A Practical Application of Market-based Mechanisms for Allocating Harvesting Tasks

Helen Harman[0000−0002−3195−2579] and Elizabeth I Sklar[0000−0002−6383−9407]

Lincoln Institute for Agri-food Technology,
University of Lincoln, Lincoln, UK
hharman@lincoln.ac.uk
esklar@lincoln.ac.uk

Abstract. Market-based task allocation mechanisms are designed to distribute a set of tasks fairly amongst a set of agents. Such mechanisms have been shown to be highly effective in simulation and when applied to multi-robot teams. Application of such mechanisms in real-world settings can present a range of practical challenges, such as knowing what is the best point in a complex process to allocate tasks and what information to consider in determining the allocation. The work presented here explores the application of market-based task allocation mechanisms to the problem of managing a heterogeneous human workforce to undertake activities associated with harvesting soft fruit. Soft fruit farms aim to maximise yield (the volume of fruit picked) while minimising labour time (and thus the cost of picking). Our work evaluates experimentally several different strategies for practical application of market-based mechanisms for allocating tasks to workers on soft fruit farms, identifying methods that appear best when simulated using a multi-agent model of farm activity.

Keywords: Task Allocation Mechanism · Multi-Agent System · Agent-Based Simulation.

1 Introduction

Due to the increasing demand for soft fruits and shortages in seasonal workers [6, 17, 27], farms are requiring innovative solutions for managing their fruit harvesting processes. Typically, on such farms, each day a harvest manager determines which fields are ready for picking and how many workers should be assigned to each field. In the field, supervisors assign tasks to individual workers, who place harvested fruit into containers ("punnets", in the case of strawberries) that must be carried (i.e. transported) to a centralised location, such as a permanent pack house or mobile trailer adjacent to the fields, where they are weighed, scanned for quality and tallied so that the picker responsible is compensated correctly. The task of transporting punnets to the central location is often performed by a worker called a "runner". This task allocation problem thus involves decisions about which tasks to assign to which workers and how many workers to assign to each role (picker and runner). In the not too distant future, robots may soon be filling gaps in the shortages of seasonal workers [4, 17, 20, 30, 32]; and therefore, robotic co-workers will need to be managed alongside the human workforce.

Multi-robot task allocation (MRTA) problems address situations in which a group of robots must work together to complete a *mission*—a set of *tasks* to be executed. A key challenge is to decide which tasks should be assigned to which robots so that the overall execution of the mission is efficient: resources are used effectively, so that time and energy are not wasted and, often, some reward is maximised. A range of methods for allocating tasks in multi-robot teams are described in the literature, for example handling heterogeneous teams of robots [26] and multi-robot swarms [24], assigning tasks dynamically [34] and limiting robots to local input from immediate neighbours [2]. Recent real-world applications include disinfecting public areas in order to reduce spread of contagious diseases [28] and delivering food [14].

In the work presented here, we posit that approaches designed to address task allocation in a multi-robot team can be adapted to manage the human workforce on a soft fruit farm. Fruit picking and transporting harvested fruit are two types of tasks that need to be allocated to workers, who are often assigned one of two roles (*picker* or *runner*, respectively). Here, we apply market-based MRTA strategies and investigate two questions: (1) What is the most efficient ratio of runners to pickers? (2) What is the most efficient strategy for allocating tasks to runners? We investigate these questions empirically, using a multi-agent based simulation that emulates the activity of human workers on a soft fruit farm.

The paper is organised as follows. Section 2 provides brief background on multi-robot task allocation problems, focussing on market-inspired approaches, and also highlights related work in *agricultural robotics*. Section 3 describes the methodology we employed for our simulation. Section 4 explains our experiment design, and section 5 presents the results of our simulation experiments. Section 6 closes with a summary of results.

2 Background

The *multi-robot task allocation (MRTA)* problem has been classified in the literature according to several taxonomies that distinguish specific features of tasks and task environments [9, 18, 21]. From that literature, the parameters that are particularly relevant for the work presented here are: static (SA) vs dynamic (DA) assignment—whether all the tasks are known at the start of a mission (static) or new ones may appear during the mission (dynamic); independent (IT) vs constrained (CT) task—whether or not the assignment of one task is dependent on the completion of another; and the further distinction between in-schedule (ID), cross-schedule (XD) and complex (CD) dependencies for CT tasks. Our soft fruit farm task allocation scenario is unusual because it combines SA and DA tasks within an XD environment (runner tasks are dependent on picker tasks and vice-versa).

Market-based mechanisms, especially *auctions*, are a popular approach to the MRTA problem [5, 13]. Auctions are typically executed in *rounds*, comprised of three phases: (1) a centralised auction manager advertises one or more tasks to a set of robots (or agents); (2) each agent determines its individual (private) valuation (cost or utility) for one or more of the announced tasks and presents that valuation in the form of a *bid* to the auction manager; and (3) then the auction manager compiles the bids and decides which tasks to assign to which agents. Multiple rounds can occur, until all the tasks advertised are assigned. One prominent auction-based method is the *sequential single-item (SSI)* method [16] (described in Section 3). SSI has been a popular choice for multi-robot task allocation, and many variants have been studied, for example *TeSSI* [25], to efficiently allocate a set of tasks with temporal constraints to a team of robots, and *sequential single-cluster (SSC)* auctions [12] for solving pick-up and delivery tasks in a dynamic environment.

One area of application for multi-robot teams that has been gaining attention recently is *agricultural robotics* [6]. State-of-the-art work includes use of autonomous robots and machine learning methods, for example to identify ripe fruit [15], map regions in need of irrigation [3]. or locate weeds [22]. A wide range of robotic solutions for picking and transporting crops are currently being developed, including harvesting sweet peppers [7, 20], and other fruiting vegetables [32], When harvesting crops, if a container has been filled, it must be transported to a storage and/or packing location. Some have evaluated hybrid human-robot solutions, where robots perform the transporting tasks while humans do the picking [4, 30, 31].

3 Methodology

In order to investigate our two research questions, we have constructed a multi-agent based simulation of operations on a soft fruit farm, where each human worker is represented by an agent. We assume that there are two different roles for workers (picker and runner), that each task can be completed by one worker on their own and that each worker performs one type of task (picking or transporting, respectively). Pickers harvest fruit in the field (in this case, a type of greenhouse called a *polytunnel*) and place the produce in punnets; and runners collect trays of full punnets and deliver them to a centralised location called a *packing station*. Our simulator was developed using MASON [23], a discrete-event multi-agent simulation library. We adapted a market-based task allocation mechanism from [29], to advertise a set of fruit picking tasks. Agents bid on these tasks and an auction manager assigns each task to the agent that presents the bid with the lowest cost.

In practice on farms, picking tasks are determined each day by inspecting the rows of crops, to discover the amount of ripe fruit they contain. In our simulation, picking tasks are represented by patches (areas) of *unoccluded* (readily visible) and *occluded* (hidden) fruits that are ripe. Transport tasks are created when a picker's schedule contains a task that will cause its capacity to be reached. According to the taxonomies cited in Section 2, we characterise picking task assignment as SA, because this is done *a priori*. Transport task assignment could be characterised either as SA, allocated before the mission when picker tasks are assigned, or DA, allocated during the mission, as pickers fill trays.

3.1 Agents

We define two roles for agents in our simulation:

- A **picker** is defined by the tuple $p = \langle v, l, s_p, c\rangle$, where l is the agent's initial location and v its navigation speed; $s_p = \langle s_o, s_u\rangle$, for which s_o is the speed at which the agent can pick occluded fruit (number of fruits per step); and s_u the agent's unoccluded fruit picking speed. When a picker has reached their capacity (c) they cannot pick any more fruits. Pickers cannot leave trays/punnets on the ground since customers are unwilling to accept fruit covered in mud, and potentially contaminated with pests and disease. They also require empty punnets to be delivered to them. Thus, the agent must wait for a runner to collect the ripe fruits and take them to the pack house.
- A **runner** navigates to a picker, collects the punnet and then returns to the pack house. Runners have a navigation speed and an initial location, i.e., $r = \langle v, l\rangle$. For runners, their initial location is always within the pack house.

3.2 Task Allocation Mechanisms

Similarly to our earlier work [11], we compare the variations in performance resulting from the application of three different auction-based mechanisms to the process of allocating picker and transporter tasks.

- *Round Robin* (RR) assigns the first task to the first agent, the second to the second agent and so forth. After a single task has been assigned to each agent, the agents are re-iterated over to assign each of them a second task. This process continues until all tasks have been assigned to an agent.
- In *Ordered Single Item* (OSI), all agents bid on the first task and the agent with lowest costing bid is assigned the task. The subsequent task is then auctioned. When all tasks are assigned, the process concludes.
- For *Sequential Single Item* (SSI), in each round all unassigned tasks are bid on by all agents. The task of the lowest costing bid is assigned to the agent who placed that bid.

3.3 Allocation of Picking Tasks

Pickers are allocated work by bidding on, winning, and thus being assigned, picking tasks. A picking task is defined as an (x, y) location and a number of ripe fruits. Before bidding begins, the list of picking tasks is sorted, highest first, by the total number of ripe fruits they contain. Pickers are sorted by picking speed, s, which is a combination of speeds for picking unoccluded, s_u, and occluded, s_o, fruits; quickest picker appears first. The cost of a picking bid is the *duration* for the agent to complete all their previously assigned tasks plus the task being auctioned. The duration of a single picking task is the sum of three components:

- The time it takes the agent to navigate to their picking location (d_v). Navigation duration is calculated by dividing the length of the path by the agent's navigation speed (v): $d_v = len(path)/v$.
- The time it takes to pick the ripe fruits (d_p). Picking duration is calculated by combining the time spent picking unoccluded fruits with the time to pick occluded fruits: $d_p = (u/s_u) + (o/s_o)$.
- The time spent waiting for a runner, but only if two conditions are met: (i) the agent's capacity will be reached whilst picking that patch; and (ii) the runner scheduling interweaves the picker scheduling (see Section 3.4).

As precise AI path planning (e.g. [10]) causes the bidding process to be computationally expensive, Euclidean distance is calculated as a proxy for the path length. If the agent has not won any tasks (yet), two Euclidean distances are summed: (i) the distance from the picker's initial location to the row in which the new task is located, and (ii) from the end of the new task's row to the location within the row of the new task. For navigating between locations within the same aisle, a single distance is measured. For patches in different aisles, three distances are summed: the distance from the previous location to the end of its row, from the row of the previous location to the end of the row containing the new location, and from that row end to the location itself. When the mission is executed, *Jump Point Search (JPS)* [10] is called to find the precise path.

If executing a task would cause a picker's capacity to be reached, it creates a *provisional transport task* whilst constructing its bid. To facilitate this, the number of fruits the agent will be holding when it completes its schedule and the time step the agent will finish on are updated each time it is assigned a task. To determine the time spent picking before the agent's capacity is reached, we assume that pickers harvest unoccluded fruits before picking the occluded fruits from a patch. Along with the navigation time, this is added to the time the picker will start the task (i.e. the timestep after its previously scheduled task will end). Ideally, a runner will take the picked fruit from the picker on the timestep directly after the picker has reached capacity. In reality, often a picker has to wait for a runner; or vice versa. If the picker's bid wins, then the transport task is no longer provisional; it is appended to a list of transport tasks. When a picker will reach capacity more than once when executing a task, multiple transport tasks are created.

3.4 Allocation of Transport Tasks

Transport tasks contain the location and timestep that a picker will reach maximum capacity. The less time a picker spends waiting for a runner, the sooner it will be able to complete its task. Therefore, the winning transport bid is the bid that causes the picker the shortest delay. If multiple bids have an equally short delay, then the bid with the shortest duration wins. For a transport bid, duration is the sum of the time it takes the runner to navigate to the picker, collect the punnet and return to the packing station. Runners are sorted by navigation speed, quickest appearing first.

Three different modes were implemented and compared for allocating tasks to runners. To differentiate between these and the mechanisms implemented for allocating picking tasks, each adds a prefix to the mechanism name (e.g. W-RR):

- *Whilst scheduling picking* (W): Runners can be scheduled as soon as a transport task is created. This enables a picker's bid to include the time they would spend waiting for the runner.
- *Post scheduling picking* (P): The auction manager can wait until all transport tasks have been created (i.e. all picking tasks have all been assigned) before scheduling the runners.
- *Whilst executing picking* (E): Runners can be scheduled during execution, which facilitates delays (differences between the scheduled duration and execution duration) to be accounted for within the runners' schedules.

The transport bid creation algorithm determines where within the runner's existing schedule the task should be placed. The algorithm iterates over all the runner's already scheduled tasks, selecting those with start time after the ideal end time of the task being auctioned and checking where the new task will fit within this selected list. A record of the location/index is kept, so that if the agent's bid wins, the task can be inserted into the schedule easily.

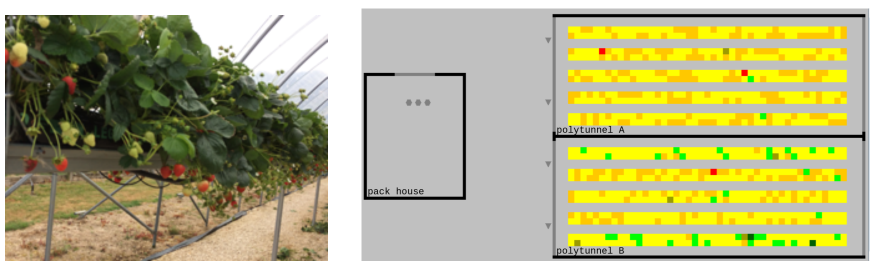
The delay to the picker, in waiting for the runner to complete its task, is calculated by finding the difference between the time the transport is required and how soon after this time the runner can arrive. If the runner can arrive on time, then the delay is the time it takes to hand over the punnet.

For the three modes (W, P and E), implementations of RR and OSI were developed. In the W and E modes, OSI and SSI are equivalent since only one task at a time is offered to the bidders. The algorithms employed to auction transport tasks are essentially equivalent to the those developed for auctioning picking tasks. In the P mode, before bidding begins, the transport tasks are sorted by the timestep at which the runner is required. Unlike the W and E modes, when a runner is assigned a task, the picker who created the task is required to update its schedule to take into account the delay. The delay amount is added to the start, end and transport-required times of all the tasks proceeding the delayed picking task. The transport-required times of the corresponding (unassigned) transport tasks are updated simultaneously. P-SSI is not performed since a runner's tasks must be in order of when a picker reaches capacity (to prevent deadlocks).

In the E mode, the transport task is only offered to the runners when the picker (actually) reaches capacity. When a runner has no tasks to execute, it will navigate to and wait in front of the polytunnels, so that it has less distance to travel when a picker reaches capacity. These locations are predefined and iterated over (then re-iterated over) to assigned them to the runners. In future work, we will consider selecting different "waiting" locations that take into consideration the current locations of the pickers.

4 Experiments

We designed a series of experiments to evaluate our research questions concerning the ratio of runners to pickers and the strategy for allocating tasks to runners. Two experimental scenarios were defined, below, and results of experiments are presented in Section 5. Two key metrics are computed: **execution time**—how long it takes to perform the tasks allocated; and **waiting time**—how long pickers spend waiting for runners. If the system is efficient, then the execution time and wait time is minimised and yield is maximised. To determine the significance of our results, we applied statistical testing and factor analysis, where appropriate. A Shapiro-Wilk test [33] was performed to check if each sample is normally distributed. If there is a greater than 90% chance that the samples are all normally distributed, an ANalysis Of VAriance (ANOVA) test [1, 8] was performed (for which the F test statistic is reported). Otherwise, Kruskal-Wallis tests [19] were run (for which the H test statistic is reported). T-test are performed when there are only two samples. The significance of results is indicated by p, the probability of the results occurring randomly.



(a) Inside the polytunnel. (b) Layout. Agents' starting locations are indicated.

Fig. 1. Our strawberry farm. See text for explanation.

We developed two scenarios, one emulating a **small farm** and one a **large farm**. Both are based on existing soft fruit farms and the data used in the scenarios come from each of these farms.

The **Small Farm** (pictured in Fig. 1a) is a small research farm. During Summer 2020, the volume of ripe fruits that were picked per row of crops were recorded. This included information on how many of the fruits were occluded from view. Data was recorded on each picking day (twice per week). In our initial experiments, there was no statistically significant difference between the results for different dates. Therefore, for the experiments presented here, we selected a single date in which a large number of fruits were harvested. The data per row was broken down into patches by adding each fruit to a randomly selected patch from the same row (as depicted in Fig. 1b). The colour of the patches, in Fig. 1b, represents the number of ripe fruits: red patches contain more ripe fruits than orange patches, which contain more than yellow patches and green indicates the patches containing low amounts of ripe fruits. As an element of randomness was included, two random distributions were produced (illustrated as *heatmaps*, like that in Fig. 1b). For this scenario, we employed a 7-agent team of workers.

The **Large Farm** replicates aspects of a commercial fruit farm and we have modeled one of their fields as an example. This single field is about 100 times as large as the small research farm. Based on data provided from this farm, we calculated the average yield per date and the average picking speed. Within our simulation of this field, the yield was uniformly distributed across patches. For all agents, navigation speed was set to ≈1 meter per timestep. The capacity of pickers is set to the volume (4000 grams) of a standard tray (which contains the punnets of picked fruits). On average, 39 pickers picked each day from this field; therefore our experiment for this scenario contains 39 agents.

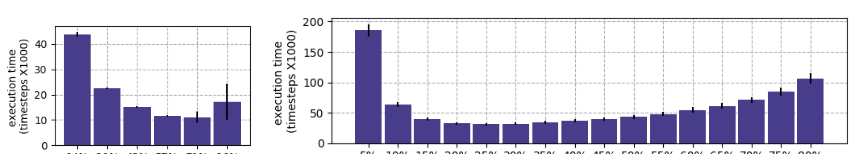
5 Results

We analyse our results by looking first at the composition of our workforce (number of pickers and transporters); second at the impact of allocating tasks to transporters at different points in the picking process; and third at the evenness of the distribution of tasks in terms of how much time pickers spend waiting for runners. In all plots presented, error bars indicate ±1 standard deviation.

The whilst scheduling pickers (W) runner mode, is more computationally expensive than the E and P modes, since for every bid that a picker creates (for which transport is required) the transportation task auction is invoked; whereas, for E and P, only the transport tasks of winning picking bids are auctioned. The deliberation time (i.e. the time it takes to allocate the tasks) of RR, OSI and SSI has previously been compared and is nominal in the scheme of the overall run time of our scenarios; thus, deliberation time is not analysed here [29].

5.1 Workforce composition

As shown in Fig. 2a, the ideal team split, for the small farm is 71% of agents deployed as runners and the remaining agents as pickers; and for the large farm is 25% of agents deployed at runners. Although the best percentages differ—due



(a) small farm
($H = 56.21, p = 0.00000$)

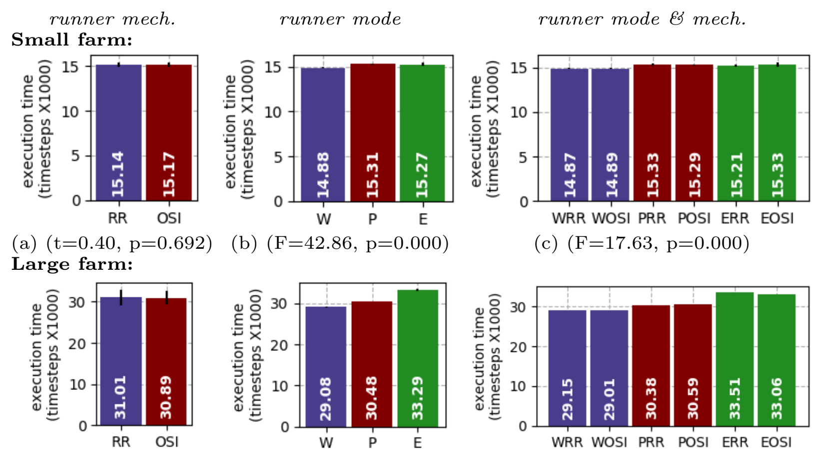
(b) large farm
($H = 91.07, p = 0.00000$)

Fig. 2. Results for **execution time** for different percentages of agents being employed as runners. The H statistic from Kruskal-Wallis tests and associated p values are shown, indicating statistically significant differences for the different ratios for both farms.

to the large difference in size between the small and large farms and workforces—the trends are the same. The two extremes (highest:lowest and lowest:highest ratios of runners:pickers) represent the worst execution times, but in both cases there is a sweet spot in the middle.

5.2 Transport task allocation mode

We computed factor analysis to compare the three transport task allocation modes: whilst scheduling picking (W), post scheduling picking (P) and whilst executing picking (E), as shown in Figure 3. For scheduling runners, overall there is no statistically significant difference in execution time between the two task allocation mechanisms (RR and OSI) (plots a and d). Scheduling the runners whilst scheduling the pickers produced a shorter execution time than the alternative modes (plots b and e). The ablated results for the runner scheduling mechanisms and modes show no statistically significant differences (plots c and f). The statistical significance is reported for the small farm; for the large farm, only one heatmap (i.e. distribution of ripe fruit) was evaluated for each run, so it is not possible to compute statistical significance. Future work will involve running over multiple heatmaps (e.g. representing different days in a picking season).



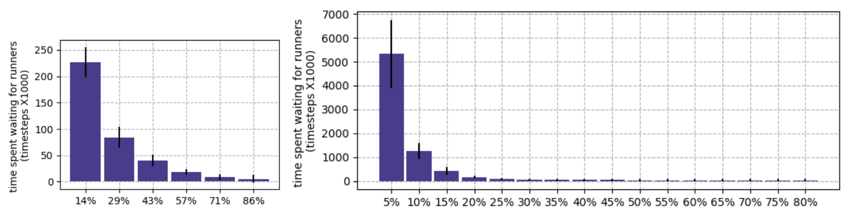
(a) (t=0.40, p=0.692) (b) (F=42.86, p=0.000) (c) (F=17.63, p=0.000)

(d) (e) (f)

Fig. 3. Factor analysis for the picker and runner mechanisms (mech.) and modes.

5.3 Cost of waiting

Finally, we compare the cost of waiting. Since pickers are more expensive than runners (i.e. the best pickers are paid higher salaries), we focus on the picker waiting time here. Figure 4 shows the *cumulative* waiting time, summed over all pickers in each run. As expected, when the percentage of runners increases, the pickers' wait time decreases; however, this also results in each picker picking a higher proportion of the fruit.



(a) small farm
($H = 64.96, p = 0.00000$)

(b) large farm
($H = 58.33, p = 0.00000$)

Fig. 4. Results for **cumulative picker waiting time** for different percentages of agents being employed as runners. The H statistic from Kruskal-Wallis tests and associated p values are shown, indicating statistically significant differences for the different ratios in both farm scenarios.

6 Discussion, Conclusions and Future Work

This paper explored the application of market-based task-allocation mechanisms to the problem of managing workers to harvest fruit. Patches of ripe fruits were auctioned to agents in one of two roles (picker or runner) using three different market-based mechanisms (RR, OSI and SSI) drawn from the MRTA literature and three different modes for assigning dependent tasks to runners (whilst scheduling picking, post scheduling picking and whilst executing picking). The comparative performance of different ratios of pickers and runners was evaluated.

Our experiments were designed to answer two questions. The first question asks what is the most efficient ratio of runners to pickers. Our results show that the ratio of runners to pickers is critical with respect to both execution and picker waiting time, and that the "sweet spot" varies depending on the size of field and workforce. The second question asks what is the most efficient strategy for allocating tasks to runners. Our results show that, for allocating tasks to runners, the *whilst scheduling pickers* (W) mode produces the best results (shortest execution times), with similar results for each of the different auction mechanisms—indicating that the critical factor is identifying the best point in the complex harvesting process to allocate transport tasks.

Current work involves deeper investigation of the large farm scenario, particularly using different distributions of ripe fruit and expanding to consider multiple fields in the allocation (e.g. using market-based mechanisms to allocate the workforce to different fields, as well as allocating tasks within a field). Ongoing collaboration with both of the farms that served as examples for the scenarios implemented here will allow farm managers to employ our task allocation methods for managing their workforce. These real-world deployments will provide additional verification of our methodology, such that predicted results from our simulation compared to actual results in the field will be presented in future work.

Acknowledgements

This work was supported by Research England [Lincoln Agri-Robotics] as part of the Expanding Excellence in England (E3) Programme.

References

1. Anscombe, F.: The validity of comparative experiments. Journal of the Royal Statistical Society, series A (General) **111**(3), 181–211 (1948)
2. Atay, N., Bayazit, B.: Emergent task allocation for mobile robots. In: Proceedings of Robotics: Science and Systems Conference (2007)
3. Chang, C.L., Lin, K.M.: Smart Agricultural Machine with a Computer Vision-Based Weeding and Variable-Rate Irrigation Scheme. Robotics **7**(38) (2018)
4. Das, G., Cielniak, G., From, P., Hanheide, M.: Discrete event simulations for scalability analysis of robotic in-field logistics in agriculture–a case study. In: IEEE International Conference on Robotics and Automation, Workshop on Robotic Vision and Action in Agriculture (2018)
5. Dias, M.B., Zlot, R., Kalra, N., Stentz, A.: Market-based multirobot coordination: A survey and analysis. Proceedings of the IEEE **94**(7) (2006)
6. Duckett, T., Pearson, S., Blackmore, S., Grieve, B., Smith, M.: Agricultural Robotics White Paper: The Future of Robotic Agriculture. https://www.ukras.org/wp-content/uploads/2018/10/UK_RAS_wp_Agri_web-res_single.pdf (last accessed 10-Mar-2020) (2018)
7. Elkoby, Z., van 't Ooster, B., Edan, Y.: Simulation analysis of sweet pepper harvesting operations. In: Advances in Production Mgt Sys: Innovative and Knowledge-Based Production Management in a Global-Local World. Springer (2014)
8. Fisher, R.A.: Statistical methods for research workers (1925)
9. Gerkey, B., Matarić, M.: A formal analysis and taxonomy of task allocation in multi-robot systems. The International Journal of Robotics Research **23**(9) (2004)
10. Harabor, D., Grastien, A.: Online graph pruning for pathfinding on grid maps. In: Proc of AAAI (2011)
11. Harman, H., Sklar, E.: Auction-based task allocation mechanisms for managing fruit harvesting tasks. In: UK Robotics and Autonomous Systems Network Annual Conference (UK-RAS) (2021)
12. Heap, B., Pagnucco, M.: Repeated sequential single-cluster auctions with dynamic tasks for multi-robot task allocation with pickup and delivery. In: Multiagent System Technologies. Springer (2013)
13. Heap, B., Pagnucco, M.: Sequential single-cluster auctions for robot task allocation. In: AI 2011: Advances in Artificial Intelligence. Springer (2011)
14. Hern, A.: Robots deliver food in Milton Keynes under coronavirus lockdown. The Guardian (12-Apr 2020), last accessed: 9-oct-2020

15. Kirk, R., Cielniak, G., Mangan, M.: L*a*b*fruits: A rapid and robust outdoor fruit detection system combining bio-inspired features with one-stage deep learning networks. Sensors **20**(1) (2020)
16. Koenig, S., Tovey, C., Lagoudakis, M., Markakis, V., Kempe, D., Keskinocak, P., Kleywegt, A., Meyerson, A., Jain, S.: The power of sequential single-item auctions for agent coordination. In: Proc of AAAI. vol. 2 (2006)
17. Kootstra, G., Wang, X., Blok, P.M., Hemming, J., Van Henten, E.: Selective harvesting robotics: current research, trends, and future directions. Current Robotics Reports (2021)
18. Korsah, G., Stentz, A., Dias, M.: A comprehensive taxonomy for multi-robot task allocation. The International Journal of Robotics Research **32**(12) (2013)
19. Kruskal, W.H., Wallis, W.W.: Use of Ranks in One-Criterion Variance Analysis. Journal of the American Statistical Association **47**(260) (1952)
20. Kurtser, P., Edan, Y.: Planning the sequence of tasks for harvesting robots. Robotics and Autonomous Systems **131** (2020)
21. Landén, D., Heintz, F., Doherty, P.: Complex Task Allocation in Mixed-Initiative Delegation: A UAV Case Study. In: Principles and Practice of Multi-Agent Systems. Springer (2012)
22. Liu, B., Bruch, R.: Weed Detection for Selective Spraying: a Review. Current Robotics Reports **1** (2020)
23. Luke, S., Cioffi-Revilla, C., Panait, L., Sullivan, K., Balan, G.: Mason: A multiagent simulation environment. SIMULATION **81**(7) (2005)
24. McLurkin, J., Yamins, D.: Dynamic Task Assignment in Robot Swarms. In: Proc of Robotics: Science and Systems Conference (RSS) (2005)
25. Nunes, E., Gini, M.: Multi-robot auctions for allocation of tasks with temporal constraints. In: Proc of AAAI (2015)
26. Parker, L.E.: Task-Oriented Multi-Robot Learning in Behavior-Based Systems. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (1996)
27. Pelham, J.: The Impact of Brexit on the UK Soft Fruit Industry. London: British Summer Fruits (2017)
28. Reuters: Robots target coronavirus with ultraviolet light at London train station. https://uk.reuters.com/article/uk-health-coronavirus-britain-robots/robots-target-coronavirus-with-ultraviolet-light-at-london-train-station-idUKKCN26E2NT (23-Sept 2020), last accessed: 9-oct-2020
29. Schneider, E., Sklar, E.I., Parsons, S., Özgelen, A.T.: Auction-based task allocation for multi-robot teams in dynamic environments. In: Proc of TAROS. Springer (2015)
30. Seyyedhasani, H., Peng, C., Jang, W.J., Vougioukas, S.G.: Collaboration of human pickers and crop-transporting robots during harvesting – Part I: Model and simulator development. Computers and Electronics in Agriculture **172** (2020)
31. Seyyedhasani, H., Peng, C., Jang, W.J., Vougioukas, S.G.: 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** (2020)
32. Shamshiri, R.R., Hameed, I.A., Karkee, M., Weltzien, C.: Robotic harvesting of fruiting vegetables: A simulation approach in V-REP, ROS and MATLAB. Proc in Automation in Agriculture-Securing Food Supplies for Future Generations (2018)
33. Shapiro, S.S., Wilk, M.B.: An Analysis of Variance Test for Normality (Complete Samples). Biometrika **52**(3/4) (1965)
34. Vail, D., Veloso, M.: Dynamic multi-robot coordination. In: Multi-Robot Systems: From Swarms to Intelligent Automata. Kluwer (2003)