POMS PRODUCTION AND OPERATIONS MANAGEMENT

PRODUCTION AND OPERATIONS MANAGEMENT

Vol. 0, No. 0, xxxx–xxxx 2021, pp. 1–20
ISSN 1059-1478 | EISSN 1937-5956 | 21 | 00 | 0001
DOI 10.1111/poms.13406

Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems

Zheng Wang
School of Maritime Economics and Management, Dalian Maritime University, No.1, Linghai Road, Dalian 116026, China,
drwz@dlut.edu.cn

Jiuh-Biing Sheu*
Department of Business Administration, National Taiwan University, No.1, Section. 4, Roosevelt Road, Taipei, 10617, Taiwan, ROC,
jbsheu@ntu.edu.tw

Chung-Piaw Teo
NUS Business School and Institute of Operations Research and Analytics, National University of Singapore, 119077, Singapore,
bizteocp@nus.edu.sg

Guiqin Xue
School of Maritime Economics and Management, Dalian Maritime University, No.1, Linghai Road, Dalian 116026, China,
xuegqmail@126.com

Intelligent part-to-picker systems are spreading across a broad range of industries as preferred solutions for agile order fulfillment, wherein mobile racks are carried by robots and moved to stations where human pickers can pick items from them. Such systems raise the challenge of designing good work schedules for human pickers; they also give rise to a new class of operational scheduling problems in human–robot coordinated order picking systems. This work studies the problem of finding a suitable robot schedule that takes into account the schedule-induced fluctuation of the working states of human pickers. A proposed model enables mobile racks with various workloads to be assigned to pickers, and schedule the racks that are assigned to every picker to minimize the expected total picking time. The problem is formulated as a stochastic dynamic program model. An approximate dynamic programming (ADP)-based branch-and-price solution approach is used to solve this problem. The developed model is calibrated using data that were collected from a dominant e-commerce company in China. Pickers' working state transitions are modeled based on data obtained from this warehouse. Counter-factual studies demonstrate that the proposed approach can solve a moderately sized problem with 50 racks in under 2 minutes. More importantly, the approach yields high-quality solutions with picking times that are 10% shorter than the solutions that did not consider schedule-induced fluctuations of pickers' working states.

Key words: robot scheduling; order picking; human–machine coordination; circadian rhythm; approximate dynamic programming
History: Received: July 2020; Accepted: February 2021 by Subodha Kumar, Vijay Mookerjee, Xiaohang Yue after 2 revisions.

1. Introduction

Order picking has long been identified as the most time-consuming and labor-intensive set of activities in warehousing. With the advent of Industry 4.0, many new technologies, such as the robotics and Internet of Things (IoT), have enabled a new wave of warehouse automation (Olsen and Tomlin, 2020). A novel part-to-picker system, involving a fleet of robots, mobile racks, working stations, and sophisticated control software, provides an innovative and promising way to improve picking performance (Weidinger et al., 2018). In a warehouse that uses such a system, robots are scheduled to carry racks to stations, where human operators pick items from the racks and place them in bins. By leaving the tedious carrying work to robots, the system is expected to reduce greatly the walking distance of human operators and improve order picking productivity.

Various logistics service providers, including the giants like Alibaba, JD Logistics, and Amazon, have more than 100,000 robots in their fulfillment centers (Demaitre, 2019). Figure 1 shows a Kiva system that is implemented in an Amazon warehouse. ABI Research has predicted that the number of commercial robots will reach four million in 2025, 20 times more than the number of robots that were deployed in 2018 (Demaitre, 2019). With fast expanding sales in the

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

Figure 1 Order Picking in a Mobile-Rack Part-to-Picker System (Designboom, 2014) [Color figure can be viewed at wileyonlinelibrary.com]

e-commerce sector, the global warehouse robotics market is projected to reach $6 billion by 2025 (Wolff, 2019).

However, warehouse automation brings new challenges to order picking operations. In the cases that were studied here, the working states of human pickers fluctuate substantially owing to natural circadian rhythms and work-induced fatigue. To perform picking tasks, robots must collaborate with human pickers in changing working states. If robots merely loaded the human pickers with an exhausting list of jobs, productivity would not be much improved, despite the investment in automation. Managers who want to optimize order picking performance should consider the working states of pickers when setting the schedules of the mobile racks; doing so requires taking a holistic view of human energy regulation, and appreciating the fact that a picker will be more effective with a suitably coordinated schedule, rather than an overwhelming and daunting one (Glock et al., 2019). Amazon recently suffered a spate of very high turnover (Lecher, 2019), which greatly reduced productivity and detrimentally affected order fulfillment. Workers complained that they were treated like robots. Around 300 workers left the Baltimore fulfillment center in a year. These events reveal that the development of warehouse automation must not ignore the needs of the human workers who are also an integral part of the system.

Researchers have recently come to appreciate the importance of considering human factors in order picking, and noted the lack of studies on this issue (Grosse et al., 2015). Shehhi et al. (2019) used Siemens Jack software to simulate the effects of picking hundreds of items an hour on order pickers and highlighted the potential for such a modern order picking operation to cause repetitive motion injuries. Grosse et al. (2015) emphasized human factors in classical order picking systems.

This paper studies the following research questions with reference to a representative case of a dominant e-commerce company in China:

1. How does a human picker's working state fluctuate over time, and does this fluctuation significantly influence the performance of a picking system?
2. How should racks be assigned to multiple pickers with different working state uncertainties to improve their picking productivity?
3. How should all of the racks that are assigned to pickers be scheduled to minimize the total expected picking time?

The above questions are very challenging because of uncertainties in the transitions among human working states after the handling of various racks. The assignment of racks to multiple pickers, each of whom has an individual state transition uncertainty, significantly affects their total expected picking time. Consequently, the real-world robot scheduling problem with a large number of racks, robots, and human pickers is difficult to solve in a limited period.

Our study of this robot scheduling problem makes three important contributions. First, data from a dominant e-commerce company in China are analyzed to identify the working state fluctuations of pickers and their impact on the performance of a human–robot coordinated picking system. Thus, this work extends the research on human–robot coordinated picking systems by incorporating the working state fluctuations of human pickers into robot scheduling to determine efficient human–robot coordinated schedules. Second, the robot scheduling problem is formulated as a stochastic dynamic program (SDP) for multiple pickers, and a branch-and-price approach to its solution is developed by transforming the SDP into multiple single-picker SDPs and using duality and approximate dynamic programming (ADP) to solve the nonlinear pricing sub-problem. Third, extensive case studies with real-world data are performed and incorporation of the state fluctuations of human pickers into robot scheduling is shown to be critical to the performance of the order picking system. Without more training and/or monetary incentives for human pickers, the presented solution captures the impact of picker state variability on picking time and outperforms the actual schedule, which did not consider pickers' state fluctuations, by reducing picking time by 10%. The approaches that are developed herein can be applied to other warehouses to reveal fluctuations of pickers' working states, incorporate those fluctuations into robot schedules, and increase the productivity of a picking system.

Note that the findings in this paper are specific to the mobile-rack warehouse from which the used

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

order picking data were obtained. The reported savings of 10% may differ for different warehouses, depending on the type of system used therein. In a recent study, Ae Lee et al. (2020) reported that an AS/RS system exposes workers to a lower risk of task-related postural stress than a system that uses moving robots. This finding implies that changes in the states of workers are less affected by the schedule in an AS/RS system than in a system using moving robots (as is considered in this paper).

2. Literature Review

Automation technology supports the integration of robotics into the daily labor-intensive processes in warehouses, but it brings with it the challenge of coordinating human–robot interaction. The productivity of warehouses with both humans and robots cannot be maximized without optimal human–robot coordination. Related studies take human factors into account in different ways.

First, the picking productivity of human workers is commonly treated either as fixed (Lee and Murray, 2019) or random with a probability distribution. Relevant studies have used a uniform distribution to model the handling and walking rates of pickers (Ruben and Jacobs, 1999), a normal distribution to model the picking and sorting time of pickers (Chen et al., 2010), an order-sequence-based picking time measurement (Boysen et al., 2017), an exponential distribution to measure the time taken by pickers to complete an order from a bin (Zou et al., 2018), and an exponential service rate for each picker to determine queue length probabilities in sequential zone picking systems (Van der Gaast et al., 2020). To smoothen human–robot coordination in a picking system, a human's actual picking behavior and the special structure of the picking system must be well understood. Bozer and White (1990) found that increasing the number of picking positions could yield a notable improvement in performance only if the mean human picking time was close to the mean machine cycle time, such that the picking system was "balanced." Khachatryan and McGinnis (2014) developed models to estimate the expected value and variance of picker travel time in a picking system with robotic and manual pickers that was based on pick-to-buffer technology. Lim (2011, 2017), Webster et al. (2012), and Lim and Wu (2014) investigated the impact of the skill levels of human pickers on the performance of special picking systems.

Another stream of research concerns the impact of human cognition, feeling, or health risk on order picking performance. Batt and Gallino (2019) used learning curves to support managerial decisions concerning order picking. They conducted a detailed econometric study of the impact of intra-bin searching on picking times, and found that incorporating pickers' experience into pick assignments could substantially improve productivity. Glock et al. (2019) investigated the spinal load on order pickers and the consequent risks of injury in order picking. In a different vein, De Vries et al. (2016) studied the impact of an exogenous incentive mechanism on human picking performance, and showed that combining a given order picking method with either a cooperation-based or a competition-based incentive system could improve productivity.

Table 1 summarizes the relevant literature and the focus of this paper. Despite the publication of many relevant studies, gaps remain in our understanding of intelligent human–robot coordinated picking systems —especially in the context of Industry 4.0, in which robots do much more than ever before (Azadeh et al., 2019, Kumar, Mookerjee, and Shubham, 2018). As observed in Table 1, this paper differs from previous works in matching robot schedules with the working states of human pickers. First, the working state of a human picker is variable, and this variability has not been addressed by the literature. The efficiency and state of a human picker change with time and robot schedules. Second, with the advent of Industry 4.0, robots are performing more tasks and a wider range of tasks. In the mobile-rack-based picking system that is considered herein, robots must "perceive" how "hungry" the pickers are, so that they can supply them with the right rack at the right time. The robots should therefore be more proactive in responding to, or even managing, the uncertain human working states.

Methodologically, this work contributes to the literature by integrating ADP into the branch-and-price framework for a large stochastic dynamic program (SDP). SDP is used to model sequential decision making in environments in which outcomes are partly random and partly under the control of a decision maker. The huge number of system states in a typical SDP makes the computation of the value function impractical or impossible in terms of both memory and time. To overcome this curse of dimensionality, several approaches to approximating the SDP value function have been proposed. Basis functions and state aggregation are the two most common approaches (Powell, 2010). Studies that are related to our work in the domains of scheduling and resource assignment have addressed the delivery vehicle dispatching problem (Minkoff, 1993), dynamic fleet management (Topaloglu and Powell, 2006), ambulance redeployment decisions in an emergency medical service system (Maxwell et al., 2010), patient scheduling for multi-appointment health care programs (Diamant et al., 2018), the last-mile ride-sharing problem

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

Table 1 Literature Review

Refs.	Problem	Human factors	Means of handling human factors
Ruben and Jacobs (1999)	Order batch construction	Handling and walking rates	Uniform distribution
Chen et al. (2010)	Flexible evaluative framework for order picking systems	Picking and sorting time	Normal distribution
Boysen et al. (2017)	Sequencing orders in mobile rack warehouse	Picking time	Dependent on number of aisle relocations
Zou et al. (2018)	Evaluating storage policies of a picking system	Time to complete an order from a bin	Exponential distribution
Van der Gaast et al. (2020)	Analysis of capacity of sequential zone picking systems	Order picking time	Exponential distribution
Bozer and White (1990)	Performance modeling for end-of-aisle order picking systems	Picking time	Deterministic and exponentially distributed
Khachatryan and McGinnis (2014)	Picker travel time modeling for a picking system with buffers	Travel time	Estimated for a picking system with buffers
Lim (2011)	Designing bucket brigades and investigating impacts of hand-off time	Work velocities	Each worker has two work velocities
Webster et al. (2012)	Analysis of impact of storage assignment on order picking line	Walking and picking speed	Speed function
Lim and Wu (2014)	Coordinating workers in U-shaped assembly line	Work rate	Workers have station-dependent rates
Batt and Gallino (2019)	Analysis of the effect of searching and learning on picking performance	Picker's experience of picking	Parametric survival model
Glock et al. (2019)	Modeling ergonomic and economic performance in order picking	Injury risk	Formula for effect of peak load on spine of a picker
De Vries et al. (2016)	Aligning picking methods, incentive systems, and regulatory focus to increase performance	Picking performance	Alignment of picking methods, incentive systems, and regulatory focus
Our work	Robot scheduling for mobile-rack warehouses	Working state	Picker's working state changes with schedule

(Agussurja et al., 2019), and the in-patient bed assignment problem (Dai and Shi, 2019). Our robot scheduling problem contains a team of agents (pickers), each of whom is a source of uncertainty with independent state transition probabilities. All agents in this paper share the same set of resources (racks). The multi-agent SDP suffers from exponential spaces of joint actions and states and thus is intractable by existing approaches. In this work, the multi-agent SDP is handled by transforming it into multiple single-agent SDPs and an ADP-based branch-and-price approach is proposed to solve the problem.

3. Transition between Working States

Order picking data over 6 days in 2019 were obtained from one of the dominant e-commerce companies in China[1]. The data record the picking operations of nine human pickers in a 2,500-square-meter mobile-rack warehouse that stores small cosmetic items. The pickers worked with about 40 robots to pick daily customer orders. This warehouse is medium-sized; such a size is common and other logistics companies have built a number of mobile-rack warehouses of the same type (Demaitre, 2019). The data include the times that the racks arrived at and left the picking stations; the items that were picked from each rack; and the layers of the racks from which the items were picked.

3.1. Observation and Assumptions

The data interestingly revealed that a picker may take significantly different amounts of time to pick the same set of items from the same rack on two consecutive days. The picking time in one instance may actually be more than double that in another. This was the case for not just a single picker but most of them. To determine the causes of changes in the states of the pickers and their different picking times, the picking operations were carefully studied, yielding the following important finding. The amounts of work required to pick from different layers of the racks can be vastly different: picking many items from high layers of a rack by climbing up and down a ladder can be quite exhausting, whereas picking items that are easily reached can be very relaxing. The sequences of the racks that were handled by the pickers were then retrieved, and picking times were longer than usual mostly following the picking of items from one or more of the "exhausting racks." The different working states of the pickers can thus be attributed to different work intensities of picking from previous racks. This finding was confirmed by the pickers.

Based on observations of operations, we make the following assumptions. First, the time that a human picker takes to pick from a rack depends on her working state. The "better" her state, the shorter her picking time. Second, the working state of a picker

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

depends on the workload in her previous rack. The greater workload for her previous rack, the higher the probability that her state will deteriorate. Given a set of racks for a picker, a different service sequence will therefore result in different working state transitions and different expected picking times. Third, the state transitions of every picker are independent of those of the others.

The workload of picking from every rack will next be quantified (Section 3.2), and then approaches to measuring picker state variability will be proposed (Section 3.3), based on the data.

3.2. Picking Workload

Some relevant studies have evaluated the time associated with picking activities in traditional picker-to-part warehouses (Glock et al., 2019). However, unlike traditional warehouses, a mobile-rack warehouse is equipped with only light items; pickers in such a warehouse do not need to walk to find items; racks can be moved to pickers by robots; a system tells pickers from which layers items are to be picked; and most of the time taken by pickers is spent moving to the right layer, retrieving items, and putting them into a bin to meet the corresponding order.

A rack in the warehouse has five layers and the height of each layer is about 0.4 m. Picking from the lowest layer requires a picker to squat and stoop, while picking from the top two layers requires a picker to step on a ladder. Therefore, the time to pick an item depends mainly on the layer of the item in the rack.

Squatting down (SD), standing up (SU), stretching an arm (SA), picking an item (PI), putting an item into a bin (PB), climbing up a ladder (CU), and climbing down a ladder (CD) were timed for multiple pickers. These times were averaged over the pickers as $\mathbb{t}_{SD}$, $\mathbb{t}_{SU}$, $\mathbb{t}_{SA}$, $\mathbb{t}_{PI}$, $\mathbb{t}_{PB}$, $\mathbb{t}_{CU}$, and $\mathbb{t}_{CD}$, respectively, yielding the times for an average picker. The time required to pick an item from the lowest layer is $\mathbb{t}_1 = \mathbb{t}_{SD} + 2\mathbb{t}_{SA} + \mathbb{t}_{PI} + \mathbb{t}_{SU} + \mathbb{t}_{PB}$, the time required to pick an item from the top two layers is $\mathbb{t}_{45} = \mathbb{t}_{CU} + 2\mathbb{t}_{SA} + \mathbb{t}_{PI} + \mathbb{t}_{CD} + \mathbb{t}_{PB}$, and the time required to pick an item from the middle two layers is $\mathbb{t}_{23} = 2\mathbb{t}_{SA} + \mathbb{t}_{PI} + \mathbb{t}_{PB}$.

Let $\mathbb{t}_r$ be the expected time for which an average picker handles rack r, and $\mathbb{t}_1, \mathbb{t}_{23}, \mathbb{t}_{45}$ be the average times required to pick an item from layers 1, 2 and 3, and 4 and 5, respectively. Given a rack from which $\mathbb{n}_1$ items in the lowest layer, $\mathbb{n}_{45}$ items in the top two layers, and $\mathbb{n}_{23}$ items in the middle two layers must be picked, $\mathbb{t}_r$ may be less than $\mathbb{n}_1\mathbb{t}_1 + \mathbb{n}_{23}\mathbb{t}_{23} + \mathbb{n}_{45}\mathbb{t}_{45}$ because a complete picking action can obtain multiple items from a single layer. The maximum number of the items that an average skilled picker can obtain from a layer is defined as $\mathbb{N}$. Then, we set $\mathbb{t}_r = \lceil \mathbb{n}_1/\mathbb{N} \rceil \mathbb{t}_1 + \lceil \mathbb{n}_{23}/\mathbb{N} \rceil \mathbb{t}_{23} + \lceil \mathbb{n}_{45}/\mathbb{N} \rceil \mathbb{t}_{45}$, where $\lceil n \rceil$ is the minimum integer that is not less than a positive real n.

A unit of picking workload is defined as the picking work that an average skilled picker can complete in a minute. The picking workload for any rack r, denoted as $\mathbb{w}_r$, can be measured in these units, yielding $\mathbb{w}_r = \mathbb{t}_r$ if $\mathbb{t}_r$ is given in minutes. Based on the data, for every rack r, the picking time $\mathbb{t}_r$ of an average skilled picker as well as the picking workload $\mathbb{w}_r$ are estimated using the parameters that are provided in Appendix S1.

Racks are now grouped by their workloads to investigate the effects of the racks with different workloads on the probabilities of state transitions of pickers. Let $\mathbb{w}_0$ be the average workload of all racks, yielding a ratio $\mathbb{w}_r/\mathbb{w}_0$ for every rack r. The ratios of all the racks fall in the range (0.25, 1.85). Without loss of generality, this range is divided into five equal parts with an interval of 0.32, and each part is associated with a workload level. These levels are named *least*, *less*, *medium*, *more*, and *most* levels, respectively. For example, a $\mathbb{w}_r/\mathbb{w}_0$ ratio of 0.7 falls into the second range (0.57, 0.89], so rack r would be at the *less* workload level. The lower the workload level of a rack, the less the time a picker takes to handle it in any given state.

3.3. Data-driven Approach to Measuring Picker State Variability

Identifying the working state of a picker when she is picking from a rack and measuring the time a picker takes in a particular state to handle a rack with a particular workload level are both difficult. To formalize the measurement, a set of data-driven approaches to identifying the working states of pickers (Section 3.3.1), grouping pickers by their state variability (Section 3.3.2), and measuring the probabilities of state transitions of pickers (Section 3.3.3), is developed.

3.3.1. Identifying Working States of Pickers. The obtained data record the racks that every picker handled in six days. To identify a picker's working state when she was handling a rack, the following productivity-related notation is defined. Let $\mathbb{a}$ and $\mathbb{b}$ be the total workload and working time, respectively, of all the pickers in the six days, and $\mathbb{c} = \mathbb{a}/\mathbb{b}$ be the average productivity of the pickers. Let $\mathbb{a}_i$ and $\mathbb{b}_i$ be the total workload and working time, respectively, of picker i in the six days. Let $\mathbb{c}_i = \mathbb{a}_i/(\mathbb{b}_i\mathbb{c})$ be the productivity coefficient of picker i, indicating how proficient she is in handling racks relative to the average picker. If $\mathbb{c}_i > 1$, then picker i is more proficient than the average. Let $\mathbb{t}_{i,r}$ and $\mathbb{t}\prime_{i,r}$ be the time picker i should have spent and actually spent, respectively, handling rack r. We set $\mathbb{t}_{i,r} = \mathbb{t}_r/\mathbb{c}_i$ because $\mathbb{t}_r$ is the picking time of

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

an average skilled picker and $\mathfrak{c}_i$ is the productivity coefficient of picker i relative to the average. Then, the picking time ratio $\mathfrak{t}'_{i,r}/\mathfrak{t}_{i,r}$ can be obtained for picker i and rack r. Given a fixed $\mathfrak{t}_{i,r}$, the ratio reflects the working state of the picker when she was handling the rack. The smaller the ratio, the shorter her actual picking time and the better her state. The ratio is computed for every rack, the number of times that the ratios fall in different ranges is determined, and the histogram in Figure 2 is thus obtained.

The histogram in Figure 2 suggests a good fit to the normal distribution with mean $\mu = 1.0$ and standard deviation $\sigma = 0.138$. (See Appendix S2 for the proof of the normal distribution hypothesis in Figure 2.) Without loss of generality, the range $(\mu - 3\sigma, \mu + 3\sigma)$ is divided into five equal parts. From left to right, these parts are associated with five working states named *best*, *better*, *normal*, *worse*, and *worst* states, respectively. For example, a $\mathfrak{t}'_{i,r}/\mathfrak{t}_{i,r}$ ratio of 0.85 falls in the range $(\mu - 1.8\sigma, \mu - 0.6\sigma]$, so picker i was in the *better* state when she was handling rack r.

3.3.2. Distinguishing Pickers by State Variability. Figure 2 displays the distribution of the picking time ratios of all nine pickers. Each picker is associated with a distribution with particular mean and variance values. In this section, their individual distributions are used to examine the differences in their working state variability, based on which their picking performances will be compared in Section 7.3.

A normality test is performed on the distribution of the picking time ratios of every picker. The values of skewness and kurtosis for the pickers are from −1.671 to 1.529; their absolute values are smaller than $z_{0.025} = 1.96$. This test provides no reason to reject the normality hypothesis. The means and variances of the distributions are then estimated, and the gap between the mean ratio of any picker and the threshold 1.0 is within 0.0036. Similarly, the average picking productivity $\mathfrak{c}_i$ of every picker is close to 1.0. These results reveal that all pickers perform similar amounts of work per unit time on average and their working efficiencies are comparable. A two-sided chi-square test is performed to determine whether a significant difference exists between the variance of a single picker and that of all pickers. (See Appendix S3 for the test statistics.) Based on the comparisons of variance and kurtosis, the pickers are divided into three groups: the first two, the median four, and the last three pickers are in groups 1–3, respectively. Figure 3 plots the normal distribution that fits the picking time ratios of each group. The figure demonstrates that the first-group pickers have the steadiest state variability since they are often in the *normal* state, while the third group of pickers exhibits the most drastic state variability because they are more prone to being in the non-*normal* states than are the pickers in the other two groups.

3.3.3. Measuring Probabilities of State Transitions of Pickers. Let $\mathbb{m}_{i,r,j}$ be a binary indicator that equals 1 if picker i spent time $\mathfrak{t}'_{i,r}$ handling rack r and the ratio $\mathfrak{t}'_{i,r}/\mathfrak{t}_{i,r}$ falls in the range of working state j, and equals 0 otherwise. Let $\mathbb{d}_{r,r'}$ be a binary indicator that equals 1 if a picker handled rack r' just after handling rack r, and equals 0 otherwise. From the data are obtained $\mathbb{m}_{i,r,j}$ and $\mathbb{d}_{r,r'}$ for any picker, rack, and working state. Let $\mathbb{p}_{i,l,j,k}$ be the probability that the working state of picker i transitions from j to k after she picks a rack with workload level ℓ. Then,

$$\mathbb{p}_{i,l,j,k} = \left(\sum_{r\in\mathcal{R}_\ell, r'\in\mathcal{R}} \mathbb{m}_{i,r,j}\mathbb{m}_{i,r',k}\mathbb{d}_{r,r'}\right) / \left(\sum_{r\in\mathcal{R}_\ell} \mathbb{m}_{i,r,j}\right),$$

where $\mathcal{R}$ is the set of all racks and $\mathcal{R}_\ell$ is the set of racks with workload level ℓ. Let $t_{i,j,r}$ be the time spent by picker i handling rack r when she is in

Figure 2 Histogram of Picking Time Ratios of All Pickers [Color figure can be viewed at wileyonlinelibrary.com]

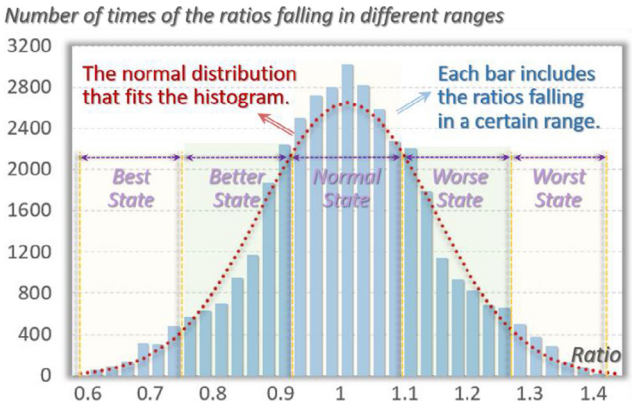
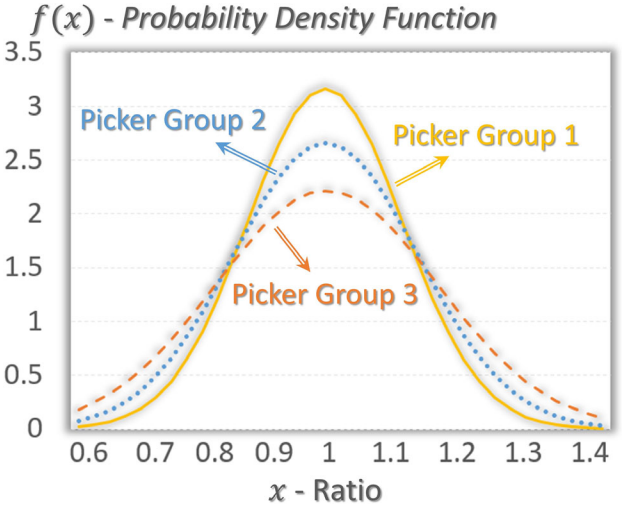


Figure 3 Normal Distribution that Fits the Ratios of Each Picker Group [Color figure can be viewed at wileyonlinelibrary.com]



Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

state j. We set $t_{i,j,r} = 0.5(\mathbb{r}_{j,0} + \mathbb{r}_{j,1})\mathbb{t}_{i,r}$, where $(\mathbb{r}_{j,0}, \mathbb{r}_{j,1}]$ is the ratio range that is associated with working state j. Appendix S4 provides an explanatory example of this approach to measuring $\mathbb{P}_{i,\ell,j,k}$.

Based on the above data-driven approach to measuring picker state variability, Appendix S5 provides statistical evidence for the assumptions that (1) a picker who handles a rack with a higher workload has a higher probability to deteriorate to an inferior state; and (2) the sequence of racks that are assigned to a picker significantly affects her picking time for these racks.

4. Robot Scheduling Model with Varying Picker State

This section formulates the robot scheduling problem as a stochastic dynamic program (SDP), based on the variability of picker states.

Multiple picking stations, each associated with a human picker, are considered. A set of known customer orders awaits the pickup service that is provided by the pickers. The racks from which the items of an order are picked have been determined. Enough robots are assumed to be available to carry the racks, as was the case in the real-world warehouse. These robots carry a rack to a picker whenever the picker has finished picking from a previous rack. Since the queue of racks at a station may increase pressure on the picker, the next rack will be sent to the station by a robot no earlier than the completion of picking from the previous rack. Hence, the queue of racks at any station is empty.

Owing to the uncertainty of state transitions, a picker may be in one of different states, each associated with a probability, after she has picked from a rack. The state probability depends on the workload of the rack from which she has just picked as well as her state before she begins picking from it. This probability also determines the expected time of picking the subsequent rack to be assigned to her. Therefore, the carrying of racks in different sequences by robots may result in different state transitions and different expected picking times for a picker.

The robot scheduling problem is considered here from a static perspective. The working state of every picker is not determined in real time whenever she picks from a rack; rather, a robot schedule that minimizes the total expected time taken by pickers to handle all of the racks is generated. A robot schedule consists of assigning racks to, and sequencing assigned racks for, multiple pickers.

Additionally, no breaks are incorporated into the service sequences of pickers. Instead, the focus is on robot scheduling during the working time of pickers, because when and if a picker requires a break, and how long a break should last, are not determined by the order picking system but are pre-determined by the worker duty schedule, according to the policy of the company.

4.1. System State

We define $s = (\vec{w}, u, \vec{m}, i)$ as the system state in a stage, where the arrow $\rightarrow$ denotes a list and the stage is defined in Section 4.2. A system state contains (i) the working states $\vec{m}$ of multiple pickers, whose transitions are independent of each other; (ii) the list of rack sequences w assigned to pickers; (iii) the remaining set of racks u yet to be assigned; and (iv) the picker i to whom a rack has just been assigned. Table 2 provides the most important notation; additional notation will be provided as required.

4.2. Stage and Action

The robot scheduling problem with R racks can be divided into R stages. In each stage, a rack is appended to the end of the service sequence of a picker. Since the racks that have not yet been assigned are in set u, a feasible action is to assign one of these racks to a picker under the following two constraints. First, if rack r has been assigned to picker i, then rack r' that is associated with the same order as r (meaning that items in the same order are found in both racks) cannot be assigned to a picker other than i. This constraint follows from the fact that all items in an order must be put into a single bin that is located at a single

Table 2 Notation Used to Define SDP

Notation	Explanation
$\mathcal{P}$	Set of P pickers $\{1, 2, \cdots, P\}$.
$\mathcal{M}$	Set of M working states $\{1, 2, \cdots, M\}$.
$\mathcal{R}$	Set of R racks to be handled $\{1, 2, \cdots, R\}$.
$\mathcal{L}$	Set of L rack workload levels $\{1, 2, \cdots, L\}$, each of which is associated with a picking workload range.
$\mathcal{R}_l$	Set of the racks with workload level $l (\in \mathcal{L})$.
l_r	Workload level of rack $r (\in \mathcal{R})$.
m_i	Working state of picker $i (\in \mathcal{P})$.
$\vec{w}_i$	Sequence of racks handled by picker $i (\in \mathcal{P})$.
w	List of P rack sequences $\{w_1, w_2, \cdots, w_P\}$, each of which corresponds to a picker.
u	Set of all racks that are to be assigned to pickers. It contains all such racks initially, and no racks at the last stage.
$\vec{m}$	List of P working states $\{m_1, m_2, \cdots, m_P\}$, each of which corresponds to a picker.
$\alpha_{i,m}$	Probability that picker $i (\in \mathcal{P})$ has the initial state $m (\in \mathcal{M})$.
$\mathbb{p}_{i,l,j,k}$	Probability that the working state of picker i transitions from j to k after she picks a rack with workload level $\ell (i \in \mathcal{P}, \ell \in \mathcal{L}, j, k \in \mathcal{M})$. Different pickers may have different probabilities $\mathbb{p}_{i,l,j,k}$ even for the same j, k, and ℓ.
$t_{i,j,r}$	Time spent by picker i handling rack r when she is in working state j.

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

station. Second, to balance the workloads of the pickers, every picker may be given a minimum workload and a maximum workload. Then, the lower bound $\underline{b}$ and the upper bound $\bar{b}$ are defined and set on the total workload associated with the racks to be handled by a picker.

Let $\mathcal{O}$ be the set of O orders each of which is associated with more than one rack. Let $\mathcal{R}_o$ be the set of racks that each need to be handled to fill order $o(\in\mathcal{O})$. Let $x_{i,r}$ be a binary variable that is 1 if rack r is handled by picker i and 0 otherwise. An action is feasible if it satisfies the following constraints. Constraint (1) ensures that two racks associated with an order must be handled by the same picker. Constraint (2) sets bounds on the workload of each picker.

$$x_{i,r} = x_{i,r'} \quad \forall i\in\mathcal{P}, o\in\mathcal{O}, r, r'\in\mathcal{R}_o, r\neq r' \tag{1}$$

$$\underline{b} \le \sum_{r\in\mathcal{R}} \mathbb{W}_r x_{i,r} \le \bar{b} \quad \forall i\in\mathcal{P} \tag{2}$$

Constraint (1) is also applicable to a case in which a rack is associated with multiple orders. In practice, such a multi-order rack is generally carried to a picking station only once so that all of the items on the rack that is associated with those orders can be picked at the same time, reducing the number of trips that must be made by the robots. Therefore, if a rack contains items in multiple orders, all of the racks for all of those orders should be assigned to a single picker. Given an example in which orders o_1 and o_2 are associated with the racks in $\{r_1, r_2\}$ and $\{r_2, r_3\}$, respectively, and rack r_2 contains items in both orders, constraint (1) requires $x_{i,r_1} = x_{i,r_2} = x_{i,r_3}$. However, the obtained data demonstrate that over 63% of orders are single-rack orders. The chance that two orders contain items on the same rack and are picked in the same batch is minuscule. Therefore, such racks and orders are excluded from the cases that are considered in Section 7.

Let $\mathcal{A}_k$ be the set of all feasible actions in stage $k(0\le k<R)$ and $A_k(r,i)$ be a feasible action in $\mathcal{A}_k$ that assigns rack r to picker i. Since state transitions of pickers are independent of each other, $A_k(r,i)$ cannot cause a state transition of a picker other than i.

4.3. Transition Equation for System State and Picker State

The transition equation for a system state from s^k to $s^{k+1}(0\le k<R)$ is formulated as Equation (3), where $m_{i'}^{k+1} \sim \mathbb{p}\left(m_{i'}^{k+1} | \ell_{r'}, m_{i'}^k\right) = \mathbb{p}_{i',\ell_{r'},m_{i'}^k,m_{i'}^{k+1}}$.

$$\begin{aligned} s^k &= \left(\left\{w_1^k, w_2^k, \ldots, w_{i\prime}^k \ldots, w_P^k\right\}, u^k, \left\{m_1^k, m_2^k, \ldots, m_{i\prime}^k, \ldots, m_P^k\right\}, i\right) \overset{A_{k+1}(r\prime, i\prime)}{\rightarrow} \\ s^{k+1} &= \left(\left\{w_1^k, w_2^k, \ldots, w_{i\prime}^k \cup \{r\prime\} \ldots, w_P^k\right\}, u^k \setminus r\prime, \left\{m_1^k, m_2^k, \ldots, m_{i\prime}^{k+1}, \ldots, m_P^k\right\}, i\prime\right) \end{aligned} \tag{3}$$

$$s^0 = \left(\vec{w}^0, \mathcal{R}, \vec{m}^0, 0\right) \tag{4}$$

$$s^R = \left(\vec{w}^R, \emptyset, \vec{m}^R, i\right) \tag{5}$$

Comparing s^k and s^{k+1} reveals that after an action $A_{k+1}(r', i')$, rack r' is removed from u^{k+1} and appended to the i'-th sequence of $\vec{w}^{k+1}$, and the i'-th element of $\vec{m}^{k+1}$ is $m_{i'}^{k+1}$, which may be the same as or different from $m_{i'}^k$.

Equation (3) specifies the state dynamics of the system, which starts from an initial state s^0, given by Equation (4), and ends at a final state s^R, given by Equation (5). In Equation (4), $\vec{w}^0$ contains P empty rack sequences; $\vec{m}^0$ is a list of pickers' initial states, and $i=0$, indicating that i does not point at any picker. Since the initial state of a picker may be uncertain, the system may have multiple initial states, each of which is associated with a different combination of pickers' initial states and a probability. In Equation (5), $\vec{w}^R$ contains all racks in $\mathcal{R}$; $\vec{m}^R$ is the set of the final states of pickers, and i is the picker who is associated with the final action. Each element w_i^R in $\vec{w}^R$ records a sequence of actions on picker i. After these actions, she may be in one of the multiple working states in $\mathcal{M}$, each of which is associated with a probability.

Let $\mathbb{p}_{i,l} = \left\{\mathbb{p}_{i,l,j,j'}\right\}_{j,j'\in\mathcal{M}}$ be the probability matrix of working state transitions of picker i after she has picked from a rack with workload level ℓ. Let $P_i\left(w_i^k\right) = \left\{p_{i,j,k}\right\}_{j\in\mathcal{M}}$ be the probabilities of the final states of picker i after the actions in w_i^k. $P_i\left(w_i^k\right)$ can be obtained from the state transition equation of picker states, formulated as Equation (6), where $\vec{\alpha}_i = \{\alpha_{i,m}\}_{m=1,2,..M}$ and $w_i^k(n)$ is the n-th rack of w_i^k. $P_i\left(w_i^k\right)$ is a list of M elements, each of which is the probability $p_{i,j,k}$ that picker i is in state j at stage k.

$$P_i\left(w_i^k\right) = \vec{\alpha}_i \prod_{n=1}^{k} \mathbb{P}_{i,\ell_{w_i^k(n)}} \quad \forall i\in\mathcal{P}, 0<k\le R \tag{6}$$

4.4. Picking Time

The expected time to pick from all of the racks must be minimized. The penalty function, $R\left(s^k, A_k(r,i)\right)$, given by Equation (7), equals the time that picker i spends handling rack r in system state s^k.

$$R\left(s^k, A_k(r,i)\right) = t_{i,m_i^k,r} \tag{7}$$

Let $v\left(s^k\right) = v(\vec{w}^k, u^k, \vec{m}^k, i^k)$ be the penalty function, which is the negative value function of the SDP in

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

system state s^k. The penalty function, $v(s^k)$, is given by the Bellman equation in Equation (8).

$$v(s^k) = \min_{v(s^{k+1}) \in \mathcal{S}} \{R(s^k, A_k(r,i)) + \mathbb{E}[v(s^{k+1})]\} \quad (8)$$

Given a system state s^k and an action on picker i, the picker may transition to any state in $\mathcal{M}$ so s^{k+1} can be one of multiple possible system states. The expectation $\mathbb{E}[v(s^{k+1})]$ over all possible system states is formulated as Equation (9).

$$\mathbb{E}[v(s^{k+1})] = \sum_{m \in \mathcal{M}} \mathbb{P}_{i,l_r,m_i^k,m} \cdot v(s^{k+1} | m_i^{k+1} = m) \quad (9)$$

From the SDP that was formulated in Section 4, the robot scheduling problem can be regarded as a novel parallel machine scheduling problem, in which every machine (picker) has respective state transition probabilities and the job assignment and sequence affect the processing time by changing the machine states, based on those probabilities. The problem is clearly NP-hard because it can be reduced to the single machine scheduling problem with sequence-dependent setup times, which has been proved to be NP-hard (Ertem et al., 2019). For a real-world case with dozens of racks and a number of pickers that have mutually independent state transitions, a huge state space must be explored to find the solution with the minimal expected time to pick from all racks. Appendix S6 provides an explanatory example of the difficulty of finding a good robot scheduling solution. Appendix S7 analyzes how the picking times associated with different rack assignment and sequencing solutions change with pickers' state variability. Section 5 provides a linear programming formulation, based on which Section 6 proposes an ADP-based branch-and-price approach to handling this challenging robot scheduling problem.

5. Linear Programming Formulation

The SDP, summarized in Equation (8), is transformed into its equivalent linear program (LP) using a fundamental result in SDP theory (Puterman, 2014). A variable $v(s^k)$ is created for each system state $s^k = (\vec{w}^k, u^k, \vec{m}^k, i^k)$. By specifying a constraint for each pair of s^k and feasible action $A_k(r,i)$, minimizing the total picking time in (8) is made equivalent to solving the following LP.

$$\max_v \left\{ \sum_{\vec{m}_0 \in \mathcal{M}^P} \alpha(\vec{m}_0) v(\vec{w}^0, \mathcal{R}, \vec{m}^0, 0) \right\} \quad (LP)$$

$$\text{s.t.} \sum_{\vec{m}_0 \in \mathcal{M}^P} \alpha(\vec{m}_0) = 1$$

$$v(s^k) \le R(s^k, A_k(r,i^k)) + \sum_{m \in \mathcal{M}} \mathbb{P}_{i^k,\ell_r,m_{i^k}^k,m} \bullet v(s^{k+1} | m_{i^k}^{k+1} = m)$$

$$\forall s^k \in \mathcal{S}, A_k(r,i^k) \in \mathcal{A}_k, 0 \le k < R$$

$$v(s^k) \le R(s^k, A_k(r,i^k)) \quad \forall A_k(r,i^k) \in \mathcal{A}_k, k = R$$

In the LP, $\alpha(\vec{m}_0) = \prod_{i \in \mathcal{P}, m_i \in \mathcal{M}} \alpha_{i,m_i}$ is the probability that is associated with a combination of possible initial states of all pickers. The last two constraints are related to the intermediate stages and the final stage, respectively. The LP includes a variable for each system state and a constraint for each state-action pair. Proposition 1 explains the number of system states in the LP.

PROPOSITION 1. *A problem with* R *racks and* $P(P \le R)$ *pickers, each of whom has* M *possible states and handles at least one rack, involves more than* $\sum_{r=0}^{R-P} (C_R^r \bullet (R-r)! \bullet C_{R-r-1}^{P-1} \bullet M^P \bullet P)$ *system states. (See Appendix S11 for proof.)*

As stated in Proposition 1, a real-world case with dozens of racks and a number of pickers would involve a huge number of system states. Solving or even constructing an LP for real-world cases, is thus impractical.

6. ADP-based Branch-and-Price Approach

The curse of dimensionality in the LP calls for a more efficient and tractable solution approach. As explained in Section 4, the problem can be solved in two stages, which are assigning racks to multiple pickers and sequencing the racks that have been assigned to every picker. Among solution approaches, the column generation approach fits the problem well because its master and sub problems nicely correspond to the two solution stages and can be used to assign columns (rack sequences) to pickers and generate good columns (rack sequences) with negative reduced costs, respectively. A column generation procedure that is embedded into a branch-and-bound framework will be developed below to solve the problem.

6.1. A Branch-and-Price Framework

Let $\mathcal{Q}_i$ be the set of all feasible rack sequences that will be handled by single picker i, and $\mathcal{V}_{i,q}$ $(i \in \mathcal{P}, q \in \mathcal{Q}_i)$ be the expected time spent by picker i handling the rack sequence q. Let $y_{i,q}$ be a binary variable that equals 1 if the sequence q of picker i is selected in the final solution and equals 0 otherwise. Let $a_{i,q,r}$ be a binary value

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

that equals 1 if the sequence q of picker i contains rack r and equals 0 otherwise. The master problem of robot scheduling can now be formulated as a sequence-based model (SBM).

$$\min \sum_{i\in\mathcal{P}}\sum_{q\in\mathcal{Q}_i} \mathcal{V}_{i,q} y_{i,q} \ (SBM)$$

$$\text{s.t.} \sum_{i\in\mathcal{P}}\sum_{q\in\mathcal{Q}_i} a_{i,q,r} y_{i,q} \geq 1 \quad \forall r\in\mathcal{R}$$

$$\sum_{q\in\mathcal{Q}_i} y_{i,q} = 1 \quad \forall i\in\mathcal{P}$$

$$y_{iq}\in\{0,1\} \ \forall i\in\mathcal{P}, q\in\mathcal{Q}_i$$

The SBM minimizes the expected picking time associated with selected sequences by ensuring that each rack must be handled at least once and each picker must be assigned exactly one sequence. The two difficulties in solving the SBM are an exponential number of columns in the SBM and the extremely time-consuming task of obtaining $\mathcal{V}_{i,q}$ for each column. To handle the first difficulty, a column generation procedure is proposed. The linear relaxation of the SBM is solved through column generation by repeatedly solving (1) a restricted master problem that is formulated as an SBM with a subset of columns $\overline{\mathcal{Q}}_i \subset \mathcal{Q}_i$ for any picker $i(\in\mathcal{P})$, and (2) a pricing sub-problem to generate a column with negative reduced cost. To handle the second difficulty, an ADP is introduced to price columns.

The SBM is an integer programming model. Even if its linear relaxation is solved to optimality by column generation, the resulting optimal solution may not necessarily be integral. Therefore, column generation is generally embedded into a branch-and-bound search framework (Gu et al., 2010, King et al., 1993) and is executed at every node of the search tree, yielding a branch-and-price algorithm (Barnhart et al., 1998, Guedes and Borenstein, 2018, Lübbecke and Desrosiers, 2005).

6.2. Pricing

Let dual variables $\pi_r (r\in\mathcal{R})$ and $\sigma_i (i\in\mathcal{P})$ correspond to the two types of constraint in SBM, respectively. The reduced cost $C_{\delta(i,q)}$ of a column $\delta(i,q)$ associated with picker i and rack sequence q can be formulated as Equation (10).

$$C_{\delta(i,q)} = \mathcal{V}_{i,q} - \sum_{r\in\mathcal{R}} \pi_r a_{i,q,r} - \sigma_i \tag{10}$$

Given a column with a subset of the racks that are assigned to picker i, the LP in Section 5 can be solved by restricting $\mathcal{S}$ and $\mathcal{A}_k$ to the sets feasible for picker i and $\mathcal{R}$ to the set of assigned racks. Then, $\mathcal{V}_{i,q}$ and q can be obtained from the solution to the restricted LP. However, the curse of dimensionality of the LP remains even for a single picker: solving the restricted LP for a column with only ten racks, five workload levels, and five possible picker states, still takes over an hour. To handle this problem, Section 6.2.1 introduces ADP to construct an approximate LP (ALP), and Section 6.2.2 develops an ALP-based pricing approach.

6.2.1 Approximate Dynamic Programming. To solve the restricted LP, ADP is introduced to approximate the values of the variables. In the restricted LP, variable $v(w_i, u, m_i, i)$ is used to indicate the expected picking time for which picker i with an initial state m_i handles the racks in u and the last assigned rack in w_i, denoted as $w_i(E)$. The approximation of the variable, denoted as $v_{ALP-i}(w_i, u, m_i)$, is formulated as Equation (11) by assuming a linear form. In Equation (11), constant θ_i is an adjustment term for picker i, and terms $\beta_{i,\ell}$ and γ_{i,m_i} are the marginal values for a rack with workload level ℓ to be handled by picker i, and state m_i of picker i, respectively.

$$v_{ALP-i}(w_i,u,m_i) \approx \theta_i + \sum_{j\in u\cup\{w_i(E)\}} \beta_{i,\ell_j} + \gamma_{i,m_i} \quad \forall i\in\mathcal{P} \tag{11}$$

Based on Equation (11), the restricted LP is reformulated as an approximate LP, denoted as ALP (i), for picker i, handling the racks in sequence q.

$$\mathcal{V}_{i,q} = \max_{\theta_i,\vec{\beta}_i,\vec{\gamma}_i} \left\{ \theta_i + \sum_{r\in q}\beta_{i,l_r} + \sum_{m\in\mathcal{M}} \alpha_{i,m}\gamma_{i,m} \right\} \ ALP(i)$$

$$\text{s.t.} \ \ \beta_{i,l} + \gamma_{i,m} - \sum_{m'\in\mathcal{M}} p_{i,l,m,m'}\gamma_{i,m'} \leq \mathbb{E}[t_{i,m,l}]_{r\in\mathcal{R}_l} \forall l\in\mathcal{L}, m\in\mathcal{M} \tag{12}$$

$$\theta_i + \beta_{i,l} + \gamma_{i,m} \leq \mathbb{E}[t_{i,m,l}]_{r\in\mathcal{R}_l} \quad \forall l\in\mathcal{L}, m\in\mathcal{M} \tag{13}$$

$$\left(\theta_i, \vec{\beta}_i, \vec{\gamma}_i\right) \in \mathbb{R} \tag{14}$$

In ALP (i), constraints (12) and (13) transform, respectively, the last two constraints of the LP in Section 5 by replacing the variable $v(\cdot)$ with the approximation $v_{ALP-i}(\cdot)$ in Equation (11) and the picking time $R(\cdot)$ spent by picker i handling rack r in state m with $\mathbb{E}[t_{i,m,l}]_{r\in\mathcal{R}_l}$. $\mathbb{E}[t_{i,m,l}]_{r\in\mathcal{R}_l} = \frac{1}{|\mathcal{R}_l|}\sum_{r\in\mathcal{R}_l} t_{i,m,r}$ is the average time spent by picker i handling a rack with workload level l in working state m. $\mathcal{R}_l$ is the set of racks with workload level l. For simplicity, $\vec{\beta}_i = \{\beta_{i,l}\}_{l\in\mathcal{L}}$ and $\vec{\gamma}_i = \{\gamma_{i,m}\}_{m\in\mathcal{M}}$ are defined for picker i. Recall that $\alpha_{i,m}$ denotes the probability that picker i has initial state m.

In the LP of Section 5, each pair of system state and action is associated with a constraint, while in ALP(i), each pair of picker state and rack level is associated

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

with a constraint, for the following reasons. (1) ALP(i) is based on the restricted LP for picker i, so (2) the system state is reduced to the working state of picker i; and (3) a feasible action is the assignment to picker i of one of the racks that have been aggregated by their workload levels.

ALP(i) is a column-dependent formulation, as different columns may have different racks in q and therefore different ALP(i) s and solutions $\left\{\theta_i, \vec{\beta}_i, \vec{\gamma}_i\right\}$. Let $x_{i,r}$ be a binary parameter that specifies whether rack r is in the sequence q that is assigned to picker i. ALP(i) can be reformulated as ALP $(i, \vec{x}_i)$, where $\vec{x}_i = \{x_{i,r}\}_{r\in\mathcal{R}}$ determines the subset of racks that is assigned to picker i.

$$\mathcal{V}_{i,q} = \max_{\theta_i, \vec{\beta}_i, \vec{\gamma}_i} \left\{\theta_i + \sum_{r\in\mathcal{R}} x_{i,r}\beta_{i,l_r} + \sum_{m\in\mathcal{M}} \alpha_{i,m}\gamma_{i,m}\right\} \; ALP\left(i, \vec{x}_i\right)$$
$$\text{s.t. Constraints}\,(12)-(14)$$

ALP $(i, \vec{x}_i)$ is of a much smaller scale than the restricted LP because ALP $(i, \vec{x}_i)$ aggregates racks by their workload levels; a rack sequence is changed to a workload level sequence; and different state-action pairs in the LP of Section 5 may be associated with the same constraint. ALP $(i, \vec{x}_i)$ has only $(L + M + 1)$ variables and $2LM$ constraints. This formulation is tractable, and the pricing subroutine that is based on it can be executed effectively.

6.2.2 Pricing based on Approximate LP. The pricing sub-problem (PSP) comprises finding a column with the minimum negative reduced cost among all pickers. The PSP is modeled below based on the reduced cost of a column, formulated in Equation (10). In the PSP, z_i is a binary variable that specifies whether a column of picker i has the minimum reduced cost, and $x_{i,r}$ is another binary variable that indicates whether rack r is assigned to picker i. We define $\vec{x} = \{x_{i,r}\}_{i\in\mathcal{P}, r\in\mathcal{R}}$ and $\vec{z} = \{z_i\}_{i\in\mathcal{P}}$. In the PSP, Equation (15) ensures that only one picker's column is selected and Eqs. (16) and (17) ensure the feasibility of a selected column, as discussed in Section 4.2.

$$\min_{\vec{x},\vec{z}} \sum_{i\in\mathcal{P}} \left(\mathcal{V}_{i,q} - \sum_{r\in\mathcal{R}} \pi_r a_{i,q,r} - \sigma_i\right) z_i \;\; (PSP)$$
$$\text{s.t.} \sum_{i\in\mathcal{P}} z_i = 1 \tag{15}$$
$$x_{i,r} = x_{i,r'} \;\; \forall i\in\mathcal{P}, o\in\mathcal{O}, r, r'\in\mathcal{R}_o, r\neq r' \tag{16}$$
$$\underline{b}\cdot z_i \le \sum_{r\in\mathcal{R}} \mathbb{W}_r x_{i,r} \le \bar{b}\cdot z_i \;\; \forall i\in\mathcal{P} \tag{17}$$
$$x_{i,r}, z_i \in \{0,1\} \; \forall i\in\mathcal{P}, r\in\mathcal{R}$$

Importantly, the objective function of the PSP is nonlinear. As formulated in ALP $(i, \vec{x}_i)$, $\mathcal{V}_{i,q}$ is not fixed but depends on $\vec{x}_i$, which is a variable set in the PSP. Then, the variables θ_i, $\vec{\beta}_i$, and $\vec{\gamma}_i$ in $\mathcal{V}_{i,q}$ must be solved simultaneously with $\vec{x}$ and $\vec{z}$ in the PSP.

To make the PSP tractable, the dual of ALP $(i, \vec{x}_i)$ is formulated as DALP $(i, \vec{x}_i)$, where $\vec{h}_i = \{h_{i,l,m}\}_{l\in\mathcal{L}, m\in\mathcal{M}}$ and $\vec{g}_i = \left\{g_{i,l,m}\right\}_{l\in\mathcal{L}, m\in\mathcal{M}}$ are the dual variables of constraints (12) and (13), respectively.

$$\min_{\vec{h}_i, \vec{g}_i} \sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} \mathbb{E}[t_{i,m,l}]_{r\in\mathcal{R}_l} \left(h_{i,l,m} + g_{i,l,m}\right) \; DALP\left(i, \vec{x}_i\right)$$
$$s.t. \sum_{m\in\mathcal{M}} h_{i,l,m} + \sum_{m\in\mathcal{M}} g_{i,l,m} = \sum_{r\in\mathcal{R}_l} x_{i,r} \;\; \forall l\in\mathcal{L}$$
$$\sum_{l\in\mathcal{L}} \left(1 - p_{i,l,m,m}\right) h_{i,l,m} - \sum_{l\in\mathcal{L}}\sum_{m'\in\mathcal{M}\setminus m} p_{i,l,m',m} h_{i,l,m'} + \sum_{l\in\mathcal{L}} g_{i,l,m} = \alpha_{i,m} \;\; \forall m\in\mathcal{M}$$
$$\sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} g_{i,l,m} = 1$$
$$\vec{h}_i \ge 0, \vec{g}_i \ge 0$$

The PSP is then transformed into the DPSP based on DALP $(i, \vec{x}_i)$. In the DPSP, the objective function integrates the objectives of the PSP and DALP $(i, \vec{x}_i)$. Constraints (18)–(20) are based on DALP $(i, \vec{x}_i)$. Constraints (21) and (22) define the relationships between $\vec{x}$ and $\vec{z}$. Constraints (15)–(17) are derived from the PSP.

$$\min_{\substack{\vec{x},\vec{z}\\ \vec{h},\vec{g}}} \sum_{i\in\mathcal{P}}\sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} \mathbb{E}[t_{i,m,l}]_{r\in\mathcal{R}_l} \left(h_{i,l,m} + g_{i,l,m}\right) - \sum_{i\in\mathcal{P}} \sigma_i z_i - \sum_{i\in\mathcal{P}}\sum_{r\in\mathcal{R}} \pi_r x_{i,r} \;\; (DPSP)$$
$$\text{s.t.} \sum_{m\in\mathcal{M}} h_{i,l,m} + \sum_{m\in\mathcal{M}} g_{i,l,m} = \sum_{r\in\mathcal{R}_l} x_{i,r} \;\; \forall i\in\mathcal{P}, l\in\mathcal{L} \tag{18}$$
$$\sum_{l\in\mathcal{L}} \left(1 - p_{i,l,m,m}\right) h_{i,l,m} - \sum_{l\in\mathcal{L}}\sum_{m\prime\in\mathcal{M}\setminus m} p_{i,l,m\prime,m} h_{i,l,m\prime} + \sum_{l\in\mathcal{L}} g_{i,l,m} = \alpha_{i,m} z_i \;\; \forall i\in\mathcal{P}, m\in\mathcal{M} \tag{19}$$
$$\sum_{l\in\mathcal{L}}\sum_{m\in\mathcal{M}} g_{i,l,m} = z_i \;\; \forall i\in\mathcal{P} \tag{20}$$
$$x_{i,r} \le z_i \;\; \forall i\in\mathcal{P}, r\in\mathcal{R} \tag{21}$$
$$z_i \le \sum_{r\in\mathcal{R}} x_{i,r} \;\; \forall i\in\mathcal{P} \tag{22}$$
$$\text{Constraints}\,(15)-(17)$$

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

$$x_{i,r}, z_i \in \{0,1\} \forall i \in \mathcal{P}, r \in \mathcal{R}; \vec{h} \geq 0, \vec{g} \geq 0$$

PROPOSITION 2. *The DPSP formulation has the same optimal objective value as the PSP formulation. (See Appendix S12 for proof.)*

The DPSP is formulated as a mixed 0-1 linear model with $P(R+1)$ binary variables, $2P{\cdot}L{\cdot}M$ linear variables, and $P(L + M + R + O + 4) + 1$ constraints if each order in $\mathcal{O}$ is associated with two racks. DPSP is a tractable formulation for real-world cases with a number of racks and pickers. It is solved to optimality using a MIP solver, returning a column with a negative reduced cost.

6.3. Branching

To handle the non-integral solution to the restricted master problem, a branching strategy that was proposed by Savelsbergh (1997), based on fixing a single variable, is used. The strategy firstly extracts the relationship $x_{i,r}$ for any picker-rack pair (i,r) from the SBM solution by setting $x_{i,r} = \sum_{q \in \mathcal{Q}_i} a_{i,q,r} y_{i,q}$. It then branches on a fractional variable $x_{i,r}$ with the fractional part closest to 0.5. Since the DPSP contains variable $x_{i,r}$, the constraint $x_{i,r} = 1$ is added to one branch and $x_{i,r} = 0$ is added to the other. In the case of a tie, the variable that is associated with the largest picking time $\mathcal{V}_{i,q}$ is selected. Using such a branching scheme, the pricing sub-problem structure is preserved. To obtain a feasible solution rapidly, the depth-first node selection strategy is applied.

6.4. Sequencing

The robot scheduling problem comprises assigning racks to pickers and sequencing the racks assigned to every picker. Using the ADP-based branch-and-price approach, a solution to the rack assignment sub-problem can be obtained. Then, the racks that are assigned to every picker can be sequenced based on the LP in Section 5. However, owing to the curse of dimensionality, the LP can only be solved for very small cases. To handle the sequencing sub-problem for large real-world cases, the policies that are implied by the dual variables of h and g in Section 6.2.2 are applied.

Given a rack assignment solution $\vec{x} = \{x_{i,r}\}_{i \in \mathcal{P}, r \in \mathcal{R}}$, the racks that are assigned to every picker are sequenced using the following procedure. Recall that $x_{i,r}$ specifies whether rack r has been assigned to picker i. First, the dual variables of $\vec{h}_i \left(= \{h_{i,l,m}\}_{l \in \mathcal{L}, m \in \mathcal{M}}\right)$ and $\vec{g}_i \left(= \left\{g_{i,l,m}\right\}_{l \in \mathcal{L}, m \in \mathcal{M}}\right)$ are calculated using DALP $(i, \vec{x}_i)$. These variables provide the distribution over the pairs of picker state and rack level. $\mathbb{p}_{i,l,m} = \left(h_{i,l,m} + g_{i,l,m}\right) / \left(\sum_{l \in \mathcal{L}} h_{i,l,m} + \sum_{l \in \mathcal{L}} g_{i,l,m}\right)$ is taken as the probability of selecting a rack with workload level l when picker i is in state m. Then, the roulette wheel strategy is applied to simulate an initial state m_0 of picker i, based on $\{\alpha_{i,m}\}_{m \in \mathcal{M}}$, and to select a rack level, based on $\left\{\mathbb{p}_{i,l,m_0}\right\}_{l \in \mathcal{L}}$. One of the racks with the selected level is appended to the sequence of picker i, and the steps of simulating the picker state and selecting a rack are repeated until all assigned racks have been appended. If the selected level has no racks, then one of the un-appended racks is selected at random. Finally, a rack sequence would be obtained. The above procedure is repeated to find multiple (such as ten) rack sequences for each picker and to return the one with the shortest expected picking time.

7. Case Studies

This section presents a numerical study of the proposed ADP-based branch-and-price approach on the robot scheduling problem using the data of the target company. The obtained data are important in the investigation of the variability of picker states, providing reasonable estimates of the parameters of the problem, such as the transition probabilities of picker states and the time for which a rack is handled by a picker in a particular working state, and evaluating the feasibility and effectiveness of the approach in real-world cases.

First, in Section 7.1, the performance of the proposed approach is tested by using various cases and comparing our solutions to the actual operations of the company. Section 7.2 will then investigate some common rack assignment and sequencing policies and compare our solutions to those generated by these policies. Section 7.3 will compare the performances of pickers with different variability of working states.

Gurobi 9.0 is used as the MIP solver. The ADP-based branch-and-price approach is coded using the C#/Gurobi API. The data are processed using MS SQL Server. All computations are conducted on a computer with an Intel I7 2.69 GHz processor and 16GB RAM.

In practice, customer orders are input into the picking system on a rolling basis. The robot scheduling problem is concerned with handling the orders in a batch, which may contain dozens of orders. When the system receives orders, it identifies the racks that will satisfy those orders. Based on the orders and racks, robots are scheduled. The orders are grouped herein by their ordering times to simulate the rolling manner in which the picking system operates and cases with

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

10–95 orders are generated. The rack(s) associated with each order are obtained from the data. Each picker is assumed to begin in the *normal* state. The lower and upper bounds on picker workload, $\underline{b}$ and $\bar{b}$, are set to $0.7\sum_{r\in\mathcal{R}}\mathbb{w}_r/P$ and $1.4\sum_{r\in\mathcal{R}}\mathbb{w}_r/P$, respectively.

7.1. Performance of Proposed Approach

First, the performance of the proposed branch-and-price approach in cases with 10 to 95 orders is tested. Table 3 presents our solutions and the corresponding operations that were actually performed according to the data; the columns R, O, P, M, and L show the numbers of racks, orders, pickers, picker states, and workload levels of racks, respectively. The total picking time, average picker state, CPU time, and the picking time gap are shown in the table. The average picker state τ is given by Equation (23), where $p_{i,j,k}$ is the probability that picker i is in state j at stage k and is determined by a given solution. Most cases in the table involve five working states of pickers and five workload levels of racks, as presented in Section 3. A few cases involve three states and three levels that can be identified from the parameters in Appendix S13. Some of the pickers in the data are selected for these cases. In each case, the numbers of pickers in the three groups presented in Section 3.3.2 are equal.

$$\tau=\frac{\left(\sum_{i\in\mathcal{P}}\sum_{j\in\{1,2\ldots,M\}}\sum_{k\in\{0,1,2\ldots,R-1\}}j\cdot p_{i,j,k}\right)}{(P\cdot R)} \tag{23}$$

Table 3 shows that the proposed branch-and-price approach performs well in a number of cases. It takes less than half a minute to generate a solution for the cases with 20 racks, whereas the LP model in Section 5 cannot even be built in an hour for the case with only ten racks. Constructing the ALP(i) model in Section 6.2.1 is also quite time-consuming because it enumerates all possible assignments of racks to pickers. The approach proposed herein can solve the cases with 120 racks in less than 20 minutes.

Table 3 also reveals that our solution from the proposed approach takes less time than the actual solution that did not consider schedule-induced fluctuations of pickers' working states, to handle the same set of racks. The average improvement of our solution in total picking times is approximately 10%. The improvement results only from the adjustments of the rack assignment and sequence for pickers. The average state of a picker in our solution is smaller, indicating that she spends more time in a better state. Since the average picker state is obtained by weighing the *best*, *better*, *normal*, *worse*, and *worst* states using integers from 1 to 5, respectively, a smaller average corresponds to a better overall state. This result demonstrates that our solution is able to detect the change in picker states and promote the state transition of pickers in a favorable direction.

7.2. Comparison of Proposed Approach and Common Policies

This section compares the proposed branch-and-price approach to some common policies that can be easily implemented in practice. Since the scheduling problem comprises assigning racks to pickers and sequencing the racks that are assigned to each picker, Section 7.2.1 first investigates common policies that are applicable to the two sub-problems, and then Section 7.2.2 compares the results obtained using the proposed approach to those obtained by combining these policies.

7.2.1 Rack Assignment and Sequencing Policies. Racks are commonly assigned to pickers in a manner that makes the workload of all pickers as equal as possible. This equal assignment policy is formulated in Appendix S14.

Given the assignment of a set of racks to a picker, two straightforward policies can be applied to

Table 3 Performance of Proposed Branch-and-Price Approach

						Our solution			Actual solution		
No.	R	O	P	M	L	Picking time (in seconds)	Average Picker State	CPU time (in seconds)	Picking time (in seconds)	Average picker state	Picking time gap
1	10	9	3	3	3	135.76	3.08	2.38	148.61	3.27	9.47%
2	20	18	3	3	3	277.83	3.14	9.28	310.14	3.35	11.63%
3	20	18	6	3	3	277.65	3.15	15.03	306.72	3.29	10.47%
4	20	18	6	5	5	277.18	3.28	20.82	311.15	3.87	12.26%
5	50	42	6	5	5	697.90	3.19	108.23	759.23	3.93	8.79%
6	50	42	9	5	5	693.70	3.23	143.84	763.68	3.90	10.09%
7	90	74	6	5	5	1259.86	3.17	487.62	1367.17	3.71	8.52%
8	90	74	9	5	5	1255.70	3.21	593.90	1374.20	3.88	9.44%
9	120	95	6	5	5	1680.40	3.17	1067.42	1854.67	3.82	10.37%
10	120	95	9	5	5	1674.33	3.20	1158.91	1827.53	3.94	9.15%

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

sequence these racks - random sequencing and rotation. Under the random sequencing policy, racks are sequenced randomly. Under the rotation policy, a rack with workload level l' is appended to the sequence if the last rack in the sequence has workload level l and l' is one level less than l or the *most* level if l is the *least*. If no rack has a workload level l', then a rack with one level less than l' is appended until all racks have been appended. This policy alternates the workload levels of racks so that racks of various workloads are distributed evenly.

7.2.2 Comparison of Policies. Two rack assignment policies and three rack sequencing policies have now been specified, and they are shown in Table 4. A problem solution can be obtained by combining an assignment policy with a sequencing policy. The six combinations of policies are Our-A/Our-S, Our-A/Random-S, Our-A/Rotation-S, Equal-A/Our-S, Equal-A/Random-S, and Equal-A/Rotation-S, respectively.

To test the performances of the policy combinations, seven picking tasks are constructed, based on the obtained data, and shown in Table 5. They have the same total picking workload but different average workload levels of racks and different numbers of racks. Due to the same total picking workload, the larger the number of racks, the lower the average rack level. In a simulation, six pickers are selected to perform the seven tasks under each policy combination. Among the six pickers, two from the first group, two from the second, and two from the third are selected. Forty-two solutions for the seven tasks under the six combinations of policies are thus obtained.

Figure 4 displays the picking time gaps between Our-A/Our-S and the other combined policies. The horizontal axis of the figure shows the average rack level. Generally, a larger average rack level corresponds to more racks with a large level. However, a rack with a large level must be scheduled with care: if such a rack is assigned to a picker in a bad state, the picking time may be long, and the picker will probably transition to a worse state, affecting her subsequent performance; if such a rack is sent to a picker in a good state, the picking time will probably be greatly reduced. Therefore, the transitions of picker states must be understood and then used to ensure that the scheduling of racks keeps up with the fluctuation of picker states. The numerical results show that the proposed approach outperforms the other policy combinations in reducing the impact of the variability of picker states on picking time.

Table 4 All Rack Assignment and Sequencing Policies

Policy name	Policy explanation
Our-A	The rack assignment policy based on our approach.
Equal-A	The equal assignment policy.
Our-S	The rack sequencing policy based on our approach.
Random-S	The random sequencing policy.
Rotation-S	The rotation sequencing policy.

Table 5 Picking Tasks with Same Workload

No.	*R*	*P*	*M*	*L*	Avg. rack level
1	87	6	5	5	1.5
2	68	6	5	5	2.0
3	58	6	5	5	2.5
4	50	6	5	5	3.0
5	44	6	5	5	3.5
6	39	6	5	5	4.0
7	36	6	5	5	5.0

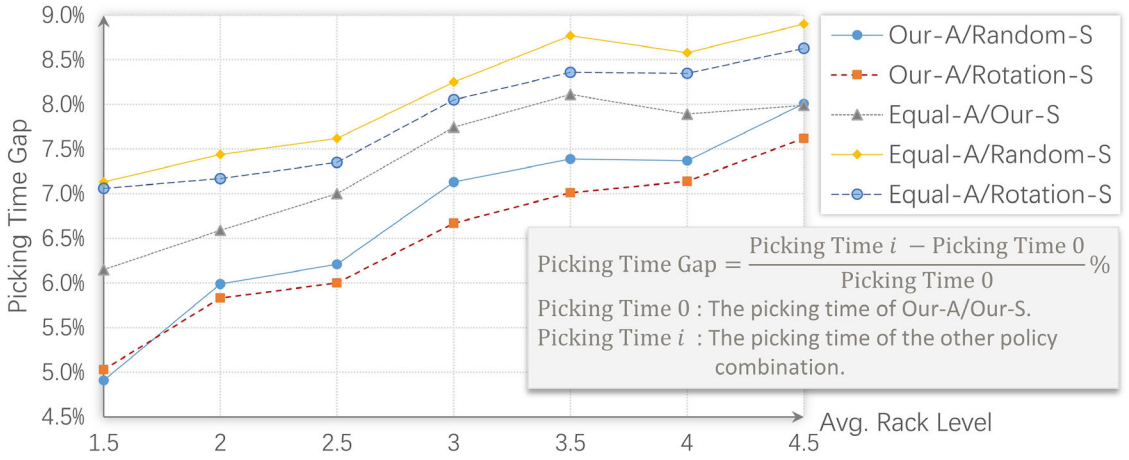
Figure 4 demonstrates that the two combinations Our-A/Random-S and Our-A/Rotation-S outperform the other three, which include the Equal-A policy. This result is not surprising because the Equal-A policy assigns racks by considering only the workloads of the racks but not the pickers' varying states. However, the states of pickers after picking from racks with different workloads may be differently distributed. The approach proposed herein captures the varying states of pickers and assigns racks to efficient pickers in a manner that reduces the overall picking time. The fundamental role of the rack assignment policy is evident. Without good rack assignment, obtaining a good solution with a short picking time is difficult, regardless of the rack sequencing policy.

Figure 4 shows that the Rotation-S policy is worse than the Our-S policy but better than the Random-S policy if these sequencing policies are combined with the same assignment policy. The Rotation-S policy is based on the assumption that a picker enters a bad state after picking from a large-level rack. Therefore, it schedules a rack with a lower level for a picker who has just picked from a large-level rack to relieve the pressure on that picker and bring her back into a good state. However, a bad state does not always follow picking from a large-level rack. The Our-S policy identifies the probabilities associated with the states of a picker more accurately and schedules racks for a picker based on the probabilities. On the contrary, the Random-S policy fails to consider the picker state at all and so makes finding a solution with a short picking time difficult to obtain.

Figure 5 displays the average picker states under various policy combinations. Generally, a worse average picker state corresponds to higher probabilities of the bad states of pickers. A larger average rack level corresponds to greater difficulty in controlling picker states because a picker's state is likely to deteriorate after she picks from a large-level rack. As shown in the figure, as the average rack level increases, Our-A/

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

Figure 4 Picking Time Gaps between Our-A/Our-S and the Other Combined Policies [Color figure can be viewed at wileyonlinelibrary.com]



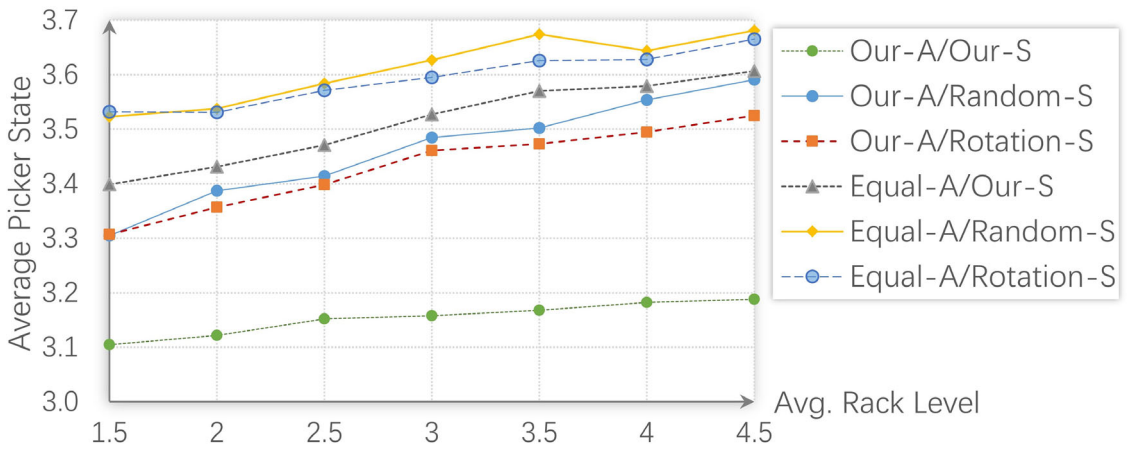
Our-S yields a steady trend in the average picker state whereas the other policy combinations yield faster worsening of states. Effective control of picker states is required to reduce picking time as much as possible. This result also explains the picking time gaps of Figure 4.

To gain a more detailed insight into picker states, the probabilities of each state are summed over all the pickers under the Our-A/Our-S and Equal-A/Rotation-S policies. Figure 6, which comprises seven pairs of columns, shows the percentage of each state for all pickers. The left column of each pair provides the state distribution under Our-A/Our-S, and the right column provides that under Equal-A/Rotation-S. The five states are colored from blue to red. As shown in the figure, pickers under Our-A/Our-S are associated with more blue and yellow and less pink and red than those under Equal-A/Rotation-S. As the average rack level increases, under Equal-A/Rotation-S, fewer states are blue and more are pink or red, explaining the worse average picker state and longer picking time under Equal-A/Rotation-S because a picker generally spends more time picking from a rack in a bad state than in a good one. In contrast, Our-A/Our-S utilizes the variability of picker states to ensure that pickers are in good states as often as possible. Without such utilization, picker states may change freely and arbitrarily.

7.3. Comparison of Performances of Pickers with Different Working State Variability

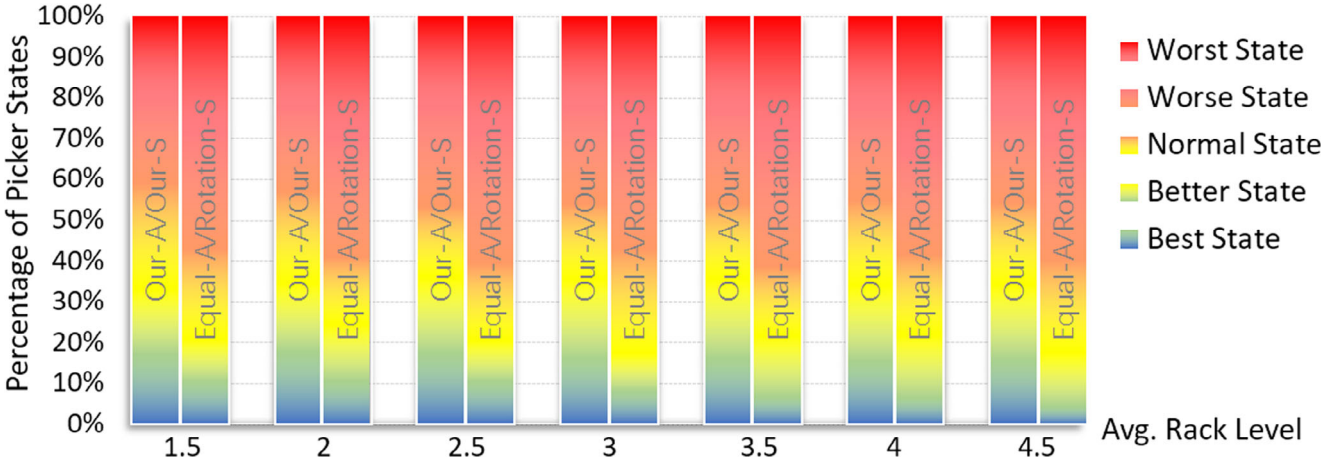
In Section 3.3.2, pickers were divided into three groups by their working state variability. As discussed in that section, their average working

Figure 5 Average Picker States under Policy Combinations [Color figure can be viewed at wileyonlinelibrary.com]



Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

Figure 6 Percentages Associated with Picker States under Policy Combinations [Color figure can be viewed at wileyonlinelibrary.com]



efficiencies are comparable, and the difference in the three groups of the pickers lies only in the degree of fluctuation of their working states. To compare the performances of pickers with different working state variability, pickers from the three groups are selected to construct five sets of pickers, which are shown in Table 6. If a group does not have enough pickers, a picker in the group can be replicated multiple times. In a simulation, each set of pickers performs one of the seven picking tasks in Table 5, under a policy combination of Our-A/Our-S or Equal-A/Rotation-S. Thus, 35 simulations are conducted under each of these two policy combinations.

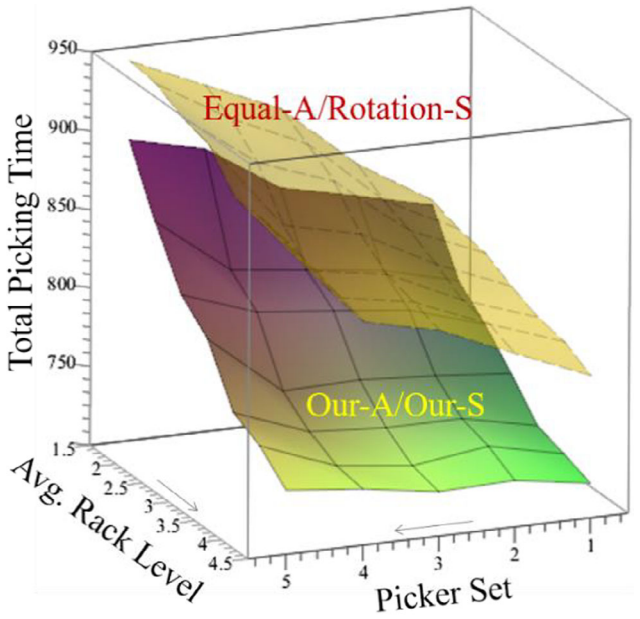
Figure 7 presents the results obtained under Our-A/Our-S and Equal-A/Rotation-S when the five selected sets of pickers perform the seven specified tasks. The two horizontal axes represent the average rack level and the set of pickers, respectively, and the vertical axis represents the total picking time. Two surfaces can be seen in the figure. The Our-A/Our-S surface is below the other surface, indicating that picking times are shorter under Our-A/Our-S. The two surfaces show different variations in picking times over average rack level and picker set. The following two observations are made:

(1) Along the picker set axis, the picking time increases slowly under Our-A/Our-S. Given a fixed average rack level, corresponding to a fixed set of racks from which items are to be picked, the picking time spent by pickers in set i is slightly less than that spent by pickers in set $i + 1$ ($1 \le i < 5$). The difference between the picking times of adjacent sets of pickers is about 1.3%. The differences between the picking times of picker sets 1 and 3, and those of sets 3 and 5 are only about 2.5%. Such a small difference reflects the comparable productivity of different groups of pickers if they handle racks based on the proposed solutions herein. However, the situation is different under Equal-A/Rotation-S. As shown in Figure 7, the Equal-A/Rotation-S surface reveals a more obvious increase in the picking times spent by different sets of

Table 6 Sets of Pickers from Different Groups

	Number of pickers from different groups		
No.	**Group 1**	**Group 2**	**Group 3**
1	6	0	0
2	3	3	0
3	0	6	0
4	0	3	3
5	0	0	6

Figure 7 Results of Picking Tasks Done by Sets of Pickers under Our-A/Our-S and Equal-A/Rotation-S [Color figure can be viewed at wileyonlinelibrary.com]



Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

pickers handling the same set of racks. The differences between the picking times of picker sets 1 and 3 and those of sets 3 and 5 are over 6.5%, and that between picker sets 1 and 5 is 13.3%. The difference in the state variability among picker groups is not absorbed by Equal-A/Rotation-S policies. These results imply that for a given set of racks, the greater the average state variability of pickers, the larger the difference in picking times between the two surfaces, and the more advantageous and applicable the proposed approach.

(2) Along the average rack level axis in Figure 7, it can be observed that a larger average rack level corresponds to a shorter total picking time, regardless of the set of pickers. This result is expected because a heavier average workload of racks generally corresponds to more items that need to be picked from a rack, more items caught by a picker in a picking operation, and less time taken by a picker to pick out the items. As the average rack level increases, the picking times of the two surfaces decrease at different rates. Given a fixed picker set, as the average rack level increases, the picking time under Our-A/Our-S falls faster than that under Equal-A/Rotation-S and the difference between the picking times increases. This result reveals the different capabilities of the two policy combinations in adjusting picker states by assigning and sequencing large-level racks, and that a larger average rack level corresponds to a greater advantage of the proposed approach over common policies.

8. Conclusion

The introduction of mobile racks and robots into warehouses provides a cost-effective solution to order picking problems by re-distributing tasks involved in the problem between humans and robots, but it also creates challenges in coordinating picking activities. In such a warehouse, robots must cooperate with human operators whose working states are prone to fluctuation. Robot scheduling in the warehouse is a hard problem because it involves assigning racks with various workloads and setting service sequences for the pickers with different variability of working states.

To achieve a robot schedule that can anticipate changes in the working states of human pickers, this work developed a data-driven approach to determining the state transition probability of each picker. The robot scheduling problem was then formulated as an SDP and a corresponding linear programming model was constructed. Due to the curse of dimensionality, the problem is reformulated as a sequence-based model that is integrated into a branch-and-price framework. An ADP and its dual are constructed to make tractable the pricing sub-problem, which is the most challenging part of the branch-and-price framework.

The ADP-based branch-and-price approach is applied to data that were provided by one of the dominant e-commerce companies in China. The proposed approach is shown to solve a case with 50 racks in under two minutes, yielding a solution with a picking time that is 10% shorter than the actual schedule, which did not consider pickers' state fluctuations. It implies that our solution enables pickers to perform about 10% picking work more during a given working period. This increased picker efficiency allows a warehouse to hire one fewer picker for every 10, providing a large cost saving especially for giant companies like Amazon, JD, and Alibaba. For a company with 25,000 pickers, this saving amounts to tens of millions of dollars annually.

8.1. Theoretical and Practical Implications

The robot scheduling problem is a multi-agent SDP in which each agent has an independent state transition distribution and a state-dependent working efficiency. The problem is NP-hard because it can be reduced to the single machine scheduling problem with sequence-dependent setup times, which has already been proved to be NP-hard. The spaces of joint actions and states grow exponentially with the numbers of jobs, agents, and their states, making the computation of the value functions of multiple agents very challenging.

Due to the mutual independence of the agents' state transitions, the problem is transformed herein into multiple single-agent SDPs; ADP is introduced to formulate single-agent value functions, and an ADP-based branch-and-price approach is developed. To handle the nonlinear function that is encountered when ADP is embedded into the pricing sub-problem, the dual of an approximate LP (DALP) and a DALP-based dual are created. The method makes the sub-problem tractable and thus ensures the efficiency of the overall solution approach. Effective methods are also developed to assign jobs to, and sequence jobs for, agents, which are the two most important parts of handling a multi-agent SDP. Herein, jobs are assigned to agents based on their comparative advantages in performing different sets of jobs; jobs are sequenced for each agent according to the probability distribution, derived from the DALP, over the pairs of agent state and job. These methods contribute considerably to finding a good job assignment and sequencing solution to the multi-agent SDP.

Multi-agent SDPs can be found in a variety of domains, such as robotics, economics, and manufacturing. These SDPs share common characteristics,

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

such as multiple agents, mutually independent state transition distributions of agents, sequence-dependent processing costs, and considerable solution challenges. The approach herein provides an innovative and effective framework for solving multi-agent SDPs with the characteristics.

This work has the following implications for managerial practices.

Implication 1. ***Robot scheduling has direct effects on the working states of human pickers, which would further influence the performance of an order picking system***. This conclusion is drawn from Section 3 and Appendix S7. The data analysis in Section 3 shows that the working states of human pickers may fluctuate with the workloads of the racks that they handle. Human working states are important since they significantly affect the performance of an order picking system. As revealed by the analysis, a picker can spend twice as long handling a rack in a bad state as in a good state, and the picking times of multiple pickers in different states can vary even more. The model-based analysis in Appendix S7 further shows that a favorable handling sequence for multiple racks by a picker can reduce picking time by 18.7%. Therefore, scheduling robots to assign racks to, and sequence racks for, pickers and to align rack workloads with the working state fluctuations of human pickers, is crucial to enhancing the performance of an order picking system.

Implication 2. ***Pickers' degrees of fluctuation of working states may vary significantly. However, this variation does not result in a large variation among their picking times if scheduling is effective***. This conclusion is drawn from Section 7.3. Case studies in that section show that as the working states of pickers fluctuate more drastically, the workers tend to spend longer picking from a set of racks, so schedules must be designed to adapt to such drastic fluctuations. Common policies fail to provide good schedules for volatile pickers, who take much longer than steady pickers. In contrast, the variation in picking times among pickers is insignificant if racks are handled based on our solutions.

Implication 3. ***Understanding fluctuations of human working states and responding to them proactively can improve the performance of human–robot coordinated systems***. This conclusion is drawn from SSections 7.1 and 7.2. These sections show that a solution that proactively responds to fluctuations of human working states outperforms the schedules from common policies and the actual schedule, which did not consider these fluctuations, with a 10% shorter picking time. This improvement does not arise from any additional incentives or penalties (such as picker training, rewards, or working condition improvements). This result is satisfactory for not only decision makers but also pickers. The substantial improvement in the performance of the picking system confirms the importance of understanding fluctuations of human working states and responding to them proactively. This finding demonstrates the promise of human circadian rhythm-based robot scheduling for human–robot coordinated systems.

8.2. Limitations and Future Directions

Several limitations of this research and future directions for research should be considered. First, the proposed approach can be generalized for scheduling picker rotations. Order picking is labor-intensive and rotations are required to enable pickers to relax. Grouping pickers and specifying rotations with different lengths based on daily picking workload the variability of every picker's working state are challenging. Second, limited by the obtained data, this work did not model working behaviors of pickers, which may play an important role in determining working state and performance. Once behavior-related data, concerning for example the videos about the operations of pickers, are obtained, the causal chain, "picker behavior → working state → picking performance" could be studied. Third, variations in the behaviors of pickers in warehousing systems among countries would also be significant. These issues are left for future research.

Acknowledgments

The authors thank the editors and anonymous reviewers for their constructive comments and invaluable contributions to enhance the presentation of this paper. This work is supported by the National Natural Science Foundation of China (71971036, 71421001, 71531002), the Major Program of Key Disciplines in Dalian (2019J11CY002), and the Key R&D project of Liaoning Provincial Department of Science and Technology (2020JH2/10100042). This work is also sponsored by the Ministry of Science and Technology of Taiwan, R.O.C. (grant no.: MOST 109-2410-H-002-076-MY3), and the Ministry of Education, Singapore, under its 2019 Academic Research Fund Tier 3 grant call (Award ref: MOE-2019-T3-1-010).

Note

[1]Due to the data confidentiality agreement signed with the company, the company name and data are not disclosed. Please contact the first author if there are any further questions related.

References

Ae Lee, J., Y. Seok Chang, W. Karwowski. 2020. Assessment of working postures and physical loading in advanced order picking tasks: A case study of human interaction with

automated warehouse goods-to-picker systems. *Work* 1–12. Preprint. https://doi.org/10.3233/WOR-203337

Agussurja, L., S. F. Cheng, H. C. Lau. 2019. A state aggregation approach for stochastic multiperiod last-mile ride-sharing problems. *Transportation Science* **53**(1): 148–166.

Azadeh, K., R. De Koster, D. Roy. 2019. Robotized and automated warehouse systems: Review and recent developments. *Transportation Science* **53**(4): 917–945.

Barnhart, C., E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, P. H. Vance. 1998. Branch-and-price: Column generation for solving huge integer programs. *Operations Research* **46**(3): 316–329.

Batt, R. J., S. Gallino. 2019. Finding a needle in a haystack: The effects of searching and learning on pick-worker performance. *Manage. Sci.* **65**(6): 2624–2645.

Boysen, N., D. Briskorn, S. Emde. 2017. Sequencing of picking orders in mobile rack warehouses. *Eur. J. Oper. Res.* **259**(1): 293–307.

Bozer, Y. A., J. A. White. 1990. Design and performance models for end-of-aisle order picking systems. *Manage. Sci.* **36**(7): 852–866.

Chen, C. M., Y. Gong, R. B. De Koster, J. A. Van Nunen. 2010. A flexible evaluative framework for order picking systems. *Prod. Oper. Manag.* **19**(1): 70–82.

Dai, J. G., P. Shi. 2019. Inpatient overflow: An approximate dynamic programming approach. *Manufacturing & Service Operations Management* **21**(4): 894–911.

De Vries, J., R. De Koster, D. Stam. 2016. Aligning order picking methods, incentive systems, and regulatory focus to increase performance. *Prod. Oper. Manag.* **25**(8): 1363–1376.

Demaitre, E. 2019. Geek+ raises another $150M for global expansion of autonomous mobile robots. Available at https://www.therobotreport.com/geek-raises-funds-autonomous-mobile-robots/ (accessed date July 01, 2020).

Designboom. 2014. 15,000 amazon kiva robots drive eighth generation fulfillment center. Available at https://www.designboom.com/technology/amazon-kiva-robots-generation-fulfillment-12-02-2014/ (accessed date December 09, 2020).

Diamant, A., J. Milner, F. Quereshy. 2018. Dynamic patient scheduling for multi-appointment health care programs. *Prod. Oper. Manag.* **27**(1): 58–79.

Ertem, M., F. Ozcelik, T. Saraç. 2019. Single machine scheduling problem with stochastic sequence-dependent setup times. *Int. J. Prod. Res.* **57**(10): 3273–3289.

Glock, C. H., E. H. Grosse, H. Abedinnia, S. Emde. 2019. An integrated model to improve ergonomic and economic performance in order picking by rotating pallets. *Eur. J. Oper. Res.* **273**(2): 516–534.

Grosse, E. H., C. H. Glock, M. Y. Jaber, W. P. Neumann. 2015. Incorporating human factors in order picking planning models: Framework and research opportunities. *Int. J. Prod. Res.* **53**(3): 695–717.

Gu, J., M. Goetschalckx, L. F. McGinnis. 2010. Solving the forward-reserve allocation problem in warehouse order picking systems. *Journal of the Operational Research Society* **61**(6): 1013–1021.

Guedes, P. C., D. Borenstein. 2018. Real-time multi-depot vehicle type rescheduling problem. *Transportation Research Part B: Methodological* **108**: 217–234.

Khachatryan, M., L. F. McGinnis. 2014. Picker travel time model for an order picking system with buffers. *IIE Trans.* **46**(9): 894–904.

King, R. E., T. J. Hodgson, F. W. Chafee. 1993. Robot task scheduling in a flexible manufacturing cell. *IIE Trans.* **25**(2): 80–87.

Kumar, S., V. Mookerjee, A. Shubham. 2018. Research in operations management and information systems interface. *Prod. Oper. Manag.* **27**(11): 1893–1905.

Lecher, C. 2019. How Amazon automatically tracks and fires warehouse workers for 'productivity'. Available at https://www.theverge.com/2019/4/25/18516004/amazon-warehouse-fulfillment-centers-productivity-firing-terminations (accessed date February 01, 2021).

Lee, H. Y., C. C. Murray. 2019. Robotics in order picking: Evaluating warehouse layouts for pick, place, and transport vehicle routing systems. *Int. J. Prod. Res.* **57**(18): 5821–5841.

Lim, Y. F. 2011. Cellular bucket brigades. *Operations Research* **59** (6): 1539–1545.

Lim, Y. F. 2017. Performance of cellular bucket brigades with hand-off times. *Prod. Oper. Manag.* **26**(10): 1915–1923.

Lim, Y. F., Y. Wu. 2014. Cellular bucket brigades on U-lines with discrete work stations. *Prod. Oper. Manag.* **23**(7): 1113–1128.

Lübbecke, M. E., J. Desrosiers. 2005. Selected topics in column generation. *Operations Research* **53**(6): 1007–1023.

Maxwell, M. S., M. Restrepo, S. G. Henderson, H. Topaloglu. 2010. Approximate dynamic programming for ambulance redeployment. *INFORMS Journal on Computing* **22**(2): 266–281.

Minkoff, A. S. 1993. A Markov decision model and decomposition heuristic for dynamic vehicle dispatching. *Operations Research* **41**(1): 77–90.

Olsen, T. L., B. Tomlin. 2020. Industry 4.0: Opportunities and challenges for operations management. *Manufacturing & Service Operations Management* **22**(1): 113–122.

Powell, W. B. 2010. Merging AI and OR to solve high-dimensional stochastic optimization problems using approximate dynamic programming. *INFORMS Journal on Computing* **22**(1): 2–17.

Puterman, M. L. 2014. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley & Sons, Hoboken, NJ.

Ruben, R. A., F. R. Jacobs. 1999. Batch construction heuristics and storage assignment strategies for walk/ride and pick systems. *Manage. Sci.* **45**(4): 575–596.

Savelsbergh, M. 1997. A branch-and-price algorithm for the generalized assignment problem. *Operations Research* **45**(6): 831–841.

Shehhi, S. A., N. King, S. Amer, J. Mohammad, A. Dhabi 2019. Modeling the ergonomics of Goods-to-Man order picking. Proceedings of the 2019 IISE Annual Conference.

Topaloglu, H., W. B. Powell. 2006. Dynamic-programming approximations for stochastic time-staged integer multicommodity-flow problems. *INFORMS Journal on Computing* **18**(1): 31–42.

Van der Gaast, J. P., R. B. De Koster, I. J. Adan, J. A. Resing. 2020. Capacity analysis of sequential zone picking systems. *Operations Research* **68**(1): 161–179.

Webster, S., R. A. Ruben, K. K. Yang. 2012. Impact of storage assignment decisions on a bucket brigade order picking line. *Prod. Oper. Manag.* **21**(2): 276–290.

Weidinger, F., N. Boysen, D. Briskorn. 2018. Storage assignment with rack-moving mobile robots in KIVA warehouses. *Transportation Science* **52**(6): 1479–1495.

Wolff, D. 2019. RightHand Robotics: Automated piece-picking solutions. Available at https://ilp.mit.edu/newsstory.jsp?id=25718 (accessed date July 01, 2021).

Zou, B., R. D. Koster, X. Xu. 2018. Operating policies in robotic compact storage and retrieval systems. *Transportation Science* **52**(4): 788–811.

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406

Appendix S1: Values of the Parameters in Section 3.2.
Appendix S2: The Proof of the Hypothesis in Section 3.3.1.
Appendix S3: The Statistics of the Two-sided Chi-square Test in Section 3.3.2.
Appendix S4: An Example for the Approach of Measuring $\mathbb{P}_{i,l,j,k}$ in Section 3.3.3.
Appendix S5: Statistical Evidences for Assumptions.
Appendix S6: Illustrative Example for the SDP in Section 4.
Appendix S7: Impact of Picker's State on Picking Time.
Appendix S8: State Transition Probabilities and Picking Times of Pickers in "equi-gap" SDP.
Appendix S9: Proof of Proposition 1 of Appendix G.
Appendix S10: Proof of Proposition 2 of Appendix G.
Appendix S11: Proof of Proposition 1 in the Manuscript.
Appendix S12: Proof of Proposition 2 in the Manuscript.
Appendix S13: Picker States and Workload Levels in Case Studies.
Appendix S14: Formulation of the Equal Assignment Policy.

Please Cite this article in press as: Wang, Z., et al. Robot Scheduling for Mobile-Rack Warehouses: Human–Robot Coordinated Order Picking Systems. *Production and Operations Management* (2021), https://doi.org/10.1111/poms.13406