International Journal of Production Research

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/tprs20

Multi-agent system optimisation in factories of the future: cyber collaborative warehouse study

Puwadol Oak Dusadeerungsikul, Xiang He, Maitreya Sreeram & Shimon Y. Nof

To cite this article: Puwadol Oak Dusadeerungsikul, Xiang He, Maitreya Sreeram & Shimon Y. Nof (2022) Multi-agent system optimisation in factories of the future: cyber collaborative warehouse study, International Journal of Production Research, 60:20, 6072-6086, DOI: 10.1080/00207543.2021.1979680

To link to this article: https://doi.org/10.1080/00207543.2021.1979680

Published online: 11 Oct 2021.

INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH
2022, VOL. 60, NO. 20, 6072–6086
https://doi.org/10.1080/00207543.2021.1979680

Multi-agent system optimisation in factories of the future: cyber collaborative warehouse study

Puwadol Oak Dusadeerungsikul, Xiang He, Maitreya Sreeram and Shimon Y. Nof

PRISM Center and School of Industrial Engineering, Purdue University, West Lafayette, IN, USA

ABSTRACT

The rapid advancement of technologies leading to automation 5.0 has challenged manufacturers preparing for factories of the future, including warehouses, which are considered a key element in supply chains. Because of technologies such as warehouse robots, Internet of Things, Internet of Services, and cyber-augmented collaboration, the traditional warehouse system structure has been changed, improving its performances significantly. The challenges, however, are how to design a system with multi-agents and technologies to reach maximum potential. In this study, a new collaborative workflow protocol for cyber collaborative warehouse, called Collaboration Requirement Planning protocol for HUB-CI (CRP-H), is developed for optimising the collaborative workflow of a warehouse multi-agent system. The two phases of CRP-H are designed to answer questions: (1) Which robot(s) should execute which task? and (2) When should this task be executed? Results show (with statistical significance) that under CRP-H, total operational cost reduces by 11.84%, and total weighted completion time reduces by 37.11%. When the system has unplanned requests, CRP-H can still reduce total operational cost by 5.70% and total weighted completion time by 10.11%. Lastly, CRP-H, which enables a human input integrated into the design, can also reduce the total operational cost even when critical information is missing.

ARTICLE HISTORY
Received 18 January 2021
Accepted 31 August 2021

KEYWORDS
Automation 5.0; Collaborative Control Theory (CCT); cyber augmentation; cyber collaborative protocol; Task Administration Protocols (TAP)

Nomenclature

Abbreviations

C2W	cyber collaborative warehouse
CCT	collaborative control theory
CRM	collaboration requirement matrix
CRP	collaboration requirement planning
CRP-H	collaboration requirement planning protocol for HUB-CI
CRS	collaborative robot schedule
FoF	factories of the future
HUB-CI	HUB for collaborative intelligence
IoT/IoS	Internet of things and internet of services
MRTA	multi-robot tasks allocation
TAP	task administration protocol
UR	unplanned request
VIPO	visual programming
WSPT	weighted shortest processing time first

Variables

$\Delta Cost$	cost saving
$\Delta Time$	time saving
γ	total weighted completion time of all package except i and i'
C_{CRP-H}	total operational cost from CRP-H
$C_{Basedline}$	total operational cost from the baseline procedure
C_{max}	makespan
C^*_{max}	optimal makespan
C_{ij}	cost for package i is stored by a robot of robot team j
K	maximum loading factor
N	number of packages
N_p	number of job in package p
p_i	processing time of package i
R	number of robot or robot team
R_j	robot j
S, S'	schedule
W_{CRP-H}	total weighted completion time from CRP-H
$W_{Basedline}$	total weighted completion time from baseline procedure
x_{ij}	indicator of package i is executed by a robot or robot team j
y_{it}	indicator of package i start at time t

CONTACT Puwadol Oak Dusadeerungsikul pdusadee@purdue.edu PRISM Center and School of Industrial Engineering, Purdue University, West Lafayette, IN, 47907, USA

This article has been corrected with minor changes. These changes do not impact the academic content of the article.

1. Introduction

Currently, humanity is at an early stage of a massive convergence of technologies. This convergence of technologies can affect the design and implementation of Factories of the Future (FoF), including warehouses and distribution centres. Warehouses and distribution centres have been considered as an important element in supply chain which needs to address perspectives and features including next-generation agility and adaptability to market demands, modular and connected systems, learning and intelligent agents, treating data as an asset, virtual system models, and digital supply chain visibility (Tate 2018). Therefore, the competitiveness of companies in the globalised supply chain would depend upon how well they address these features, characteristics, and challenges.

In this article, a study of the Cyber Collaborative Warehouse (C2W) operations is examined. Robots have been utilised in warehouses to execute tasks such as picking, packing, storing, and retrieving packages. While there have been policies designed for optimising routing decisions for a single robot in a smart warehouse (He, Aggarwal, and Nof 2018), there is limited literature on multi-robot task allocation and scheduling. Moreover, Internet of Things and Internet of Services (IoT/IoS) devices will be the important elements in the future warehouse as they can minimise data collection and validation time (Cao et al. 2019). Without a model to manage massive data generated in real-time by IoT/IoS, and to optimise and harmonise system agents, the system will work ineffectively.

To accomplish the above objective, a Collaboration Requirement Planning protocol for HUB-CI (CRP-H) is designed, developed, and validated in this research. HUB-CI is a hub for Collaborative Intelligence, inspired by CCT, the Collaborative Control Theory (Nof 2007; Nof et al. 2015), and has been introduced and developed at the Purdue PRISM Center since 2008 (Devadasan, Zhong, and Nof 2013; Zhong, Wachs, and Nof 2014; Nair 2019). HUB-CI is a brain-inspired model for addressing problems in agents' collaboration and data management. Its objective is to manage massive information exchanges among distributed agents in real-time. Based on the current information and intelligence, it can optimise and harmonise operations instructed by human operators. This hub is designed to receive commands from human agents via a user interface and develop a plan for robots that are managed by executive middleware. A CRP-H, which manages local information obtained from system agents (humans, robots, warehouse shelves, etc.), enables collaboration among agents in the system.

The contribution of the research is the development of a new protocol for C2W enhancing the ability of HUB-CI. Moreover, an algorithm for scheduling multi-robot collaboration, called Collaborative Robots Scheduling (CRS), is developed to support the new protocol. Two theorems that ensure the system's performance are proven in this article. Lastly, the design analysis shows the benefit of implementing the original concept of C2W compared to current practices.

In this article, Section 2 summarises the background and previous work related to the problem. Section 3 defines the characteristics of C2W, and Section 4, to solve the problem, presents the design of CRP-H. Section 5 discusses the experiment conducted to validate the performance of CRP-H and summarises the results. Finally, Section 6 discusses the conclusion and future research directions.

2. Literature review

In order to improve the performance of warehouse operation, robots have been utilised for tasks such as storage and retrieval. Such warehouse operation is a critical value-adding activity in the supply chain to attain excellence in terms of customer service, lead times, and cost (Liu, Yu, and Liu 2006). With increasing interaction and collaboration between agents (e.g. robots), there is a growing demand for real-time e-services for communication and information validation. Intelligent agents for future warehouse operation, such as autonomous robots, must be able to process information and decisions from their environment, share the workflows with their peers to overcome difficulties due to shortage and lateness of data, improve efficiency by integrating real-time information (Wang, Chen, and Xie 2010) and decrease work-related errors and conflicts in systems.

The primary motivation for adopting multi-robot control in warehouse is the possibility of reducing production costs by having robots working faster and in parallel (Brogårdh 2007), and in overall work tasks and resource collaboration (Nof 1999). Collaborative work has been proven to reduce processing time, enable better task assignment and performance allocation, and thus improves the efficiency, timeliness, and quality of operations. Instead of having a single, powerful, and complicated robot, a group of small yet simple robots is easier to implement (Khamis, Hussein, and Elmogy 2015) and yields superior performance (Nof et al. 2015). When collaborating machine sets share the same functionality, the overlap of machine capability introduces backup resources to the system. The benefit is especially significant at process bottlenecks, in which cases a single failure can disrupt the entire workflow. Moreover, the overlap of machine capability through collaboration adds another dimension of flexibility due to the additional tasks that can be performed by the collaborating machine set (Rajan and Nof 1996). Importantly, having multiple small and simple robots can be cheaper to implement than having a single powerful and complicated robot (Nof 2007; Khamis, Hussein, and Elmogy 2015).

In Part 12, Robotics Terminology, of the *Handbook of Industrial Robotics* (Nof 1999) defines a Collaborative Multi-Robot System as a system where robots exploit the ability to complete tasks independently or through collaboration. In such systems, tasks such as storing packages in warehouse can be assigned either to individual robots or collaborative teams of robots with enhanced capabilities (Ceroni and Nof 1999). The challenging problem in a collaborative multi-robot system is the Multi-Robot Task Allocation (MRTA) problem, especially when it comes to heterogeneous, unreliable robots equipped with different types of sensors and actuators (Khamis, Hussein, and Elmogy 2015). This problem can be seen as finding the task-to-robot assignment to achieve the overall system goals. The following challenges, therefore, need to be addressed when designing a solution to an MRTA problem (Parker 1999): (1) How to assign a set of tasks to a set of robots? (2) How the robot teaming is coordinated efficiently and reliably? (3) How to make the robot teams adapt autonomously to dynamic changes in the environment?

To systematically address and solve the challenges, there are several collaborative tools designed for operation, knowledge, and information sharing (Durugbo 2016; Zhong et al. 2015). Task Administration Protocol (TAP) (e.g. Nof et al. 2015; Tkach, Edan, and Nof 2017), HUB-CI (e.g. Zhong, Wachs, and Nof 2014; Nair 2019), and Collaboration Requirement Planning (CRP) (e.g. Rajan and Nof 1996; Velásquez and Nof 2008) are among key tools that shown promise results. TAP is the workflow optimisation protocol for managing a multi-agent system. HUB-CI, as extended in this work to optimise and harmonise agent actions across domains, is a tool designed for cyber-augmented collaboration between physical and virtual agents, while previous HUBs were limited to virtual interactions between agents (McLennan and Kennell 2010). Lastly, CRP is a hierarchical decision-support strategy for the collaborative multi-robot system. CRP is the process of generating a consistent and coordinated global execution plan for a set of tasks to be completed by a multi-agent system based on the task collaboration requirements and interactions (Rajan and Nof 1996).

While there is considerable work relating to multi-robot collaboration, the design of the warehouse to support automation 5.0 has been far less studied. To support automation 5.0, a new approach that can incorporate new technologies to improve service level and minimise operational cost is necessary. Hence, the C2W is first defined for supporting future supply chain systems. Then, an original approach based on the combination of TAP, HUB-CI, and CRP is proposed for improving and collaborating operations in C2W.

3. Problem description: the cyber collaborative warehouse (C2W)

The main operations in a warehouse are package storage and retrieval (Bartholdi and Hackman 2008). Storage is the process of putting-away items to storage locations; Retrieval is the process of items picking process from their storage location. Because of the symmetry, in this article, only storage operations are discussed.

Moreover, the objectives of the C2W are not only to store all the packages with minimum cost but to provide flexibility for responding to new requests. Multiple robots with different capabilities are selected to work in C2W as it allows more flexibility and cost-effectiveness at the same time. In C2W, new information is collected continuously via IoT/IoS devices. Also, packages differ by their relative priorities. For example, higher priority packages can have more critical time constraints than regular priority packages.

For the rest of the article, the following model is used: The C2W contains two unique types of robots: R_1 andR_2. Hence, there are three possible robot teaming options:R_1, R_2, and $R_1 + R_2$. Moreover, to reflect the reality, R_1, R_2, and $R_1 + R_2$ have different cost and capability to store packages. Two types of C2W operations are (1) Planned operation and (2) Unplanned operation, categorised based on whether the arrival of an operational request is planned or unplanned. Also, five package types in C2W are distinguished based on either robot or team's ability to accomplish them. The following sections explain the operational types and task types as well as cost/time associated with them.

3.1. C2W operational types

Planned operation and Unplanned operation are the two types of operations in C2W. Because of the dynamic environment in warehouses, not every operation can be scheduled in advance. There is a high potential to receive additional requests during a normal operational process. Hence, the warehouse must be able to deal with each unplanned request. The details for each operational type are as follows.

(1) *Planned operation*. Under normal operation, the operation starts with a receipt of the task list, indicating the type of package, storage location, and priority. The task list is a list of packages to store, usually received in advance according to truck schedule and material planning. The type of package information specifies the handling procedure, indicating which type and number of robot(s) are required. The storage location is the destination of the package within the warehouse. Naturally, the farther the distance of the storage location from the receiving dock, there will be higher operational cost and longer time required. Lastly, the priority value of a given package indicates its priority of storage. The packages with higher priority need to be processed earlier than others.

(2) *Unplanned operation*. During a storage process, the system may receive additional operations, called Unplanned Requests (UR). The UR must be integrated into the current plan to complete all tasks. Typically, re-optimising the entire array of the remaining tasks along with the UR would provide the best results. However, given that the re-optimising process usually comes with additional cost and time, the C2W should have the decision support systems in place to ensure that an entire schedule re-optimisation is not required.

3.2. C2W package types

In the operational process (either Planned or Unplanned operation), there are five possible types of packages based on specific robot handling capabilities, which are explained as follows.

Type 1. No-bottleneck: The simplest package, which can be executed by either R_1, R_2 or the collaborative team, $(R_1 + R_2)$. This is the most common package type in warehouses.

Type 2. Robot 1 is critical: R_1 is necessary for storing this package. The storage can either be completed by R_1 or by the collaborative team, $(R_1 + R_2)$.

Type 3. Robot 2 is critical: Similar to Type 2, R_2 is necessary for storing this package. The storage can either be completed by R_2 or by the collaborative team, $(R_1 + R_2)$.

Type 4. Mandatory: The package requires both robots $(R_1 + R_2)$ working but cannot be accomplished by either individual robot. Example: Large or heavy package(s) which cannot be carried by either robot alone, but can be carried when they work collaboratively.

Type 5. Optional: The storage of this package can be executed by either R_1 or R_2 individually, but not collaboratively as a team. Examples: Package(s) where the storage location has space constraints such as narrow aisles, where either single robot can path.

3.3. Operational cost and time

The cost (and time) to execute each operation by the robots or their collaborative teams is defined by the Collaboration Requirement Matrix (CRM). As an assumption, individual robot execution cost (and time) is strictly cheaper (and slower) than when executed by a team. CRM contains c_{ij} which represent cost for package i stored by robot (or team) j. For package types 1, 2, and 3, if a robot team stores a package, the operational cost is the total operational costs executed by each individual robot, which is strictly higher.

4. Proposed methodology

In this section, the approach, called Collaboration Requirement Planning protocol for HUB-CI (CRP-H), is discussed. The CRP-H improves from work by Dusadeerungsikul et al. (2019). The objective of CRP-H is to increase warehouse's performance by applying cyber collaboration to the system for synchronising system agents. The proposed approach is as follows.

4.1. Cyber collaborative warehouse system architecture

The C2W system has three main agents; human operators, robots, and warehouse shelves (indicated by IoT/IoS devices). The system receives input from human operators via a spatial-visual programming software called VIPO (Huang et al. 2020). VIPO allows human operators to input information to the system in the spatial context within an interface, which can minimise the learning time. Also, with VIPO, human (as an intelligence agent) can fill in missing information (Dusadeerungsikul and Nof 2019) for packages which contain incomplete data such as priority or location to store. In such cases, it is up to human operators to use their expertise to provide missing data for the packages, e.g. appropriating priority, updating storage location, and add packages to the queue once the information is complete. An output from VIPO is a computer script that indicates the type of package, location to store, and priority. The output will feed to HUB-CI, which can manage tasks in real-time and decide how to maximise system performance.

HUB-CI maintains CRP-H, which has two main modules; Optimiser (CRP-I) and Harmoniser (CRP-II). The Optimiser is responsible for assigning a package to a robot or a robot team (one to one or one to many matching). The Harmoniser takes care of sequencing and

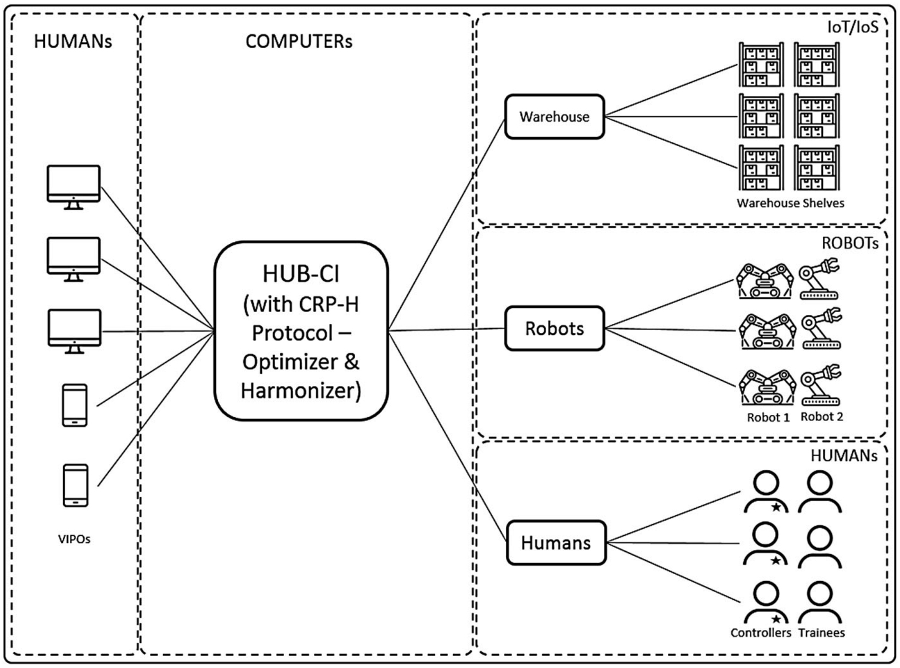

Figure 1. Cyber collaborative warehouse system architecture.

scheduling a robot or robot team to store packages under dynamically changing constraints. After the inputs are processed by HUB-CI, utilising CRP-H, the generated plan will be distributed to warehouse shelves, robots, and human users. The system architecture is presented in Figure 1.

4.2. CRP-H protocol design

For effective coordination, a Collaboration Requirement Planning protocol for HUB-CI, called CRP-H, is developed in this section. The CRP-H is the workflow optimisation and collaboration protocol in HUB-CI for package allocation and storage scheduling/sequencing. Figure 2 presents the CRP-H and its components.

As mentioned before, CRP-H has two modules, Optimiser (CRP-I) and Harmoniser (CRP-II). The following section will describe each part of CRP-H in detail.

The objectives of a C2W system are (1) Store given packages with the lowest total operational cost (solved by package allocation); (2) Minimise the total weighted completion time; (3) Minimise makespan. The second and third objectives are solved by robot schedule. Also, considering that the system can receive unexpected/emergency requests (UR) which can often impact the current schedule, it is advantageous to minimise total weighted completion time as the objective function since the system will complete the higher priority package earlier. Thus, if there is a new request (i.e. UR) during operation, it will create a lesser impact on the overall performance.

4.2.1. Optimiser (CRP-I)

To achieve the first objective of C2W (store packages with the lowest total operational cost), the Optimiser (CRP-I) is applied. The Optimiser aims to minimise the total operation cost for planned operations by assigning package(s) to the suitable agent. The mathematical model for Optimiser is presented as follows.

4.2.1.1. Mathematical model for optimiser.

Let

$$c_{ij} = \textit{cost for package } i \textit{ is performed by a robot or team of robots } j$$

$$x_{ij} = \begin{cases} 1 \textit{ if package } i \textit{ is performed} \\ \quad \textit{by a robot or team of robots } j \\ 0 \textit{ otherwise} \end{cases}$$

$$R = \textit{Number of robot or team of robots}$$

$$N = \textit{Number of packages}$$

$$N_p = \textit{Number of packages of type } p$$

$$K = \textit{Maximum loading factor} = \max\left(\frac{N}{R}, \max_p(N_p)\right)$$

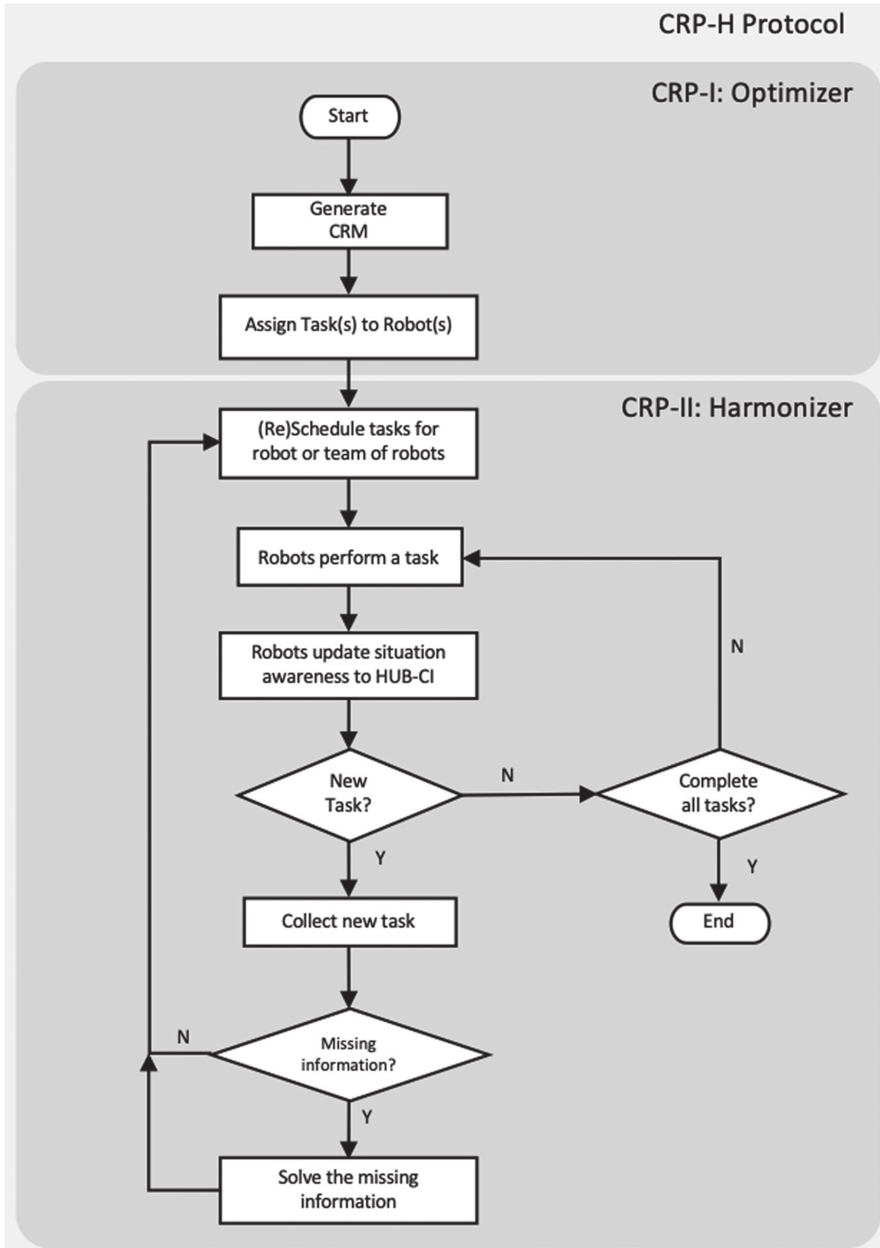

Figure 2. CRP-H protocol.

Objective function

$$min\ z = \sum_i \sum_j c_{ij} x_{ij} \tag{1}$$

s.t.

$$\sum_j x_{ij} = 1; \forall i \tag{2}$$

$$\sum_i x_{ij} \leq K; \forall j \tag{3}$$

$$x_{ij} \in \{0, 1\}, \forall i, j$$

$$i = 1, 2, \ldots, N$$

$$j = 1, 2, \ldots, R$$

As mentioned before, the objective of the Optimiser is to minimise the total operational costs presented in Equation (1). c_{ij} is the operational cost defined in the CRM. The CRM will update continuously with the new information received from IoT/IoS agents and human operators.

The first constraint, Equation (2), ensures that all tasks will be executed by a robot or robot team. The second constraint, Equation (3), ensures that the robot or robot team will not be overloaded. Because in CRP-H, not only total operational cost is considered, time to complete each task is also important. K which is the maximum loading factor helps the Optimiser balance tasks among robot and robot team. Without Equation (3), it might be a scenario that one robot is overloaded and affects the second and third objectives of CRP-H (minimise total weighted completion time and minimise makespan).

In the planned operation, the input data (package, location, and priority) are received in advance. In addition, Optimiser requires relatively larger computational power and processing time than Harmoniser (CRP-II) due to the volume of data being processed; hence it is initiated ahead of the actual operation.

The Optimiser output is the assignment of the package(s) to a robot or robot team with respect to cost (c_{ij}). Two types of assignments from the Optimiser are (1) Collaborative assignment package and (2) Non-collaborative assignment package. A collaborative assignment package is for ($R_1 + R_2$); a non-collaborative assignment package is for a single robot (R_1 or R_2). The output, however, does not indicate the sequence of tasks for each team of agents. Therefore, the Harmoniser is necessary.

4.2.2. Harmoniser (CRP-II)

The second phase of CRP-H is Harmoniser. The Harmoniser has the main objective of sequencing given tasks for each robot or robot team, to minimise makespan (C_{max}) and total weighted completion time. The objective is selected to ensure that in case of unexpected conditions such as UR, robot delays, operation conflicts, and errors, the high priority packages are scheduled as early as possible to minimise losses due to re-schedule. The mathematical model for Harmoniser is presented as follows.

4.2.2.1. Mathematical model for harmoniser.

The math ematical model for Harmoniser is as follows.

Let

$$y_{it} = \begin{cases} 1 \ \textit{if package i starts at time t} \\ 0 \ \textit{otherwise} \end{cases}$$

$$p_i = \textit{proces}\sin g \textit{ time of package } i$$

$$w_i = \textit{priority of package } i$$

$$C_{max} = \sum_i p_i$$

Objective function

$$\min z = \sum_i \sum_t w_i(t + p_i) y_{it} \tag{4}$$

s.t.

$$\sum_{t=0}^{C_{\max}-1} y_{it} = 1; \forall i \tag{5}$$

$$\sum_{i=1}^{N} \sum_{u=(\max(t-p_i),0)}^{t-1} y_{iu} = 1; \forall t \tag{6}$$

$$y_{it} \in \{0, 1\}; \forall i, t$$

$$i = 1, 2, \ldots, N$$

$$t = 0, 1, 2, \ldots, C_{\max} - 1$$

The objective function, Equation (4), ensures the minimisation of the total weighted completion time. p_i is the processing time of package i. In practice, p_i can be calculated from travel time plus storing time of package i. w_i is the priority of package i which can be set according to the importance of the package. The first constraint, Equation (5), ensures that any package has a single starting point (any t). Also, C_{max} in Equation (5) can be calculated in advance as an input of the model. The second constraint, Equation (6), ensures only one package can be executed at a time t. By solving the above mathematical model, a schedule that minimises the total weighted completion time is generated. Furthermore, the minimisation of makespan is ensured by having a non-delay schedule. Note that, in the Harmoniser phase, cost (c_{ij}) is not considered. The reason is because costs are considered during the Optimiser phase. The Harmoniser aims to sequence packages for each robot or robot team.

The solution, however, needs a relatively large computational power and time due to the nature of NP-hard problems when the problem size is scaled up. Therefore, an algorithm for Harmoniser, called Collaborative Robots Scheduling (CRS), is introduced.

4.2.2.2. Collaborative robots scheduling (CRS) algo rithm.

The Harmoniser provides real-time control and adaptation based on new information. Harmoniser is executed at the local agent(s) levels to provide efficient responsiveness. Solving the mathematical model presented, which contains a large number of decision variables ($t \times y_{it}$) requires powerful computational power for real-time execution. The Collaborative Robots Scheduling (CRS) algorithm is developed to ensure that an optimal schedule is achieved locally at a relatively lower computational burden.

The CRS utilises the advantages of the Weighted Shortest Processing Time first (WSPT) algorithm that yields the optimal solution for the total weighted completion time problem. In addition, CRS helps combine multi-levels of WSPT to provide the optimal makespan. The CRS algorithm is defined as follows.

CRS algorithm	
Step 1.	Schedule collaborative assignment packages, according to WSPT (called collaborative schedule).
Step 2.	Schedule non-collaborative assignment packages, according to WSPT (called non-collaborative schedule).
Step 3.	Combine the collaborative schedule with the non-collaborative schedule by setting the release time of the non-collaborative schedule equal to the makespan (C_{max}) of the collaborative schedule. In other words, the starting time of a non-collaborative schedule is after all collaborative schedules are completed.
Step 4.	Terminate algorithm.

From the CRS, it can be shown that the algorithm can yield (1) Optimal schedule for collaborative robots and (2) Optimal makespan of the collaborative robot schedule as follows.

Theorem 4.1: *The optimal schedule for collaborative robots: The CRS yields an optimal total weighted completion time for each robot and robot team.*

Proof: see Appendix 1 ■

Theorem 4.2: *The guarantee optimal makespan for collaborative robot schedule*

The optimal makespan C^*_{max} of the system is

$$C^*_{\max} = \sum_{k \in R_1+R_2} p_k + \max\left(\sum_{i=R_1} p_i, \sum_{i'=R_2} p_{i'}\right) \tag{7}$$

Proof: see Appendix 2 ■

5. Experiments and results

To validate the newly designed protocol and algorithm, three simulations experiments are constructed by coding in MATLAB. The first experiment presents the system performance during the normal condition; all packages arrive in advance with priority 1 (the lowest priority) to 10 (the highest priority) and no UR. The second experiment shows the situation where UR happens randomly during the operation. Lastly, the third experiment presents a situation where some critical information is missing.

5.1. Experiment 1: performance analysis in planned operations

Experiment 1 compares the designed protocol with a standard warehousing procedure (baseline) for the normal operations. The system description is as follows.

5.1.1. System description

Agents: Two robots (R_1 and R_2) and one robot team ($R_1 + R_2$); $R = 3$.

Tasks: 100 packages with different priorities are available immediately at the start (N = 100).

Operation procedures: Two types of operation procedures.

(1) CRP – H: The designed protocol
(2) Baseline procedure: Randomly assign packages to robot or robot team with First Come First Serve (FCFS) scheduling.

5.1.2. Experiment 1 results

The results from random package to agent assignment with FCFS rule are used as the design baseline. Table 1 and Figure 3 summarise two performance metrics based on the results of 100 operation runs. The average total operation cost and the average weighted completion time of the CRP-H are 11.84% and 37.11% lower than the baseline, respectively. At significance level 0.05, two sample standard t-tests ($p < 0.0001$) confirm that the CRP-H significantly outperforms the baseline both in terms of total operation cost and total weighted completion time. The difference of 37% in the average total weighted completion time is meaningful because it reflects the influence of the task priority (weight) on the total weighted completion time under the CRP-H logic on the scheduling and prioritising. It is interesting to note that even though in the experiment there is only 1% lower makespan, as both schedules from CRP-H and the baseline are non-delay schedules, CRP-H yields a statistically significant lower makespan ($p < 0.0001$) than the baseline. Importantly, the makespan from the CRP-H has met the guaranteed optimal makespan from Theorem 4.2.

Next, to see the impact of savings from the CRP-H, the cost and time savings are calculated, as shown in Equations (8) and (9).

Cost saving

$$\Delta Cost = C_{Basedline} - C_{CRP-H} \tag{8}$$

where $\Delta Cost$ = Cost saving, $C_{Basedline}$ = Total operation cost from the baseline procedure, C_{CRP-H} = Total operation cost from CRP-H.

Time saving

$$\Delta Time = W_{Basedline} - W_{CRP-H} \tag{9}$$

where $\Delta Time$ = Time saving, $W_{Basedline}$ = Total weighted completion time from the baseline procedure, W_{CRP-H} = Total weighted completion time from CRP-H.

Figure 4 shows the cost and time savings at a various number of packages. The results show that cost savings

Table 1. Results of Experiment 1 based on 100 tasks in the queue.

	CRP-H	Baseline	Difference %
Average Total Operational Cost ($)	4916.02 (235.97)	5576.10 (267.65)	11.84%∗
Average Total Weighted Completion Time (sec)	1.61×10^5 (2.01×10^4)	$2.56\times10^5(2.87\times10^4)$	37.11%∗
Average Makespan (sec)	889 (15.91)	901 (18.57)	1.33%∗

Note: Standard deviations are given in parentheses; ∗ – Statistically significant at ($p < 0.0001$).

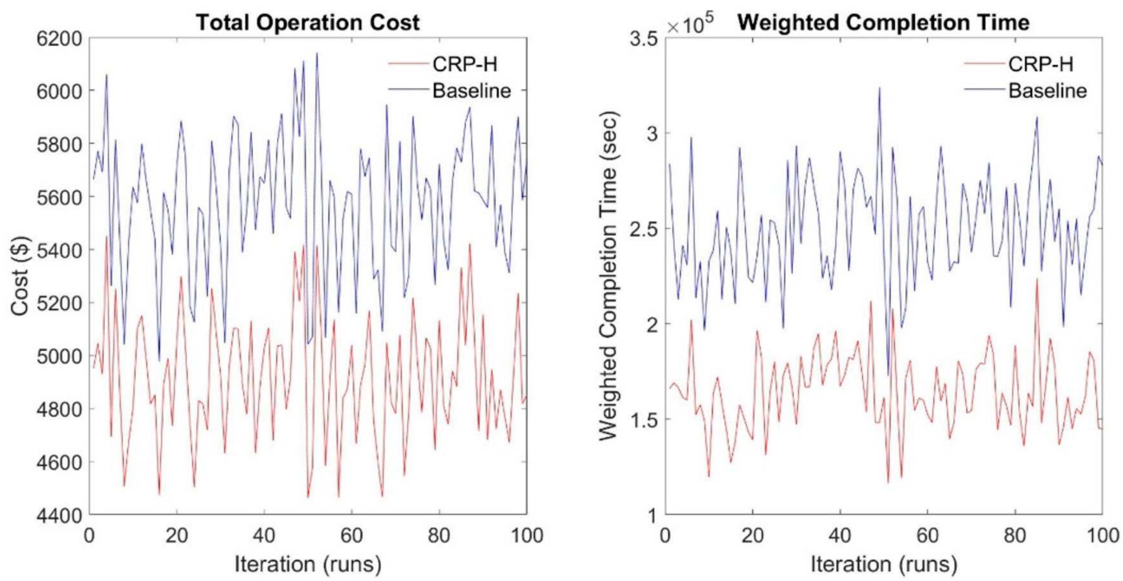

Figure 3. Experiment 1: Performance metrics at 100 packages.

are linearly proportional ($R^2 = 0.997$) to the increasing number of packages, and, interestingly, the time savings are in a polynomial, non-linear relationship ($R^2 = 0.981$) with the number of packages. The possible explanation is that while cost saving impacts independently from other packages, time saving creates a snowball effect, which impacts the later package in queue. In other words, with time saving, the following packages will be completed faster, creating greater time saving to the overall system. The results also validate the robustness of the CRP-H in a multi-robot task allocation problem, as both cost and time savings are always positive.

5.2. Experiment 2: performance analysis with unplanned operations

Experiment 2 compares the performance of the CRP-H with the baseline when unplanned operations happen (e.g. UR). The designed protocol is tested against the standard warehousing procedure to deal with unplanned operations (Baseline). The system description is as follows.

5.2.1. System description

Agents: Two robots (R_1 and R_2) and one robot team ($R_1 + R_2$); $R = 3$.

Tasks: 90 packages available immediately with different priorities ($N = 90$) and 10 packages randomly added after the operation begins (UR = 10).

Operation procedures: Two types of operation procedures.

(1) CRP – H: The designed protocol will re-schedule a new task, according to Harmoniser.
(2) Baseline procedure: Utilising Optimiser for the package assignment with FCFS for UR.

5.2.2. Experiment 2 results

In Experiment 2, 10 packages are added to the queue after the Optimiser has been executed. The goal of the experiment is to understand the impact of dynamic changes in the C2W environment. Table 2 and Figure 5 compare the performance metrics from the results of 100 operation runs. Results show that, at a significant level of 0.05, the average operational cost of CRP-H and the weighted completion time of CRP-H is lower than the baseline. In addition, CRP-H also provides a statistically significant lower makespan than the baseline. Note that the makespan by the CRP-H in experiment 2 is larger than the guaranteed bound from Theorem 4.2. The makespan from the theorem captures only planned operation while the actual makespan reflects the URs.

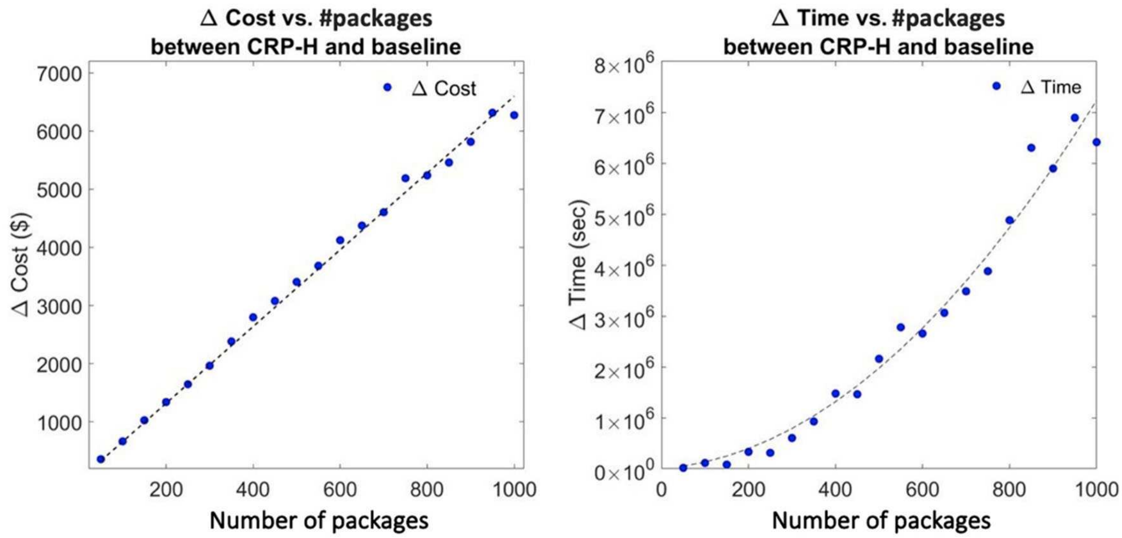

Figure 4. Experiment 1: Cost and time savings at different number of packages.

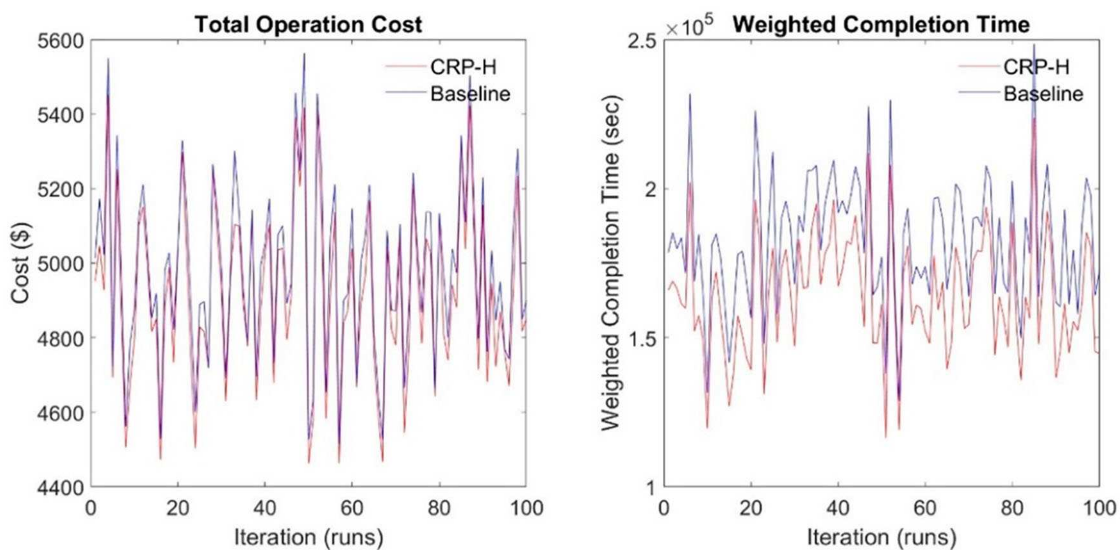

Figure 5. Experiment 2: Performance metrics of 100 packages, with 10 packages added while robots are in operation.

5.3. Experiment 3: performance analysis with human operator

In experiment 3, human experts are involved in the decision-making process. Human is an intelligence agent who can add or revise missing information such as package priority or wrong location via the user interface. The CRP-H with human integration features is validated against the non-human procedure. The system description is as follows.

5.3.1. System description

Agents: Two robots (R_1 and R_2) and one robot team ($R_1 + R_2$); $R = 3$.

Tasks: 100 packages available immediately with 10 packages without priority ($N = 100$).

Table 2. Results of Experiment 2 based on 100 packages, with 10 packages added while robots are in operation.

	CRP-H	Baseline	Difference %
Average Total Operational Cost ($)	4983.22 (219.01)	5284.31 (234.49)	5.70%$*$
Average Total Weighted Completion Time (sec)	1.69×10^5 (1.91×10^4)	1.88×10^5 (2.11×10^4)	10.11%$*$
Average Makespan (sec)	897 (7.32)	911 (8.75)	1.54%$*$

Note: Standard deviations are given in parentheses; $*$ – Statistically significant at ($p < 0.0001$).

Table 3. Results of Experiment 3 based on 100 packages with 10 packages without priority.

	CRP-H	Baseline	Difference %
Average Total Operational Cost ($)	5071.68 (239.46)	5437.81 (267.65)	6.73%$*$
Average Total Weighted Completion Time (sec)	1.61×10^5 (2.00×10^4)	1.93×10^5 (2.12×10^4)	16.58%$*$
Average Makespan (sec)	893 (6.98)	899 (11.35)	0.67%$*$

Note: Standard deviations are given in parentheses; $*$ – Statistically significant at ($p < 0.001$).

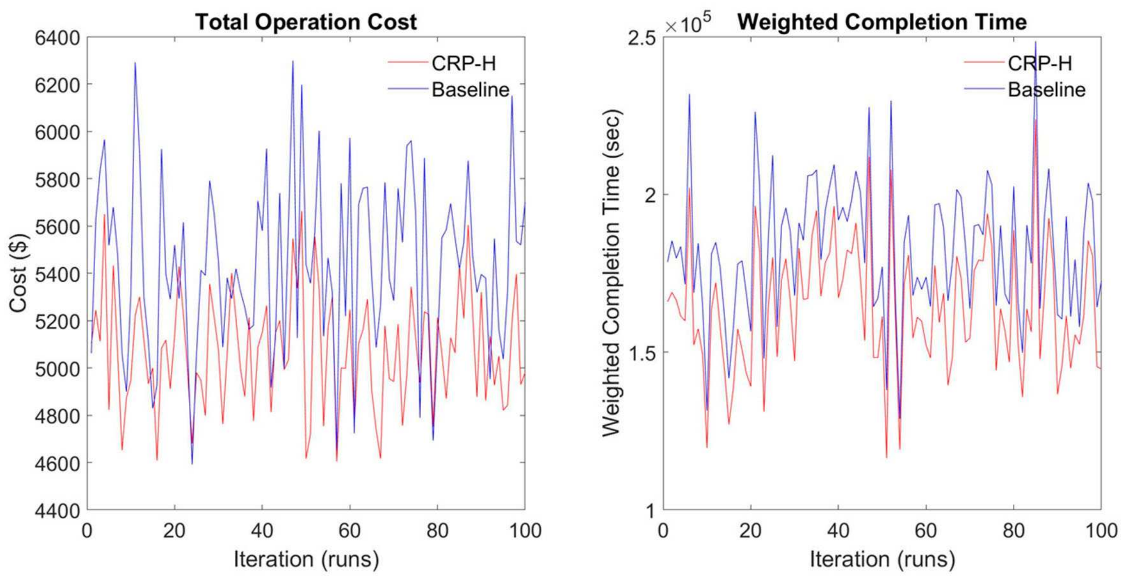

Figure 6. Experiment 3: Performance metrics at 100 packages with 10 packages without priority.

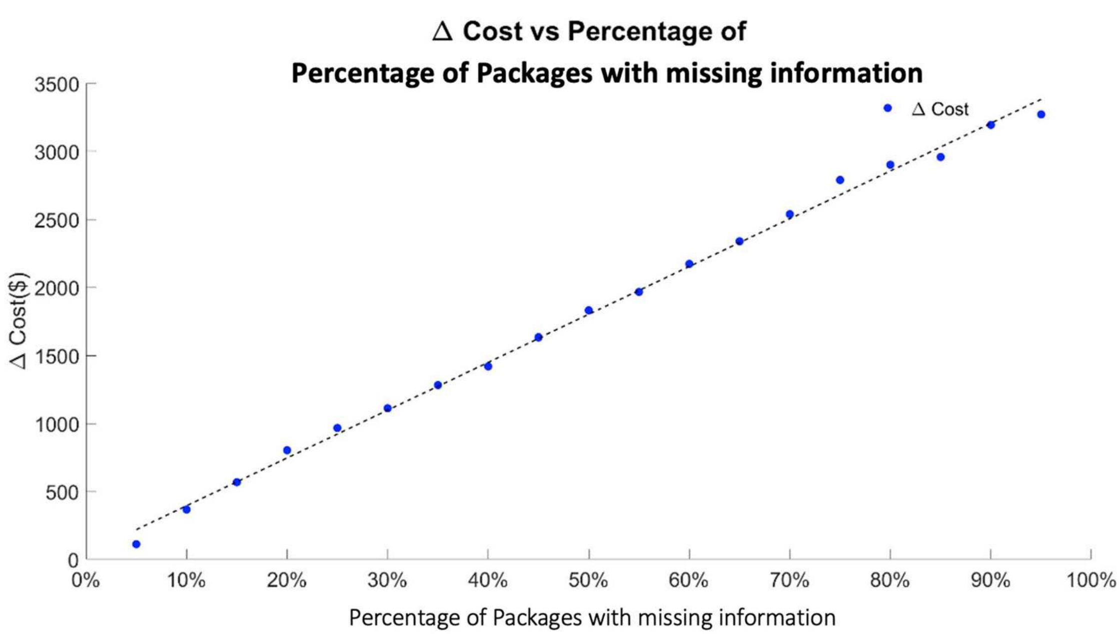

Figure 7. Cost differences between CRP-H and baseline when percentage of packages with missing information increases.

Operation procedures: Two types of operation procedures.

(1) CRP – H: The designed protocol with a human operator.
(2) Baseline procedure: The designed protocol without a human operator, packages with missing information will be given a default priority (no priority).

5.3.2. Experiment 3 results

In Experiment 3, a certain percentage of packages arrive with missing information, such as priority or deadlines. The experiment studies the impact of human intervention in the packages allocation and sequencing problem. For this experiment, the human agent assigns the missing priority of packages during the Harmoniser process, compared to the baseline state where these tasks are added at the end of the queue with minimal priority. The results, shown in Table 3 and Figure 6, suggest that CRP-H significantly improves both the total operational cost and weighted completed time compared to the baseline ($p < 0.0001$). An additional human involvement cost for assigning priority of packages is assumed to be zero since the involvement can be considered part of routine human tasks. On the other hand, if there is an additional human involvement cost, the cost should be lower than a cost threshold (the difference between total operation cost from CRP-H and baseline) to be considered as a cost-effective situation.

Figure 7 shows the cost difference at various percentages of packages with missing information. The results show the cost differences are linearly proportional ($R^2 = 0.998$) to the increasing percentage of packages with missing information, suggesting that with an increasing percentage of packages with missing information, CRP-H improves cost reduction via augmented stabilisation of performance.

6. Conclusion and discussion

Fast improvement and changes of technologies have challenged emerging, future factories, warehouses, and service systems. To be ready for automation 5.0, the key preparation is to design a multi-agent system to work smoothly and effectively, taking advantage of cost-effective new advanced technologies.

In this article, a new protocol for HUB-CI called CRP-H is designed, developed, and validated to address such challenge. By studying a C2W, two operation types and five package types are defined to represent the warehouse operations. CRP-H, which aims to address operation in C2W, is composed of two main parts: Optimiser (CRP-I) for package(s) to robot(s) assignment optimisation; and Harmonisation (CRP-II) for storage sequencing and scheduling. Optimiser operates at the global level, which requires high computational time. In contrast, Harmoniser operates at the local agent level, which has only limited computational power, responding to dynamic changes in the warehouse, and hence CRS algorithm is introduced.

Two theorems are presented in this article. Theorem 4.1, the optimal schedule for collaborative robots, proves that the CRS algorithm provides the optimal schedule for collaborative robots in the warehouse. Theorem 4.2, the guaranteed optimal makespan for a collaborative robot schedule, provides the optimal makespan of the collaborative robot schedule.

Based on the three experiments and their assumptions, observations indicate that the CRP-H protocol can deliver superior performance in terms of total operation cost, makespan, and total weighted completion time compared to a common practice in today's operations. Lower operational cost is enabled by the use of Optimiser, which optimally assigns package(s) to the robot(s). Moreover, total weighted completion time and makespan are minimised because of the Harmoniser, which can update the execution schedule in real-time, based on the ongoing cyber collaborative connectivity with IoT/IoS devices' timely information. In addition, the experimental results also show that with human operators, the system becomes more robust via augmented stabilisation, given that the versatility of human decision-making is appropriately applied in cases of missing data.

Considering practical implications, the CRP-H can save both money and time for a company without additional investment. Optimiser can be performed in the background before operations begin. Harmoniser requires a relatively small computational power due to the simple rules of the algorithm and can support the system by adjusting the operation during the ongoing process in response to UR. Additionally, with the design allowing humans involved, the system can overcome unexpected situations, such as missing data, with minimal incremental cost. On the other hand, if the additional cost due to human operators (e.g. specialised worker cost) exceeds the difference between CRP-H and the baseline case, then it is not attractive to maintain human operators. It, however, must be mentioned that if the human input were not included, the tasks would either incur hidden costs from erroneous storage or cost from retrieving packages from the distant locations, both of which are undesired outcomes.

Based on this work, and to extend it, researchers will pursue the following directions.

(1) Research machine learning algorithms to improve CRP-H. Learning can adapt to new data collected

during operation and can deal with more complex situations and tasks in a warehouse.

(2) Explore situations where two robots are not sufficient. A warehouse might be too large for two robots, or tasks may require more than two robots to be completed effectively. Hence, a more complex protocol to synchronise and optimise multi-robots operation workflow is necessary.

(3) Apply other simulation tools such as Gazebo and ROS to include more realistic scenarios of a warehouse. Also, examine the more complex situation, such as when conflicts and errors are prone to occur and cause major delays in certain operations, or when the agents have boundaries such as limited operating hours or limited energy.

Acknowledgements

Thanks to A. S. Nair for the initial background research. An earlier short version of this article was presented at ICPR-25, Chicago, IL, in August 2019 (Dusadeerungsikul et al. 2019).

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

The research reported here has been partially supported by the Purdue University PRISM Center for Production, Robotics, and Integration Software for Manufacturing & Management, and the National Science Foundation's Award [1839971] FW-HTF: Collaborative Research: Pre-Skilling Workers, Understanding Labor Force Implications and Designing Future Factory Human-Robot Workflows Using a Physical Simulation Platform.

Notes on contributors

Puwadol Oak Dusadeerungsikul received a Ph.D. degree in Industrial Engineering from Purdue University in 2020. He also received M.S. degree in Industrial Engineering at Georgia Institute of Technology in 2015. Currently, he is a postdoctoral researcher at the PRISM Center, Purdue University. His research interests include network routing problems, scheduling problems, logistics and supply chain management, and operations research.

Xiang He received the B.S. in Materials Engineering from McGill University. He attended M.S. Industrial Engineering program at Purdue University. His research interests include production system design and supply chain management.

Maitreya Sreeram is a graduate student from Purdue University, having completed his M.S. in Industrial Engineering. His research is focused primarily on collaboration in human-in-the-loop systems. His background is in Mechanical Engineering, which he has a Bachelor's degree from IIT Hyderabad.

Shimon Y. Nof, Ph.D., D.H.C., is Professor of Industrial Engineering, Purdue University, and held visiting positions at MIT and at universities in Chile, EU, Hong Kong, Israel, Japan, Mexico, and Taiwan. He is the Director of the NSF – and industry-supported PRISM Center (Production, Robotics and Integration Software for Manufacturing & Management) linked with PGRN (PRISM Global Research Network); recent Chair of the IFAC Coordinating Committee 'Manufacturing & Logistics Systems'; recent President, Secretary General, and current Board member of IFPR (International Federation of Production Research), Fellow of IFPR and of the IIE (Institute of Industrial Engineers), and inaugural member of Purdue's Book of Great Teachers. He is co-inventor of five automation patents; the author, coauthor and editor of thirteen books, including the *Handbook of Industrial Robotics* 1st and 2nd editions, the *International Encyclopedia of Robotics*, (both winners of the 'Most Outstanding Book in Science and Engineering' Award,) *Information and Collaboration Models of Integration*, *Industrial Assembly*, and *Springer Handbook of Automation*.

References

Bartholdi, J. J., and S. T. Hackman. 2008. *Warehouse & Distribution Science: Release 0.89*. Atlanta: Supply Chain and Logistics Institute.

Brogårdh, T. 2007. "Present and Future Robot Control Development—An Industrial Perspective." *Annual Reviews in Control* 31 (1): 69–79. doi:10.1016/j.arcontrol.2007.01.002.

Cao, Y., F. Li, K. Huo, Z. Xu, W. Zhong, and K. Ramani. 2019. "V. Ra: An in-situ visual authoring system for robot-IoT task planning with augmented reality." In Proceedings of the 2019 on Designing Interactive Systems Conference, San Diego, CA, USA.

Ceroni, J. A., and S. Y. Nof. 1999. "Robotics Terminology." In *Handbook of Industrial Robotics*, edited by S. Y. Nof, 1259–1317. New York: Wiley.

Devadasan, P., H. Zhong, and S. Y. Nof. 2013. "Collaborative Intelligence in Knowledge Based Service Planning." *Expert Systems With Applications* 40 (17): 6778–6787. doi:10.1016/j.eswa.2013.06.048.

Durugbo, C. 2016. "Collaborative Networks: A Systematic Review and Multi-Level Framework." *International Journal of Production Research* 54 (12): 3749–3776. doi:10.1080/00207543.2015.1122249.

Dusadeerungsikul, P. O., and S. Y. Nof. 2019. "A Collaborative Control Protocol for Agricultural Robot Routing with Online Adaptation." *Computers & Industrial Engineering* 135: 456–466. doi:10.1016/j.cie.2019.06.037.

Dusadeerungsikul, P. O., M. Sreeram, X. He, A. Nair, K. Ramani, A. J. Quinn, and S. Y. Nof. 2019. "Collaboration

Requirement Planning Protocol for HUB-CI in Factories of the Future." *Procedia Manufacturing* 39:218–225.

He, Z., V. Aggarwal, and S. Y. Nof. 2018. "Differentiated Service Policy in Smart Warehouse Automation." *International Journal of Production Research* 56 (22): 6956–6970. doi:10.1080/00207543.2017.1421789.

Huang, G., P. S. Rao, M. Wu, M. Wadhawan, S. Y. Nof, S. Chandrasegaran, K. Ramani, and A. J. Quinn. 2020. "VIPO: Spatial-Visual Programming with Functions for Robot-IoT Workflows." Proceeding of ACM CHI 2020.

Khamis, A., A. Hussein, and A. Elmogy. 2015. "Multi-robot Task Allocation: A Review of the State-of-the-Art." In *Cooperative Robots and Sensor Networks 2015*, edited by A. Koubâa, and J. R. Martínez-de Dios, 31–51. Cham: Springer International Publishing.

Liu, G., W. Yu, and Y. Liu. 2006. "Resource Management with RFID Technology in Automated Warehouse System." *Proceeding of ACM CHI 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Honolulu, Hawai'i, USA.

McLennan, M., and R. Kennell. 2010. "HUBzero: A Platform for Dissemination and Collaboration in Computational Science and Engineering." *Computing in Science & Engineering* 12 (2): 48–53. doi:10.1109/MCSE.2010.4.

Nair, A. S. 2019. "A HUB-CI model for networked telerobotics in collaborative monitoring of agriculture greenhouses." M.S. Thesis, Purdue University.

Nof, S. Y. 1999. "Robot Ergonomics: Optimizing Robot Work." In *Handbook of Industrial Robotics*, edited by S. Y. Nof, 2nd ed., 603–644. New York: Wiley.

Nof, S. Y. 2007. "Collaborative Control Theory for e-Work, e-Production, and e-Service." *Annual Reviews in Control* 31 (2): 281–292.

Nof, S. Y., J. Ceroni, W. Jeong, and M. Moghaddam. 2015. *Revolutionizing Collaboration Through e-Work, e-Business, and e-Service*. Berlin: Springer.

Parker, L. E. 1999. "Cooperative Robotics for Multi-Target Observation." *Intelligent Automation and Soft Computing* 5 (1): 5–19. doi:10.1080/10798587.1999.10750747.

Rajan, V. N., and S. Y. Nof. 1996. "Cooperation Requirements Planning (CRP) for Multiprocessors: Optimal Assignment and Execution Planning." *Journal of Intelligent and Robotic Systems* 15 (4): 419–435. doi:10.1007/BF00437605.

Tate, P. 2018. "Factories of the Future: Five Expert Perspectives: Factories of the Future." Accessed April 11, 2019. https://portal.manufacturingleadershipcouncil.com/2018/02/06/factories-future-five-expert-perspectives/

Tkach, I., Y. Edan, and S. Y. Nof. 2017. "Multi-sensor Task Allocation Framework for Supply Networks Security Using Task Administration Protocols." *International Journal of Production Research* 55 (18): 5202–5224. doi:10.1080/00207543.2017.1286047.

Velásquez, J. D., and S. Y. Nof. 2008. "A Best-Matching Protocol for Collaborative e-Work and e-Manufacturing." *International Journal of Computer Integrated Manufacturing* 21 (8): 943–956. doi:10.1080/09511920802014896.

Wang, H., S. Chen, and Y. Xie. 2010. "An RFID-Based Digital Warehouse Management System in the Tobacco Industry: a Case Study." *International Journal of Production Research* 48 (9): 2513–2548. doi:10.1080/00207540903564918.

Zhong, H., R. R. Levalle, M. Moghaddam, and S. Y. Nof. 2015. "Collaborative Intelligence – Definition and Measured Impacts on Internetworked e-Work." *Management and Production Engineering Review* 6 (1): 67–78. doi:10.1515/mper-2015-0009.

Zhong, H., J. P. Wachs, and S. Y. Nof. 2014. "Telerobot-enabled HUB-CI Model for Collaborative Lifecycle Management of Design and Prototyping." *Computers in Industry* 65 (4): 550–562. doi:10.1016/j.compind.2013.12.011.

Appendices

Appendix 1: Proof of Theorem 4.1 (The optimal schedule for collaborative robots)

At each particular robot (and robot team), the problem becomes $1||\sum w_i C_i$ problem.

Let package i has priority w_i, the time to complete the storage operation p_i, and task i' has priority $w_{i'}$, the time to complete the storage operation $p_{i'}$ so that $\frac{w_i}{p_i} < \frac{w_{i'}}{p_{i'}}$.

Suppose schedule S', which starts at time t, contradicted with the WSPT rule. There will be at least one pair of packages such that $\frac{w_i}{p_i} < \frac{w_{i'}}{p_{i'}}$. However, package i is placed before i' (Figure A8). The total weighted completion time of S' is $\gamma + w_i(t + p_{i'}) + w_{i'}(t + p_i + p_{i'})$ where γ is the total weighted completion time of all packages except i and i'.

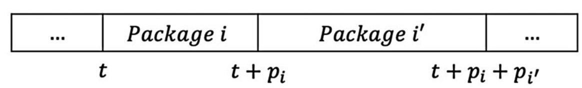

Figure A8. Schedule S'.

If task i and i' are interchanged and produce schedule S (Figure A9), the total weighted completion time of S is $\gamma + w_i(t + p_{i'}) + w_{i'}(t + p_i + p_{i'})$. Note that γ of S and S' are the same as all packages, except i and i' remain unchanged.

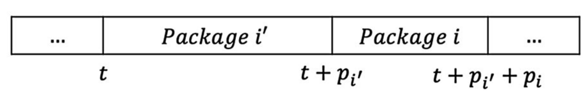

Figure A9. Schedule S.

Because $\frac{w_i}{p_i} < \frac{w_{i'}}{p_{i'}}$, then $w_i(t + p_{i'}) + w_{i'}(t + p_i + p_{i'}) > w_i(t + p_{i'}) + w_{i'}(t + p_i + p_{i'})$ and the new schedule, which follows WSPT, has a strictly lower objective function value.

Moreover, since a non-collaborative assignment package requires a strictly longer processing time (by definition of the package) and have strictly lower priority (because it uses only a single robot), to schedule the storage operation optimally, the non-collaborative schedules must be released after the completion of the collaborative schedule. ■

Appendix 2: Proof of Theorem 4.2 (The guarantee optimal makespan for collaborative robot schedule)

There are two sub-schedules for the makespan; makespan form collaborative schedule and from non-collaborative schedule.

First, consider a robot team ($R_1 + R_2$). Regardless of the sequence of storage operation, the makespan of the robot team equals to the total processing time of all storage operations assigned to the robot team ($\sum\limits_{k \in R_1 + R_2} p_k$).

Next, consider a single robot agent (R_1 and R_2). Each robot can work individually, and the makespan of all single robot agent become maximum total completion time of all robots ($\max(\sum_{R_1} p_i, \sum_{R_2} p_{i'})$). Also, the earliest time that each robot can start working is immediately after the collaboration schedule is done. Therefore, the optimal makespan for the robotic schedule is:

$$C^*_{\max} = \sum_{k \in R_1 + R_2} p_k + \max\left(\sum_{i = R_1} p_i, \sum_{i' = R_2} p_{i'}\right) \qquad \blacksquare$$