The International Journal of Advanced Manufacturing Technology (2024) 134:3201–3216
https://doi.org/10.1007/s00170-024-14302-3

ORIGINAL ARTICLE

Achieving productivity and operator well-being: a dynamic task allocation strategy for collaborative assembly systems in Industry 5.0

Martina Calzavara[1] · Maurizio Faccio[1] · Irene Granata[1] · Alberto Trevisani[1]

Received: 23 May 2024 / Accepted: 17 August 2024 / Published online: 29 August 2024

Abstract
Collaborative robots, or cobots, offer a unique combination of productivity and flexibility that has led to significant growth in adoption over the past decade. Moreover, recently, there has been a shift towards a human-centered design of the workspace, known as one of the drivers of Industry 5.0, which prioritizes the well-being of operators. To achieve this, various human factors such as ergonomics, mental workload, personal skills, and capabilities need to be considered in the workspace design, and their impact on system productivity must be evaluated. The integration of a human and a cobot in the same workplace can affect the performance of the human operator, as the perception of the cobot can impact their work. This highlights the importance of taking human factors into account, as a lack of consideration in these aspects has contributed to the failure of many implementations. To link the objectives of productivity, flexibility, and human factors consideration, a dynamic real-time multi-objective task allocation strategy for collaborative assembly systems is developed. This approach considers the different characteristics of the resources and optimizes for two objectives, makespan, and energy expenditure of the operator. By using this approach, it is possible to modify the behavior of the cobot by reallocating tasks between the two resources based on the operator's current needs. In other words, if the operator appears too stressed due to time constraints or their energy rate level is too high, some of their assigned tasks can be transferred to the cobot. This helps to maintain a balanced system while reducing the operator's stress.

Keywords Human-centered design · Multi-objective task allocation · Cobot systems · Human factors · Human-robot collaboration

1 Introduction

Collaborative robots, or cobots, have gained substantial traction in the last decade due to their distinct advantages, combining the productivity of automatic machines with the flexibility of manual systems, particularly in assembly setups [1]. Notably, their adaptability to new designs and production volume changes sets them apart from traditional robots specialized for specific product variants [2]. Cobots can work alongside human operators without safety fences, enhancing production efficiency and negating the need for additional safety measures [3].

The contemporary shift towards a human-centered design in workspaces, prioritizing operator well-being, underscores the importance of considering human factors such as ergonomics, mental workload, skills, and capabilities [4, 5]. The integration of human operators with cobots in the same workplace introduces considerations for the operator's performance and perception of the cobot, emphasizing the need to account for human factors to avoid implementation failures [6, 7].

Connecting productivity, flexibility, and human factors, a task allocation strategy for collaborative assembly systems becomes crucial [8]. This strategy, factoring in resource characteristics, optimizes for varied objectives, ensuring efficient task assignment to maximize productivity and resource utilization. A well-designed task allocation strategy fosters improved collaboration between human and robotic resources, creating a harmonious work environment and enhancing overall system performance.

To achieve this, dynamic task allocation in collaborative assembly systems considers resource characteristics, optimizing for makespan and operator energy expenditure

✉ Irene Granata
irene.granata@phd.unipd.it

[1] Department of Management and Engineering, University of Padua, Stradella San Nicola 3, Vicenza 36100, Italy

to align with human-centered workplace goals [9]. While traditional offline task allocation may offer optimal global solutions, it contradicts Industry 5.0's emphasis on flexibility, necessitating a shift towards online and dynamic task allocation to accommodate evolving operator needs [10]. The proposed work combines static and dynamic allocation, digitally capturing the human operator's position, tasks execution times and updating the energy expenditure in real-time. This leads to the digitalization of the human operator which is a more effective solution to improve the collaboration, rather than estimating his/her position inside the work area [11, 12], and allows to reassign the tasks among the resources considering cobot availability and operator's needs variations.

This paper contributes by reviewing the state of the art in Sect. 2, introducing the optimization model for dynamic task allocation in Sect. 3, detailing the developed architecture setup in Sect. 4, and presenting extended testing results in Sect. 5, ultimately concluding with insights into the overall work in Sect. 6.

2 Literature review

The task allocation problem in collaborative systems, which involves the interaction between collaborative robots (cobots) and human operators, has become increasingly significant due to the complex and dynamic nature of these environments. In dynamic task allocation, tasks are continuously reassigned based on real-time capabilities and evolving needs, striving to maintain balance and optimize system performance. This approach contrasts with adaptive task sharing, which generally involves minimal adjustments during execution and depends more on the immediate capabilities and decisions of human operators [13].

Early research in dynamic task allocation includes Chen et al. [14], who developed a genetic algorithm for real-time subtask allocation aimed at balancing assembly time and cost. Their approach provided a foundational framework for addressing the complexities of dynamic task allocation. Pupa et al. [15] proposed a dynamic scheduling system that considered resource requests but lacked real-time monitoring of objective function values, limiting its effectiveness in rapidly changing environments. Liu et al. [16] highlighted the superiority of adaptive task allocation over static strategies, focusing on the importance of minimizing communication overhead and idle time. In the realm of Industry 5.0, recent advancements have integrated operator well-being into the task allocation process, with Messeri et al. [17] using deep neural networks to predict operator fatigue and Merlo et al. [18] proposing an ergonomic assessment index for real-time evaluations, reflecting a broader emphasis on human factors in collaborative environments.

In the industrial context, the assembly line balancing problem (ALBP) is a central challenge, involving the allocation of tasks to workstations to optimize productivity. The Simple Assembly Line Balancing Problem (SALBP) represents a basic form of this problem, characterized by simplifying assumptions that have been relaxed over time to accommodate various problem variants. A significant challenge within ALBP is the binary nature of task-station assignments. Literature on dynamic balancing and work-sharing has addressed this issue by allowing tasks to be performed at multiple stations, which can lead to increased throughput or reduced cycle times [19].

Fractional task allocations, where tasks are distributed across parallel stations, offer an alternative to traditional binary assignments. While fractional allocations can enhance throughput, they require considerable changes to the layout and increased worker training costs [20]. Existing literature does not adequately address the costs and trade-offs associated with fractional allocations, and task-sharing as a specific variant of ALBP remains underexplored. This paper introduces the Fractional Allocation Assembly Line Balancing Problem (FA-ALBP) and examines how internal storage costs relate to fractional allocations, providing a deeper understanding of the trade-offs involved [21].

Furthermore, balancing tasks to evenly distribute workloads among workstations is crucial in assembly lines. Static line balancing, which involves predetermined task allocations, contrasts with dynamic line balancing (DLB), which allows for flexible task allocation to improve efficiency. Work-sharing strategies, such as floating-worker lines (FRL) and floating-work lines (FKL), introduce flexibility by enabling task sharing or the movement of workers between stations. These strategies can improve throughput but require significant changes in layout and training practices. This paper proposes a mathematical model for work-sharing in DLB and compares the performance of FRL and FKL strategies [22].

Recent research has also highlighted the merits of human-robot cooperative assembly applications. The synergy between the precision, repeatability, and strength of robots and the intelligence and flexibility of humans offers significant advantages, particularly in small-scale production settings where leanness, re-configurability, and adaptability are crucial [23]. New projects and products have emerged to exploit the potential of these hybrid systems, and ongoing research addresses newly arising issues such as safety [24].

Current industrial deployments typically separate human and robot work areas to ensure operator safety, which often leads to suboptimal accommodation of both production entities. Robotic cells are designed without considering ergonomic positioning, while human workplaces fail to address robotic reachability constraints. To address these challenges, systematic design and deployment tools are needed for human-robot task sharing applications. These tools should effectively allocate production tasks based on the intrinsic characteristics of humans and robots and generate detailed alternative cell layouts that efficiently accommodate these task allocations. Enhanced methods are also required for evaluating the ergonomic impact of different task assignments and optimizing individual activities, such as motion and path planning [13].

Despite progress in multi-criteria planning and task assignment in both automated and human-based production systems [25], research specifically addressing hybrid systems remains limited. Efficient planning can reduce assembly time and costs, but existing approaches often overlook layout considerations and product specifications. For instance, [26] discussed methods for evaluating resource availability, suitability, and human safety while anticipating future production scenario changes. Petzoldt et al. [27] examined task allocation and cell configuration using intelligent search algorithms, while other decisional architectures have explored socially acceptable human-robot cooperation [28].

Michalos et al. [29] introduce a multi-criteria method for assigning tasks to humans and robots while considering the spatial layout of the assembly workplace. Using CAD models of products to extract assembly sequences, an intelligent algorithm generates and examines alternative planning scenarios. The method is implemented as a web-based tool and applied to an automotive case study, demonstrating its effectiveness in real-world settings. The paper concludes with an outlook on future research directions, highlighting the need for continued innovation to bridge gaps in real-time, cost-effective, and multi-objective task allocation strategies.

Despite the advancements in dynamic and adaptive task allocation in collaborative systems, several significant gaps remain in the research and practical implementation of these strategies. The primary gaps identified in the current literature include

- Real-time implementation: While various studies have developed algorithms and methods for dynamic task allocation, there remains a lack of effective real-time implementations. Most research focuses on theoretical models or simulations without translating these models into practical, real-world applications. This gap highlights the need for practical systems that can handle real-time data and adapt dynamically to changing conditions in industrial environments.
- Cost-effectiveness: Many current solutions for task allocation are expensive to implement and maintain. The research often lacks a focus on developing cost-effective solutions that are feasible for small and medium-sized enterprises (SMEs). This issue is exacerbated by the high costs associated with advanced sensors, computational resources, and the integration of dynamic allocation systems into existing workflows.
- Human factors integration: While some research has begun to address human factors, such as operator fatigue and ergonomic assessments, there is still a significant gap in integrating these factors seamlessly into dynamic task allocation frameworks. Ensuring that systems are not only efficient but also enhance worker satisfaction and well-being remains a challenge [17, 18].
- Multi-criteria optimization: Current approaches often fail to adequately address multi-criteria optimization, which involves balancing various factors such as efficiency, cost, and ergonomic impact. The development of comprehensive models that consider multiple criteria simultaneously and provide actionable insights for both human and robotic task allocation is still limited.

In summary, while there have been significant advancements in the field of dynamic and adaptive task allocation for human-robot collaboration, ongoing research must focus on bridging these gaps. Addressing these issues will involve developing real-time, cost-effective solutions that integrate human factors and optimize for multiple criteria to enhance the efficiency and effectiveness of collaborative systems in real-world applications.

3 Task allocation for collaborative workspace

This section outlines the primary objective of the proposed static task allocation method, which will be the input of the resources in the dynamic rescheduling. The processes that can benefit from this system are typically those of assembly or production, in which the cycle must be repeated several times within the working shift [30].

It minimizes the makespan which, in a production system, refers to the total time required to complete all tasks that must be performed [31]. The makespan is a crucial factor in determining the system's productivity, as a lower makespan implies a higher quantity of pieces produced or assembled within a given timeframe. The importance of makespan is further highlighted by the fact that it forms the basis of all scheduling problems [32], and minimizing it can significantly enhance a company's competitiveness in the market by reducing the time required to provide its products.

This work incorporates makespan as an objective function through the variable ms to ensure optimal system throughput. By minimizing the value of ms, the proposed task allocation method aims to efficiently allocate tasks that minimize the overall makespan, thus maximizing the number of products produced or assembled in a given timeframe. This enhances productivity and contributes to increased profitability and competitiveness in the market.

3.1 Static task allocation

The model used here includes the standard formulation for the makespan [33] since the aim of this paper is to use the resolution of a static task allocation problem as input for the dynamic task allocation one. Either way, the model is here briefly recalled. It includes the scheduling of J tasks that have to be done by K resources. The output is a binary variable, $x_{j,k}$, defined as follows:

$$x_{j,k} = \begin{cases} 1 \text{ if the task } j \text{ is performed by the resource } k \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

and it is obtained through the minimization of the makespan (2).

$$\min\ ms = \min\left(\max \sum_{j=1}^{J}(S_{j,k} + T_{j,k})\right) \tag{2}$$

where $S_{j,k}$ is the starting time of the task j and $T_{j,k}$ is its execution time when it is performed by the resource k.

The model is subjected to the following constraints:

$$\sum_{k=1}^{K} x_{j,k} = 1 \quad \forall j \tag{3}$$

$$x_{j,k} \in \{0, 1\} \quad \forall j, k \tag{4}$$

$$\sum_{j=1}^{J} x_{j,k} \geqslant 1 \quad \forall k \tag{5}$$

$$x_{j,k} = 0 \quad \forall j \in U_k \tag{6}$$

Equations 3 and 4 are respectively the *occurrence* and *integrality* constraints that assure that each task is performed by only one resource. This is done in order to consider each task as a single, not divisible job. Equation 5 it is necessary to guarantee that both resources have at least one task assigned to them; otherwise, the cell would not be collaborative, and Eq. 6 is the technological constraint for the tasks that cannot be performed by one or the other resource.

The precedence constraint was not introduced since in such scenarios where tasks are independent of each other and do not have to be performed in a particular order, introducing precedence constraints may unnecessarily restrict the resources' flexibility to perform tasks most efficiently.

However, it is crucial to consider the operator's well-being when implementing dynamic task reallocation strategies, which can be achieved by taking into account the operator's energy expenditure. Various metrics are available to evaluate human fatigue [34], with the selection depending on the specific type of physical stress being monitored. Commonly used metrics include muscular fatigue, cardiovascular indicators, and respiratory measures.

By considering energy expenditure as a measure of fatigue, it is possible to assess the overall metabolic demand placed on the operator during task execution. Energy expenditure provides a holistic view of the operator's physiological strain, encompassing the collective effort exerted across the entire body rather than focusing on isolated muscle groups or specific physiological systems.

The study of energy expenditure was first introduced by Garg et al. [35], who proposed an approach to evaluate the metabolic rate for manual jobs and walking movements, including various human aspects, such as age, body weight, gender, height, the weight of the loads, and more. Estimating energy expenditure is crucial in evaluating ergonomic risks [36], as it includes metrics such as duration, level, and repetitiveness of body works that are indicators of the stress caused by physical jobs [37]. To measure the energy expenditure required to complete a task, this work adopts the approach proposed by [38], which measures the energy needed by a resource k to complete task j ($e_{j,k}$).

3.2 Dynamic task allocation

As soon as the process starts, i.e., the resources start to perform the tasks assigned through the static task allocation, the system begins to monitor the operator's position, and task times, here called $t_{j,1}$. This is a fundamental step for the correct functioning of the system since each task is assigned to a specific position in the space, meaning that through the change of the resource position, it is possible to exactly measure the start time and the end time of each task. In other words, it is necessary to ensure that each task is assigned to a specific position in the space. This means that the resource required to perform the task has to be placed in the specific location that corresponds to that task, and only when it moves to another location, the task is completed and the next is started.

The system monitors in real-time the before-mentioned variables for each task. However, if there is a necessity, the dynamic rescheduling is done at the end of the current cycle, generating a new task allocation for the next ones. The processes that can benefit from this system are typically those of assembly or production, in which the cycle must be repeated several times within the working shift. The new task allocation is generated for the following cycles with respect to the one in which the rescheduling condition was verified, in order not to change the sequence of the tasks while the resources are performing them.

Here, various scenarios can happen. By defining the operator's energy expenditure rate $\dot{E}_j$ for each task as the ratio between the energy required for that task and its completion time, it is possible to evaluate if the reached value exceeds the threshold fixed at $\dot{E}_{th,\max} = 4.2927\ kcal/\min$ [39]. However, in order to evaluate also the residual energy effects of the previous tasks, it is necessary to include, in the evaluation of the energy expenditure, the recovery function [40]. This last follows Eq. 7:

$$R_{j-1}(\tau_{j-1}) = \int_0^{\tau_{j-1}} \dot{E}_{j-1} \cdot e^{-\mu\tau_{j-1}} d\tau \tag{7}$$

where $R_{j-1}(\tau_{j-1})$ is the residual fatigue, function of the energy rate $\dot{E}$ and of the parameter μ. That is the integral of the energy decrease rate, after the task $j-1$ if the recovery time τ_{j-1} has passed. The recovery time can be both a reaction time or an idle time or even a specific amount purposefully included to give the operator the required rest allowance as better described later.

Accordingly, the energy expenditure for the task j becomes

$$E_j = e_{j,1} + R_{j-1} \tag{8}$$

consequently, the energy expenditure rate becomes

$$\dot{E}_j = \frac{E_j}{t_{j,1}} \tag{9}$$

where $t_{j,1}$ is the actual time the operator required to complete the task j.

If, at the end of the cycle, the energy threshold is met, it can be necessary to introduce the rest allowance time τ, as mentioned before, to let the operator's energy expenditure rate return to its resting value of $\dot{E}_R = 1.86\ kcal/\min$ [39]. This time is evaluated as in Eq. 10:

$$\tau = \frac{ln(\dot{E}) - ln(\dot{E}_R)}{\mu} \tag{10}$$

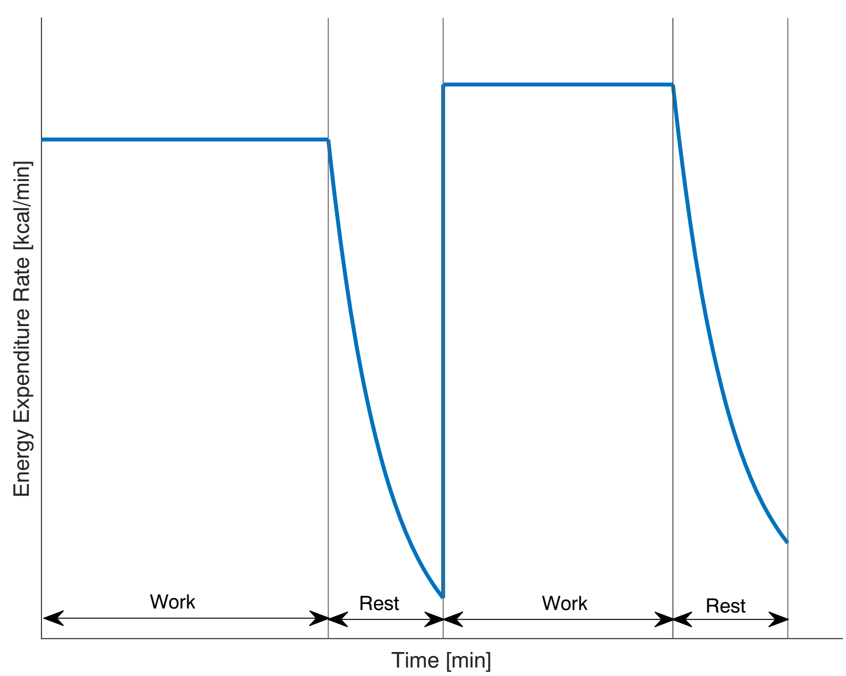
An example of the energy accumulation and recovery process is shown in Fig. 1.

Along with that, a new rescheduling is carried out to relieve the operator from the tasks that required too much effort by assigning them to the cobot if possible Eq. 6. However, if the energy lower threshold $E_{th,\min}$ is reached, the scheduling returns to the original one.

Conversely, another scenario can happen, related to exceeding of the task times. Consequently, for the time rescheduling, when the operator exceeds the makespan, a thorough assessment is conducted to determine the disparity between the total time allocated to the cobot and consumed by the operator. The primary objective is to allocate an appropriate quantity of tasks to the cobot, considering only those that conform to the technological constraint, thereby reinstating a harmonious equilibrium. Conversely, if the operator demonstrates an accelerated pace, the system will reassign certain tasks to maintain a comparable level of resource saturation between the operator and the cobot.

As for the energy, also a time minimum threshold is used: the new task allocation is given as input to the resources and kept until the actual working time of the operator reaches

Fig. 1 Energy accumulation and recovery



$T_{th,\min}$. This is done in order to keep the system as balanced as possible.

These two lower thresholds are used as parameters to return to the original task allocation in order not to disadvantage too much the required productivity. In fact, the new scheduling, which removes tasks from the operator's sequence, can be unbalanced and so increase too long the makespan, risking that the productivity requirements are not satisfied, meaning the number of pieces to complete during the working shift is not met.

Within this context, the presented approach can be proposed to facilitate the reassignment of tasks while minimizing the disruption to the operator's existing workflow. It seeks to ease the burden on the operator while also preserving the original task assignments as much as possible.

For a better understanding, the presented strategy is summarized in the Algorithm 1.

Algorithm 1 Monitoring and task allocation.

$StartSystem$
while Shift not completed **do**
Require: OP & C task scheduling
  while cycle not completed **do**
    Measure $t_{j,1}$ by monitoring OP position
    Update $\dot{E}_j$
  end while
  if $\dot{E}_j = E_{th,min}$ or $\sum t_{j,1} < ms_{th,min}$ **then**
    Revert to the original scheduling
  end if
  if $\dot{E}_j > \dot{E}_{th,max}$ **then**
    if $j \in U_2$ **then**
      Assign these tasks to the cobot
      Evaluate operator's rest allowance
    end if
  end if
  if $\sum t_{j,1} > \sum t_{T,2}$ **then**
    $\Delta = \sum t_{j,1} - \sum T_{j,2}$
    $min(\Delta)$ by reallocating tasks
  end if
end while

4 System implementation

The proposed architecture, shown in Fig. 2, is designed to be used in a collaborative work environment, where a human operator and a cobot work together nearby.

In this system where a human operator and a cobot work together, it is crucial to track their positions so that tasks can be allocated dynamically. Each position is associated with a task, and as soon as the operator changes position, the next task starts. This helps measure the exact execution time of the task at hand and update the related energy expenditure.

To track the positions of the human operator and the cobot, a markerless motion capture system is used. This system is non-intrusive and does not require the user to wear any special markers or sensors. Instead, it relies on an Intel RealSense camera, which has an RGB sensor and two sensors for stereophotogrammetry. This camera is chosen for its cost-effectiveness and accurate depth data [1].

The OpenPose library is used for real-time recognition of body joint positions. It uses advanced neural network models to accurately detect and track the movements of human

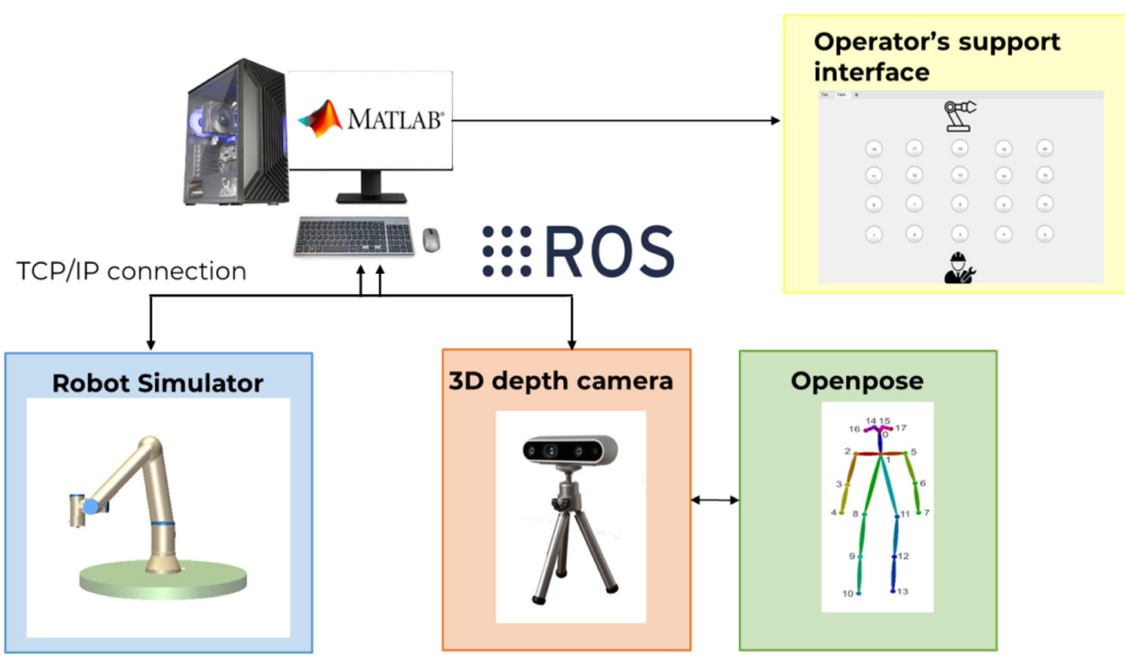


Fig. 2 Developed system components

Fig. 3 Input/output of the developed system

body parts. The BODY-25 model provides 25 key-points for each person detected in the frame, allowing for detailed tracking of each person's movement and position. To assign three-dimensional coordinates to each key-point, the system superimposes the 25 key-points on the depth frame produced by the camera [41].

To ensure real-time performance, a DELL-ALIENWARE R11 with an Intel Core i7-10700KF CPU 3.80GHz and 32 GB of RAM is used. The middleware Robotic Operating Systems (ROS) is used to achieve a frequency rate of 30fps. ROS provides a framework for developing and deploying software in a modular and scalable manner, which makes it easier to integrate different components of the motion capture system. The high-performance computing hardware used in this architecture ensures that the system can process data quickly and accurately, even when tracking multiple people and objects in real-time.

Finally, all the data provided by the camera, and the collaborative robot, such as positions and speeds, are managed by MATLAB (MathWorks) software [23], in order to have centralized control. Moreover always with Matlab, as can be seen from the figure, the resources have the input through a user interface developed, by which it is possible to receive input, control the cobot, and check the progress of the process (i.e., which tasks have been completed and which remain to be performed).

Overall, this dynamic rescheduler, whose input and output are also shown in Fig. 3, is a powerful tool for enabling collaborative work between human operators and cobots. By tracking the positions of the human operator and the cobot in real-time, the system can dynamically allocate tasks based on the current location and movement of each person and object. This can help to increase productivity, reduce errors, and improve safety in collaborative work environments.

5 Algorithm testing

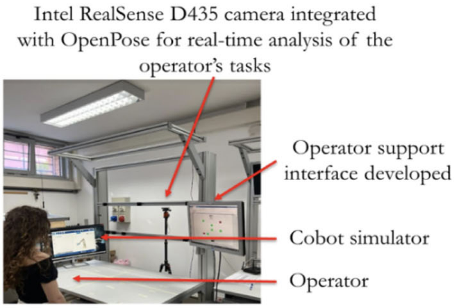
In this section, the algorithm presented is tested. In the testing, as in the model described before, the precedence constraint was not introduced. In such scenarios where tasks are independent of each other and do not have to be performed in a particular order, introducing precedence constraints may unnecessarily restrict the resources' flexibility to perform tasks most efficiently. In these cases, it may be better to allow the resources to perform tasks in any order, which can lead to faster task completion times and higher overall productivity.

Table 1 displays the tasks duration, with T_{op} the operator tasks times, T_c the cobot tasks times, and energy consumption, e_{op} of an assembly process comprising J=20 tasks. These values are sourced from [42]. The process is the assembly of a self-priming pump. This is made of a preassembly phase, a painting task, and a finishing phase, including cover refinement, quality, and packaging. The analysis is here focused on the preassembly in which most of the entire assembly, with the most fatigued tasks, is realized. Some values in the table are marked as "-" to indicate that they are infeasible for the resources utilized, as stated by the technological constraint (6). Moreover, Fig. 4 shows the setup with the developed interface.

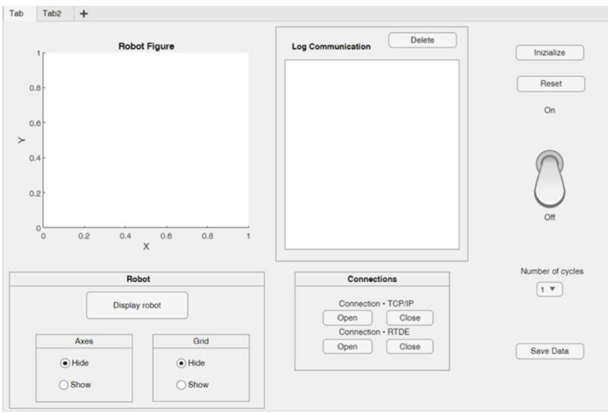
The resulting task allocation from the makespan minimization, along with the required operator's energy expenditure, are shown in Table 2, in which "OP" indicates the tasks the operator has to perform, while "C" the tasks assigned to

Table 1 Tasks time and energy

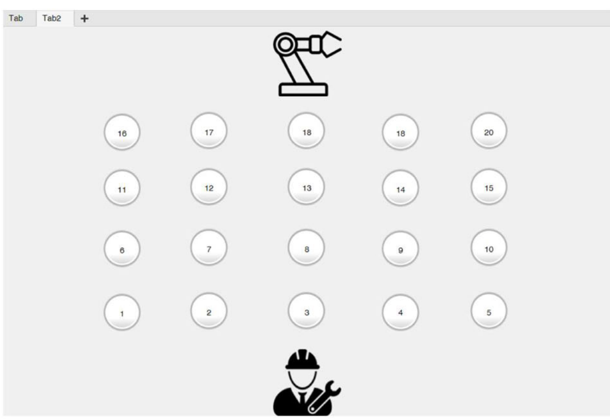
Task	T_{op} [min]	T_c [min]	e_{op} $[kcal]$
1	0.40	0.56	1.54
2	0.44	–	1.63
3	0.40	0.56	1.35
4	0.42	–	1.58
5	0.60	0.84	1.76
6	0.64	–	2.15
7	0.44	0.62	1.42
8	0.08	–	0.20
9	0.44	0.62	1.50
10	0.39	–	1.31
11	–	0.31	–
12	0.60	0.42	1.76
13	–	0.29	–
14	0.44	0.31	1.52
15	–	0.41	–
16	0.15	0.11	0.33
17	–	0.25	–
18	0.73	0.51	1.49
19	–	0.13	–
20	0.39	0.27	1.32



(a) Setup of the experimental test



(b) Operator supporting interface for control



(c) Operator supporting interface for tasks

Fig. 4 Setup of the experimental test with the operator's supporting interface

the cobot. This was obtained through the use of the Solving Constraint Integer Programs (SCIP) optimizer in the MATLAB (Mathworks) environment, since it is one of the fastest non-commercial solvers for mixed integer programming (MIP).

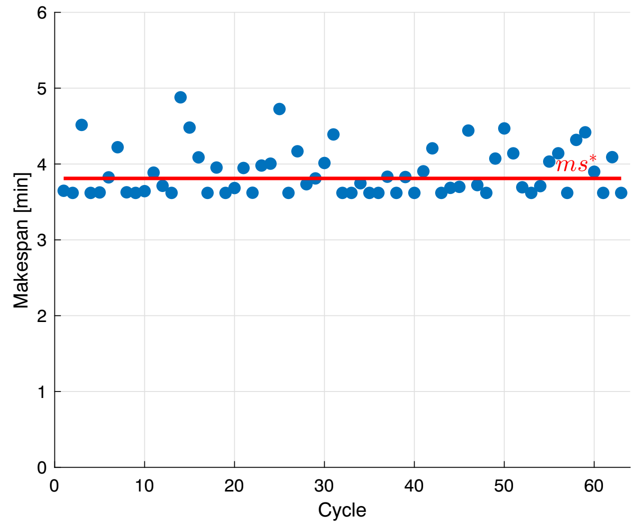


Fig. 5 Makespan during the cycle with no rescheduling

The task allocation was fed as input to the resources using the previously described approach, and a rigorous testing protocol was implemented for a continuous period of 4 h. The duration of the testing period was determined based on the standard industry practice of an $8-h$ work shift with a $1-h$ break in the middle. This break allows sufficient time for the operator to recover from the accumulated fatigue, effectively starting a new shift-like cycle. In this time range, the expected result is around 63 in assembled pieces.

5.1 No rescheduling

The first scenario studied was a traditional system without any rescheduling opportunities, without the introduction of energy evaluation, and with no rest allowance, as typically happens in an industrial environment. The result obtained in terms of productivity is shown in Fig. 5, where the makespan and its rated value (ms^*) are displayed.

The obtained average makespan, denoted as $\bar{ms}$, was found to be 3.89 min, indicating that the productivity standards were adhered to without any significant variation. This result suggests that the production process was efficiently managed, resulting in consistent performance. However, when considering the energy expenditure of the operator (E), a notable increment above the rated value (E^*) is observed. Figure 6 depicts this trend, revealing an approximately 23% increase in energy expenditure. This value, E, is obtained by summing the energy expenditure values obtained through (8)

Table 2 Makespan and energy values with the task division of the proposed solution

ms [min]	E [$kcal$]	OP	C
3.81	12.94	[1,2,3,4,5,6,7,8,10]	[9,11,12,13,14,15,16,17,18,19,20]

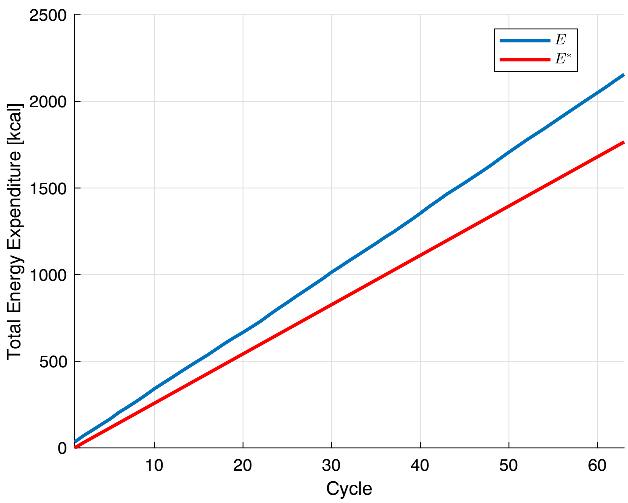


Fig. 6 Operator's total energy expenditure during the cycles with no rescheduling

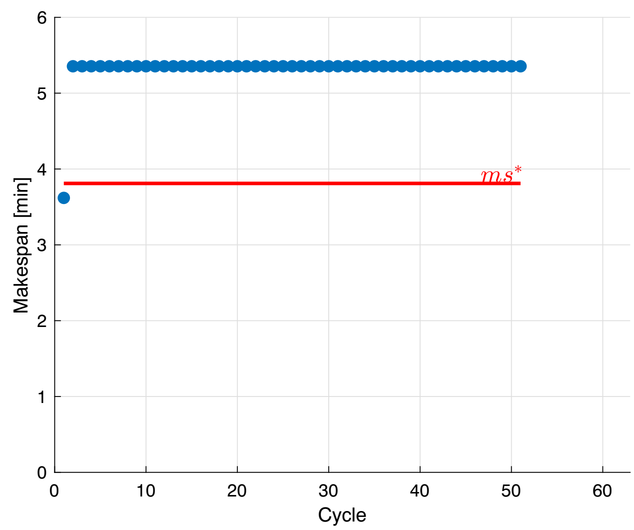


Fig. 7 Makespan during the cycles with only the energy rescheduling

and compared with the rated one, E^*, that is the value if the operator does the input scheduling for all the cycles.

While the absence of rescheduling may not have an immediate impact on long-term production, overlooking the energy expenditure and neglecting the need for rest allowance for the operator can have adverse consequences. This substantial increase in energy expenditure can lead to over-fatigue among the workforce, which in turn may result in a decline in both the operator's well-being and overall performance. It is crucial to recognize that the operator's physical and mental well-being directly influences their productivity and job satisfaction. Failure to address the increased energy demands and the necessity for rest periods can have detrimental effects on the operator's health and overall performance in the long run.

5.2 Only energy rescheduling

On the other hand, if the focus is only on the operator's well-being by minimizing energy expenditure, the makespan obtained is shown in Fig. 7. After the first cycle, the makespan is always the same because the system kept the same assignment between the operator and cobot, which was the one that minimized the energy.

The average makespan is $\bar{ms} = 5.32$ min, with a 40% increment with respect to the rated one. As a direct consequence, the production decreases from 63 expected pieces to 51 pieces actually assembled. Naturally, instead, since the rescheduling, in this case, is based on the overcome of the energy expenditure, this last is significantly lower than the rated average value, of about 25%, as shown in Fig. 8.

Based on this finding, it can be of help to proceed to explore the potential benefits of introducing the rescheduling, which considers both time and energy variability, focusing on how it can enhance the operator's working sustainability and productivity in accordance with the principles of Industry 5.0.

5.3 Makespan and energy dynamic rescheduling

The thresholds employed for this testing are presented in Table 3, and the value of parameter $\mu = 1.5$ for the recovery function was derived from [43], given that the activities have comparable intensity and duration. This value is typically linked to the characteristics of the operator; however, it is possible to adopt an average value, as done, representative of the average operator.

As planned and serving as an initial outcome, the conducted tests within the designated time frame have demon-

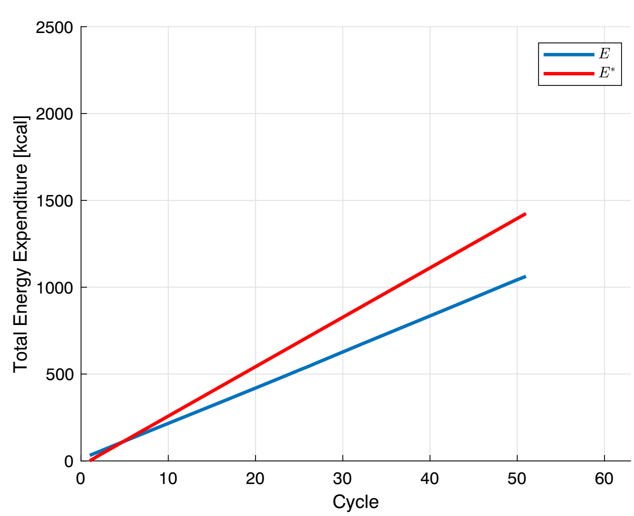


Fig. 8 Total energy expenditure during the cycles with only the energy rescheduling

Table 3 Thresholds for the rescheduling

$T_{th,\min}$	$\dot{E}_{th,\min}$	$\dot{E}_{th,\max}$
$0.5 \cdot ms^*$	$0.5 \cdot \dot{E}_{th,\max}$	$4.2927\ [kcal]$

strated the attainment of the expected number of produced pieces. Figure 9 substantiates this observation, which graphically depicts the achieved makespan during the cycles, accompanied by the minimum operator's time threshold $T_{th,\min}$ and the rated makespan ms^*.

In accordance with the previously outlined model, the makespan incorporates the requisite rest time τ if the energy expenditure rate exceeds the predetermined maximum threshold. Notably, despite the inclusion of rest time, the productivity requirements have been duly respected, as evidenced by the average makespan recorded during the testing phase amounting to $\bar{ms} = 3.9$ min, a marginal increment of merely 0.09 min in comparison to the rated value. It is noteworthy to emphasize that despite the inclusion of a rest time at the end of the cycle when necessary, the productivity standard is consistently met. This achievement was made possible through the implementation of time rescheduling, which dynamically altered task allocation a total of 9 times during the conducted tests, thereby facilitating the balancing of the resources. The majority of these rescheduling instances occurred towards the end of the shift, a period when the operator experienced heightened fatigue.

An equally balanced result has been achieved in terms of the average energy expenditure rate on the tasks, as shown in Fig. 10.

In the figure, the operator's average energy expenditure rate throughout the shift is reported, along with the maximum

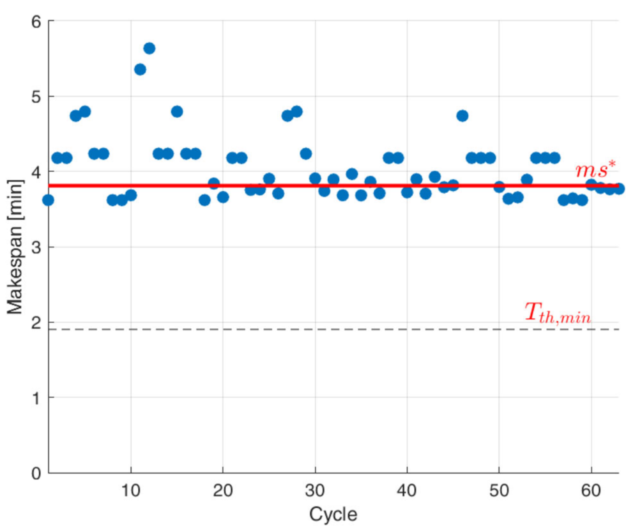


Fig. 9 Makespan during the cycles with the dynamic rescheduling

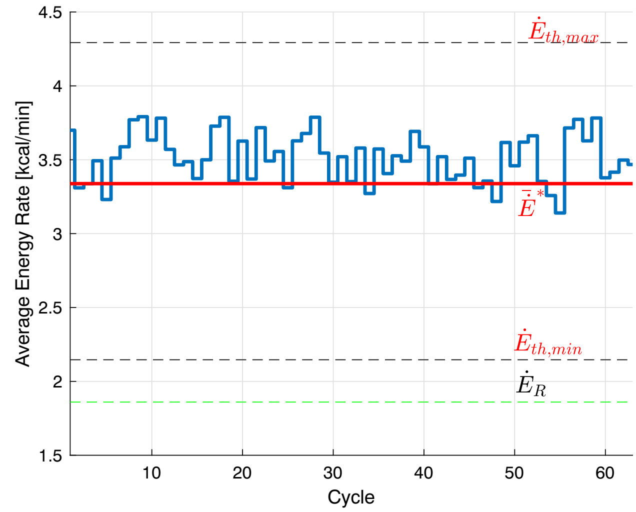


Fig. 10 Average energy expenditure rate during the cycles with the dynamic rescheduling

and minimum thresholds ($\dot{E}_{th,\max}$, $\dot{E}_{th,\min}$), the rest rate ($\dot{E}_R$), and the rated average energy expenditure rate ($\bar{\dot{E}}^*$). On average, the observed trend remains within the thresholds and closely approximates the nominal rate. However, it is important to note that adherence to the average threshold does not imply that the threshold was not surpassed during individual cycles, as illustrated in Fig. 11.

At the local level, within each cycle, the energy expenditure rate may exceed either threshold, necessitating a system rescheduling. This proactive adjustment ensures that subsequent cycles impose a reduced task load on the operator, allowing for recuperation from the exertion experienced in the preceding cycle. Consequently, the strategic

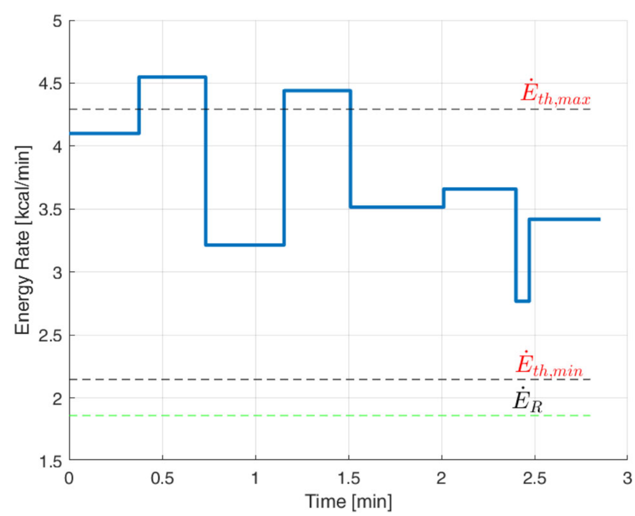


Fig. 11 Energy expenditure rate during a single cycle with the dynamic rescheduling

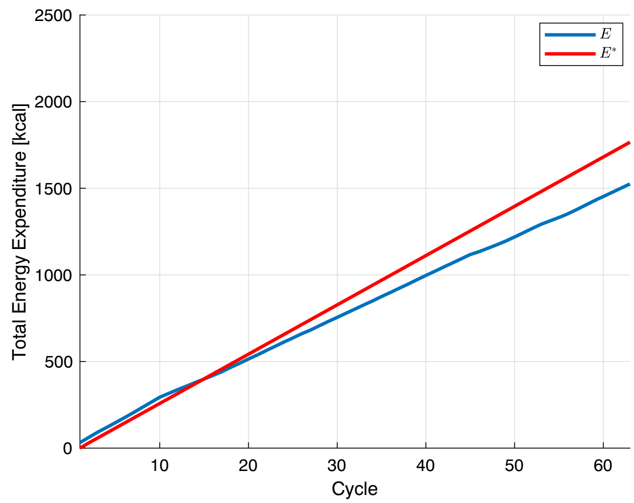


Fig. 12 Average total energy expenditure during the cycles with the dynamic rescheduling

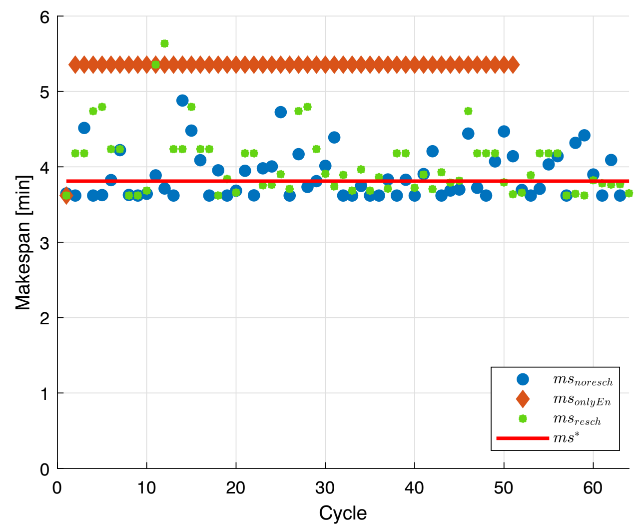


Fig. 13 Comparison between no rescheduling, only energy rescheduling, and complete rescheduling for the makespan

re-scheduling based on energy thresholds effectively mitigates operator fatigue and facilitates a more sustainable workflow.

This is also confirmed by the total energy expenditure trend during the cycles, as shown in Fig. 12, which is even lower than the rated value.

The success of maintaining the energy expenditure within acceptable bounds is attributed to the rescheduling approach driven by energy thresholds. This adaptive strategy accommodates local variations in the energy expenditure rate, enabling task redistribution and enabling the operator to recover from exertion, thus optimizing overall operational efficiency.

5.4 Overall results

The goodness of this approach is also confirmed by Figs. 13 and 14, in which are reported, respectively, the comparison for the three cases just discussed for the makespan and for the energy expenditure.

In both figures, the blue curves represent the case without any rescheduling which show that the makespan is closer to the rated value, although the energy expenditure is bigger with respect to its rated value. The orange curves show the case in which only the energy rescheduling is applied, showing a decrease in productivity in order to keep the energy expenditure smaller, while the compromise is shown by the green curves that represent the makespan and energy expenditure rescheduling scenario. In this last case, as previously discussed, both the makespan and the total energy are close to their rated value, maintaining productivity, nevertheless the inclusion of the resting times, and reducing the effort required to the operator, improving the well-being.

5.5 Sensitivity analysis

In this section, some of the parameters previously used are varied to evaluate if and how they influence the result and usefulness of the proposed system. In particular, the parameters considered are the makespan and energy expenditure minimum thresholds and the strictness of the technological constraint.

By keeping the same set of feasible tasks but tightening the thresholds to 75% of their maximum values, the results obtained, always in a test of 4 h of duration, are shown in Fig. 15.

Despite the apparent similarity to the previous case, especially since the average value of the makespan is $\bar{ms}$ =

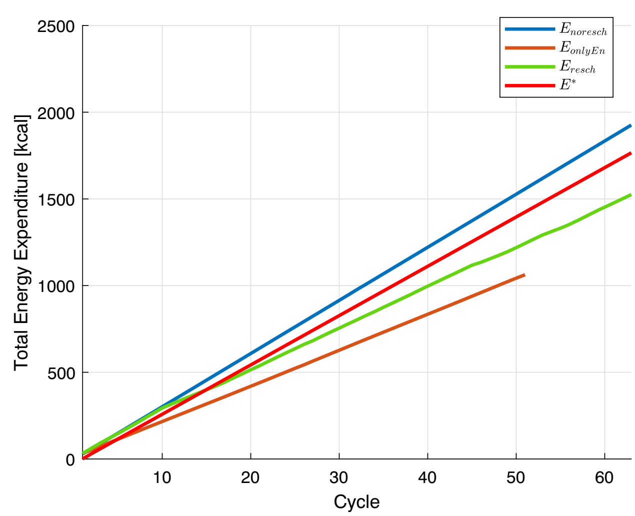
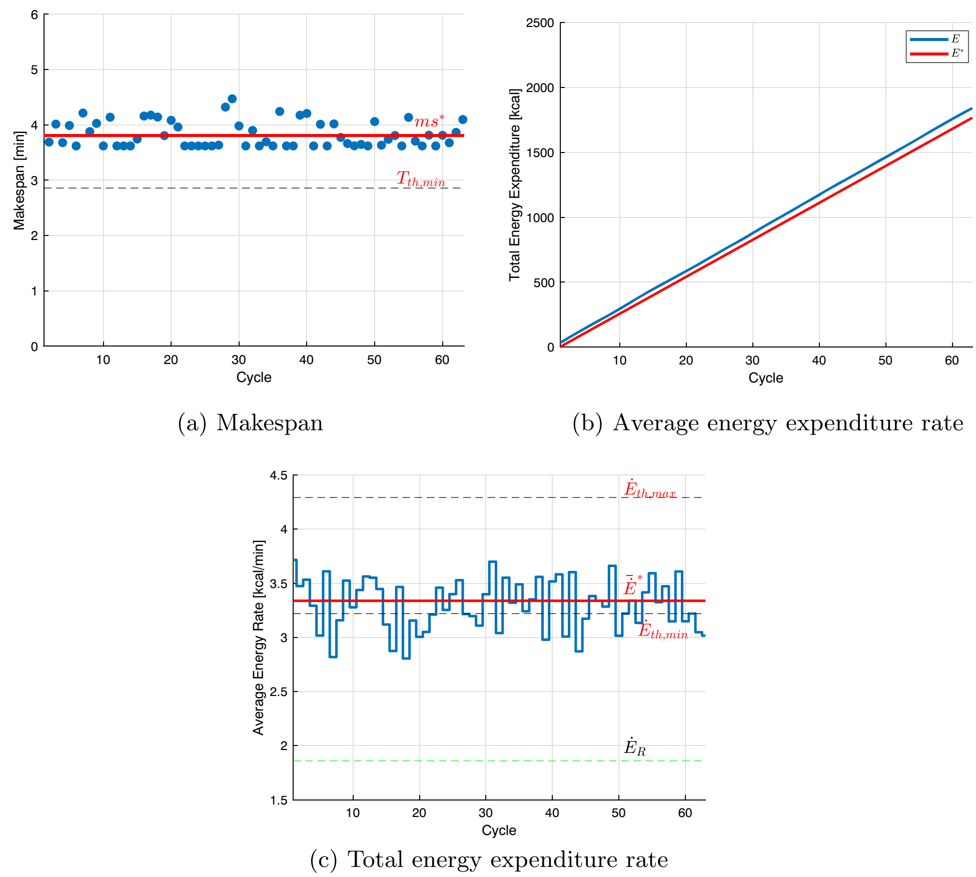


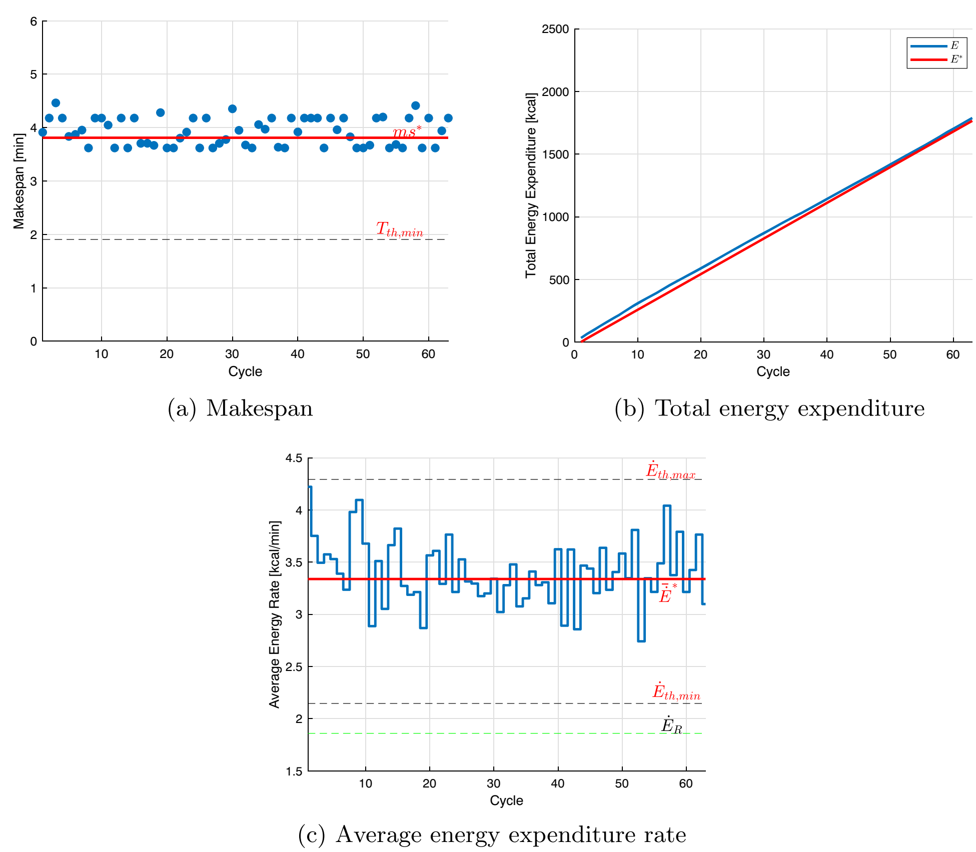
Fig. 14 Comparison between no rescheduling, only energy rescheduling, and complete rescheduling for the energy expenditure

Fig. 15 Makespan, total energy expenditure and average energy expenditure rate during the cycles with minimum thresholds equal to 75% of the maximum ones

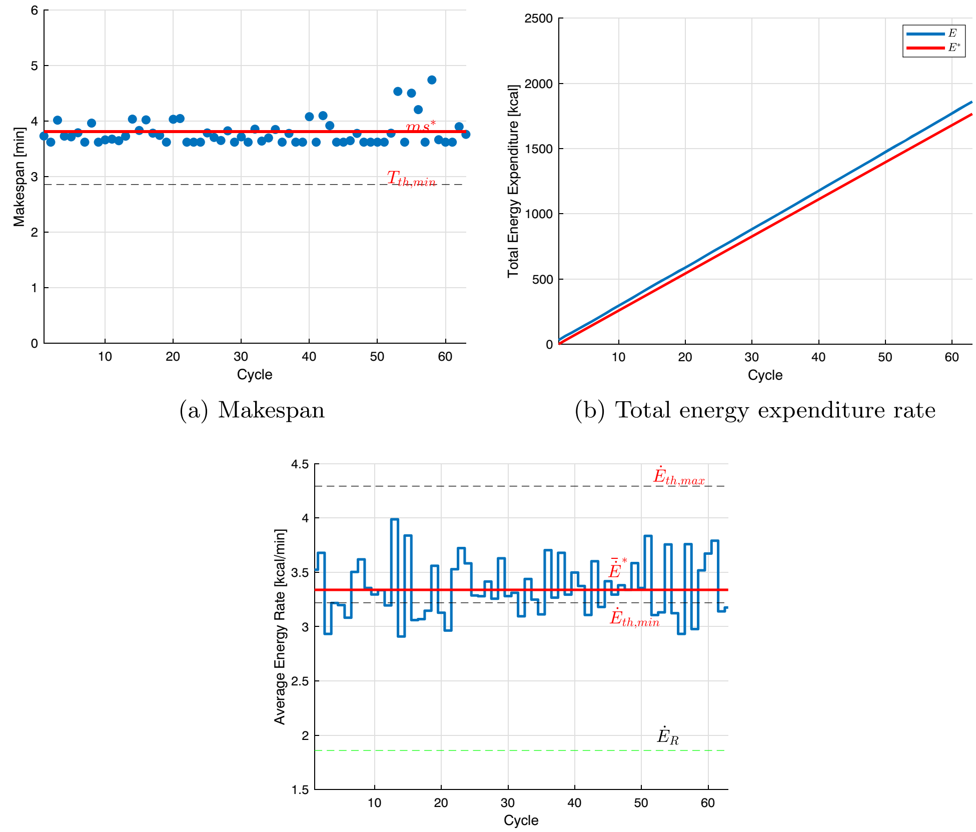


(a) Makespan (b) Average energy expenditure rate (c) Total energy expenditure rate

Fig. 16 Makespan, total energy expenditure and average energy expenditure rate during the cycles with minimum thresholds equal to 50% of the maximum ones



(a) Makespan (b) Total energy expenditure (c) Average energy expenditure rate



(a) Makespan

(b) Total energy expenditure rate

(c) Average energy expenditure rate

Fig. 17 Makespan, total energy expenditure and average energy expenditure rate during the cycles with minimum thresholds equal to 75% of the maximum ones

3.82 min (Fig. 15), a notable distinction emerges, attributable to the stringent nature of the minimum energy threshold. As evidenced in Fig. 15, this heightened strictness restricts the occurrence of rescheduling events. Consequently, the total energy expenditure increase becomes evident, surpassing both the rated average and the scenario characterized by a smaller minimum threshold. The average energy expenditure rate further supports this observation (Fig. 15), where it is evident that the minimum threshold imposed is excessively restrictive, hindering the system's ability to effectively redistribute task allocation and optimize resource utilization, which could have also alleviated the operator's workload.

Analogous findings can be extrapolated when imposing an even stricter technological constraint on the resources, limiting their capability to perform only 12 tasks instead of the previously considered 15. In this scenario, the initial task allocation remains unchanged. Nevertheless, due to the substantial impediment in executing the majority of the tasks assigned to the operator, the cobot's involvement in undertaking critical tasks may be significantly diminished, particularly when confronted with higher lower thresholds. This discernible effect is effectively depicted in Figs. 16 and 17, which respectively illustrate the case with lower thresholds set at 50% and 75% of their higher counterparts, thereby corroborating the significant influence of threshold levels.

By imposing a more rigorous technological constraint on the system, the resulting impact is twofold. Firstly, the inability to perform a significant proportion of the tasks assigned to the operator restricts the cobot intervention, leading to a diminished allocation of critical tasks. Consequently, the cobot's role in balancing the workload is curtailed, compromising the optimization of resource utilization.

These observations underscore the substantial impact of threshold levels on task allocation dynamics. A more stringent technological constraint not only impedes task execution but also hinders the cobot's ability to intervene effectively. Consequently, the findings highlight the critical role played by threshold levels in shaping resource allocation strategies and optimizing operational efficiency.

6 Conclusions

The shift towards a human-centered design in the workspace, known as a pillar of Industry 5.0, emphasizes the importance of considering human factors in collaborative assembly systems. Integrating human and robotic resources can impact the performance of human operators, making it crucial to address factors such as ergonomics, mental workload, and capabilities to ensure system productivity and operator well-being.

Neglecting these aspects has been a common reason for implementation failures in such systems [7, 44].

To achieve productivity, flexibility, and human factor objectives, a task allocation strategy able to adapt to human variability can be developed for collaborative assembly systems. This strategy real-time optimizes task assignment to resources, considering their characteristics, to maximize productivity and resources saturation. It also takes into account the comfort and physical strain on the human operator, promoting a harmonious work environment.

A well-designed task allocation strategy considers makespan and the operator's energy expenditure as objectives, aligning with the goals of productivity and creating a human-centered workplace. Traditional static allocation, developed offline, may provide optimal global solutions but can be time-consuming and inflexible. In contrast, a dynamic task allocation approach enables effective collaboration by adapting to the evolving needs of operators in real-time. This work combines static and dynamic allocation, leveraging digitalization of the human operator's role to improve collaboration and consider real-time human variability.

For this purpose, this paper presented a new architecture with a strategy for re-allocating the tasks among the resource during the working shift. The findings of the study highlight the significance of time rescheduling and energy thresholds in optimizing task allocation and resource balancing within the system. The implementation of time rescheduling, particularly towards the end of the shift when operator fatigue is more pronounced, proves instrumental in promoting resources' equilibrium and minimizing disruptions to the operator's workflow.

The analysis of energy expenditure rates and thresholds reveals crucial insights. The observed adherence of the average energy expenditure rate to the prescribed thresholds indicates the successful management of operator workload, ensuring it remains within acceptable bounds. Moreover, the use of rescheduling based on energy thresholds allows for task redistribution, facilitating the operator's recovery from exertion, reducing the total energy consumption, and enhancing overall operational efficiency.

However, the study also highlights the impact of stringent minimum energy thresholds on task allocation dynamics. In scenarios where the minimum threshold is particularly strict, rescheduling opportunities become limited, resulting in higher energy expenditure compared to the rated one and scenarios with less restrictive thresholds. This emphasizes the importance of balancing threshold levels to optimize operator performance and minimize energy consumption.

The investigation further demonstrates the sensitivity of the system's performance to variations in input parameters. Different thresholds and constraints significantly influence the allocation of tasks and the involvement of the cobot, with implications for workload distribution and resource utilization.

Overall, this work provides insights into the relative importance and impact of various factors on task allocation, operator fatigue, and operational efficiency. These findings contribute to the body of knowledge on resource allocation strategies in human-robot collaboration systems, informing decision-making processes and guiding future research and development efforts in the field.

Future research in human-robot collaboration systems should focus on several key areas to enhance system effectiveness and adaptability. One important area is the development of alternative metrics for evaluating physical stress. This involves identifying and utilizing metrics that accurately reflect the physiological demands of specific tasks, such as those requiring particular limb stress or unique ergonomic considerations. Such metrics will enable a more precise assessment of physical fatigue, thereby improving task allocation strategies and enhancing operator well-being.

Additionally, expanding rescheduling strategies to include multiple objectives can significantly benefit both productivity and operator health. Future studies should aim to incorporate considerations such as minimizing monotonous tasks and optimizing task sequencing to reduce cognitive load. By integrating these objectives, rescheduling strategies can be better aligned with broader goals, promoting a balanced approach that addresses both efficiency and human factors.

Another promising direction is the exploration of fractional task allocations. This involves examining the implications of distributing tasks across parallel stations, which can potentially enhance throughput but also necessitates changes to layout design, worker training, and cost management. Understanding the trade-offs associated with fractional allocations, including their impact on system flexibility and performance, is crucial. Research should investigate how fractional tasks can be integrated with dynamic task allocation strategies to improve overall system functionality.

Furthermore, there is a need to focus on real-time implementation and adaptability of task allocation systems. Developing practical, cost-effective solutions that integrate human factors and multi-criteria optimization is essential. Current research often relies on theoretical models or simulations, so there is a gap in translating these models into real-world applications. Addressing this gap will involve creating systems capable of processing real-time data and adapting dynamically to changing conditions, ensuring that they are both practical and responsive.

By addressing these areas, future research can advance the field of human-robot collaboration, improving both system performance and operator satisfaction while achieving greater productivity in industrial settings.

Author Contributions All authors contributed to the study conception and design. Material preparation, data collection, and analysis were performed by all the authors. The first draft of the manuscript was written by Irene Granata, and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding Open access funding provided by Università degli Studi di Padova within the CRUI-CARE Agreement. This study was carried out within the PNRR research activities of the consortium iNEST (Interconnected North-Est Innovation Ecosystem) funded by the European Union Next-GenerationEU (Piano Nazionale di Ripresa e Resilienza (PNRR) - Missione 4 Componente 2, Investimento 1.5 - D.D. 1058 23/06/2022, ECS_00000043). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

Declarations

Conflict of Interest The authors declare no compeing interests.

References

1. Boschetti G, Faccio M, Granata I, Minto R (2023) 3d collision avoidance strategy and performance evaluation for human-robot collaborative systems. Comput & Ind Eng 179:109225
2. Faccio M, Bottin M, Rosati G (2019) Collaborative and traditional robotic assembly: a comparison model. Int J Adv Manuf Technol 102(5):1355–1372
3. Takata S, Hirano T (2011) Human and robot allocation method for hybrid assembly systems. CIRP annals 60(1):9–12
4. Romero D, Stahre J (2021) Towards the resilient operator 5.0: the future of work in smart resilient manufacturing systems. Procedia CIRP 104:1089–1094
5. Faccio M, Granata I, Menini A, Milanese M, Rossato C, Bottin M, Minto R, Pluchino P, Gamberini L, Boschetti G, Rosati G (2022) Human factors in cobot era: a review of modern production systems features. J Intell Manuf. https://doi.org/10.1007/s10845-022-01953-w
6. Fletcher SR, Johnson T, Adlon T, Larreina J, Casla P, Parigot L, Alfaro PJ, del Mar Otero M (2020) Adaptive automation assembly: identifying system requirements for technical efficiency and worker satisfaction. Comput & Ind Eng 139:105772
7. Ghani KA, Jayabalan V (2000) Advanced manufacturing technology and planned organizational change. J High Technol Manag 11(1):1–18
8. Kulic D, Croft E (2005) Anxiety detection during human-robot interaction. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 616–621
9. Granata I, Faccio M, Maretto L (2023) Dynamic task allocation for collaborative robot systems. In: Proceedings of the changeable, agile, reconfigurable and virtual production conference and the world mass customization & personalization conference, pp 808–816. Springer
10. Leng J, Sha W, Wang B, Zheng P, Zhuang C, Liu Q, Wuest T, Mourtzis D, Wang L (2022) Industry 5.0: prospect and retrospect. J Manuf Syst 65:279–295
11. Kong XT, Luo H, Huang GQ, Yang X (2019) Industrial wearable system: the human-centric empowering technology in Industry 4.0. J Intell Manuf 30(8):2853–2869
12. Mandolini M, Favi C, Germani M, Marconi M (2018) Time-based disassembly method: how to assess the best disassembly sequence and time of target components in complex products. Int J Adv Manuf Technol 95(1):409–430
13. Schmidbauer C, Schlund S, Ionescu TB, Hader B (2020) Adaptive task sharing in human-robot interaction in assembly. In: 2020 IEEE international conference on industrial engineering and engineering management (IEEM), pp 546–550. https://doi.org/10.1109/IEEM45057.2020.9309971
14. Chen F, Sekiyama K, Cannella F, Fukuda T (2013) Optimal subtask allocation for human and robot collaboration within hybrid assembly system. IEEE Trans Autom Sci Eng 11(4):1065–1075
15. Pupa A, Van Dijk W, Secchi C (2021) A human-centered dynamic scheduling architecture for collaborative application. IEEE Robot Autom Lett 6(3):4736–4743
16. Liu D, Wei Q, Wang D, Yang X, Li H (2017) Adaptive dynamic programming with applications in optimal control
17. Messeri C, Bicchi A, Zanchettin AM, Rocco P (2022) A dynamic task allocation strategy to mitigate the human physical fatigue in collaborative robotics. IEEE Robot Autom Lett 7(2):2178–2185
18. Merlo E, Lamon E, Fusaro F, Lorenzini M, Carfì A, Mastrogiovanni F, Ajoudani A (2023) An ergonomic role allocation framework for dynamic human-robot collaborative tasks. J Manuf Syst 67:111–121
19. Scholl A, Klein R (1999) Balancing assembly lines effectively-a computational comparison. Eur J Oper Res 114(1):50–58
20. Renna P (2024) Design model for assembly lines including fractional tasks and parallel workstations. International J Inter Design Manuf (IJIDeM) 18(1):177–190
21. Lopes TC, Brauner N, Magatão L (2022) Assembly line balancing with fractional task allocations. Int J Prod Res 60(5):1569–1586
22. Jeong I-J, Jeon S (2021) Balance ability of a work-sharing line using floating workers and its comparison with floating work strategy. Int J Prod Res 59(19):5903–5912
23. Granata I, Faccio M, Boschetti G (2024) Industry 5.0: prioritizing human comfort and productivity through collaborative robots and dynamic task allocation. Procedia Comput Sci 232:2137–2146
24. Kokotinis G, Michalos G, Arkouli Z, Makris S (2024) A behavior trees-based architecture towards operation planning in hybrid manufacturing. Int J Comput Integr Manuf 37(3):324–349
25. Riedelbauch D, Henrich D (2019) Exploiting a human-aware world model for dynamic task allocation in flexible human-robot teams. In: 2019 International conference on robotics and automation (ICRA), pp. 6511–6517. IEEE
26. Makris S, Makris S (2021) Allocation of manufacturing tasks to humans and robots. Cooperating Robots for Flexible Manufacturing, pp 373–380
27. Petzoldt C, Niermann D, Maack E, Sontopski M, Vur B, Freitag M (2022) Implementation and evaluation of dynamic task allocation for human-robot collaboration in assembly. Appl Sci 12(24): 12645
28. Sipsas K, Nikolakis N, Makris S (2021) Dynamic assembly planning and task assignment. In: Advanced human-robot collaboration in manufacturing, pp 183–210. Springer, Berlin

29. Michalos G, Spiliotopoulos J, Makris S, Chryssolouris G (2018) A method for planning human robot shared tasks. CIRP J Manuf Sci Technol 22:76–90
30. Faccio M, Granata I, Minto R (2024) Task allocation model for human-robot collaboration with variable cobot speed. J Intell Manuf 35(2):793–806
31. Tayali HA (2021) Manufacturing scheduling strategy for digital enterprise transformation. Emerging challenges, solutions, and best practices for digital enterprise transformation. IGI Global, Hershey, Pennsylvania, USA, pp 104–119
32. Pinedo M, Hadavi K (1992) Scheduling: theory, algorithms and systems development. In: Gaul W, Bachem A, Habenicht W, Runge W, Stahl WW (eds) Operations research proceedings 1991. Springer. Berlin, Heidelberg, pp 35–42
33. Calzavara M, Faccio M, Granata I (2023) Multi-objective task allocation for collaborative robot systems with an Industry 5.0 human-centered perspective. The International Journal of Advanced Manufacturing Technology, pp 1–18
34. Ma L, Chablat D, Bennis F, Zhang W (2009) A new simple dynamic muscle fatigue model and its validation. Int J Ind Ergon 39(1):211–220
35. Garg A, Chaffin DB, Herrin GD (1978) Prediction of metabolic rates for manual materials handling jobs. Am Ind Hyg Assoc J 39(8):661–674
36. Stecke KE, Mokhtarzadeh M (2022) Balancing collaborative human-robot assembly lines to optimise cycle time and ergonomic risk. Int J Prod Res 60(1):25–47
37. Anwer S, Li H, Antwi-Afari MF, Umer W, Wong AYL (2021) Evaluation of physiological metrics as real-time measurement of physical fatigue in construction workers: state-of-the-art review. Journal of Construction Engineering and Management 147(5)
38. Granata I, Faccio M, Calzavara M (2023) Energy expenditure and makespan multi-objective optimization for cobots systems design. Procedia Comput Sci 217:126–135
39. Price A (1990) Calculating relaxation allowances for construction operatives-part 1: metabolic cost. Appl Ergon 21(4):311–317
40. Calzavara M, Persona A, Sgarbossa F, Visentin V (2019) A model for rest allowance estimation to improve tasks assignment to operators. Int J Prod Res 57(3):948–962
41. Cao Z, Hidalgo Martinez G, Simon T, Wei S, Sheikh YA (2019) Openpose: realtime multi-person 2D pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence
42. Boschetti G, Faccio M, Granata I (2022) Human-centered design for productivity and safety in collaborative robots cells: a new methodological approach. Electronics 12(1):167
43. Valentina V, Fabio S, Martina C, Alessandro P (2018) Fatigue accumulation in the assignment of manual material handling activities to operators. IFAC-PapersOnLine 51(11):826–831
44. Castrillón ID Cantorna AIS (2005) The effect of the implementation of advanced manufacturing technologies on training in the manufacturing sector. Journal of European Industrial Training