Received: 26 July 2021 | Revised: 29 April 2022 | Accepted: 15 June 2022

DOI: 10.1002/rob.22106

RESEARCH ARTICLE

WILEY

A strawberry harvest-aiding system with crop-transport collaborative robots: Design, development, and field evaluation

Chen Peng | Stavros Vougioukas | David Slaughter | Zhenghao Fei | Rajkishan Arikapudi

Davis, Department of Biological and Agricultural Engineering, University of California, Davis, Davis, California, USA

Correspondence

Chen Peng, Department of Biological and Agricultural Engineering, University of California, Davis, Davis, CA, USA.
Email: penchen@ucdavis.edu

Funding information

U.S. Department of Agriculture; National Institute of Food and Agriculture

Abstract

Mechanizing the manual harvesting of fresh market fruits constitutes one of the biggest challenges to the sustainability of the fruit industry. During manual harvesting of some fresh-market crops like strawberries and table grapes, pickers spend significant amounts of time walking to carry full trays to a collection station at the edge of the field. A step toward increasing harvest automation for such crops is to deploy harvest-aid collaborative robots (co-bots) that transport empty and full trays, thus increasing harvest efficiency by reducing pickers' non-productive walking times. This study presents the development of a co-robotic harvest-aid system and its evaluation during commercial strawberry harvesting. At the heart of the system lies a predictive stochastic scheduling algorithm that minimizes the expected non-picking time, thus maximizing the harvest efficiency. During the evaluation experiments, the co-robots improved the mean harvesting efficiency by around 10% and reduced the mean non-productive time by 60%, when the robot-to-picker ratio was 1:3. The concepts developed in this study can be applied to robotic harvest-aids for other manually harvested crops that involve walking for crop transportation.

KEYWORDS

agricultural robotics, mobile robots, optimization, systems design

1 | INTRODUCTION

Mechanizing the manual harvesting of fresh market fruits constitutes one of the biggest challenges to the sustainability of the fruit industry. Depending on the commodity, labor for manual harvesting can contribute up to 60% of the yearly operating costs per acre (Bolda et al., 2016). Additionally, recent studies indicate that the farm labor supply cannot meet demand in many parts of the world because of socioeconomic, structural, and political factors (Charlton et al., 2019; Guan et al., 2015). Robotic harvester prototypes are being developed and field-tested for high-volume, high-value crops such as apples (Silwal et al., 2017), kiwifruit (Williams et al., 2020), sweet pepper (Arad et al., 2020), and strawberries (Xiong et al., 2020). However, the developed robots have not, to date, successfully replaced the judgment, dexterity, and speed of experienced pickers at a competing cost; the challenges of high fruit picking efficiency and throughput remain largely unsolved (Bac et al., 2014).

As an intermediate step to full automation, mechanical labor aids have been introduced to increase worker productivity by reducing workers' non-productive times. For example, orchard platforms eliminate the need for climbing ladders and walking to unload fruits in bins (Baugher et al., 2009; Fei & Vougioukas, 2021). Autonomous vehicle prototypes have been developed to assist in bin management in orchards (Bayar et al., 2015; Ye et al., 2017), to reduce the need for forklift operators.

In strawberry production, mobile conveyors have been introduced to reduce the time pickers spend walking to get the produce from the plants to the designated loading stations and return to resume picking (Rosenberg, 2003). However, such conveyors are specific to strawberries and cannot be adapted to other crops. Furthermore, their adoption has been very slow, partly because of their questionable profitability, due to high purchase cost and limited efficiency gains. Two reasons for their inadequate efficiency are: (1) row-turning in the field is time-consuming because of their large size, and (2) because conveyors move slowly to accommodate slower pickers, often resulting in underutilization of faster pickers (Rosenberg, 2003).

The walking time to carry harvested crops constitutes a significant non-productive part of the harvesting cycle for several fresh-market crops, like strawberries (Figure 1), raspberries, blackberries, and table grapes. For strawberries, walking time has been measured to reach up to 22% of the total harvest time (Khosro Anjom et al., 2018); higher inefficiencies are often reported, anecdotally.

In this study, a collaborative robotic system (aka, co-robotic system) was investigated to assist in such harvesting operations by transporting trays, with strawberries as a case study. During the proposed robot-aided harvesting, each picker walks inside a furrow, harvests ripe fruits, and puts them in a standard-sized tray located on a special instrumented cart (Figure 2a), in the same way as in all-manual harvesting. These carts are equipped with load cell sensors to measure the weight of the tray, and a GNSS (Global Navigation Satellite System) module to record the geodetic locations of the carts (Khosro Anjom et al., 2018). The cart sends data wirelessly in real-time to a computer in the field (we refer to it as the "operation server"). Software running on the server predicts when and where a tray will become full (Khosro Anjom & Vougioukas, 2019). A full tray results in a tray-transport request to the scheduling software running on the server, which dispatches a team of crop-transport robots to serve those requests. The robots travel between the collection station and pickers to bring empty trays (Figure 2b). The picker walks a small distance to the robot, loads the full tray, gets an empty tray, and pushes a button to command the robot to travel back to the collection station (Figure 2c).

Given the large sizes of commercial harvesting crews (e.g., strawberry harvesting in California involves crews of 20–40 people) and the expected cost and complexity of transporting and deploying one robot per worker, our work explored an operational scenario in which a crew of pickers is served by a smaller team of robots. The direct implication of this approach is that the robots are a *shared resource*, with each robot serving multiple pickers.

Given that robots travel at relatively low speeds for safety purposes (in our case, 0.5–1.5 m/s), and that the distance to a picker can be up to 100 m long, robot sharing among the workers may introduce non-productive waiting delays between the time when a tray becomes full, and a robot arrives. Peng and Vougioukas (2020) used simulations of strawberry harvesting to show that, if the time and place of each picker's next-tray transport request are known, predictive dynamic robot scheduling—dispatch robots before trays become full—can be used to minimize the expected waiting time introduced by the robots and increase the efficiency of the crew.

However, the deployment of harvest-aid robots in real-world commercial harvesting settings must address two additional important issues. First, the time and location of a worker's next-tray transport request is not known, but can only be predicted with some degree of uncertainty (Khosro Anjom & Vougioukas, 2019). Therefore, a *stochastic scheduling approach that incorporates uncertainty* must be used; otherwise, large waiting times may arise and the crew's mean efficiency when robots are used can become worse than all-manual harvesting (Bertsimas & Van Ryzin, 2017). Second, minimizing the expected waiting time of the entire crew (equivalently, maximizing expected crew efficiency) is not enough. The robot team should

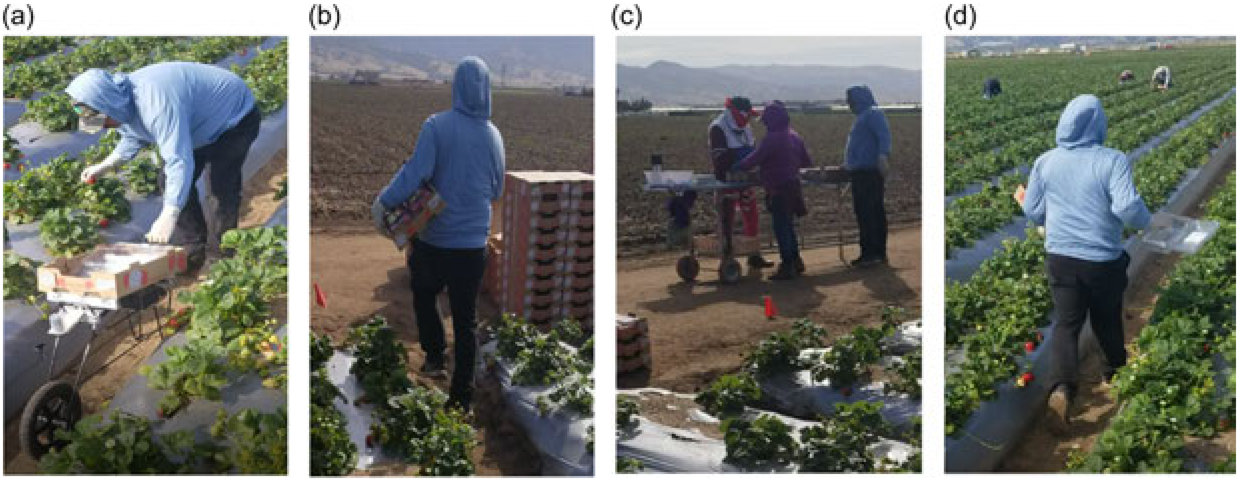


FIGURE 1 The working cycle of manual harvesting in the open commercial field: (a) the picker is picking strawberries inside the furrow, and places them in clamshells inside a tray laying on a cart; (b) the picker transports the full trays to the collection station on the headland; (c) the picker loads the trays in the collection station; (d) the picker takes an empty tray back to resume picking. [Color figure can be viewed at wileyonlinelibrary.com]

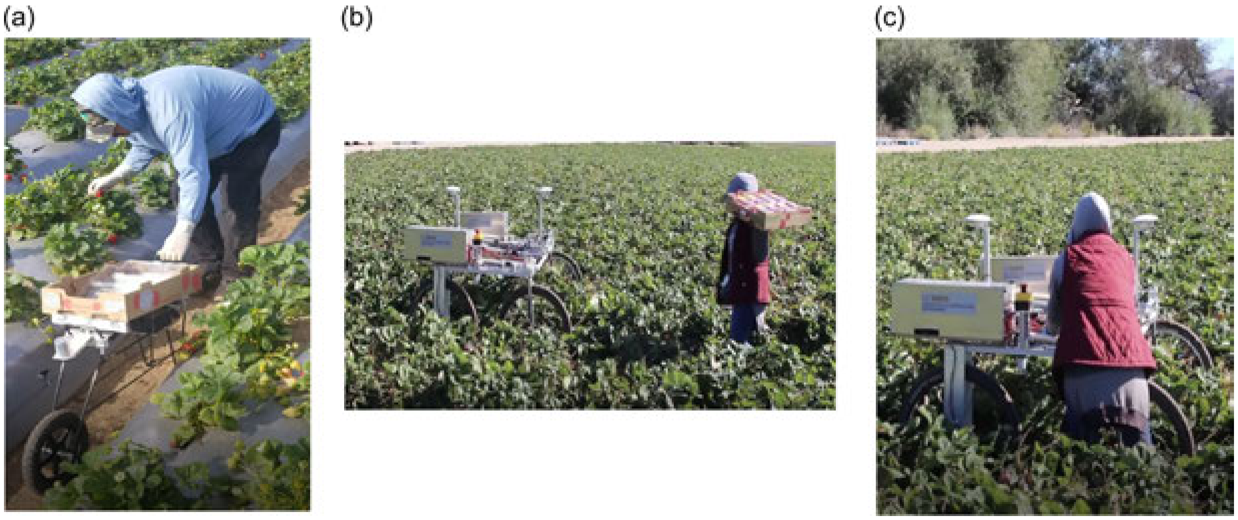


FIGURE 2 The working cycle of co-robotic harvesting in the open commercial field: (a) the picker is picking strawberries inside the furrow in the same way as in all-manual harvesting; (b) the picker walks a small distance to the serving robot; (c) the picker loads the trays on the robot. [Color figure can be viewed at wileyonlinelibrary.com]

also aim that *each worker benefits—or at least is not hindered—by robot operation, every time they need to transport a tray*; otherwise, pickers may not agree to work with the robots, as individuals may lose income, although the crew—as a whole—will collect more trays.

The main contributions of this paper are that it develops an integrated approach to address both the above issues and evaluates the approach in extensive simulation experiments and in real-world commercial strawberry harvesting. In particular:

(1) A stochastic scheduling framework (Multiple Scenario Approach—MSA) was adopted, and algorithms were developed to apply the MSA to online stochastic scheduling of harvest-aid robots. The MSA is more of a 'framework' to solve stochastic scheduling problems under uncertainty, rather than a specific algorithm. Depending on the application, one has to develop the specific "scenario solutions" (optimization routines) and the "consensus function" that work for the specific problem at hand (our contributions). The MSA samples the uncertainties that are inherent in the prediction of transport requests and generates many possible scenarios for the spatiotemporal distribution of the requests. It then computes an optimal or near-optimal schedule for each (deterministic) scenario and synthesizes a final schedule—that is executed by the robots—that is near-optimal under the exiting uncertainties.

(2) The capability of the robotic system to reject a tray-transport request was added to a harvesting simulator and to the scheduling problem formulation. Request rejection results in no robot being sent to a picker if the picker's expected waiting time is longer than the predicted time it would take this picker to walk and deliver the tray. This way, each picker can only benefit from the operation of the robots.

(3) Harvest simulation experiments—using a calibrated simulator—were performed to select key system parameters and evaluate our approach, and field experiments were done during commercial strawberry harvesting to assess system performance.

The rest of the paper is organized as follows. Section 2 surveys the previous work on robotic harvest-aids and the related work on in-field logistics. In Section 3, the physical implementation of the co-robotic system is presented, with just enough detail to provide the reader with an understanding of the system's architecture and technical implementation. Section 4 presents the implementation of the MSA and the mathematical modeling of predictive stochastic scheduling in the context of co-robotic harvesting, and how the resulting stochastic scheduling problem is solved in an online fashion. Section 5 presents the experimental design and the analysis of the experimental results from simulations, and from real-world commercial strawberry harvesting using the developed system. Finally, Section 6, presents the main conclusions of this study and suggests directions for future research.

2 | RELATED WORK

To our knowledge, the concept of using a team of collaborative robots to carry trays during harvesting was initially proposed by Vougioukas et al. (2012). In their work, the problem of carrying trays from different locations in an orchard was formulated as a Vehicle-Routing Problem and solved using mixed-integer programming. This concept was explored further in the context of the "FRAIL-Bots" research project, under the National Robotics Initiative (USDA-REEIS, 2013), and a stochastic strawberry harvesting simulator was developed by Jang (2018) that used finite state machines (FSMs) to model picker and robot activities, and reactive scheduling for robot control. In this study, random variables, such as the time required to fill a tray, the picker walking speeds during picking and carrying trays were identified from human pickers' harvesting activities and used to compute state transitions and when and where trays become full. *Reactive scheduling* policies (robots travel to pickers once trays are full), such as First-Come-First-Served, Shortest and Longest-Processing-Time were implemented, and their resulting harvest efficiencies were evaluated for various picker-robot ratios using Monte Carlo simulation. The term "reactive" describes an operation in which a resource is allocated to a task only after the scheduler receives the task request (Blazewicz et al., 2019). In the context of tray-transport robots, reactive scheduling refers to the situation where a robot starts traveling to a picker when—or after—the picker's tray becomes full. Similarly, Das et al. (2018) used FSMs with stochastic picking parameters in a discrete event simulator (DES) to evaluate the economic viability and scalability of strawberry tray-transporting robots, when greedy or load-balancing reactive dispatching policies are used.

In contrast to reactive scheduling, *predictive* scheduling policies incorporate information about the current state and the future demand (Ritzinger et al., 2016). In the context of harvesting, "future demand" refers to knowledge about when and where a worker's currently used tray will become full, giving rise to a tray-transport request. If the time and location are known in advance, a robot can be dispatched—and start moving toward that location—before the tray becomes full; hence, waiting times due to robot travel can be reduced or eliminated.

The collaborating workers and robots can be thought of as a dynamic system with a "state" that has discrete and continuous components. The DES-based approaches mentioned above capture the discrete components of this state (e.g., a worker is "picking" or "transporting"; a robot is "idle" or "transporting"), but do not model the continuous components that include worker and robot positions in the field, and the weights of the trays. As a result, these discrete-event harvesting models cannot be used for online *predictive* scheduling. This realization has led to the development of a *hybrid systems* model for harvesting (Seyyedhasani et al., 2020a). Inside each operating state, difference equations with stochastic parameters were used to model the filling of the trays (mass transfer) and the positions of the pickers (motion); the stochastic parameters were identified from human pickers' harvesting activities and the simulation was calibrated against real-world harvest operations.

Based on a hybrid-systems simulator, Seyyedhasani et al. (2020b) showed that when tray-transport robots were scheduled reactively, picker waiting time was reduced when the robot-to-pickers ratio increased. However, above a certain ratio, adding more robots did not reduce further the waiting time. The reason is explained as follows. In reactive scheduling, a robot that is idle at the collection station—after delivering its previous tray—will start traveling to a picker when the picker's request is received. However, the average distance of a picker from the collection station can be large, in the order of 30 m or more (Peng & Vougioukas, 2020). So, even if there are always several idle robots waiting to serve a picker and hence there is no delay due to lack of available robots, it will take a significant amount of time for a robot to travel to the picker and get the tray. This time cannot be reduced by adding robots. However, if request prediction is possible, the robot can start its travel earlier and thus it will be much closer to the picker when the tray is actually ready to be transported.

Peng and Vougioukas (2020) used a hybrid systems simulator to explore predictive scheduling. In their work, a predictive scheduler was developed that was assumed to know exactly the times and locations of the tray-transport requests. The scheduler used exact and heuristic methods to compute the optimal schedule. The results showed that when the ratio of robots to pickers was high enough, predictive scheduling—with perfect knowledge—increased the harvesting efficiency of all-manual harvesting much more than reactive scheduling (24% vs. 15%).

In reality, the locations and times of tray-transport requests contain uncertainty because of stochastic picker work-rate and varying—unknown—yield density (Khosro Anjom & Vougioukas, 2019). Uncertainty can be detrimental for predictive scheduling algorithms that assume perfect information (Bertsimas & Van Ryzin, 2017; Blazewicz et al., 2019). Hence, a dynamic stochastic scheduling algorithm is needed to account for the prediction uncertainty and improve the scheduling performance.

Harvest-aid robots that carry trays bear similarity to the use—and scheduling—of autonomously guided vehicles (AGVs) in flexible manufacturing systems (FMS). A typical FMS consists of work machines that feed on materials and parts and produce product components; a material handling system (MHS) that moves materials and parts, and a central control computer, which orchestrates material movements and production flow (Buzacott & Yao, 1986). A modern MHS will utilize AGVs to perform material transport operations, such as carrying materials to the machines' input buffers, removing parts from the machine output buffers, transporting them to other machines (es (for further processing), and delivering finished parts from output buffers to a collection station. Typically, the AGVs are scheduled and dispatched based on the product dispatch times of the work machines, which are stochastic, due to reasons such as machine breakdowns, stochastic processing times, and unexpected releases of high priority jobs (Ghaleb et al., 2020). Existing approaches used to solve the FMS scheduling problem include genetic algorithms, stochastic dynamic programming, integer programming, heuristic algorithm and so on (Yadav & Jayswal, 2018). Similarly, in robot-aided harvesting, the unknown and spatially variable crop distribution and the highly stochastic harvesting speeds of the human pickers result in stochastic "processing time," that is, the time it takes a picker to fill a tray (which must be transported) is stochastic. However, the harvesting problem is different, because the workers are not stationary—like work machines are—so the transport requests are stochastic both temporally, as well as spatially. Additionally, workers can transport their own trays, and therefore, the transport robots can afford to reject transport requests, something that AGVs cannot do, in an FMS setting. Hence, the scheduling paradigms for AGVs operating in manufacturing environments cannot be applied directly in our application.

The crop-transport robot operation also bears resemblances to automated warehouse logistics systems, in which humans and robots work together. Given an order for a certain product, a mobile robot is dispatched to the location of the rack that contains the product, it lifts the rack and moves it to a station where human workers pick items and place them in boxes (Weidinger et al., 2018). The robots relieve workers from walking unproductive long distances in the warehouse and increase the order processing productivity (Olsen & Tomlin, 2020). In such warehouses, many orders may arrive over short time intervals, driven by an online ordering system (Li et al., 2020). The robot scheduler must handle the dynamic product demand, while considering the stochastic and time-varying work speed of human operators, which affects the time when a robot must return to a worker to retrieve the rack and move it to its original location (Pasparakis et al., 2021; Wang et al., 2021). Wang et al. (2021) developed an online prediction system that predicts the workers' picking rates using stochastic models and schedules the mobile robots to serve them with stochastic dynamic programming; the algorithm balances picker workloads and improves the overall productivity. In automated warehouses, just like in robot-aided harvesting, the transport requests are dynamic and robot scheduling is coupled to human working speeds. However, in contrast to our problem, the dispatch destinations of the mobile robots are known/deterministic and stationary; only the return times to the workers are stochastic. In our application, both the locations of the robot destinations and the required times of arrival are stochastic, because they are coupled spatiotemporally to the pickers' stochastic harvesting activities. To handle the spatiotemporal uncertainty of transport requests, we adopted a fast and suboptimal scenario sampling-based framework, the MSA, which was originally developed for the partially dynamic vehicle routing problem with time windows (Bent & Van Hentenryck, 2004b). In the planning stage, MSA samples the uncertainties and generates multiple scenarios (possible spatio-temporal distributions) for the tray-transport requests. Given each sampled scenario, deterministic scheduling was solved quickly and computed a near-optimal schedule for the robots. At the decision instant, the actual schedule that is followed by the robots is computed from the sample scenarios using a consensus function (Bent & Van Hentenryck, 2004a) that was developed for the specific application. This schedule is the most consistent with the optimal schedules of all the sampled scenarios. This methodology can be adapted and applied to different online stochastic combinatorial optimization problems by building two case-dependent modules: a scenario sampling function to generate multiple deterministic scheduling scenarios, and a scheduling solver for each sampled deterministic scenario (Pillac et al., 2013).

3 | DESIGN OF THE CO-ROBOTIC HARVESTING SYSTEM

Strawberries are planted in rows with furrows between them to accommodate human and machine traffic (Figure 3a). The field headlands are used for collection/parking/inspection stations and traffic of people, forklifts, and trucks involved in the handling and transportation of the harvested crops. A typical harvesting block consists of approximately 80–120 rows, of about 100 m in length. Before harvesting, collection stations and empty trays are placed at one side of the field, at the headland. The crew (say, N people) start picking as a team in the first N rows on the left or right side of the block. Although there may be several collection stations, the one that is closest to the crew is the active one. To reduce the walking needed to transport full trays to the collection station, the standard harvesting practice is to have workers start picking from the middle of the field block and walk outward, toward one of its edges. Once that section of the block is harvested, the collection stations are moved to the other edge, and the other section (half of the block) is harvested. This method essentially "splits" a harvesting block into two

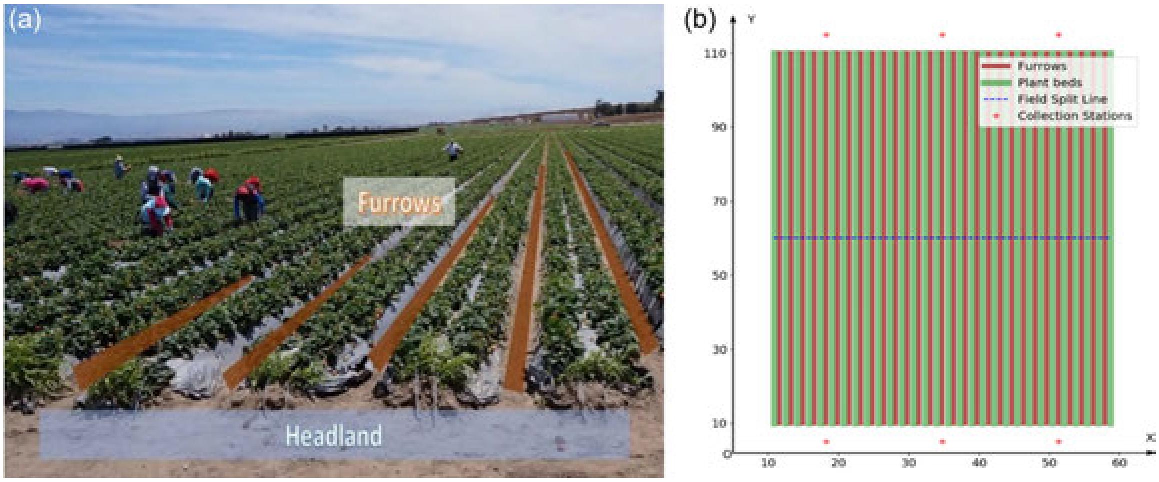


FIGURE 3 (a) Layout of a typical raised-bed strawberry field; (b) schematic figure of the strawberry harvesting field block with two sections (upper and lower); furrows; plant beds; field split line, and collection stations (Peng & Vougioukas, 2020). [Color figure can be viewed at wileyonlinelibrary.com] sections (an example is given in Figure 3b, where there is an upper and a lower block, above and below the blue dotted line, respectively). The field can be modeled using points at the edges of each furrow; two points to represent the middle line, and points for each collection station.

In the envisioned crop-transport robotic aided harvesting, each picker enters an unoccupied furrow to start picking strawberries selectively from the plants on the raised beds on each side of that furrow. When a certain fill ratio (percent of tray filled—FR) is reached (Peng & Vougioukas, 2020), the picker will press a request push-button on their instrumented cart. The button allows a picker to decide for themself if they want a robot to carry their tray. For example, if a picker prefers to walk to deliver a tray—to take a break from stooped work—they can do so. The button also helps to establish a simple communication protocol between the workers and the robotic system: a transport request is initiated and is either accepted by the system or rejected. The automated weighing system is used by the transport-request prediction module, after the button has been pressed. The scheduling system signals the picker (LEDs on their instrumented carts) if their transporting request will be served by a dispatched robot or rejected by the system. If the picker will be served, the dispatched robot starts from the active collection station, drives with an empty tray to the assigned picker's full tray location, waits for the picker to switch empty and full trays, and takes the full tray back to the active collection station, where it waits for the next dispatching. If a tray-transport request is rejected, the picker transports the full tray to the collection station, just like in manual harvesting.

The co-robotic harvest-aiding system comprises three subsystems: instrumented carts, robots (aka FRAIL-Bots: **F**ragile c**R**op h**A**rvest-a**I**ding mobi**L**e ro**bots**—USDA REEIS 2013), and an operation server. The standard carts used by the pickers weigh approximately 2.2 kg and our instrumented carts have the same form factor and are not significantly heavier, to be accepted by pickers (4 kg max). Small amounts of data from each cart are transmitted wirelessly to the field computer over distances that span typical fields (>300 m) at rates of approximately 1 Hz, using LoRa (Zourmand et al., 2019). The communication is bidirectional since the scheduler must inform pickers (by turning on a light on their cart) if their requests are rejected or will be served.

A Wi-Fi router is located at the collection station and the robots communicate in real-time over Wi-Fi with the operation server to send their state and receive dispatching commands and reference paths. The robot operation is not expected to be affected by temporal signal drops and varying latency that often arise in environments with several Wi-Fi networks and communication over long distances (this was confirmed in our field experiments). The reasons are the following. First, in our application, there is typically no other Wi-Fi network in the area, as we operate outdoors, in fields that are far away from urban infrastructure. Second, when the robots travel inside the furrows and are farther away from the collection station—and the Wi-Fi router—they do not need or use the Wi-Fi network. The robots rely on low-latency, high-speed Wi-Fi only when they travel in—and share the space of—the headland, because coordination and collision avoidance execute centrally, on the operation server. However, in the headland,

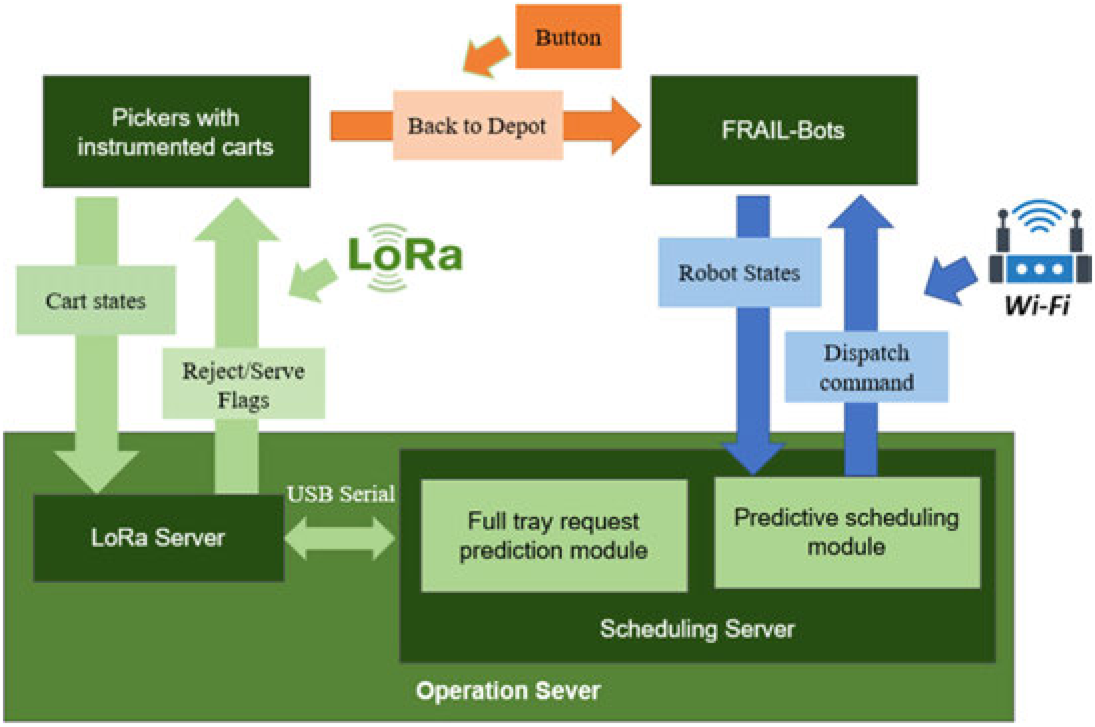


FIGURE 4 Diagrams of harvest-aiding system components and their communications [Color figure can be viewed at wileyonlinelibrary.com]

the distances between robots and the Wi-Fi router are small (a few tens of meters), and in our experiments, we did not observe any issues. (A more robust and scalable solution could be to distribute the coordination and collision avoidance computations on the individual robots.)

The complexity of the software running on the carts, robots and operation server requires a distributed software architecture that can provide real-time performance. The system architecture is shown in Figure 4.

The LoRa server, scheduling server, and FRAIL-Bots communicate with each other using a ROS network (Figure 4) that utilizes different physical layers. On the scheduling server, the tray request prediction module receives data from the cart and generates predictions of full tray requests. Given the subscribed ROS messages of robots' states and the predicted full tray requests, the predictive scheduling module calculates an optimized schedule. Then, the scheduling module publishes the dispatching commands, which are received by the available robots. After the robots arrive at the predicted full tray location, the picker will load their harvested full tray and take the empty tray from the robot. Then they press a button on the robot to command it back to the collection station. When a robot arrives back at the station, it will wait for a worker at the station to unload the full tray and replace it with an empty tray.

3.1 | Robot-aided harvesting simulator

Before deploying the actual system, a simulator is an important tool for iterating and investigating the system design. In our previous work (Peng & Vougioukas, 2020), a discrete-time hybrid systems model was developed to model and simulate the activities and motions of all agents involved in robot-aided harvesting. In this study, the model was enhanced to incorporate uncertain transport requests and tray-transport request rejections. The activities of a picker during robot-aided harvesting were classified into 14 discrete operating states/modes (Table 1), and the operations of a tray-transport robot into nine states (Table 2). The operating states of pickers and robots and the possible transitions amongst them are shown in Figure 5. Next, the changes in the picker and robot FSMs are described in detail.

In the picker FSM, pickers need to transport the trays themselves if they receive the request rejections. In FSM of robots, it may happen that the robot is dispatched to a row where the served picker cannot fill their tray and take the half-filled tray to the next unharvested row. In this case, the robot drives back to the collection station to wait for the next dispatching command.

Request rejections are integrated into the operation of pickers. When the rejection flags are received, the pickers will transport the full tray by themselves as in manual harvesting. The relevant pickers' states are “Transport-Full-Tray-Furrow,” “Transport -Full-Tray-Headland,” “Idle-In-Queue,” “Empty-Tray-Back-Headland,” and “Empty-Tray-Back-Furrow.” If the picker is served by a scheduled robot, they will wait at their full tray locations to exchange trays from the coming robot. In this case, the states after picking are “Waiting-For-Robot” and “Exchange-Tray.” The time spent between the full tray instant, and the starting instant of next tray picking is denoted as “non-productive” time.

The rest of the picker states and the dynamics inside each state are the same as in our previous work (Seyyedhasani et al., 2020a). The stochastic parameters were estimated experimentally, as in (Peng & Vougioukas, 2020).

TABLE 1 States defined to represent a picker's operating states during robot-aided harvesting

Operating state	Action
Start	A picker leaves the collection station with an empty tray in hand, to start picking.
Walk-Empty-Tray-Headland	A picker walks with an empty tray on the headland, toward an empty (unoccupied) furrow.
WALK-Empty-Tray-Furrow	A picker walks inside an empty (unoccupied) furrow with an empty tray until the field's split line is reached.
Picking	A picker is picking inside a furrow, with direction from the field split line toward the collection station.
Waiting-For-Robot	A picker waits (idle), with a full tray, for a robot to come.
Exchange-Trays	A picker takes the empty tray brought by the robot and places a full tray on the robot.
Walk-Partly-Full-Tray-Headland	A picker takes partly full tray on the headland, toward an empty (unoccupied) furrow.
Walk-Partly-Full-Tray-Furrow	A picker takes a partly full tray inside an empty (unoccupied) furrow until the field's split line is reached.
Transport-Full-Tray-Furrow	A picker takes a full tray inside a furrow toward the headland
Transport -Full-Tray-Headland	A picker takes a full tray on the headland toward the collection station
Idle-In-Queue	A picker waits in a line at the collection station to deliver her/his full tray and receive an empty tray.
Empty-Tray-Back-Headland	A picker walks in the headland—toward the last full tray furrow - carrying an empty tray, to continue harvesting.
Empty-Tray-Back-Furrow	A picker walks back to the last full tray location with an empty tray, to continue harvesting.
STOP	A picker stops picking after the last tray is picked up by a robot.

TABLE 2 States defined to represent a robot's operating states during robot-aided harvesting

Operating state	Action
Start	A robot at the collection station starts operation with no tray on it.
Available	A robot with one empty tray on it is waiting at the collection station to be dispatched to a tray-transport request.
Transp-Empty-Tray-to-Dispatch-Location	A Robot travels from a collection station—carrying an empty tray—toward the dispatched location.
Wait-At-Dispatch-Location	A robot arrives at the location of the tray-transport request and waits for the picker to finish harvesting.
Drive-To-Full-Tray-Location	A robot drives to picker's full tray location after served picker fills the full tray in its dispatched row.
Empty-Tray-Back	A robot runs back to collection stations as the served picker cannot fill the tray in its dispatched row
Exchange-Trays	A robot is idle while the picker exchanges the empty tray with a full tray.
Transp-Full-Tray-Back	A robot travels toward the collection station to deliver a full tray.
Idle-In-Queue	A robot with a full tray waits in a queue at the collection station to have its tray unloaded, and an empty tray loaded.
Stop	A robot stops its operation at the collection station after the last tray has been unloaded.

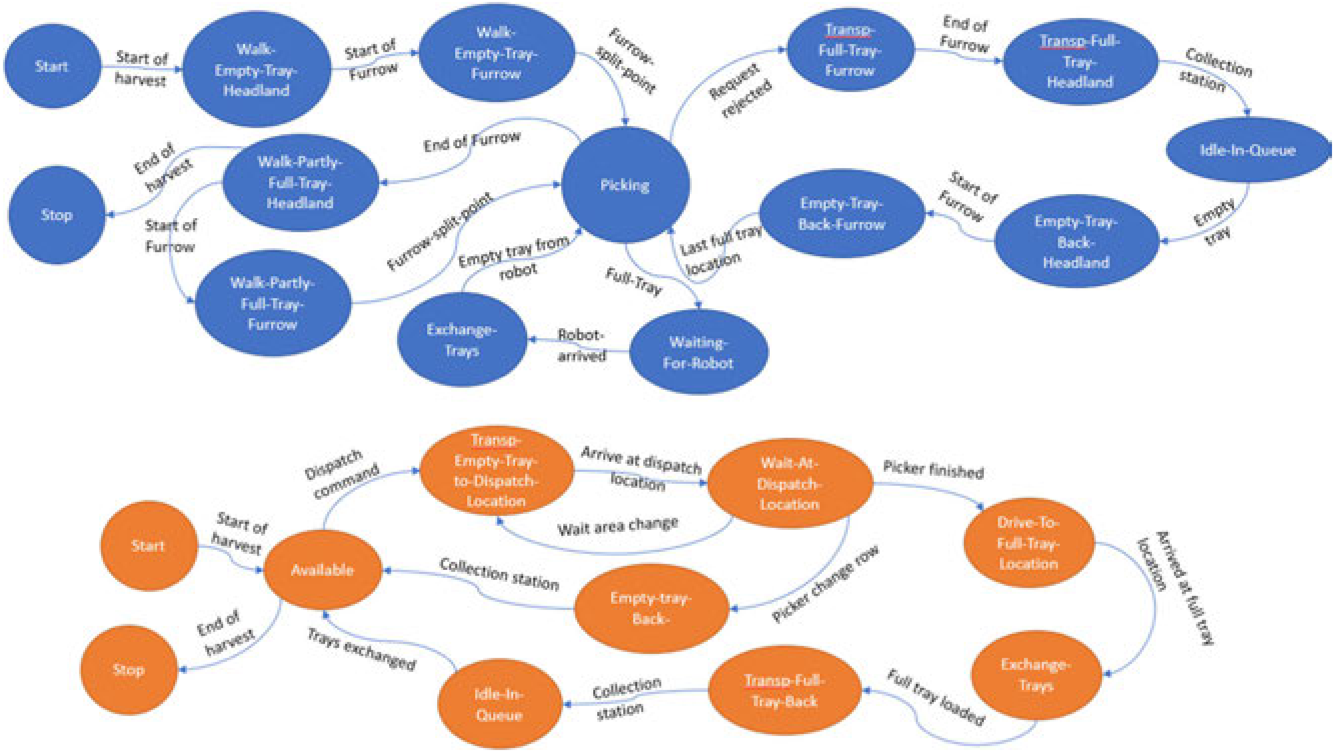


FIGURE 5 State diagram of picker states and transport robot states during human-robot collaborative harvesting [Color figure can be viewed at wileyonlinelibrary.com]

Since the predictions of tray-transport requests contain uncertainty, the exact location when the tray will become full is not known exactly. Therefore, instead of sending the robot to the predicted location, a safety distance of 5 meters from the predicted full-tray location was introduced for the robots' goal points. The pickers need to walk this small distance to load the tray onto the robot and take an empty tray back. The other states of the crop-transport robots are updated as in (Peng & Vougioukas, 2020).

3.2 | Subsystem I: Instrumented picking carts

The instrumented picking cart was modified and fabricated from a standard strawberry harvest cart (Figure 6). During harvesting, the weight of the strawberries inside the tray located on the cart is measured using two load cells underneath the supporting frame. The cart's geographical coordinates are received from a GNSS receiver module (Piksi-Multi, Swift navigation, US) that incorporates WAAS

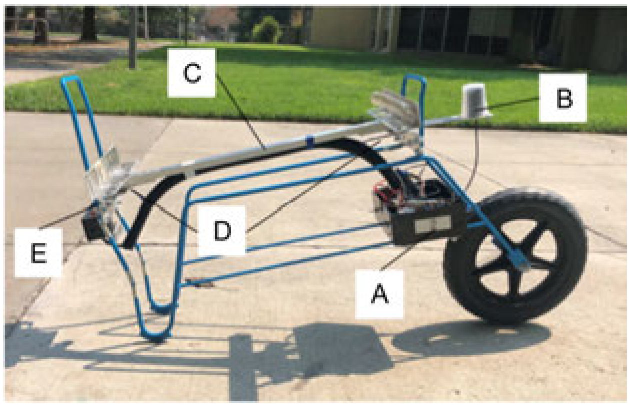


FIGURE 6 The instrumented picking cart: A. Control box with Arduino Due, LoRa module, battery, and SD card logger inside, Piksi Multi GPS unit; B. GPS antenna; C. Supporting frame on the top of load cells; D. Load cells; E. Momentary push button, yellow and red LEDs. [Color figure can be viewed at wileyonlinelibrary.com]

(Wide Area Augmentation System) corrections in real-time. An IMU (BM160, Bosch, German) is integrated on the Piksi-Multi to measure the instantaneous motion of the cart which is used to filter the weight measurements. A momentary contact button is available for the picker to notify the system that she/he wants to be served by the robot. Two LEDs (red and yellow) are used as indicators to communicate informative signals to the picker: a yellow LED turns on when the tray transport request has been assigned to a robot; the red LED turns on if the request is rejected, and the picker needs to transport the tray by themself. An SD card module installed on the control board is used to store all the sensor data during harvesting.

The GPS epoch, geodetic coordinates, tray weight, and button state are recorded in the SD card at the GPS epoch update frequency, that is, at 10 Hz. The same data are assembled into messages and are transmitted wirelessly at a rate of 1.6 Hz to the operation server, using a wireless LoRa module (RFM96W LoRa Radio; Adafruit). LoRa is a low-power wide-area network protocol, which uses license-free sub-gigahertz radio frequency bands ("LoRa", Zourmand et al., 2019).

3.3 | Subsystem II: FRAIL-Bot

Two identical crop-transport robots (aka FRAIL-Bots) were designed and built for this study (Figure 7). The bill of materials for each robot is approximately 10,000 USD; fabrication cost is not included. Constraints related to budget, available time and field deployment restricted the number of robots to two. Still, the stochastic scheduling approach and the entire system is applicable—and can be tested—with two robots, and reasonable robot-picker ratios can be achieved by using a crew size of six to eight people. The robots are designed to straddle the bed and occupy two furrows when driving inside the field. To avoid any interference of the robot with pickers in adjacent rows, pickers need to be spaced two furrows apart (with one empty furrow between them). This arrangement was acceptable by the growers and the pickers (and was used anyway during the 2020 crop year because of the COVID-19 pandemic, for physical distancing purposes).

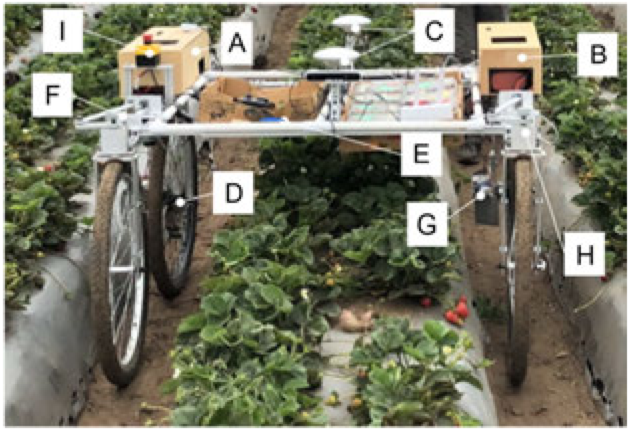


FIGURE 7 Components of FRAIL-Bot: A. control box-I with a mini-computer, battery-I, motor controllers for the two rear driving motors and two steering motors B; Control box-II with two GPS modules; C. GPS antennas; D, G. DC motors with gearbox and incremental encoders; E. Return button; F, H. Steer-driving system; I. Emergency button. [Color figure can be viewed at wileyonlinelibrary.com]

Each robot works under supervised autonomy and its collaborative operation is governed by a FSM (Peng & Vougioukas, 2020). The hardware components for the FRAIL-Bot are labeled in Figure 7. The robot weighs approximately 50 kg, and it is driven by two DC motors with gearboxes and incremental encoders attached to the rear wheels (D and E). The steering system is integrated with two screw drives and angle sensors attached on their rotation axis (F and H). Two GPS module antennas (Swift navigation, USA) are installed for getting the position and heading of the robot in open fields (C). An emergency stop button (I) is installed on the side of the robot to stop the driving system. A return button (E) on the front of the robot is used by the pickers to signal the robot that the full tray has been loaded and the robot must drive back to the collection station. The electronic devices including batteries, mini-computer (Intel NUC; Intel Inc.), driving motor controllers for rear-wheel motors, steering motor controllers, and two GPS modules are installed inside two wooden boxes (A and B).

The software architecture on each assigned FRAIL-Bot is shown in Figure 8. The FSM node first subscribes to the schedule message from the operation server including the dispatching time and dispatching location, from the operation server (explained in Section 3.4). When the dispatching time is reached, the navigation node generates the planning path from the current location of the robot to the assigned location inside the row. Given the planned path and the current robot location and heading, the path tracking module continuously outputs the control command to the motion control node, which converts the motion command of the robots to the commands of driving motors. A robot localization node fuses the subscribed sensor information, including GPS measurements, IMU data, and wheel odometry to estimate the pose and publishes the current pose of the robot into the ROS network.

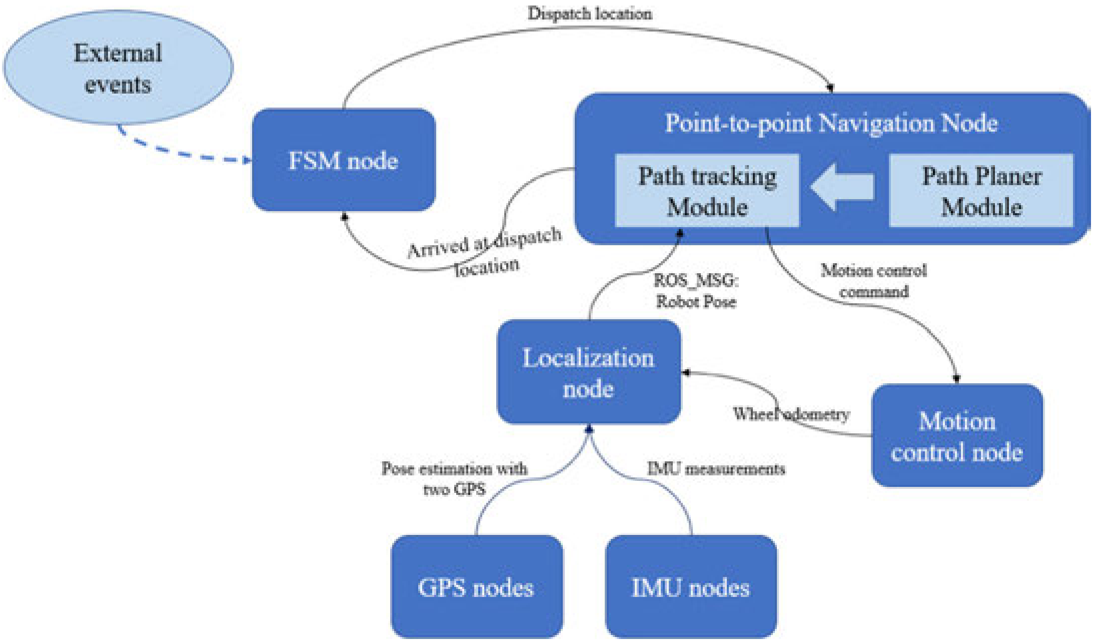


FIGURE 8 The architecture of the FRAIL-Bot software under ROS [Color figure can be viewed at wileyonlinelibrary.com]

The operation of each FRAIL-Bot is guided by a FSM introduced in our previous work (Peng & Vougioukas, 2020). The detailed implementation of each ROS node on the FRAIL-Bot in Figure 8 was elaborated in the PhD dissertation (Peng, 2021).

3.4 | Subsystem III: Operation server

The hardware of the operation server subsystem is composed of two parts: a LoRa server board and a scheduling server computer (Figure 9). The LoRa server board is connected to the server computer by a USB cable. It collects the cart states from the distributed instrumented carts in the field and publishes the received states as ROS messages. The FRAIL-Bots publish their states in the ROS network through a local Wi-Fi. The scheduling server module, running on the server computer, integrates the cart states and robot states to formulate and publish an online schedule message in the ROS network. FRAIL-Bots directly subscribe to the ROS schedule messages and execute the dispatching decisions. The LoRa server module also subscribes to the ROS schedule messages and transmits them to each instrumented cart through LoRa.

Each functional module was packaged into one ROS node shown as Figure 9. The function of each ROS node was introduced as follows. The detailed implementation of these nodes was elaborated in the PhD dissertation (Peng et al., 2020).

a. Cart-states-pub node: It receives the data of cart states from the LoRa module on the server board, converts the geodetic coordinates of each cart to field map coordinates, packages the data into ROS messages, and advertises them to the server computer through a USB cable.
b. Server-reject-sub node: It subscribes to the “serve” or “reject” flags advertised by the predictive scheduling node and is transmitted to each cart via LoRa.
c. Tray-request-prediction node: It subscribes to the cart messages from the LoRa server board and updates the prediction of picking parameters, harvesting rate, and moving speed while picking. The predictive requests are generated and published on the ROS network when a certain fill ratio of the tray is reached, and the request button is pressed by the pickers.
d. FRAIL-Bot-scheduling node: It subscribes to the robot states from the FSM node running on each FRAIL-Bot, and to the predictive tray transport requests from the tray-request-prediction node. Given these data, this node runs a stochastic predictive scheduling algorithm and advertises dispatching commands to the FRAIL-Bots, as well as the rejection and serve flags to the LoRa server boards. The online solver of this node is explained in Section 4.
e. FRAIL-Bots' coordination node: It functions as the traffic management for the robots in the shared area of the headland.
f. Operation visualization node: This node subscribes to the messages from multiple nodes of different modules for visualization of the cart/robot states. It also provides some user interface to tune the parameters during field operation.

4 | SCHEDULING OF FRAIL-BOTS UNDER STOCHASTIC REQUESTS

In our previous work (Peng & Vougioukas, 2020), the predictive scheduling of crop-transport robots assumed perfect knowledge of future requests and was modeled as a parallel machine scheduling problem (PMSP) with a release time constraint. Also, the scheduler

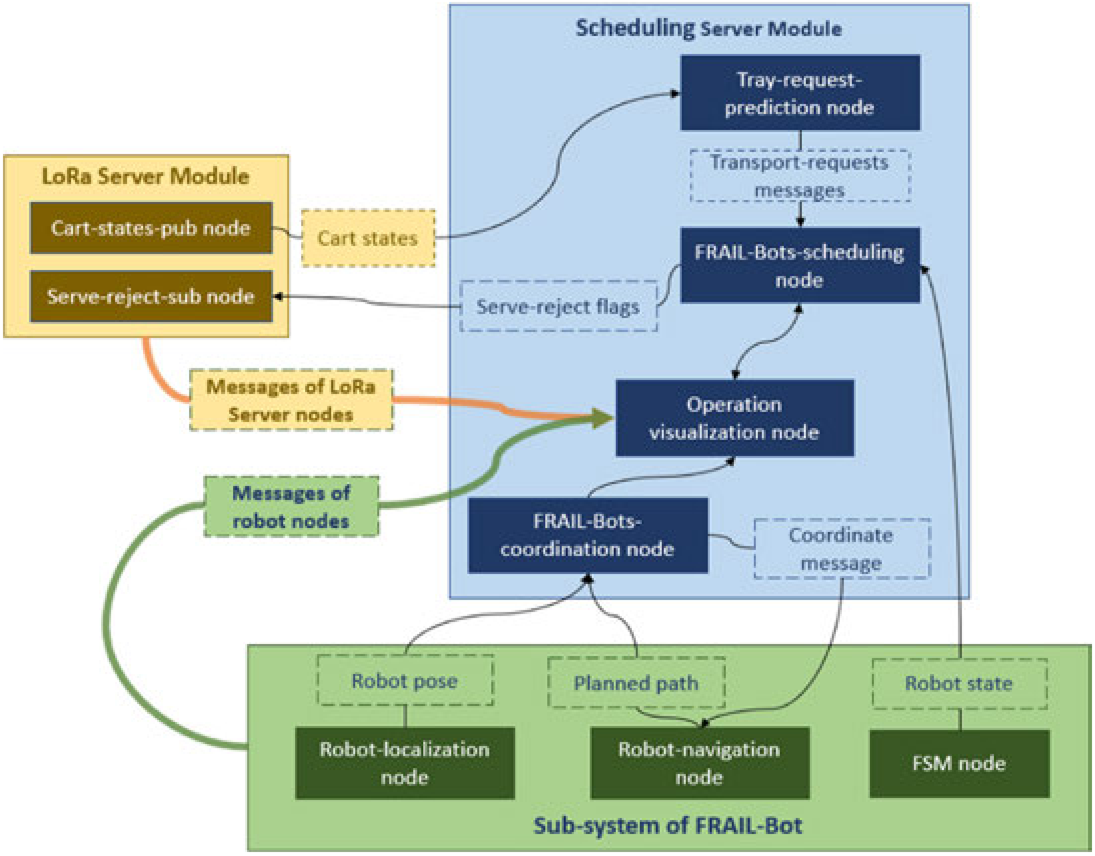


FIGURE 9 The system architecture of the operation server modularized as ROS nodes [Color figure can be viewed at wileyonlinelibrary.com]

served all the requests (no rejections), and its objective was to minimize the total waiting time of all requests.

In this study, two major changes were made to the scheduling problem: (1) the scheduler takes into consideration the inherent uncertainty in the predictions of the tray-transport requests, and (2) the scheduler is allowed to reject transport requests. The first change was necessary because the time needed to fill a tray and the corresponding distance traveled by the picker cannot be known in advance, because they are random variables that follow stochastic distributions. The second change was also necessary, because, if all requests are served, a picker may wait for a robot to arrive, even if the waiting time is longer than the time it would have taken the picker to walk and deliver the tray themselves. Such a policy would be inefficient, and not acceptable by the pickers, who are paid based on the number of trays collected; long waiting times lead to fewer trays and lower pay.

The objective of the scheduler is to minimize the expected total non-productive time of all the transport requests. The formulation of the scheduling problem is given next.

In robot-aided harvesting, each picker from a set $\mathcal{S}^{\mathcal{P}} = \{P_1,\ P_2,\ \ldots,\ P_Q\}$ of Q pickers harvests fruits in a tray that lies on a picking cart. A team of M identical transport robots $\mathcal{S}^{\mathcal{F}} = \{F_1,\ F_2,\ \ldots,\ F_M\}$ brings empty trays to the picker and carries the full tray to a collection station; the station's coordinates $\boldsymbol{L}^{s}$ are known. The robot scheduling algorithm has access to a set of predicted tray-transport requests $\mathcal{S}^{\mathcal{R}}$, where $\mathcal{S}^{\mathcal{R}} = \{\mathcal{R}_1,\ \mathcal{R}_2,\ \ldots,\ \mathcal{R}_N\}$, $0 \leq\ N \leq \mathrm{Q}$.

Let us assume that at an instant t_0, $\mathcal{R}_i$ is different from the deterministic request that contains the following (known) information: (1) a prediction distribution of the remaining time interval $\aleph(\Delta t_i^f)$ with respect to t_0 until the tray becomes full of harvested fruit, (2) the predicted moving speed along the row $\aleph(v_i^y)$ while picking, and (3) the current location of the picker $\boldsymbol{L}_i$. $\aleph(\Delta t_i^f)$ is calculated from recent measurements from the load cells and $\aleph(v_i^y)$ is computed from recent GPS readings. The main methodology for building these predictions is explained in this study (Khosro Anjom & Vougioukas, 2019). The distribution of $\aleph(\Delta t_i^f)$ and $\aleph(v_i^y)$ followed Gaussian distributions. $\aleph(\Delta t_i^f)$ was achieved by linear regression model to predict the value of full tray time at the weight of the tray capacity. Mean of $\aleph(v_i^y)$ was obtained by linear regression to estimate the slope parameter and standard deviation of $\aleph(v_i^y)$ was obtained from the standard error of the regression coefficient.

A fast and near-optimal approach, MSA (Pillac et al., 2013), was adopted and adapted to incorporate the dynamic stochastic predictive requests in the computation of the schedule, assuming a limited computational power is available in this agricultural simulation. To implement MSA, two application-dependent functions must be setup: (1) a function GET-SAMPLES $(\mathcal{S}^{\mathcal{R}}, M)$ which returns a set of M deterministic scenarios $\mathcal{S}^{\xi} = \{\mathcal{S}^{\xi_1}, \mathcal{S}^{\xi_2}, \ldots, \mathcal{S}^{\xi_M}\}$. Each scenario $\mathcal{S}^{\xi_i}$

TABLE 3 Definitions of symbols used in the modeling of deterministic predictive scheduling

$\boldsymbol{L}^s$:	the collection station location;
$\boldsymbol{L}_i^f$	the full tray location in the field frame.
D_{si}:	one-way traveling distance, the Manhattan distance from $\boldsymbol{L}^s$ to $\boldsymbol{L}_i^f$ along the path;
Δt_i^u:	the corresponding robot's one-way travel time calculated by D_{si} and robot speed;
Δt^L:	time needed for the picker to take the empty tray from the robot and load the full tray on the robot (and then resumes picking);
Δt^{UL}:	the time needed at the collection station to unload the full tray from the picker/robot and return an empty tray to the picker/robot;
Δt_i^p:	The total processing time required by a robot to serve request R_i and be available to serve another request;
Δt_i^r:	release delay of request Ri, the greatest value that eliminates robot idle time at $\boldsymbol{L}_i$. $\Delta t_i^r = max\Big(\Delta t_i^f - \Delta t_i^u\Big), 0\Big)$
Δt_k^A:	The robot is available to be dispatched again, after a time interval Δt_k^A. $\Delta t_k^A = 0$, if the robot is available at the collection station;
t_{ki}^d:	The dispatch time instant of robot F_k to the request R_i, which is no earlier than $t_0 + \Delta t_i^r$.

contains a set of N sampled requests. Each of the deterministic requests R_i is sampled from predictive transport request distributions $\aleph(\Delta t_i^f), \aleph(v_i^y)$ of $\mathcal{R}_i$ in $\mathcal{S}^{\mathcal{R}}$; (2) a function OPTIMAL-SCHEDULE ($\mathcal{S}^{\xi_i}$) which returns an optimal schedule given a deterministic sampled scenario $\mathcal{S}^{\xi_i}$. The schedule includes the request rejections to some pickers and serving order for the remaining requests (3) a *consensus* function that combines all the individual scenario solutions into a single execution plan. For the function GET-SAMPLES ($\mathcal{S}^{\mathcal{R}}, M$), the Monte Carlo sampling method was used to get the M sampled scenarios from two distributions $\aleph(v_i^y)$ and $\aleph(\Delta t_i^f)$.

In a sampled scenario $\mathcal{S}^{\xi_i}$, each deterministic predictive request R_i is composed of two sampled components, Δt_i^f and v_i^y. Given these components, a deterministic full tray location $\boldsymbol{L}_i^f$. Can be calculated. The variables needed to formulate the scheduling problem—including request rejections—are listed in Table 3, and are discussed in greater detail in our previous work that did not address request rejection (Peng & Vougioukas, 2020).

The pickers' requests may be rejected by the scheduler. In this case, they need to transport the full tray themselves and their self-transporting behavior is modeled following our previous work (Seyyedhasani et al., 2020a). If the picker transports the tray themselves the total time Δt_i^T, required to deliver the full tray and take an empty tray back to resume picking is shown as (Equation 1). Δt_i^{up} is the one-way travel time interval from full tray location $\boldsymbol{L}_i$ to $\boldsymbol{L}^s$ by the picker in R_i. Δt_i^{up} is calculated based on D_{si} and an estimated picker self-transport speed v_p^i from the historic data of pickers. Δt^{UL} is assumed to be constant depending on the crew management in the harvesting field.

$$\Delta t_i^T = 2\,\Delta t_i^{up} + \Delta t^{UL}. \tag{1}$$

The tray completion time instant, t_i^{Cp} if the full tray is transported by the picker himself, is shown in (Equation 2) with Δt_i^T representing the estimated tray-transport time by the picker.

$$t_i^{Cp} = t_i^f + \Delta t_i^T. \tag{2}$$

If the request is served by a robot F_k, the time instant, t_i^{CR} to resume picking is expressed as (Equation 3). The picker can start picking the next tray after the robot arrives at the full tray location and the full tray is exchanged with the empty tray from the robot.

$$t_i^{CR} = t_{ik}^d + \Delta t_i^u + \Delta t^L. \tag{3}$$

The nonproductive time, Δt_i^N of R_i can be calculated as (Equation 4). The objective of the modeled problem is to minimize the mean of the nonproductive time of all the pickers. In the objective function, both t_i^{Cp} and t_i^{CR} are represented by t_i^C which is decided by the decision variables. All the above-mentioned time intervals and instances are labeled on the timelines in Figure 10.

$$\Delta t_i^N = t_i^C - \Delta t_i^f. \tag{4}$$

The function OPTIMAL-SCHEDULE ($\mathcal{S}^{\xi_i}$) was developed to compute the solution for each scenario. First, the exact solution is computed using integer programming to get the best possible solution. Second, a fast and suboptimal heuristic policy is implemented to get a near-optimal solution in less time, so that the pickers do not need to wait for long time caused by the scheduling computation. The results of the exact and heuristic solutions are compared using the proposed performance metrics in Section 5.

4.1 | Scenario solution using integer programming

The deterministic predictive scheduling problem of each sampled scenario was modeled using an integer linear program. The scenario scheduling problem is different from our previous work (Peng & Vougioukas, 2020) in that the scheduler can now reject transport

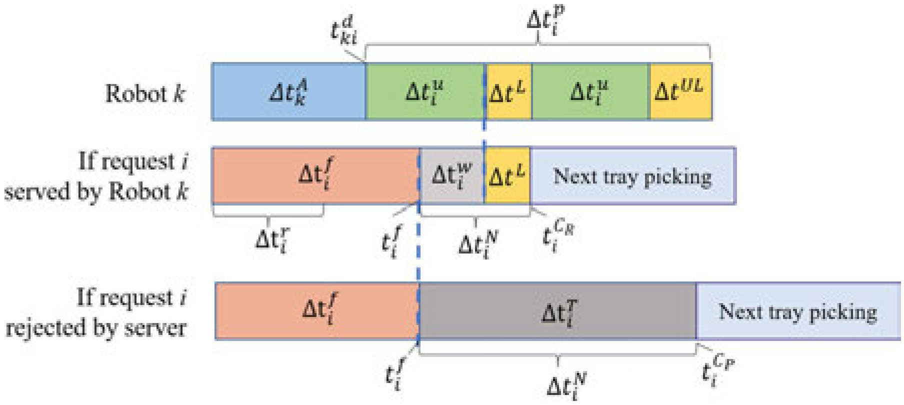


FIGURE 10 Timelines of request $\mathbf{R_i}$ served by robot $\mathbf{F_k}$ or rejected by the scheduler [Color figure can be viewed at wileyonlinelibrary.com]

requests. $\mathcal{S}^{\mathcal{T}}$ is used to represent the discretized time set, {1, 2, 3…, *TB*}. *TB* is the upper bound makespan of all requests (from t_0 to $t_0+\max_i\{t_i^C\}$). For this problem, the upper bound *TB* can be expressed as (Equation 5). It is easy to prove that the completion time of any request cannot be larger than the maximum completion time of self-transporting; otherwise, that request should be transported by the picker themselves.

$$TB\le t_0+\max_i\left\{\Delta t_i^f\right\}+\max_i\left\{\Delta t_i^T\right\}. \tag{5}$$

The decision variable is defined as χ_{ikt}, where *i* is the index of request R_i. *k* is the index of the serving robot if $1\le k\le M$; $k=M+1$ means that the picker transports the tray himself. *t* is the index of discrete-time instant. χ_{ikt} is equal to 1 if R_i is served by a robot F_k ($1\le k\le M$) or transported by the picker himself ($k=M+1$) at the time instant *t*. The problem can be modeled using integer linear programming (ILP) as follows:

$$\min\sum_{i=1}^{N}\Delta t_i^N \text{s. t.}$$

$$\sum_{k=1}^{M}\sum_{t=1}^{\Delta t_i^r}\chi_{ikt}=0,\ R_i\in\mathcal{S}^{\mathcal{R}}, \tag{6}$$

$$\sum_{i=1}^{N}\sum_{t=1}^{\Delta t_k^A}\chi_{ikt}=0,\ 1\le k\le M, \tag{7}$$

$$\sum_{k=1}^{M}\sum_{t=1}^{t_i^f}\chi_{i(M+1)t}=0,\ R_i\in\mathcal{S}^{\mathcal{R}}, \tag{8}$$

$$\sum_{k=1}^{M+1}\sum_{t=1}^{TB}\chi_{ikt}=1,\ R_i\in\mathcal{S}^{\mathcal{R}}, \tag{9}$$

$$\sum_{k=1}^{M}\sum_{t=max\left(1,t-\Delta t_i^P\right)}^{t}\chi_{ikt}\le 0,\ R_i\in\mathcal{S}^{\mathcal{R}}, \tag{10}$$

$$t_i^C=\sum_{k=1}^{M}\sum_{t=1}^{TB}\left(t+t_i^U+\Delta t^L\right)\chi_{ikt},\ R_i\in\mathcal{S}^{\mathcal{R}}, \tag{11}$$

$$t_i^C=\sum_{t=1}^{TB}\left(t+t_i^T\right)\chi_{i(M+1)t},\ R_i\in\mathcal{S}^{\mathcal{R}}, \tag{12}$$

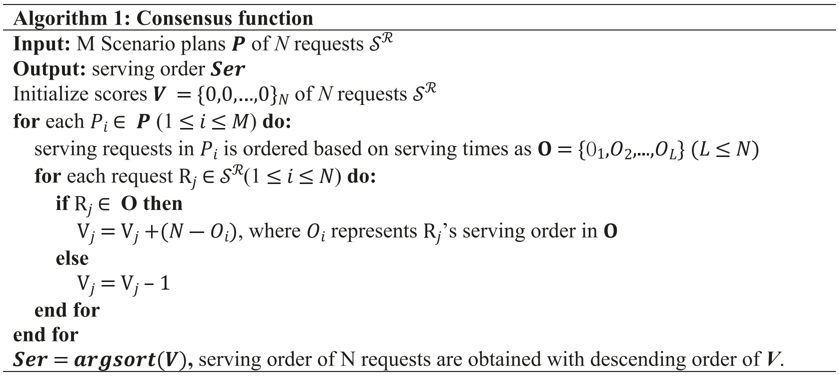
$$\Delta t_i^N=t_i^C-\Delta t_i^f,\ R_i\in\mathcal{S}^{\mathcal{R}}. \tag{13}$$

The objective function is the sum of the non-productive time of all requests and the required constraints are explained as follows. In (Equation 6), it represents that any request cannot be served by a robot before their release constraints. (Equation 7) means that the robot's start serving time cannot be earlier than their initial available time. If the tray is transported by the picker her/himself, the start time cannot be earlier than the full tray instant t_i^f as (Equation 8). (Equation 9) represents that all requests must be served either by a robot or by the picker. (Equation 10) shows that any request can be served by only one robot (preemption is not allowed). (Equation 11) expresses the tray completion time of the request served by the robots, while (Equation 12) is the tray completion time served by the pickers themselves. (Equation 13) shows the non-productive time of request R_i.

As mentioned above, predictive scheduling of crop-transport robots is a variant of the Parallel Machine Scheduling Problem (PMSP). Following symbol notations defined by Lawler et al. (1993), the problem is referred to as $Pm|r_i|\sum C_i$, where *Pm* represents identical parallel machines, r_i means that the *i*th job cannot be processed until its release time, and $\sum C_i$ represents that the objective criterion is to minimize the sum of the completion times of all jobs. It has been shown that this problem is NP-hard in a strong sense and hence the optimal solution cannot be obtained in polynomial time (Du et al., 1991). In this paper, the modeled ILP was solved by a commercial solver (Gurobi Optimization, LLC., 2020) at the cost of long computation.

4.2 | Scenario solution with heuristic policy

In this study, a heuristic policy, namely, the shortest release time with long process time first (SRLPT) (Phillips et al., 1998), is proposed to achieve a fast but suboptimal result in each sampled deterministic scenario. The requests reaching the release constraint ($\Delta t_i^r=0\Delta t_i^r=0$) will enter a scheduling pool and the request with the longest process time in the pool is ordered to be served by the first available robot. The non-productive time of those requests with large self-transport time can be reduced significantly by the service of available robots. The requests that are expected to have a shorter transport time are served late, because even if they are rejected, the non-productive time will not be that large. The requests with a full tray location less than 5 m away from the end of the row are rejected, as those pickers only need to walk a small distance back and forth to resume picking. The performance comparison between the heuristic policy and ILP is shown in Section 5.4.1.



```
Algorithm 1: Consensus function
Input: M Scenario plans $\boldsymbol{P}$ of $N$ requests $\mathcal{S}^{\mathcal{R}}$
Output: serving order $\boldsymbol{Ser}$
Initialize scores $\boldsymbol{V} = \{0,0,...,0\}_N$ of $N$ requests $\mathcal{S}^{\mathcal{R}}$
for each $P_i \in \boldsymbol{P}$ $(1 \le i \le M)$ do:
    serving requests in $P_i$ is ordered based on serving times as $\mathbf{O} = \{0_1, O_2, ..., O_L\}$ $(L \le N)$
    for each request $\mathrm{R}_j \in \mathcal{S}^{\mathcal{R}}(1 \le i \le N)$ do:
        if $\mathrm{R}_j \in \mathbf{O}$ then
            $\mathrm{V}_j = \mathrm{V}_j + (N - O_i)$, where $O_i$ represents $\mathrm{R}_j$'s serving order in $\mathbf{O}$
        else
            $\mathrm{V}_j = \mathrm{V}_j - 1$
    end for
end for
$\boldsymbol{Ser} = \boldsymbol{argsort}(\boldsymbol{V})$, serving order of N requests are obtained with descending order of $\boldsymbol{V}$.
```

4.3 | Consensus function

In the MSA framework, given the current pool of solutions (scheduling plans) for the multiple sampled scenarios, the role of the consensus function is to select/synthesize a plan most similar to the current pool of plans. Bent & Van Hentenryck (2004b) proposed a consensus function for a partially dynamic vehicle routing problem with time windows, and applied it successfully for online packet scheduling in computer networks (Bent & Van Hentenryck, 2004a). The heuristic idea behind the consensus function is the least-commitment approach, a well-known approach in the artificial intelligence community (Bent & Van Hentenryck, 2004a). By choosing jobs that occur more often, the consensus algorithm computes a solution that is as consistent as possible with the optimal solutions of the pool of scenarios.

In this study, we developed a consensus function for our application; its pseudo-code is shown in Algorithm 1. After all the deterministic scenarios are solved with the function of OPTIMAL-SCHEDULE, the scheduling plan of each scenario is converted to a serving order based on their scheduled serving times. A score function is defined for each request in one scenario. If the request is rejected, the score value of that request is counted as −1. If the request is served by a robot in the order of O_i among all the serving requests in that scenario, the score of that request is counted as $(N - O_i)$. The score of each request is obtained by adding the scores of the requests among all the sampled scenarios. The consensus serving order is the descending order of the scores of all the requests in $\mathcal{S}^{\mathcal{R}}$. The available robots were dispatched to the first request in the consensus order at the instant when the expected release time of that request is reached. The rejection flags are sent to the pickers if they are not served by the robot at the instant when their trays are full. The scheduler will update the scheduling plan only when there are robots available and new transport requests entering the set.

5 | EXPERIMENTS AND RESULTS

5.1 | Evaluation metrics

In both the all-manual and robot-aided strawberry harvesting trials, the productive time per tray—denoted as Δt_i^{ef}—is defined in the same way: it is the time required by a picker to fill the ith tray to its capacity, starting from an empty tray. Productive time includes picking and walking to relocate to a new furrow to resume picking when the tray cannot be finished in the current furrow. Non-productive time per tray—denoted as Δt_i^{fe}—is defined as the time interval that is not spent picking or relocating to pick from another furrow. In manual strawberry harvesting, Δt_i^{fe} includes the picker's walking time to transport the full tray to the unloading station, the waiting time in a queue to deliver the tray and get an empty one, and the walking time required to return to the previous position to resume picking. In contrast, in robot-aided harvesting, Δt_i^{fe} is the sum of the time, Δt_i^{w}, the picker spends waiting for a robot to arrive, plus the time, Δt^{L}, needed to place the full tray on the robot and take an empty tray from the robot. Δt_i^{w} is highly dependent on the robot scheduling policy, whereas Δt^{L} is small and is assumed to be constant. ΔT^{fe} represents the average non-productive time of all the trays measured in an experiment.

The mean harvesting efficiency, E_{ff}, when harvesting N trays with or without robots, is defined as the averaged sum of ratios of productive time over total time spent for each tray; it is calculated by (Equation 14):

$$E_{ff} = \frac{1}{N}\sum_{i=1}^{N}\frac{\Delta t_i^{ef}}{\Delta t_i^{ef} + \Delta t_i^{fe}}. \tag{14}$$

ΔT^{fe} and E_{ff} can be used to evaluate the overall performance of all-manual and robot-aided harvesting. Inspired by ASAE field efficiency standards for machinery (ASAE, 2009), this efficiency metric does not include any off-field times required by workers or robots to support the harvesting operation, such as the time needed to travel to the field, prepare for work, or store equipment.

5.2 | Simulation experiments

Two sets of simulation experiments were performed, one before the field harvest experiments (pre-harvest set) and one after (post-harvest set). The first set of simulation experiments had three goals. The first goal was to evaluate the effect of request rejections on scheduling performance. The second goal was to compare the performance difference between the heuristic (SRLPT) and exact (ILP) optimization algorithms introduced in Section 4.1. The third goal was to select the number of scenarios that the MSA would sample when executing in real-time, during field experiments. It is known that when more scenarios are sampled, the MSA solution improves, but takes a longer time to compute; hence, a tradeoff must be reached. The second set of simulation experiments used the data collected during the field experiments to investigate how the harvest-aid system's efficiency scales when the robot/picker ratio stays the same, but the crew and field sizes increase.

All the simulation experiments were performed using a Monte-Carlo simulator that was adapted from our previous work (Peng & Vougioukas, 2020). In this study it was shown that simulating the harvest process by running 100 Monte-Carlo runs was sufficient to provide 1% precision in the simulation results. The main underlying assumption in our analyses of the results is that the 100 sampled means of each evaluated metric were normally distributed.

The parameters used in the simulations are described next. During the simulations (and the field experiments) the robot speed was 0.8 m/s on the headland and 1.5 m/s inside the furrows. The effects of different robot speeds were studied in (Peng & Vougioukas, 2020) and (Peng, 2021), and it was shown that higher robot speeds improve system performance significantly. However, relatively low values were selected in this study, primarily for safety purposes, since the robots would travel close to humans during the field experiments.

Another important parameter that is used by the scheduler during simulation and real-world deployment is the tray fill-ratio (FR) threshold. The fill ratio was defined in (Peng & Vougioukas, 2020), as the current weight of a tray divided by the tray's maximum weight (capacity). In this study it was proved that predicting tray-transport requests and adding them to the scheduler's queue too early—when FR is below some threshold—does not improve the scheduling performance (although it increases computational load). Also, the prediction uncertainty decreases as the tray gets closer to being full (Khosro Anjom & Vougioukas, 2019), and therefore, tray-request predictions and scheduling should only be performed when FR is greater than a threshold. The effects of different robot speeds and FRs were studied in (Peng & Vougioukas, 2020) and (Peng, 2021), and the FR threshold for the range of robot speeds used in this study was determined to be 0.7.

The last set of parameters that is used in the simulation and by the scheduler during simulation and field deployment relates to the pickers' work or harvesting parameters (one-tray-picking time, one-tray picking distance, and picker walking speed parameters), which are stochastic. The distributions of the work parameters of the crew were estimated from experimental data, as shown in Section 5.4.2. The random parameters of the requests were generated following the distributions measured by Khosro Anjom et al. (2019): the mean of bias for the full tray time prediction was less than 10% of one tray picking time, and the standard error of the prediction was 30 s.

After determining all the parameters, in pursuit of the first goal, the effect of request rejections was investigated for robot teams of various sizes, using exact optimization algorithms. For scheduling without request rejections, we applied the exact Branch and Bound Search algorithm developed in our previous work (Peng & Vougioukas, 2020). For scheduling with request rejections, we applied the ILP solver introduced in Section 4.1. To pursue the second goal, the ILP and SRLPT single-scenario optimal scheduling solvers were implemented in MSA's OPTIMAL-SCHEDULE module and compared for an increasing number of robots (4–12 robots) and a typical crew size of 25 pickers. Toward the third goal, the harvesting efficiency achieved with the MSA was computed when the number of scenarios ranged from 1 to 80, with eight robots and 25 pickers. The efficiency results from solution pairs that used different numbers of scenarios (e.g., 30 vs. 50 scenarios) were compared using Tukey's honest significant difference (HSD) tests, to determine the number of scenarios to use in the field experiments, that is, the one beyond which the performance did not improve significantly. The simulation results and their analyses are presented in Section 5.4.1.

5.3 | Field experiment design

The main goal of the field experiments was to evaluate the savings of the harvest-aiding system in commercial strawberry harvesting, with a crew of professional pickers. The parameters used during robot deployment (e.g., robot speeds, picker work parameters) were the same as the ones used in the simulation experiments. The scheduler generated 50 scenarios and used the SRLPT heuristic with an FR threshold of 0.7 to solve each one, based on the outcome of the simulations (see Section 5.4.1).

On November 10 and 11, 2020, the harvest-aiding system was evaluated using two robots with a crew of six professional pickers in a commercial field near Lompoc, CA shown as Figure 11. All the field experiments were conducted under the UC Davis Institutional Review Board (IRB) compliance protocol “IRB 575389-8.” Each day, the pickers' working schedule was divided into two (2) sessions. The first session was from 8:00 a.m. to 11:00 a.m. and the second session was from 11:30 a.m. to 2:30 p.m. In the first session, on November 10, our system was set up and tested on the mapped field while the crew harvested the

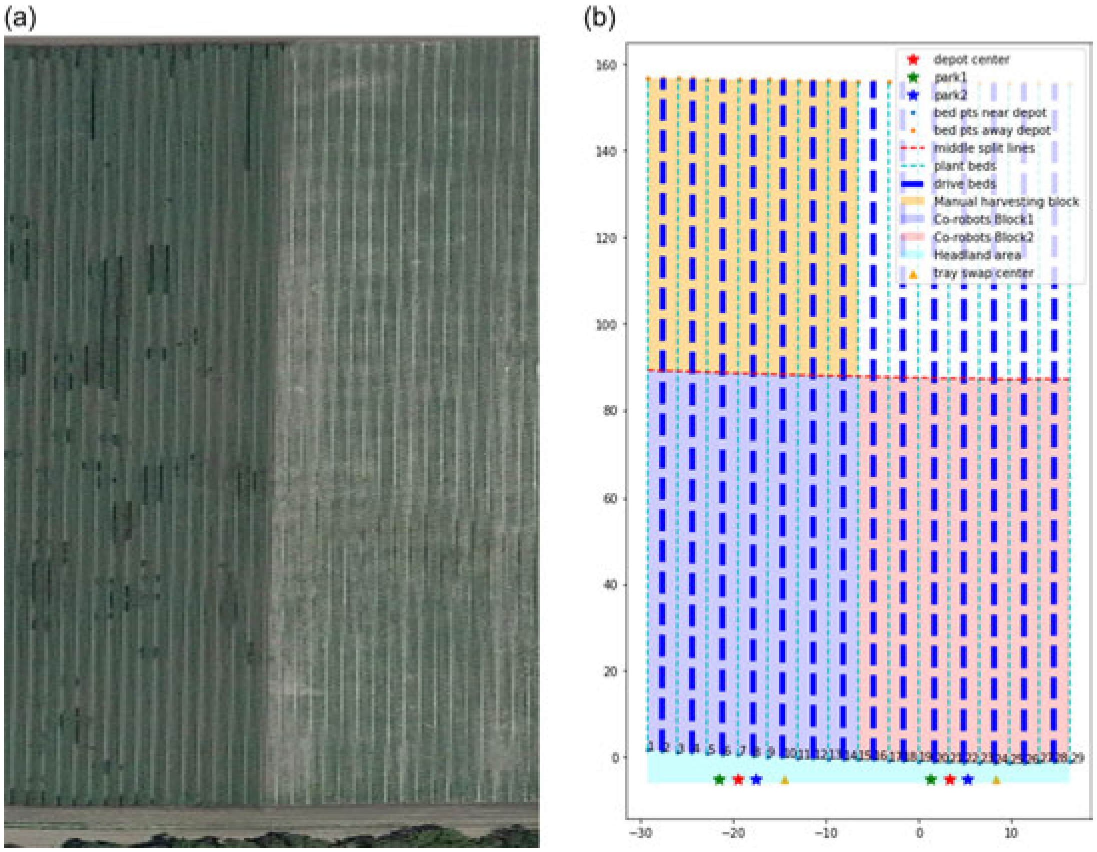


FIGURE 11 (a) Satellite picture of commercial field block near Lompoc, CA from Google Maps. (b) Map of the field block built with RTK: blue and orange shaded areas were evaluated on November 1. [Color figure can be viewed at wileyonlinelibrary.com]

orange-colored field block (Figure 11b) in their usual manner, using our instrumented carts. The pickers collected the strawberries in 500-g carton box trays. The gross mass of a full tray was around 4.5 kg (10 lbs). Their harvesting data was collected and saved on the SD card modules of the carts. From the data in that session, their walking speeds when transporting full trays were estimated. This data was used to estimate the performance of manual harvesting.

In the second and first sessions of November 10 the first session of November 11, all six pickers started harvesting from the field's middle line and moved toward the unloading station in their typical harvesting manner. The crew harvested with the assistance of the two robots (blue area for November 10th and red area for November 11 on Figure 11b). When the co-robotic harvest-aiding system was used, the harvesting data was recorded into ROS files on the server laptop, as well as in the SD cards of the carts. The evaluation results of our harvest-aiding co-robotic system were obtained from these two sessions. Non-parametric tests were used to compare the performance of manual and co-robotic harvesting.

During robot-aided harvesting the pickers were asked to press the request button on the cart once they filled six out of the eight clamshell boxes in their tray. It was explained to them that if their transport requests were accepted by the robots, the yellow LED on their cart would turn on. In this case, they were instructed to wait for a robot, in case they filled their tray and a robot had not arrived. The scheduling system dispatched robots to serve the requests by solving online the stochastic predictive scheduling problem. The robots were scheduled and dispatched to the predicted full-tray locations inside the rows. Upon arrival at the commanded location, the robots would stay still until the picker placed their tray on the robot and pressed a button that sent the robot back to the collection station.

The system components are shown in Figure 12. The full-empty tray swap location was approximately 5 m away from the depot center where the server laptop was located (red star in Figure 11a), on its right-hand side, facing the field.

In the first session on November 10, 33 trays of fruit were harvested manually in the field block. In the second session on November 10, 41 trays were harvested by the 6 pickers working with the co-robotic harvest-aiding system. In the first session on November 11, the same picking crew of 6 people harvested 24 trays of fruit using the harvest-aiding system.

5.4 | Analysis of results

The results were combined and presented in this section. First, the simulation results are presented and analyzed in Section 5.4.1. Then, the results from field experiments are presented in Sections 5.4.2 and 5.4.3.

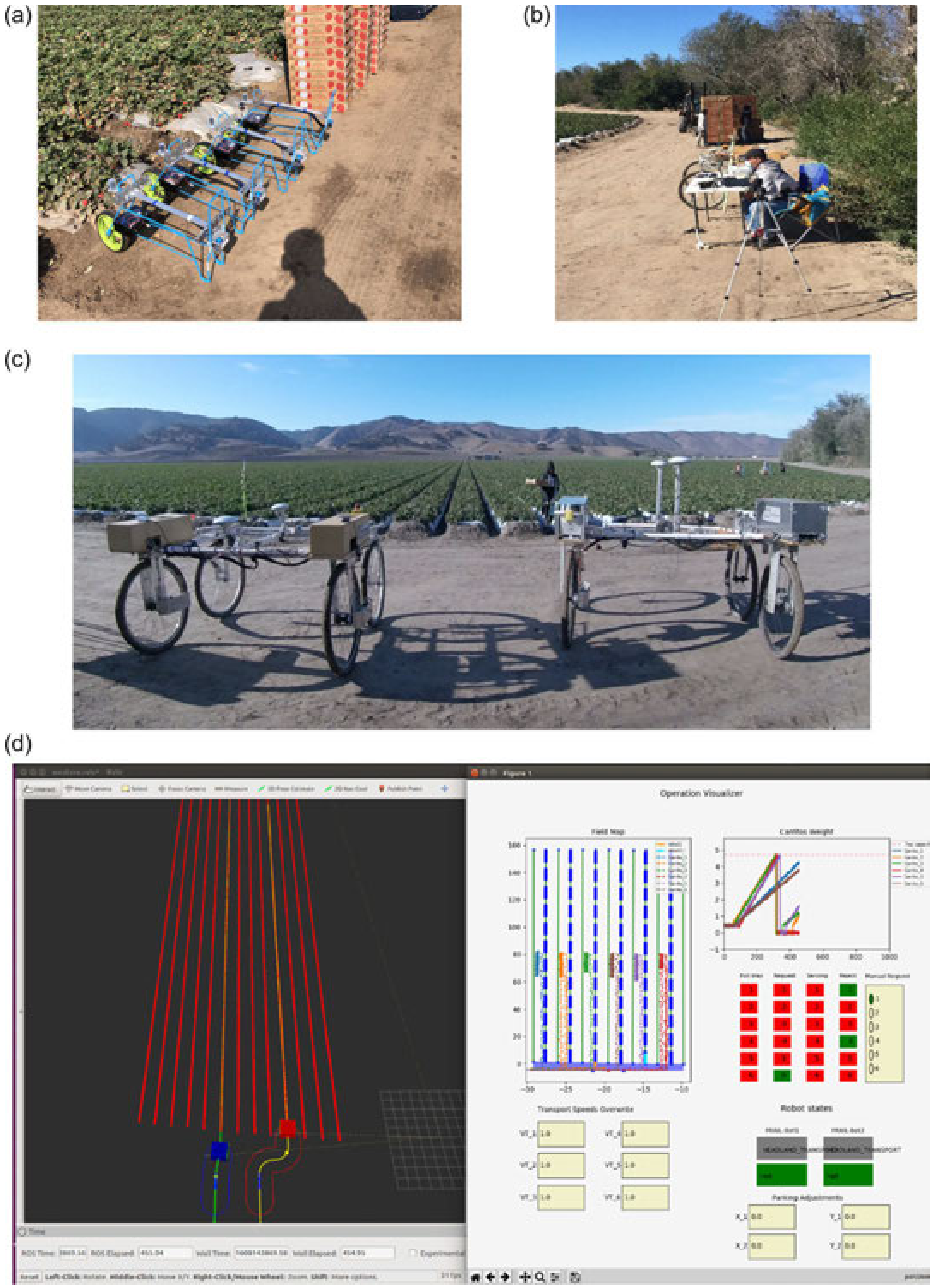


FIGURE 12 Components of robotic harvest-aiding system: (a) Instrumented carts; (b) collection station (aka, depot center) with scheduling server; (c) two FRAIL-Bots parked at the depot center; (d) scheduling server user interface built with Python Matplotlib and ROS RVIZ for visualization and monitoring of the robots' motions. [Color figure can be viewed at wileyonlinelibrary.com]

5.4.1 | Pre-harvest simulations: Results and analysis

First, simulation experiments were performed to compare the performance of the scheduling policies with request rejections and without rejections. Figure 13 shows that the 95% confidence interval of the harvest efficiency of the request rejection policy was always (robot/picker ratio greater than 4:25) much higher than that of the manual harvesting. In contrast, when all requests must be served,

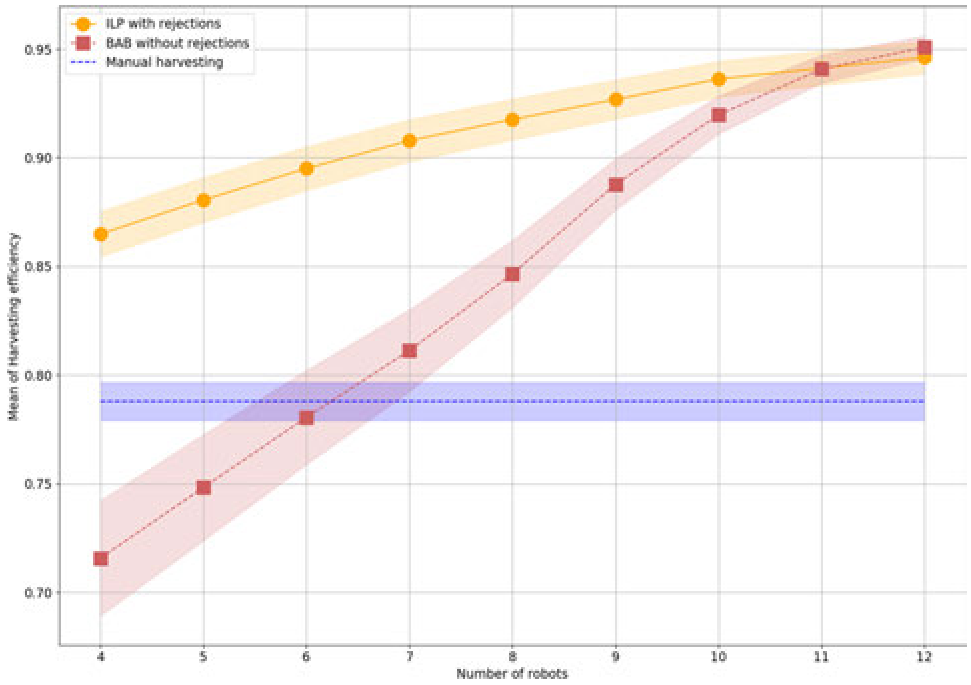


FIGURE 13 Mean (points) and its 95% confidence interval (shaded area) of harvesting efficiency as a function of the number of robots for the scheduler with request rejections and without request rejections. [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 4 *p* Values of *t*-test of scheduling efficiencies with/without request rejections for 10, 11, and 12 robots.

Number of robots	10	11	12
p value	5.24e−11	0.823	0.141

robot-aided harvesting was better only when the robot/picker ratio was higher than 7:25.

Also, the 95% confidence intervals of the two mean efficiencies intersected when the number of robots was greater than 10. This fact indicates that the predictive scheduler performs better with request rejections than without rejections, when the robot/picker ratio is smaller than 10/25. To verify this rigorously, the mean efficiencies of the two policies were compared using *t*-tests, for 10, 11, and 12 robots. The null hypothesis for the *t*-tests was that the mean efficiencies of the two policies have no significant differences in the robot ratios of 10/25, 11/25 and 12/25. The α value for the tests was set to 1% (Type I error). The results in Table 4 show that when there were fewer than 10 robots (robot/picker ratio is less than 1:2.5) the harvesting efficiency with request rejections performed significantly better than the efficiency without request rejections. However, when the number of robots was 11 or more, there was no significant difference, as the *p* value was greater than 0.05.

The main reason is that, when the robot/picker ratio is smaller, the tray-transport request queue tends to be larger, and request rejection increases the robot availability, thus allowing faster service (reducing queue size). However, at high ratios, the tray-transport request becomes smaller, and request rejection has a smaller effect. At the limit of 1:1 ratio, there would be no tray-transport requests queue, since a robot would always be available immediately for each picker, and both policies would yield the same performance. The harvest efficiency curves are not expected to change if the two policies were to be applied to bigger and different crews or larger fields, because the size of the tray-transport request queue depends on the robot/picker ratio; however, the exact robot/picker ratio where the two policies result in the "same" performance may change.

Next, we evaluated the performance of the proposed heuristic policy (SRLPT) and ILP policy (Peng & Vougioukas, 2020), using harvesting efficiency as our evaluation metric. The comparison E_{ff} is shown in Figure 14 for 25 pickers and an increasing number of robots.

t-Tests were applied for the mean efficiencies of the two policies under different robot-picker ratios. The null-hypothesis of each *t*-test was that the mean efficiencies of the two policies were the same. The α value for the tests was set to 5% (Type I error). The results are presented in Table 5, where one can see that the performance of two policies was not significantly different when the robot/picker ratio was larger than 1:3 (8/25), as all *p* values of the *t*-test results were over 0.05. The main reason is that, when the robot/picker ratio is small, the robots constitute a "scarce" highly utilized resource, the tray-transport request queue is larger, and optimal versus suboptimal scheduling makes a difference. However, at higher ratios, suboptimal scheduling does not affect the system efficiency as much; at the limit of 1:1 ratio, there would be no tray-transport requests queue, since a robot would always be available immediately for each picker, and both policies would yield the same

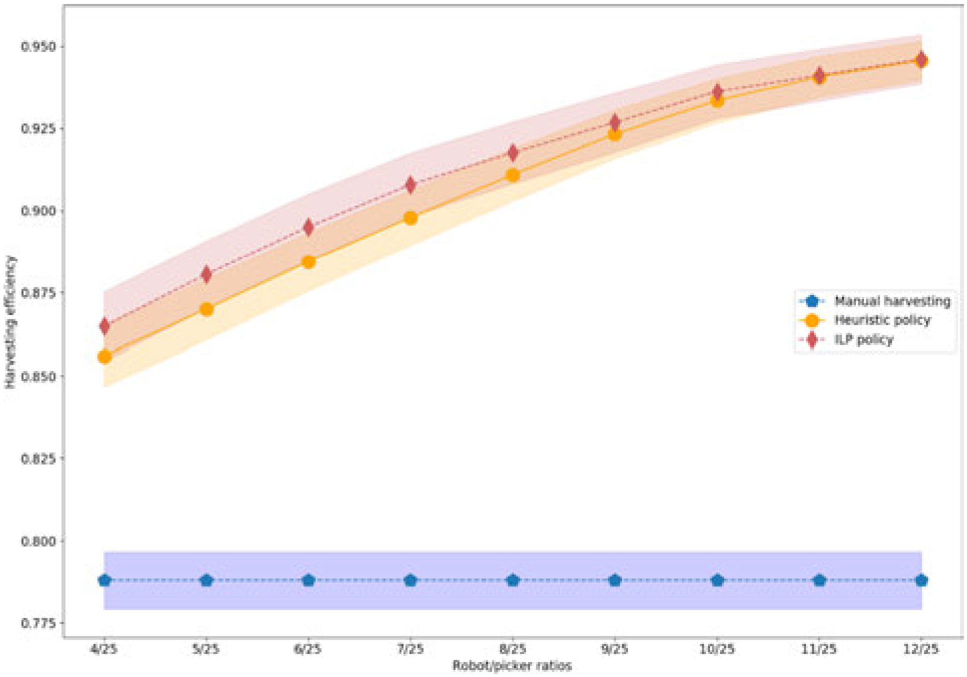


FIGURE 14 Mean values of harvesting efficiency (points) and their 95% confidence interval (shaded areas) as a function of the number of robots for the scheduler with ILP and heuristic SRLPT. [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 5 p Values of t-tests for the efficiencies of two policies in different robot/picker ratios

Robot/picker ratio	4/25	5/25	6/25	7/25	8/25	9/25	10/25	11/25	12/25
p values	0.030	0.015	0.013	0.024	0.055	0.118	0.140	0.379	0.488

performance. The ILP and SRLPT efficiency curves are not expected to change when applied to bigger and different crews or larger fields, since the size of the tray-transport request queue depends on the robot/picker ratio; however, the exact robot/picker ratio where the curves become the "same" may change. The 1:3 ratio was used in our field experiments.

Finally, we investigated the performance of the MSA scheduler as a function of the number of sampled scenarios, given the experimentally measured distribution of the request prediction uncertainty from the work of Khosro Anjom et al. (2019). The mean values of harvesting efficiency and their 95% CI (shaded areas) as a function of the number of sampling scenarios are shown in Figure 15.

From the results, one can see that as the number of scenarios in MSA increases, the mean harvesting efficiency (red curve) also increases, until the number reaches 50. However, when more than 50 scenarios are considered, efficiency and non-productive time plateau. Tukey's HSD tests were used for different sampling scenarios {10, 20, 30, 40, 50} to examine their efficiency differences; the results are presented in Table 6. The null hypothesis was that the mean efficiency of MSA in different sampling scenarios was the same. The alpha value for the tests was set to 5% (Type I error). From the combinations of {30, 40}, {30, 50} and {40, 50} in Table 6, one can see that the scheduling performance of MSA did not show significant improvement when the number of sampling scenarios was over 30. One expects that this number will increase if the uncertainty in request prediction increases (e.g., due to a larger, non-homogeneous crew or large variation in yield). However, the same method to estimate this number can be applied, once the pickers' harvesting parameters are known.

Based on the above results, the number of scenarios for the field experiments was set to 50, a conservative value that was expected to perform well, even under increased uncertainty. The mean computation time for a schedule with 50 scenarios was approximately 5 s (on the Intel Core i7-3770@3.40 GHZ laptop used as a server), which was adequate for real-time operation.

5.5 | Results of harvesting parameters

The distributions of the one-tray-picking time, one-tray picking distance, and picker walking speed parameters were generated from the data. The single-tray picking time and single-tray picking distance parameters were assumed to be dependent on the geospatial fruit distribution, for the same picking crew.

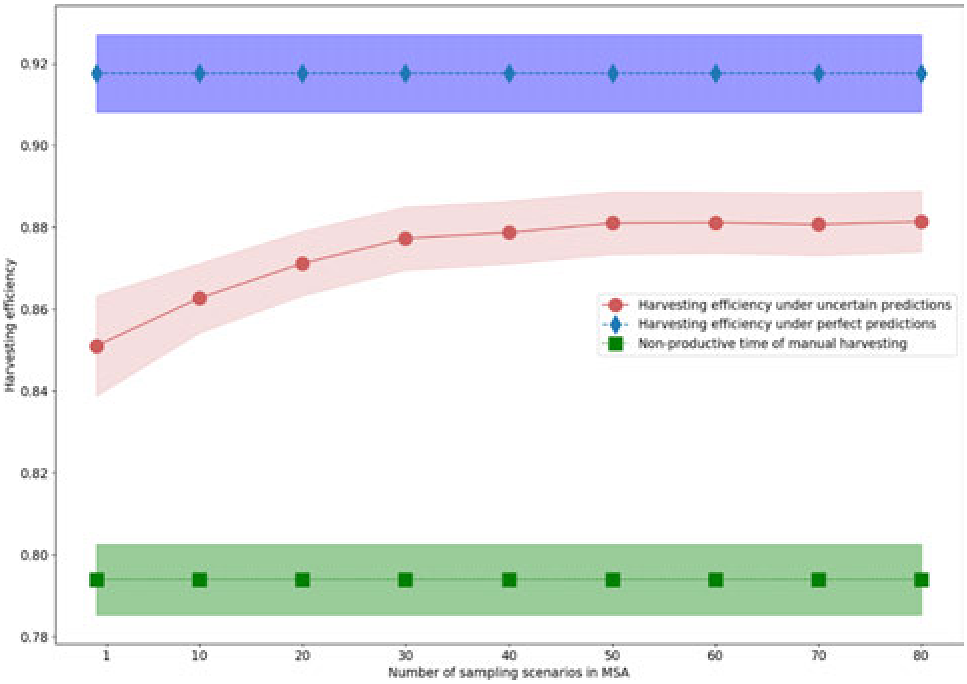


FIGURE 15 Mean values of harvesting efficiency (points) and their 95% confidence interval (shaded areas) as a function of the number of sampling scenarios, under uncertain transport request predictions. The harvesting efficiency (blue lines) under perfect transport request predictions and the manual harvesting efficiency (green lines) are also presented. [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 6 Tukey's HSD results of harvesting efficiencies for different sampling scenarios in MSA

Scenarios comparing combinations		Adjusted *p* values	Null rejections
10	20	0.0012	True
10	30	0.0013	True
10	40	0.0010	True
10	50	0.0005	True
20	30	0.0022	True
20	40	0.0031	True
20	50	0.0020	True
30	40	0.7631	False
30	50	0.0752	False
40	50	0.1221	False

Abbreviations: HSD, honest significant difference; MSA, Multiple Scenario Approach.

Figure 16 shows the distributions of the one-tray picking time of the crew for the co-robotic harvesting blocks of November 10 (session 2) and November 11 (session 1). The Mann–Whitney rank test was used to compare the two distributions. The null hypothesis was that for randomly selected values from the distribution of harvesting time per tray on November 10 and 11, the probability of the selected values on November 10 being greater than November 11 was equal to the probability of selected values on November 11 being greater than November 10. The significance level of *p* values (*α*) for rejecting the null hypothesis (Type I error) was chosen as 1%. The calculated *p* value was 2.13e−9, so the distributions of the two days differed significantly. A comparison of their mean values shows that, on average, the harvest crew took a longer time to harvest one tray on November 11 (894.62 s) than on November 10 (548.71 s).

Figure 17 shows the distributions of the one-tray picking distance, for the 2 days. The p value from the Mann-Whitney rank test of the two distributions was 2.21e−7, so they were significantly different. On average, the pickers moved a longer distance to collect a tray of strawberries on November 11 (33.31 m) than November 10 (17.92 m).

During manual harvesting, each picker would take their filled tray to the collection station, attach a sticker with their personal barcode on the tray, take an empty tray and walk back to the field to resume picking. Based on our observations, the pickers took around 8 s to stick their barcode on the tray and take an empty tray. Thus, the walking time to deliver a tray can be estimated by subtracting these 8 s from Δt_i^{fe}. The exact locations $\boldsymbol{L}_i^f$ when a picker starts walking are detected by the weight change; the corresponding time instants are $t_i^{\{end\}}$. The transport distance was computed from the coordinates of the collection station and $\boldsymbol{L}_i^f$. Hence, each picker's walking speed was estimated from the computed transport distance and the measured time interval. The manual harvesting data from the first session of November 10 was used to estimate the mean walking speeds of the six pickers, as shown in Table 7.

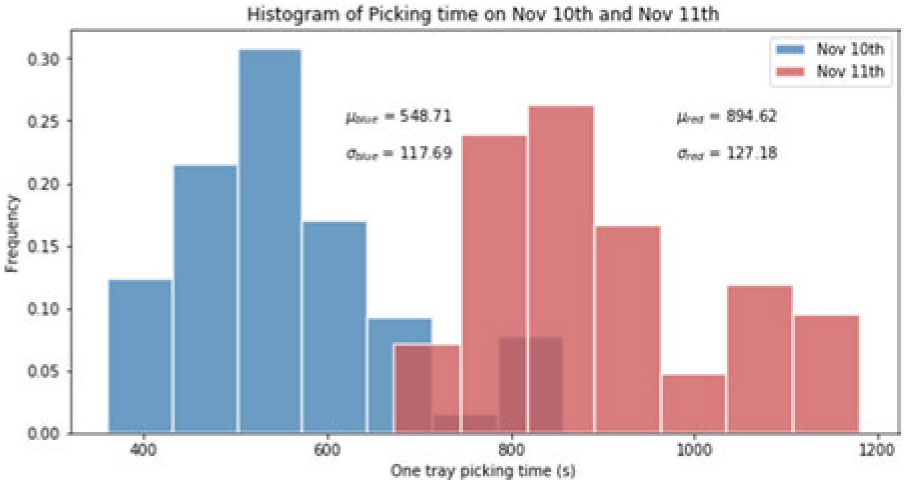


FIGURE 16 Histogram of the time it took pickers to fill one tray, on November 10 and 11 [Color figure can be viewed at wileyonlinelibrary.com]

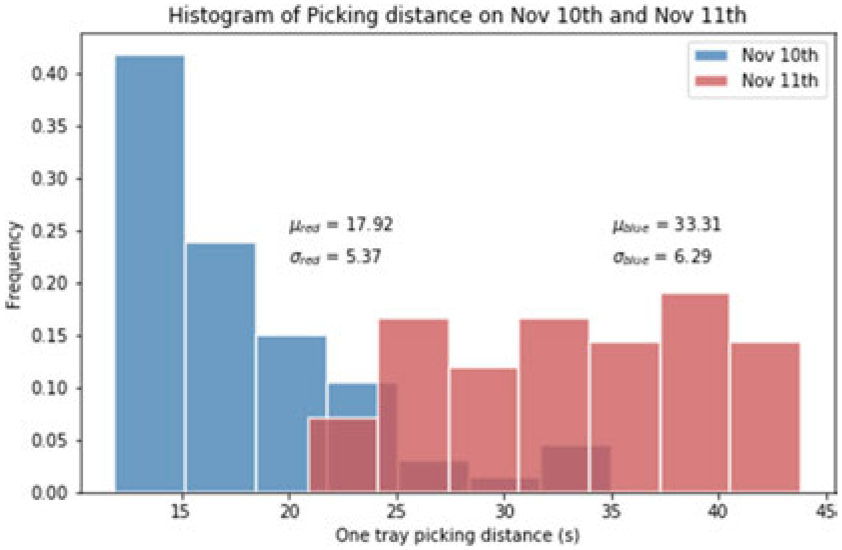


FIGURE 17 Histogram of the distance traveled by pickers to fill one-tray, on November 10 and 11 [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 7 The estimated mean walking speeds of the pickers, when they transport full trays to the collection station

Picker ID#	Sample mean of walking speed (m/s)	Sample standard deviation of walking speed (m/s)	Number of manual transport measurements
1	0.78	0.05	5
2	0.49	0.10	5
3	0.81	0.06	6
4	0.74	0.11	5
5	0.91	0.07	6
6	1.02	0.03	6

5.5.1 | Co-robotic harvesting performance

The methodology described in previous work (Khosro Anjom & Vougioukas, 2019) was used to compute the time instants (Figure 17) that are necessary to compute the evaluation metrics. The time when the picking of the *i*th tray started ($t_i^{\{start\}}$) was computed by detecting the moment when the measured mass became larger than 550 g (500 g is the weight of an empty carton box). The time when the picking of the *i*th tray was completed ($t_i^{\{end\}}$) was identified by detecting a drop greater than 1000 g in the measured mass of the tray, after the measured mass exceeded 4000 g. A tray-transport request time instant corresponded to a change in the state of the request button from "0" to "1." The productive interval Δt_i^{ef} for the *i*th tray was calculated by (Equation 15) and the non-productive interval Δt_i^{fe} was calculated using (Equation 16). The efficiency of the ith tray was calculated by (Equation 17) (Figure 18).

$$\Delta t_i^{ef} = t_i^{\{end\}} - t_i^{\{start\}}, \tag{15}$$

$$\Delta t_i^{fe} = t_{i+1}^{\{start\}} - t_i^{\{end\}}, \tag{16}$$

$$E_{ff_i} = \frac{\Delta t_i^{ef}}{\Delta t_i^{ef} + \Delta t_i^{fe}}. \tag{17}$$

Obviously, it is impossible to have the picking crew re-harvest manually a field block that was harvested using the robots. Ideally, a large trial would use a large field and divide it into smaller blocks and

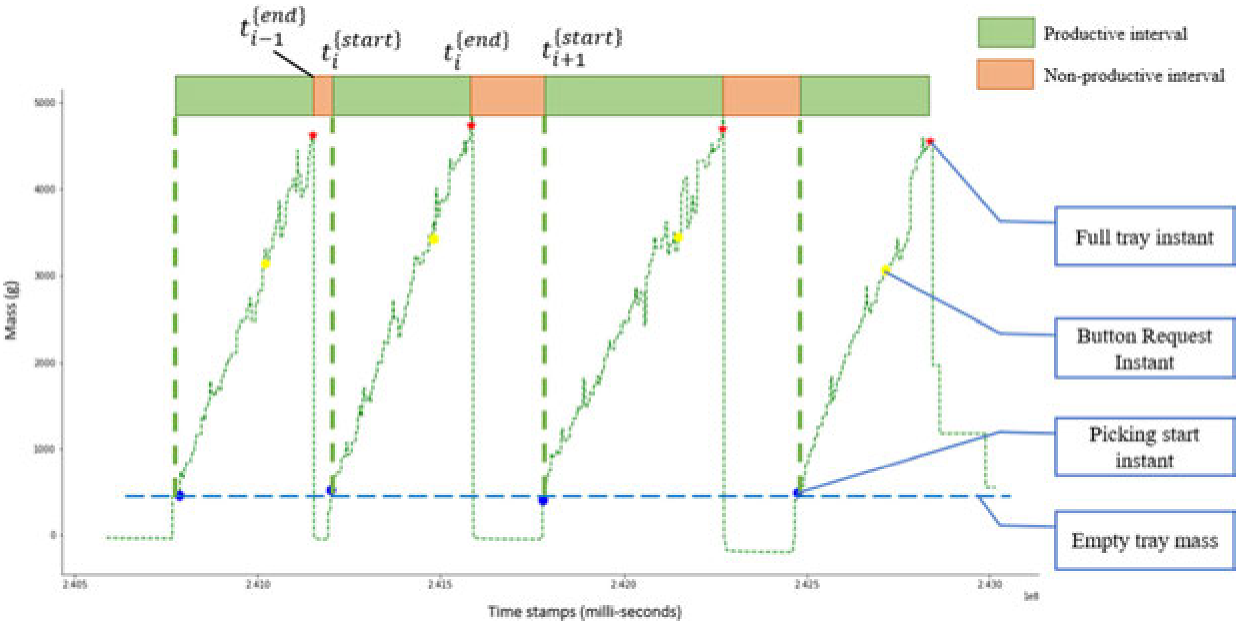


FIGURE 18 Visualization of tray mass measurements and time stamps from the collected data: blue dots represent picking start points; yellow dots represent tray-request events (button pressed); red stars represent picking end points. [Color figure can be viewed at wileyonlinelibrary.com]

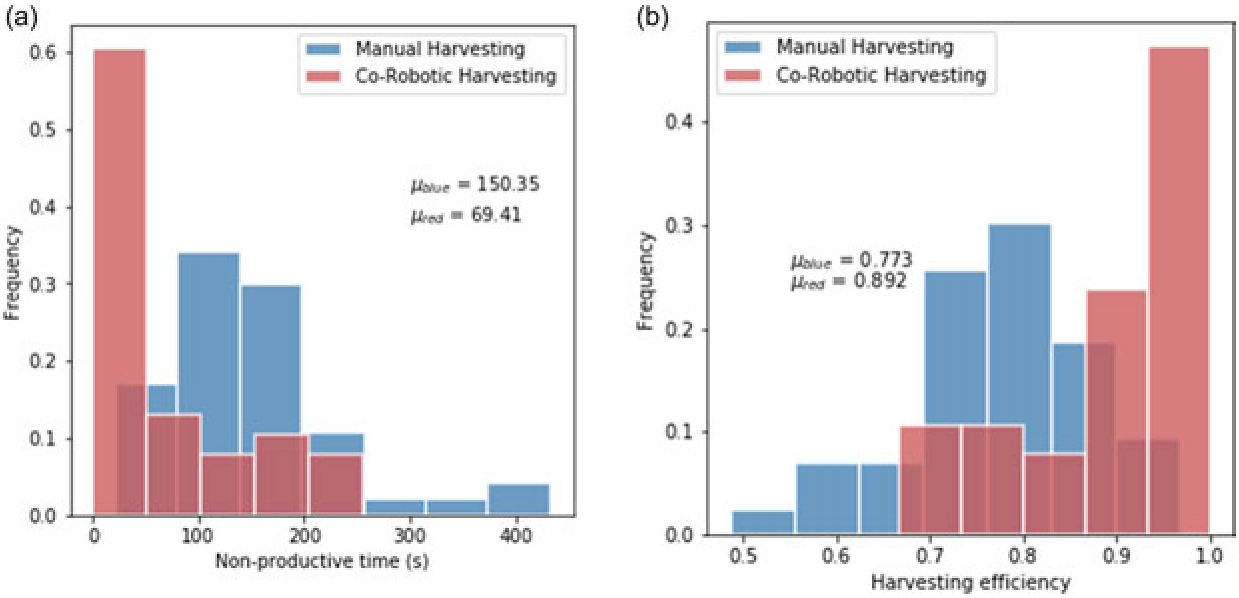


FIGURE 19 Harvesting performance on November 10: (a) histogram of non-productive time of co-robotic and manual harvesting; (b) histogram of harvesting efficiency of co-robotic and manual harvesting. [Color figure can be viewed at wileyonlinelibrary.com]

then randomly assign manual and robotic treatments to the blocks. However, such a large trial was not acceptable by the grower; commercial harvesting is a costly operation that is planned based on the weather, crop condition, labor availability, and customer demand. Hence, an estimation of the manual harvesting efficiency was made for the same field block that was harvested using robots, using the pickers' estimated walking speed and the measured locations where their trays had filled up.

TABLE 8 Mann–Whitney rank test results for the means of the measured and estimated non-productive time and harvesting efficiency of the co-robotic and manual harvesting, respectively, on November 10.

Item	Mean non-productive time	Mean harvesting efficiency
p value	3.778e−6	1.3705e−6

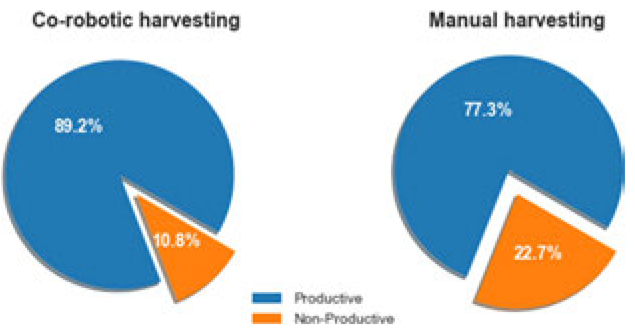


FIGURE 20 Comparison between the mean of harvesting efficiency of co-robotic and manual harvesting, based on experimental data, on November 10. [Color figure can be viewed at wileyonlinelibrary.com]

As mentioned above, the location L_i^f of each tray can be indexed from the instant $t_i^{\{end\}}$ the tray becomes full. Thus, the non-productive time and the manual harvesting efficiency for each tray were estimated from L_i^f, the collection station position and the estimated walking speeds (from Section 3.1). In summary, the picking crew's harvesting performance with the co-robotic harvest-aiding system was measured directly, and the manual harvesting performance of the same picking crew for the same field block was estimated indirectly, using the above method.

Given the significant difference in the yields of the two fields, which manifested itself in very different one-tray picking time and distance statistics, the harvesting performances of the two days were evaluated separately. The frequency histograms of non-productive time for the manual harvesting and co-robotic harvesting on November 10 are shown in Figure 19a, and the histograms of harvesting efficiency on that day are shown in Figure 19b. The *p* value of Mann–Whitney testing results of the performance data (non-productive time and harvesting efficiency) of manual harvesting and co-robotic harvesting on that day are shown in Table 8. Based on the calculated *p* values, the manual and co-robotic distributions of the non-productive time and efficiency are significantly different, with a significance level at 1%.

Their mean values are shown in the figure. The mean non-productive time of co-robotic harvesting was reduced by more than 50% of the manual harvest non-productive time. The mean co-robotic harvesting efficiency was 89.2%, 12% higher than the manual harvesting efficiency in absolute terms. These results are shown in the corresponding pie charts, in Figure 20.

Similarly, the performance distributions of manual and co-robotic harvesting on November 11 are shown in Figure 21, and the Mann–Whitney rank test results are shown in Table 9. Based on the

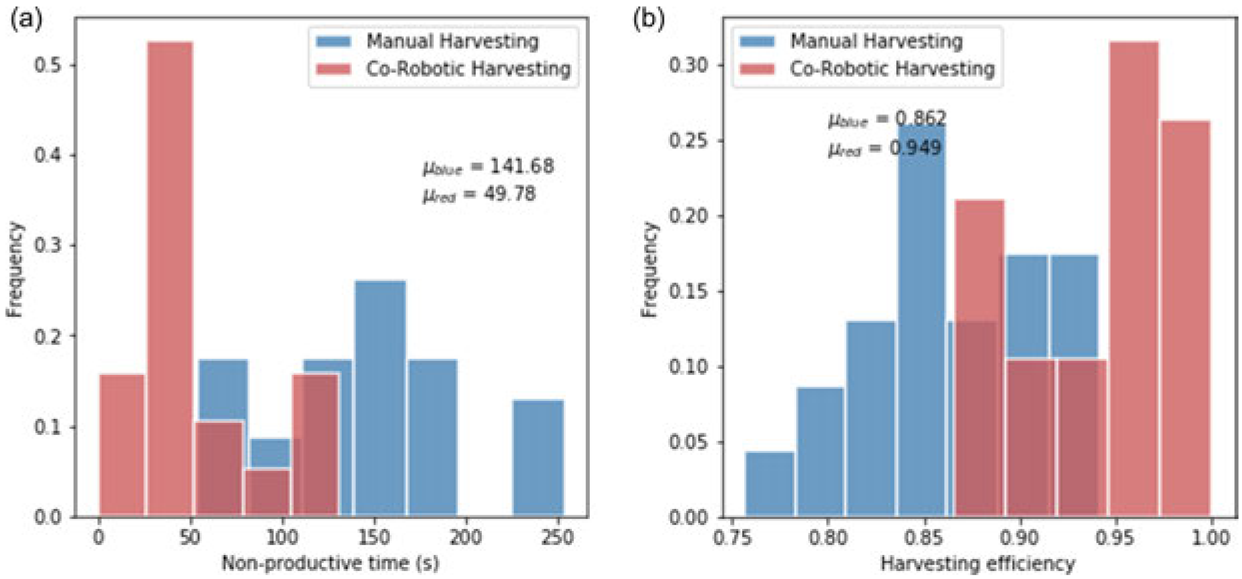


FIGURE 21 Harvesting performance on November 11: (a) histogram of non-productive time of co-robotic and manual harvesting; (b) histogram of harvesting efficiency of co-robotic and manual harvesting. [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 9 Mann–Whitney rank test results for the measured and estimated non-productive time and harvesting efficiency of the co-robotic and manual harvesting, respectively, on November 11.

Item	Non-productive time	Harvesting efficiency
T score	6.253	6.073
p value	2.409e−7	3.723e−7

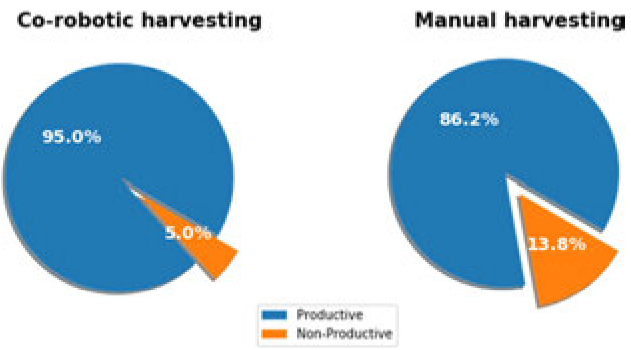


FIGURE 22 Comparison between the mean of harvesting efficiency of co-robotic and manual harvesting, based on experimental data, on November 11. [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 10 Mann–Whitney testing results for harvesting performances between each pair of picker crew sizes in manual harvesting and co-robotic harvesting

	Co-robotic harvesting				Manual harvesting			
	6	12	24	36	6	12	24	36
6	-	0.1042	0.092	0.0001	-	0.0015	0.0012	0.0001
12	-	-	0.235	0.0002	-	-	0.208	0.0048
24	-	-	-	0.0006	-	-	-	0.0052

calculated *p* values, the manual and co-robotic distributions of the non-productive time and efficiency are significantly different, at a significance level of 1%. From Figure 21b, one can see that the mean non-productive time with the robots was 33% lower than that of manual harvesting. Figure 22 shows that the mean harvesting efficiency after introducing the robots increased to 94.9% (from 86.2% during all-manual harvesting), an improvement of 8.8%, in absolute terms.

5.5.2 | Post-harvest simulations: Results and analysis

After the harvest field experiments, the recorded workers' harvesting parameters (reported in 5.4.2) were used by our simulator to explore how the efficiency gains of the harvest-aid system scale, when the size of the crew—and the field—increase. Manual and robot-aided harvesting were simulated for crew sizes of 6, 12, 24, and 36 people; larger crew sizes are very rarely used in commercial strawberry harvesting. A larger crew would only be used if the field were wider, so the width of the field was set to 10 rows times the number of pickers (6 pickers harvested 60 rows during our field experiments). The length of the rows was fixed at 100 m, because this length is the maximum row length in commercial strawberry production in CA. The parameters used in the simulations that relate to robot operation and scheduling were the same ones used during the field experiments. We also performed Mann–Whitney statistical tests of the harvesting efficiencies for different pairs of crew sizes to rigorously determine if the efficiencies are significantly different, at a significance value of 1%. Figure 22 shows the manual and co-robotic harvesting efficiencies for 6, 12, 24, and 36 pickers and Table 10 shows the statistical test results. The figure shows that the robot-aided harvest efficiency seems to drop slightly, as the crew size increases. The statistical tests suggest that the change is not significant when the

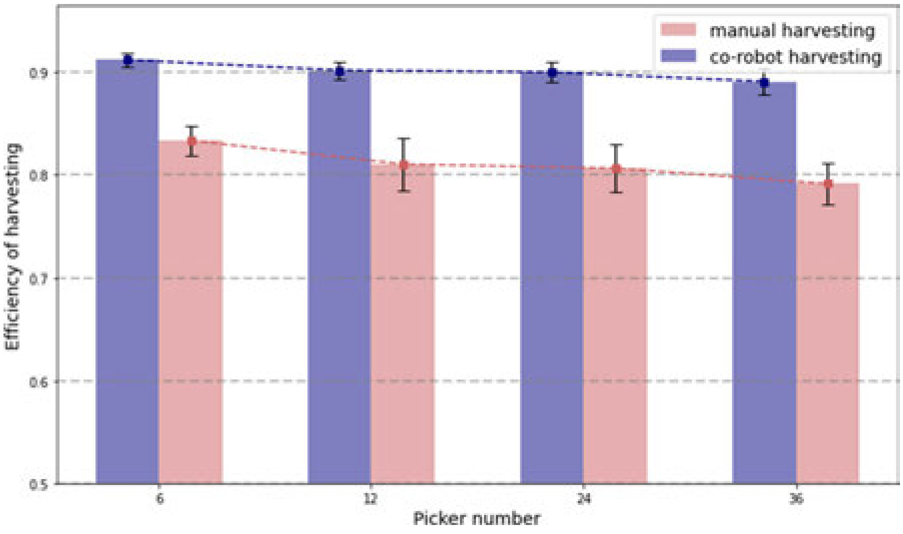


FIGURE 23 Simulation results for picker crew with size of 6, 12, 24, and 36. The blue bars are co-robotic harvesting efficiency with 1/3 picker ratios and red bars are manual harvesting performance for that picker crew. [Color figure can be viewed at wileyonlinelibrary.com]

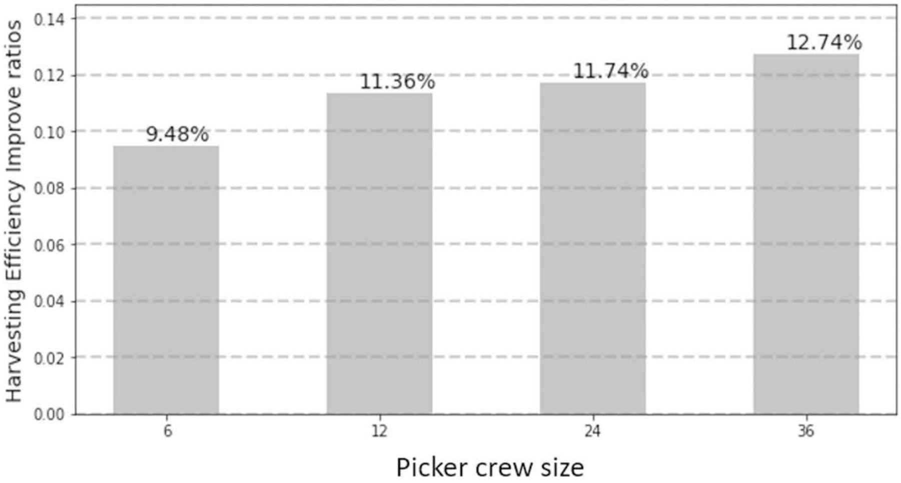


FIGURE 24 Efficiency gains for increasing picker crew sizes

crew increases from 6 to 12 or 24, or from 12 to 24. However, an increase to a crew's size of 36 lowers significantly the harvest efficiency of the robotic system. Similarly, the efficiency of manual harvesting decreases as the size of the crew increases. The explanation behind the trends of manual and co-robotic harvesting efficiencies shown in Figure 23 is given next.

Since each picker occupies one row, the width of the area covered by the rows that are being harvested at any point in time is larger when the crew is larger. In commercial production, the collection station is placed in the headland as close as possible to the middle row of the currently harvested area, to reduce walking in the headland. So, larger crews result in wider harvesting areas, which in turn results in larger average travel distance from a row's end to the collection station in the headland. Workers and robots must travel more as crew size increases, and as a result, the efficiencies of both manual and robot-aided harvesting drop. During the specific experiment, the robot travel speed was 1.5 m/s inside row and 0.8 m/s on the headland, whereas the average estimated picker walking speed was smaller and ranged from 0.49 to 1.02 m/s (Table 7). For this reason, *the efficiency gain of the co-robotic system increased as the crew and field size increased*; the efficiency gains are shown in Figure 24.

6 | CONCLUSIONS AND FUTURE WORK

This study presented the development of a harvest-aid system with a team of co-robotic crop-transport robots, and strawberry harvesting as a target application. Dynamic predictive robot scheduling was mathematically modeled as a dynamic stochastic scheduling problem with uncertain (tray-transport) requests. The problem was solved with a multiple scenario-sampling approach (MSA), which computed near-optimal solutions in real-time. Simulation experiments were performed to select system parameters and explore the efficiency gain as the harvest crew and field become larger. The harvest-aid system was built and deployed successfully during commercial harvesting.

The simulations showed that the harvest efficiency of stochastic predictive scheduling with tray-transport request rejections was much higher than that of the manual harvesting, even at a robot/picker ratio as low as 4:25. Scheduling with request rejections was also significantly better than without rejections when the robot/picker ratio was smaller than 10:25, and the same when more robots were used. Even if these specific ratios change under different harvesting scenarios (e.g., larger diverse crews, wider fields), rejecting tray-transport requests when their manual delivery is predicted to take less time than robotic delivery is a sensible policy, which was shown to perform better than manual harvesting and at least as well as scheduling without rejections.

Also, the experimental results showed that during low-yield season, the harvest-aid robotic system operating with the parameters described in this study reduced the non-productive time of the pickers by over 60% and improved the harvesting efficiency by an average of 10.35% (8.8% and 11.9%) at a robot/picker ratio equal to 1:3. Finally, simulations based on the data collected during the field harvest experiments showed that the efficiencies of manual and co-robotic harvesting are expected to drop as the crew and field size increase, because the average distance from the exit of a field row to the tray collection station increases. However, if the robots are programmed to travel faster than the pickers in the headland, the efficiency gain of co-robotic harvesting is expected to increase as the crew and field size increase.

Future research will focus on two key improvements, which stem from observations and lessons learned during the field experiments. First, the capacity of the current FRAIL-Bot design is more than one tray, and therefore, the robot does not need to return to the unloading station immediately after getting one tray from a picker. Serving multiple pickers before returning to the collection station is expected to increase efficiency. The scheduling problem when capacity is larger than one is more complex and would need to be solved in real-time. This extension would render the methodology presented in this paper applicable to a wider class of crop-transport in-field logistics applications.

Second, the interaction of human pickers and FRAIL-Bots needs to be improved. For example, the robots were programmed to stop five meters away from the predicted location of the transport request, for safety purposes. Sometimes (e.g., when the fruit load was low) the prediction of the location of the request was too "aggressive," that is, the robot stopped too close to the picker. The picker needed to move farther to fill their tray, but the robot was blocking their way. Other times (e.g., when the fruit load was high), the prediction was overly "conservative," and the picker had to walk a longer distance to the robot. In a couple of instances, we had to stop the robot, because it seemed that it might stop "behind the picker," that is, it would collide with them. Better use of existing sensor data and the addition of perception modalities (e.g., onboard cameras) to estimate the operating states of the pickers could enhance human-robot collaboration and safety.

Finally, more commercial harvesting experiments in different fields, in different seasons (low, mid, high yield), with various crew sizes and compositions (slow, fast pickers) and numbers of robots would be needed to assess the range of improvements in harvesting efficiency when the robots are deployed. The requirements for human and equipment resources and the cost for undertaking such a long-term experiment were prohibitively high to incorporate in this study, but we plan to initiate such an extension project in the near future.

ACKNOWLEDGMENTS

Part of this study was funded by the National Institute of Food and Agriculture (NIFA) grant 2020-67021-30759, under the National Robotics Initiative; NIFA Hatch/Multi-State Grant 1001069, and the National Institute for Occupational Safety and Health (NIOSH) award U54OH007550.

DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

ORCID

Stavros Vougioukas http://orcid.org/0000-0003-2758-8900
Zhenghao Fei http://orcid.org/0000-0001-9132-354X

REFERENCES

Arad, B., Balendonck, J., Barth, R., Ben-Shahar, O., Edan, Y., Hellström, T. et al. (2020) Development of a sweet pepper harvesting robot. *Journal of Field Robotics*, 37(6), 1027–1039. Available from: https://doi.org/10.1002/rob.21937

ASAE, S. (2009) ASAE D497. 6 Agricultural Machinery Management Data. ASAE. St. Joseph, 1–8.

Bac, C.W., Henten, E.J., van, Hemming, J. & Edan, Y. (2014) Harvesting robots for high-value crops: state-of-the-art review and challenges ahead. *Journal of Field Robotics*, 31(6), 888–911.

Baugher, T., Schupp, J., Lesser, K., Harsh, R.M., Seavert, C., Lewis, K. et al. (2009) Mobile platforms increase orchard management efficiency and profitability. *Acta Horticulturae*, 824, 361–364. Available from: https://doi.org/10.17660/ActaHortic.2009.824.42

Bayar, G., Bergerman, M., Koku, A.B. & Konukseven, E.I. (2015) Localization and control of an autonomous orchard vehicle. *Computers and Electronics in Agriculture*, 115, 118–128. Available from: https://doi.org/10.1016/j.compag.2015.05.015

Bent, R. & Van Hentenryck, P. (2004a, June) The value of consensus in online stochastic scheduling. *Proceedings of the 14th International Conference on Automated Planning and Scheduling, ICAPS*, 2004, 1, 219–226.

Bent, R.W. & Van Hentenryck, P. (2004b) Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52(6), 977–987. Available from: https://doi.org/10.1287/opre.1040.0124

Bertsimas, D.J. & Van Ryzin, G. (2017) A stochastic and dynamic vehicle routing problem in the Euclidean plane. *JSTOR*, 39(4), 601–615. Available from: http://www.jstor.org/stable/171167

Blazewicz, J., Ecker, K.H., Pesch, E., Schmidt, G., Sterna, M. & Weglarz, J. (2019) *Handbook on Scheduling*. Available from: https://doi.org/10.1007/978-3-319-99849-7.

Bolda, M., Tourte, L., Murdock, J. & Sumner, D. (2016) Sample costs to produce strawberries and harvest strawberries: central Coast Region, Santa Cruz & Monterey Counties, University of California.

Buzacott, J.A. & Yao, D.D. (1986) Flexible manufacturing systems: a review of analytical models. *Management Science*, 32(7), 890–905.

Charlton, D., Vougioukas, S. & Rutledge, Z. (2019) Innovations for a shrinking agricultural workforce. *Choices*, 34(2), 1–8.

Das, G.P., Cielniak, G., From, J. & Hanheide, M. (2018) Discrete event simulations for scalability analysis of robotic in-field logistics in agriculture—a case study. *ICRA 2018 Workshop on Robotic Vision and Action in Agriculture*.

Du, J., Leung, J.Y.-T. & Young, G.H. (1991) Scheduling chain-structured tasks to minimize makespan and mean flow time*. *Information and Computation*, 236, 219–236.

Fei, Z. & Vougioukas, S.G. (2021) Co-robotic harvest-aid platforms: real-time control of picker lift heights to maximize harvesting efficiency. *Computers and Electronics in Agriculture*, 180, 105894. Available from: https://doi.org/10.1016/j.compag.2020.105894

Ghaleb, M., Zolfagharinia, H. & Taghipour, S. (2020) Real-time production scheduling in the Industry-4.0 context: addressing uncertainties in job arrivals and machine breakdowns. *Computers and Operations Research*, 123, 105031. Available from: https://doi.org/10.1016/j.cor.2020.105031

Guan, Z., Wu, F., Roka, F. & Whidden, A. (2015) Agricultural labor and immigration reform. *Choices*, 30, 1–9.

Investigation on the harvest-aid robot scheduling problem and the implementation of its simulation platform. Davis: University of California (M.Sc. thesis).

Khosro Anjom, F. & Vougioukas, S.G. (2019) Online prediction of tray-transport request time using mechanistic grey box models for improved scheduling of robotic strawberry harvest-aids. *Biosystems Engineering*, 188, 265–287. Available from: https://doi.org/10.1016/j.biosystemseng.2019.10.025

Khosro Anjom, F., Vougioukas, S.G. & Slaughter, D.C. (2018) Development and application of a strawberry yield-monitoring picking cart. *Computers and Electronics in Agriculture*, 155(August), 400–411. Available from: https://doi.org/10.1016/j.compag.2018.10.038

Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G. & Shmoys, D.B. (1993) Sequencing and scheduling: algorithms and complexity. *Handbooks in Operations Research and Management Science*, 4(C), 445–522. https://doi.org/10.1016/S0927-0507(05)80189-6

Li, Z., Vatankhah, A., Jiang, J., Zhong, R.Y. & Xu, G. (2020) A mechanism for scheduling multi robot intelligent warehouse system face with dynamic demand. *Journal of Intelligent Manufacturing*, 31, 469–480.

Olsen, T.L. & Tomlin, B. (2020) Industry 4.0: opportunities and challenges for operations management. *Manufacturing and Service Operations Management*, 22(1), 113–122. Available from: https://doi.org/10.1287/msom.2019.0796

Pasparakis, A., de Vries, J. & de Koster, M.B.M.R. (2021) In control or under control? Human-robot collaboration in warehouse order picking. *SSRN Electronic Journal*. Available from: https://doi.org/10.2139/ssrn.3816533

Peng, C. (2021) *Predictive scheduling of collaborative mobile robots for improved crop-transport logistics of manually harvested crops*. Doctoral dissertation, University of California, Davis. arxiv:2111.09959.

Peng, C. & Vougioukas, S.G. (2020) Deterministic predictive dynamic scheduling for crop-transport co-robots acting as harvesting aids. *Computers and Electronics in Agriculture*, 178. Available from: https://doi.org/10.1016/j.compag.2020.105702

Peng, C., Vougioukas, S.G., Fei, Z. & Gatten, B. (2020) *System-level description and evaluation of a robot-aided strawberry harvesting system*. Bio Automation Lab. Available from: https://doi.org/10.13140/RG.2.2.11636.17281

Phillips, C., Stein, C. & Wein, J. (1998) Minimizing average completion time in the presence of release dates. *Mathematical Programming, Series B*, 82(1–2), 199–223. Available from: https://doi.org/10.1007/BF01585872

Pillac, V., Gendreau, M., Guéret, C. & Medaglia, A.L. (2013) A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1), 1–11. Available from: https://doi.org/10.1016/j.ejor.2012.08.015

Ritzinger, U., Puchinger, J. & Hartl, R.F. (2016) A survey on dynamic and stochastic vehicle routing problems. *International Journal of Production Research*, 54(1), 215–231. Available from: https://doi.org/10.1080/00207543.2015.1043403

Rosenberg, H.R. (2003) Adjusting to technological change in strawberry harvest work. *Giannini Foundation of Agricultural Economics*, University of Calirfornia.

Seyyedhasani, H., Peng, C., Jang, W. & Vougioukas, S.G. (2020a) Collaboration of human pickers and crop-transporting robots during harvesting–Part I: model and simulator development. *Computers and Electronics in Agriculture*, 172, 105324.

Seyyedhasani, H., Peng, C., Jang, W. & Vougioukas, S.G. (2020b) Collaboration of human pickers and crop-transporting robots during harvesting—Part II: simulator evaluation and robot-scheduling case-study. *Computers and Electronics in Agriculture*, 172, 105323.

Silwal, A., Davidson, J.R., Karkee, M., Mo, C., Zhang, Q. & Lewis, K. (2017) Design, integration, and field evaluation of a robotic apple harvester. *Journal of Field Robotics*, 34(6), 1140–1159. Available from: https://doi.org/10.1002/rob.21715

USDA-REEIS. (2013) NRI: Small: FRAIL-bots: Fragile cRop hArvest-aIding mobiLe robots. Available at: https://reeis.usda.gov/web/crisprojectpages/1000459-nri-small-frail-bots-fragile-crop-harvest-aiding-mobile-robots.html

Vougioukas, S., Spanomitros, Y. & Slaughter, D. (2012) Dispatching and routing of robotic crop-transport aids for fruit pickers using mixed integer programming. *American Society of Agricultural and Biological Engineers Annual International Meeting 2012, ASABE 2012*, 4(12), 3423–3432. Available from: https://doi.org/10.13031/2013.41895

Wang, Z., Sheu, J.-B., Chung-Piaw, T. & Guiqin, X. (2021) Robot scheduling for Mobile-Rack warehouses: human-robot coordinated order picking systems. *Production and Operations Management*, 31(1), 98–116. Available from: https://doi.org/10.1111/poms.13406

Weidinger, F., Boysen, N. & Briskorn, D. (2018) Storage assignment with rack-moving mobile robots in KIVA warehouses. *Transportation Science*, 52(6), 1479–1495. Available from: https://doi.org/10.1287/trsc.2018.0826

Williams, H., Ting, C., Nejati, M., Jones, M.H., Penhall, N., Lim, J.Y. et al. (2020) Improvements to and large-scale evaluation of a robotic kiwifruit harvester. *Journal of Field Robotics*, 37(2), 187–201. Available from: https://doi.org/10.1002/rob.21890

Xiong, Y., Ge, Y., Grimstad, L. & From, P.J. (2020) An autonomous strawberry-harvesting robot: design, development, integration, and field evaluation. *Journal of Field Robotics*, 37(2), 202–224. Available from: https://doi.org/10.1002/rob.21889

Yadav, A. & Jayswal, S.C. (2018) Modelling of flexible manufacturing system: a review. *International Journal of Production Research*, 7543, 1–24. Available from: https://doi.org/10.1080/00207543.2017.1387302

Ye, Y., Wang, Z., Jones, D., He, L., Taylor, M.E., Hollinger, G.A. et al. (2017) Bin-dog: a robotic platform for bin management in orchards. *Robotics*, 6(2), 1–17. Available from: https://doi.org/10.3390/robotics6020012

Zourmand, A., Hing, A.L.K., Hung, C.W., & AbdulRehman, M. (2019, June) Internet of things (IoT) using LoRa technology. In 2019 *IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)* (pp. 324–330). IEEE.

How to cite this article: Peng, C., Vougioukas, S., Slaughter, D., Fei, Z. & Arikapudi, R. (2022) A strawberry harvest-aiding system with crop-transport collaborative robots: Design, development, and field evaluation. *Journal of Field Robotics*, 1–27. https://doi.org/10.1002/rob.22106