to a mixed-model;
- deterministic and integer operations times; this is done in order to avoid the influence of stochastic effects;
- each task is performed by only one resource.

The proposed method aims to investigate the effects of a distance constraint on a task allocation for human-robot collaborative systems. Indeed, due to safety reasons, the collaborative robot(s) speed should be reduced when the distance between the resources is lower than a certain threshold.

Since this scenario leads to a decrease in the cobots performance, it is reasonable to search for a task allocation process that considers the distance between the resources during the process in order to maximize the system performance. Hence, additional hypothesis have been introduced to better focus on the aim of the proposed work and highlight the influence of the distance:

- the assembly line is composed by a single workstation consisting of one human operator and one cobot, i.e., $K = 2$;
- the considered collaborative resources share workplace and task time;
- the distance between the resources is measured as the distance between their current tasks, thus, considering the space occupied by the robot and the human operator at micro-level (Faccio et al., 2020) during the task completion.

Model description

The objective function of the proposed model is the minimization of the makespan ms in a collaborative workstation composed by a cobot and a human operator. This means that is a non linear model for a task allocation with a single objective minimization.

$$min\ ms = \sum_{t=0}^{T}\sum_{k=1}^{K}(t + t_{jk})x_{jkt} \tag{1}$$

Subject to:

$$\sum_{t=0}^{T}\sum_{k=1}^{K} x_{jkt} = 1 \quad \forall j \tag{2}$$

$$t^{*}_{jk} = t_{jk} \cdot (1 + f_s \cdot d_{ij} \cdot \sum_{t} x_{jkt}) \quad \forall j, i \quad k = 2 \tag{3}$$

$$x_{jkt} = 0 \quad \forall j \in U_k \tag{4}$$

$$x_{jkt} \in \{0, 1\} \quad \forall j, k, t \tag{5}$$

where x_{jkt} is the optimization variable:

$$x_{jkt} = \begin{cases} 1 \text{ if the task } j \text{ is performed by the resource } k \\ \quad \text{at the time } t \\ 0 \text{ otherwise} \end{cases} \tag{6}$$

The time dimension t is considered as done by Boschetti et al. (2021b), to evaluate ms, Eq. (1). The guarantee that each task is executed by a single resource is given by the Eq. (2). The distance constraint introduced in this work is defined in Eq. (3), considering the cobot as the resource $k = 2$ and where d_{ij} is defined given the values of D_{ij} as:

$$d_{ij} = \begin{cases} 1 \text{ if the distance between task } i \text{ and } j \\ \quad \text{is less than } d_s \\ 0 \text{ otherwise} \end{cases} \quad \forall i, j \tag{7}$$

where d_s is the minimum safety distance between the resources, defined on the basis of the reaction and stopping distance of both resources, as defined by the standards and better described in the following section. The distance constraint allows us to relate the distance between the resources with the performance of the system. Indeed, as the distance between the resources increases, the robot is capable of moving at a higher speed, effectively reducing the task completion time by reducing the picking and assembly time. Hence, in this way it is possible to minimize the makespan while also increasing the safety distance between the resources.

Lastly, the technological constraints are introduced in Eq. (4), which limits the possible solutions by considering that the resource k may not be able to carry out some tasks j. These tasks are included in the set U_k, whose elements u_{jk} are the tasks j that cannot be executed by resource k, differentiating between different types of resources. This is characteristic of collaborative assembly systems, where the layout is characterized by resources that are not homogeneous. However, by considering comparable task times and capabilities, it is possible to apply the model to traditional and homogeneous scenarios.

Safety distance and speed increment

In this paper, collaborative robots with *speed and separation monitoring* are considered, as described by the standards. The regulation ISO/TS 15066:2016 (2016) provides the definition of three zones where the interaction between the operator and the cobot can take place:

- great distance between cobot arm and operator where they can both work at their nominal speed because contact risk is low;
- medium distance where cobot has to reduce its speed because the risk increases with the distance reduction;
- small distance where cobot is expected to stop due to a high probability of impact.

Cobot speed is then limited taking into account its overall reaction times, thus establishing a stopping distance as:

$$d_s = S_h + S_r + S_s + C + Z_d + Z_r \tag{8}$$

where S_h, S_r, and S_s are the safety distances due to operator reaction time, robot reaction time, and the robot stop time, respectively; C, Z_d, and Z_r are terms related to uncertainties or factors that are not easily predictable. Moreover, the parameter C refers to robots or machines that are not collaborative, so it can not be correct to include it in the evaluation of ds, Marvel and Norcross (2017), since cobots should work without any additional device. Therefore, a safety parameter ν is defined, which increases the values of distances to enclose these factors. This results in:

$$d_s = (S_s + S_r + S_h) \cdot \nu \tag{9}$$

These values are used to evaluate a safety distance, below which a cobot speed reduction is required. With the speed reduction, also the stop distance will be smaller and so the cobot is in a safe condition, improving the safety of the system.

Regarding the separation distance due to operator reaction time, the regulation defines it as:

$$S_h = \int_t^{t+T_r+T_s} v_h(x)dx = v_h \cdot (T_r + T_s) \tag{10}$$

where T_r is the cobot reaction time and T_s is the cobot stop time.
Considering the established override speed v_{max} and the rated speed v_r, it is easy to determine:

$$T_s = T_{s,max} \cdot \frac{v_r}{v_{max}} \tag{11}$$

where $T_{S,max}$ is the stop time at maximum speed, usually identified in the robot data-sheet. From this, it is possible to calculate:

$$\begin{aligned} S_s &= v_r \cdot T_s \\ S_r &= v_r \cdot T_r \end{aligned} \tag{12}$$

Similarly, it is possible to evaluate the maximum collaborative speed by considering the transferred energy during contact as described by the regulation, and which is defined as:

$$E = \frac{1}{2}\beta v_{rel}^2 \tag{13}$$

where β is the reduced mass of the two-body systems, while v_{rel} is the relative speed between the operator and the cobot.

From that it is possible to derive the maximum collaborative speed of the cobot as:

$$v_c = v_{rel} - v_h \tag{14}$$

Model application and comparison

To apply the model, proper assumptions have been made, considering both the product and process characteristics.

Regarding the product characteristics, we defined the following assumptions:

- components' size is defined on the basis of small parts that can be manipulated with hands;
- tasks layout are typically arranged in a grid to have a uniform distribution and to avoid the introduction of other variables;
- the number of tasks is defined in order to be compatible with the proposed layout and safety distance.

Regarding the process characteristics, we defined:

- tasks precedence are not considered (completely parallel tasks). This is done in order not to introduce other variables, since the level of tasks parallelization influences the performance (Boschetti et al., 2021b);
- the operator's hands speed is calculated from DeGoede et al. (2001);
- time for assembly tasks for both resources is obtained from Faccio et al. (2019);
- cobot speed decrement f_s, and so cobot time increment, has been defined considering the ratio between $v_r = 0.25\ m/s$ (usually considered as a collaborative speed (Faccio et al., 2020)) and the maximum collaborative speed $v_c = 0.32\ m/s$ obtained from Eqs. (13) and (14), with $E = 0.49\ J$, $\beta = 0.6\ kg$ and $v_h = 1.6\ m/s$ as in [2];
- the minimum safety distance between the resource is set to $d_s = 80\ cm$ because of Eq. (12), using $T_{s,max} = 1.2\ s$, typical (worst) stop time for a collaborative robot, $T_r = 0.1\ s$, a typical refresh rate of a vision system with a safety parameter $\nu = 1.1$. This value is acceptable since it is comparable to the reach of a commercial collaborative cobot in normal conditions of use.

These values are reported in Table 2, for better understanding.

The following output parameters are considered:

- makespan ms, i.e. the total time required to complete all the tasks of the assembly process;
- collaboration index $c\%$, whose formulation has been adopted from Boschetti et al. (2021b), i.e. the shared time T_{coll} normalized with respect to the total assembly time ms:

$$c_{\%} = \frac{T_{coll}}{ms} \quad [0, 1] \tag{15}$$

Table 2 Process characteristics values for the case study

Parameter	Values
v_r	0.25 m/s
v_c	0.3 m/s
v_h	1.6 m/s
$T_{s,max}$	1.2 s
T_r	0.1 s
ν	1.1
d_s	80 cm
f_s	28%
E	0.49 J
β	0.6 kg

Comparison between the models without and with safety constraint: a case study

This section compares the models without and with the safety constraint. The first one does not change cobot speed according to the operator position, hence, the cobot speed is set to a minimum collaborative value to ensure the operator safety. On the other hand, the second one introduces the safety constraint based on the minimum safety distance (Eq. 12), so the cobot can work faster when it is far from the operator. Both models do not consider the precedence constraint.

Starting from the parameters previously described, $J = 12$ tasks are considered, with a grid layout and for each one the time required by the operator and the time required by the robot are considered. The values are shown in Table 3.

To solve the problem, we adopted the Solving Constraint Integer Programs (SCIP) framework in the MATLAB (Mathworks) environment. We adopted this framework since it is one of the fastest non-commercial solvers for mixed integer programming (MIP); indeed, it was possible to solve each simulation in about 2 minutes.

The task allocations obtained from both models are shown in Figs. 1 and 2.

Figures 1a and 2a represent tasks assigned to the operator, “OP” with an orange bar and those assigned to the robot with a green bar, “R” along with the amount of collaboration “C” represented by a blue bar. In both figures, red areas represent the tasks that do not allow keeping a safe distance as they are performed simultaneously despite their proximity. Figures 1b and 2b represent the positions of the tasks the resources have to perform, in particular, orange squares are operator’s tasks while blue circles are cobot tasks. In both the figures, the tasks performed simultaneously that do not allow to keep a safe distance, are circled in red.

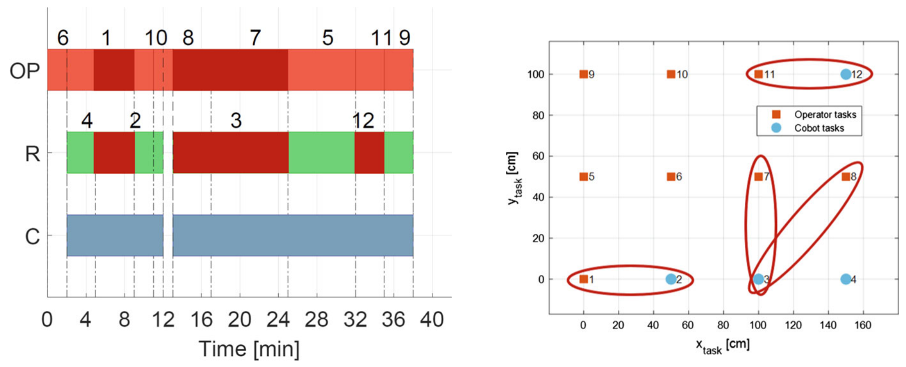
It can be seen that the makespan ms is smaller with the new model by about 15%, which is achieved by rising the cobot speed when the safety distance is ensured, in accordance to the standards. On the other hand, the traditional model needs to reduce the cobot speed to ensure the safety requirements at all times. In this way, Figures 1, 2 and 3 show that the time in which the resources are below the safety distance is decreased, i.e, the cobot works at the faster speed for more time, so the total time required to do all the tasks is lower. It is important to specify that this method does not reduce the number of occurrences of two near tasks but the total time where the distance between the resources is lower than d_s. From Fig. 3, in fact, it is possible to see that with the safety constraint the time in which the two resources are below the safety distance, decreases from 19 min to 6 min, that means that it goes from being the 50% of the makespan, to being the 18% of it, i.e. a reduction of 32% was obtained.

In order to generalize this case, several analysis have been carried out where the times of the tasks have been varied, both for the operator and for the robot. The results are shown in Fig. 4, where on the x−axis is reported an average time of the tasks, while on the y−axis the makespan ms, showing a decrease in the makespan regardless of the mean task times.

Table 3 Tasks time and position

Task	Operator task time [min]	Robot task time [min]	x_{task} [cm]	y_{task} [cm]
1	7	10	0	0
2	8	5	50	0
3	8	9	100	0
4	4	2	150	0
5	7	5	0	50
6	2	10	50	50
7	8	8	100	50
8	6	10	150	50
9	3	7	0	100
10	2	2	50	100
11	3	9	100	100
12	9	10	150	100

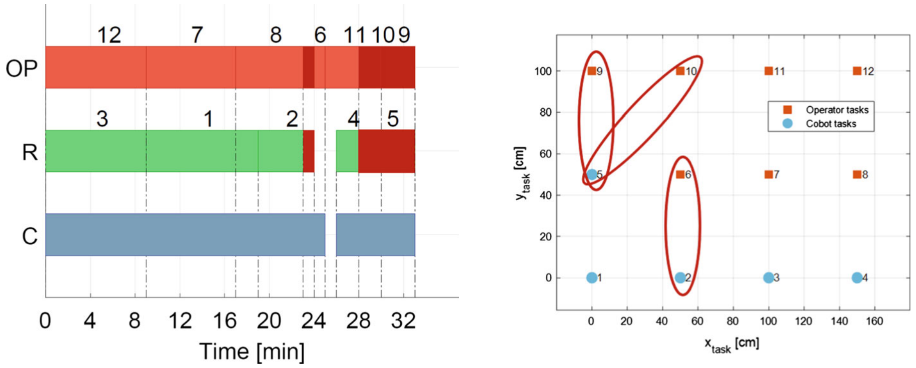
Fig. 1 Task allocation without safety constraint



(a) Task allocation and Collaboration obtained without safety constraint

(b) Tasks assigned to the operator and to the cobot

Fig. 2 Task allocation with safety constraint



(a) Task allocation and Collaboration obtained with safety constraint

(b) Tasks assigned to the operator and to the cobot

Influence of the number of tasks

In this section, the influence of the number of tasks on the makespan *ms* and on the collaboration index $c_{\%}$ is investigated. The tests are done keeping all the other parameters constant, i.e., $d_s = 80\ cm$, $f_s = 0.28$, whereas only the number of tasks has changed from $J = 9$ to $J = 20$. A number of tasks greater than this value was not considered due to the computational time with the solver used. Indeed, for $J < 12$ the average computation time was less than 60 s, for $13 < J < 17$ was about 120 s, and for J defined between 18 and 20 it was necessary to increase the time limit to 3 minutes. However, for $J > 20$ it is suggested to solve the model through the use of a heuristic algorithm since the solver was not able to find any solution in a limited amount of time.

The results are shown in Fig. 5 for makespan performance and in Fig. 6 for the percentage of collaboration.

Figure 5 shows a comparison between the two models, with a significant decrease in *ms* with the new one. From the obtained data, an interpolation is made: the data has been fitted with different curves, like power with two terms, spline, or sum of sine but the best one is 2-nd degree polynomial form, with a bisquare robustness. These curves show that as the number of tasks increases the difference in makespan between the two models is increasing, from 3% with $J = 9$ to almost 18% with $J = 20$.

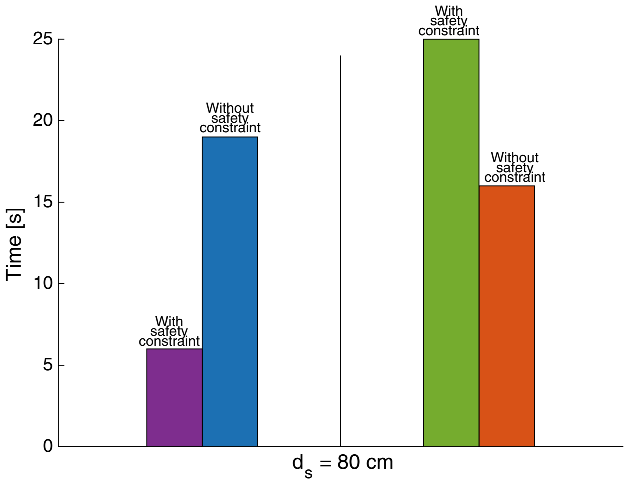


Fig. 3 Time where the resources are below and over the safety distance

Figure 6 shows the influence of the number of tasks on $c\%$, where the proposed method provides better result, improving the collaboration between the two resources. Similarly

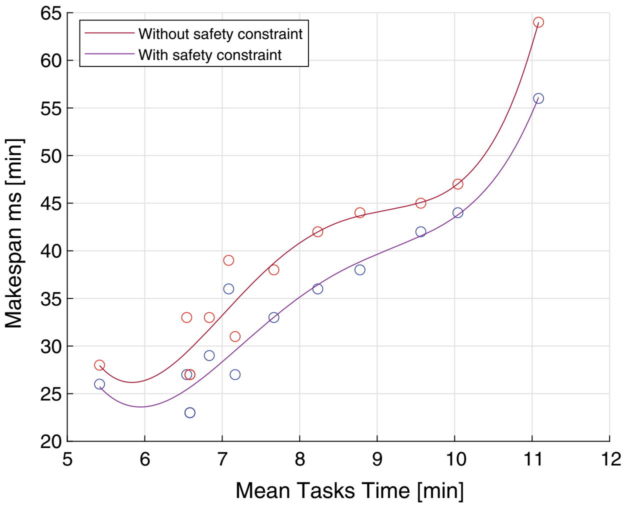


Fig. 4 Makespan performance as a function of average tasks time

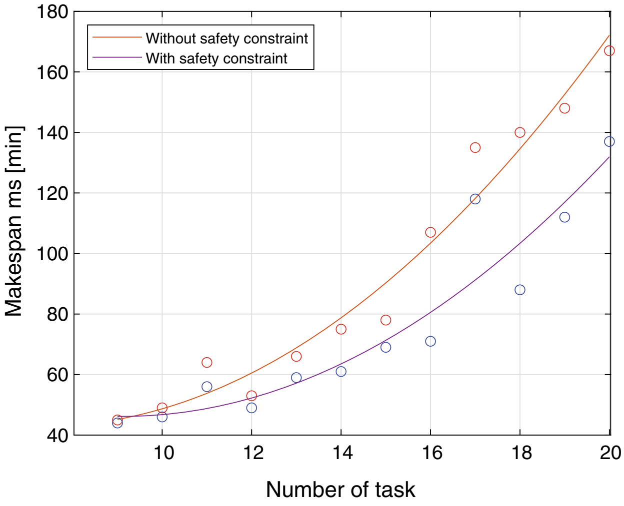


Fig. 5 Effect of the *number of tasks* on *makespan ms*

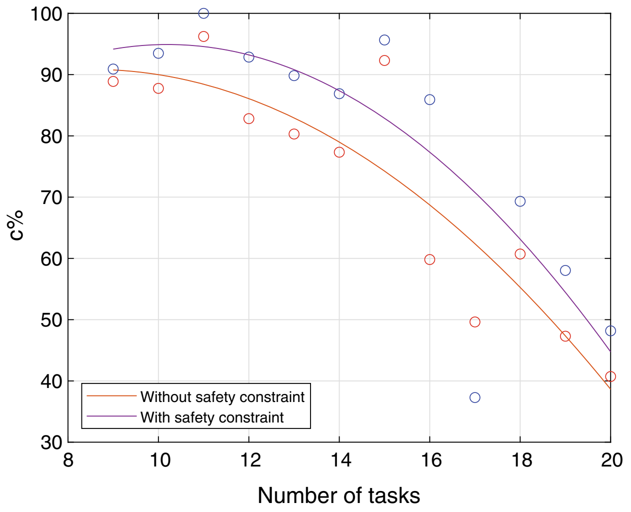


Fig. 6 Effect of the *number of tasks* on the *collaboration index c%*

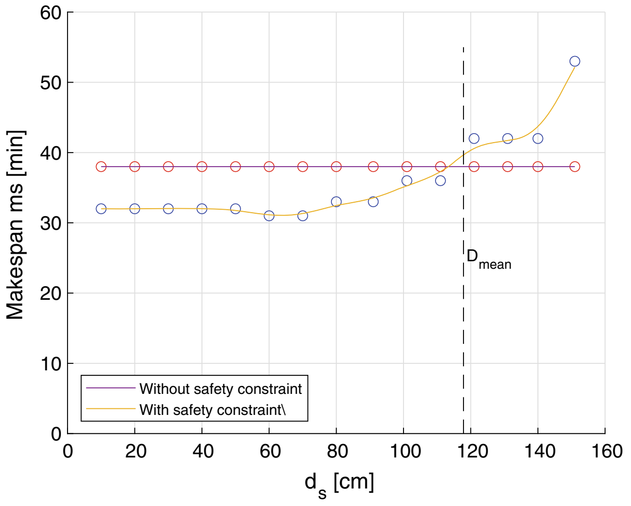


Fig. 7 Effect of the *minimum safety distance* d_s on *makespan ms*

to the ms, also here the best fitting model is a 2-nd degree polynomial form with bisquare robustness.

Influence of the minimum safety distance

Similarly, the safety distance d_s influences ms. Hence, the model is tested changing the safety distance from $d_s = 10\,cm$ (totally unsafe) to $d_s = 150\ cm$ (includes all the tasks positions). In this analysis only the safety distance has been varied (and so the robot speed since they are related as described above), while the other parameters are kept constant, using the ones defined in Table 2 and in Table 3

As it is possible to see in Fig. 7, the method with the safety constraint ensures a smaller makespan, about 20% less, than the other. This happens as long as the safety distance is less than the distance between the workstations of the tasks, evaluated through the average distance D_{mean} between them. As we can see from the interpolated data, obtained with smoothing spline, when the safety distance is above D_{mean}, the constraint on the safety distance no longer holds and the model tends not to assign tasks to one resource in order to respect it, effectively increasing the makespan. In addition, it is possible to see that ms is strongly related to ds, since less this value is, smaller is ms because also the cobot speed is affected by d_s. Increasingly reducing d_s, an higher speed can be kept for more time.

Starting from this, it is possible to analyse which is the optimal safety distance, in order to have the maximum decrement of the makespan. To do that, the difference Δ_{ms} between ms without safety constraint and ms with the safety constraint, that is here called ms_s, can be investigated, Eq. 16

$$\Delta_{ms} = ms - ms_s \tag{16}$$

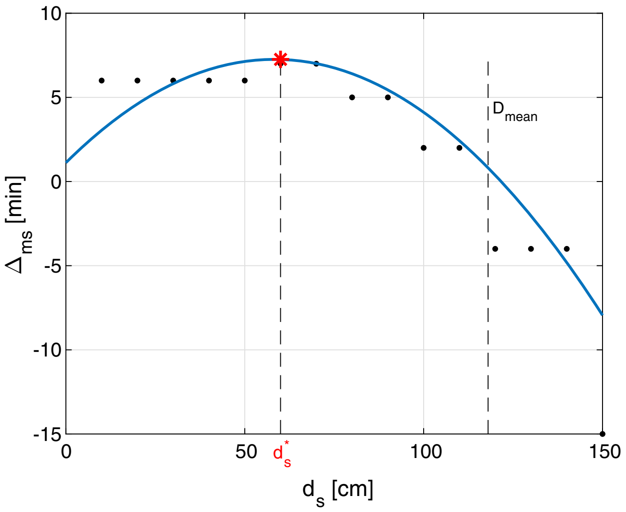


Fig. 8 Δ_{ms} trend as function of d_s, with d_s^*

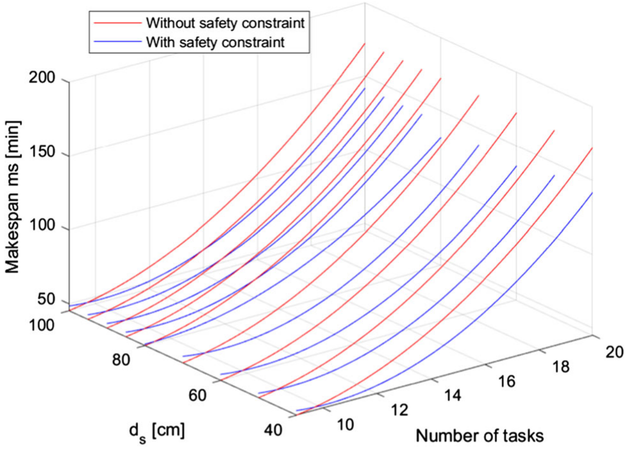


Fig. 9 Effect of the *number of tasks* and d_s on *ms*

Its trend is shown in Fig. 8 where, from the interpolated data (with $R^2 \simeq 0.91$), it is possible to see that the maximum difference is with $d_s^* \simeq 60\ cm$. This can have high practical implication because d_s is function of the cobot speed, meaning that it is possible to choose the optimal speed to have the maximum decrement of the makespan.

Overall results

The overall result is presented in Fig. 9, where the influence of both the number of tasks and the minimum safety distance d_s is shown. In particular x-axis represents the number of tasks, y-axis represents d_s and z-axis represents ms for both models. This figure is obtained with the same 2-nd degree polynomial interpolation as before.

From this figure it is possible to derive the Fig. 10, where the axis and the interpolation are the same as Fig. 9. It represents the difference between ms obtained without the safety constraint minus ms obtained with it.

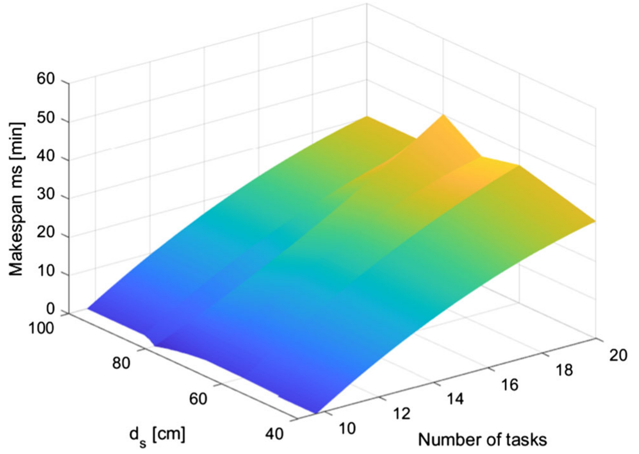


Fig. 10 Surface plot from data interpolation of the difference between ms obtained without and with safety constraint

The results show that these dimensions have to be taken into account in the design of the task allocation in order to improve the system performance. Moreover, since it is possible to observe a steeper slope along the number of tasks, it is possible to presume a greater influence of this parameter on the total time required.

Conclusion

Task allocation and safety problems are two of the biggest issues in collaborative robotics since there are typically two resources, i.e., the operator and the cobot, that have to interact to achieve high performance but at the same time, the workspace has to ensure a proper level of safety.

However, the models proposed so far, have focused more on one of the aforementioned issues and, consequently, highlighted a trade-off between productivity and safety, unless introducing additional devices.

As a result of this gap, a new model for collaborative assembly systems with a safety constraint is proposed, aiming to minimize the total time required to complete all the tasks. The novelty of the proposed model is that it takes into account the distance between the resources and it modulates the speed of the cobot accordingly. Thus, it is allowed to keep higher speeds, although collaborative, for longer, since this model, thanks to the safety constraint, maximizes the time in which the resources are sufficiently far, i.e. above the safety distance. This has led to a decrease in the makespan, up to almost 18%, since the task allocation developed reduces the time during which the two resources are below the safe distance. Moreover, evaluating the effect that different safety distances have on the makespan, it is possible to identify the one that allows to have the maximum decrement of ms with respect to the value obtained without the safety constraint and so to choose optimal cobot speed.

The proposed method did not consider stochastic times and precedence in order not to limit the space of solutions but they will be developed in future investigations. Hence, in the next development of this model, time variability will be introduced. This is due to the fact that operators can spend a variable amount of time performing tasks, thus, a certain degree of uncertainty may arise. Another future search can involve the introduction of a precedence diagram for all those products that have a specific assembly process and, finally, a dynamic task allocation with real-time control of the cobot will be developed to adapt its position to the operator's one and to ensure maximum productivity at the same time.

Funding Open access funding provided by Università degli Studi di Padova within the CRUI-CARE Agreement.

Data availability The authors confirm that the data supporting the findings of this study are available within the article or its supplementary materials.

References

Azzi, A., Battini, D., Faccio, M., Persona, A.: Sequencing procedure for balancing the workloads variations in case of mixed model assembly system with multiple secondary feeder lines. International Journal of Production Research 50(21), 6081–6098 (2012).

Bautista, J., & Pereira, J. (2007). Ant algorithms for a time and space constrained assembly line balancing problem. *European journal of operational research,177*(3), 2016–2032.

Bettoni, A., Montini, E., Righi, M., Villani, V., Tsvetanov, R., Borgia, S., Secchi, C., Carpanzano, E.: Mutualistic and adaptive human-machine collaboration based on machine learning in an injection moulding manufacturing line. Procedia CIRP 93, 395–400 (2020).

Bogner, K., Pferschy, U., Unterberger, R., & Zeiner, H. (2018). Optimised scheduling in human-robot collaboration-a use case in the assembly of printed circuit boards. *International Journal of Production Research,56*(16), 5522–5540.

Boschetti, G., Bottin, M., Faccio, M., Maretto, L., Minto, R.: The influence of collision avoidance strategies on human-robot collaborative systems. IFAC-PapersOnLine 55(2), 301–306 (2022).

Boschetti, G., Bottin, M., Faccio, M., & Minto, R. (2021). Multi-robot multi-operator collaborative assembly systems: a performance evaluation model. *Journal of Intelligent Manufacturing,32*(5), 1455–1470.

Boschetti, G., Faccio, M., Milanese, M., & Minto, R. (2021). C-alb (collaborative assembly line balancing): a new approach in cobot solutions. *The International Journal of Advanced Manufacturing Technology,116*(9), 3027–3042.

Browne, J., Dubois, D., Rathmill, K., Sethi, S. P., Stecke, K. E., et al. (1984). Classification of flexible manufacturing systems. *The FMS magazine,2*(2), 114–117.

Byner, C., Matthias, B., Ding, H.: Dynamic speed and separation monitoring for collaborative robot applications-concepts and performance. Robotics and Computer-Integrated Manufacturing 58, 239–252 (2019).

Chen, F., Sekiyama, K., Cannella, F., Fukuda, T.: Optimal subtask allocation for human and robot collaboration within hybrid assembly system. IEEE Transactions on Automation Science and Engineering 11(4), 1065–1075 (2013).

Da Silveira, G., Borenstein, D., & Fogliatto, F. S. (2001). Mass customization: Literature review and research directions. *International journal of production economics,72*(1), 1–13.

DeGoede, K. M., Ashton-Miller, J. A., Liao, J. M., & Alexander, N. B. (2001). How quickly can healthy adults move their hands to intercept an approaching object? age and gender effects. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences,56*(9), M584–M588.

Edmondson, N. & Redford, A. (2002). Generic flexible assembly system design. Assembly automation

El Zaatari, S., Marei, M., Li, W., Usman, Z.: Cobot programming for collaborative industrial tasks: An overview. Robotics and Autonomous Systems 116, 162–180 (2019).

Faccio, M., Bottin, M., & Rosati, G. (2019). Collaborative and traditional robotic assembly: a comparison model. *The International Journal of Advanced Manufacturing Technology,102*(5), 1355–1372.

Faccio, M., Minto, R., Rosati, G., & Bottin, M. (2020). The influence of the product characteristics on human-robot collaboration: a model for the performance of collaborative robotic assembly. *The International Journal of Advanced Manufacturing Technology,106*(5), 2317–2331.

Fechter, M., Seeber, C., Chen, S.: Integrated process planning and resource allocation for collaborative robot workplace design. Procedia CIRP 72, 39–44 (2018).

Galin, R., Meshcheryakov, R., Kamesheva, S. & Samoshina, A. (2020). Cobots and the benefits of their implementation in intelligent manufacturing. In: IOP Conference Series: Materials Science and Engineering, vol. 862, p. 032075. IOP Publishing

Gerbers, R., Wegener, K., Dietrich, F. & Dröder, K. (2018). Safe, flexible and productive human-robot-collaboration for disassembly of lithium-ion batteries. In: Recycling of Lithium-Ion Batteries, pp. 99–126. Springer

Gualtieri, L., Rauch, E., Vidoni, R., & Matt, D. T. (2020). Safety, ergonomics and efficiency in human-robot collaborative assembly: design guidelines and requirements. *Procedia CIRP,91*, 367–372.

Inkulu, A.K., Bahubalendruni, M.R., Dara, A. & SankaranarayanaSamy, K. (2021). Challenges and opportunities in human robot collaboration context of industry 4.0-a state of the art review. Industrial Robot: The international journal of robotics research and application

International Federation of Robotics. (2020). In: IFR Press Conference. https://ifr.org/downloads/press2018/Presentation_WR_2020.pdf

ISO/TS 15066:2016. (2016). Robots and Robotic Devices' Collaborative Robots. International Organization for Standardization

Johannsmeier, L., & Haddadin, S. (2016). A hierarchical human-robot interaction-planning framework for task allocation in collaborative industrial assembly processes. *IEEE Robotics and Automation Letters,2*(1), 41–48.

Klumpp, M., Hesenius, M., Meyer, O., Ruiner, C., Gruhn, V.: Production logistics and human-computer interaction-state-of-the-art, challenges and requirements for the future. The International Journal of Advanced Manufacturing Technology 105(9), 3691–3709 (2019).

Krüger, J., Lien, T. K., & Verl, A. (2009). Cooperation of human and machines in assembly lines. *CIRP annals,58*(2), 628–646.

Liu, H., Wang, L.: Gesture recognition for human-robot collaboration: A review. International Journal of Industrial Ergonomics 68, 355–367 (2018).

Malik, A.A. & Bilberg, A. (2019). Complexity-based task allocation in human-robot collaborative assembly. Industrial Robot: the International Journal of Robotics Research and Application

Marvel, J. A., & Norcross, R. (2017). Implementing speed and separation monitoring in collaborative robot workcells. *Robotics and computer-integrated manufacturing,44*, 144–155.

Michalos, G., Kousi, N., Karagiannis, P., Gkournelos, C., Dimoulas, K., Koukas, S., Mparis, K., Papavasileiou, A., & Makris, S. (2018). Seamless human robot collaborative assembly-an automotive case study. *Mechatronics,55*, 194–211.

Michalos, G., Spiliotopoulos, J., Makris, S., & Chryssolouris, G. (2018). A method for planning human robot shared tasks. *CIRP journal of manufacturing science and technology,22*, 76–90.

Müller, R., Vette, M., Geenen, A.: Skill-based dynamic task allocation in human-robot-cooperation with the example of welding application. Procedia Manufacturing 11, 13–21 (2017).

Nikolakis, N., Kousi, N., Michalos, G., Makris, S.: Dynamic scheduling of shared human-robot manufacturing operations. Procedia CIRP 72, 9–14 (2018).

Palleschi, A., Hamad, M., Abdolshah, S., Garabini, M., Haddadin, S., Pallottino, L.: Fast and safe trajectory planning: Solving the cobot performance/safety trade-off in human-robot shared environments. IEEE Robotics and Automation Letters 6(3), 5445–5452 (2021). DOI: 10.1109/LRA.2021.3076968.

Pearce, M., Mutlu, B., Shah, J., & Radwin, R. (2018). Optimizing makespan and ergonomics in integrating collaborative robots into manufacturing processes. *IEEE transactions on automation science and engineering,15*(4), 1772–1784.

Ranz, F., Hummel, V., & Sihn, W. (2017). Capability-based task allocation in human-robot collaboration. *Procedia Manufacturing, 9*, 182–189.

Simões, A.C., Soares, A.L., Barros, A.C.: Factors influencing the intention of managers to adopt collaborative robots (cobots) in manufacturing organizations. Journal of Engineering and Technology Management 57, 101574 (2020).

Surdilovic, D., Schreck, G., & Schmidt, U. (2010). Development of collaborative robots (cobots) for flexible human-integrated assembly automation. In: ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics), pp. 1–8. VDE

Takata, S., & Hirano, T. (2011). Human and robot allocation method for hybrid assembly systems. *CIRP annals,60*(1), 9–12.

Tan, J.T.C., Duan, F., Kato, R., Arai, T. & Hall, E. (2010). Collaboration planning by task analysis in human-robot collaborative manufacturing system. INTECH Open Access Publisher

Tsarouchi, P., Matthaiakis, A.S., Makris, S., Chryssolouris, G.: On a human-robot collaboration in an assembly cell. International Journal of Computer Integrated Manufacturing 30(6), 580–589 (2017).

UNI EN ISO 13855: 2010. (2010), Safety of machinery. UNINFO Standards for the Information Technology and related application

UNI EN ISO 10218-1: 2012. (2012). Robots and robotic devices-safety requirements for industrial robots-part 1: Robots. UNINFO Standards for the Information Technology and related applications

UNI EN ISO 10218-2: 2011. (2011). Robots and robotic devices-safety requirements for industrial robots-part 2: Robot systems and integration. UNINFO Standards for the Information Technology and related applications

UNI EN ISO 13855: 2010. (2010). Safety of machinery. UNINFO Standards for the Information Technology and related applications

UNI EN ISO 13855:2010. (2010). Safety of machinery-positioning of safeguards with respect to the approach speeds of parts of the human body. UNINFO Standards for the Information Technology and related applications

Vicentini, F.: Terminology in safety of collaborative robotics. Robotics and Computer-Integrated Manufacturing 63, 101921 (2020).

Villani, V., Pini, F., Leali, F., Secchi, C.: Survey on human-robot collaboration in industrial settings: Safety, intuitive interfaces and applications. Mechatronics 55, 248–266 (2018).