OR Spectrum
https://doi.org/10.1007/s00291-022-00695-8

ORIGINAL ARTICLE

Cost-neutral reduction of infection risk in picker-to-parts warehousing systems

Maximilian Löffler[1] · **Michael Schneider**[1] · **Ivan Žulj**[2]

Received: 22 September 2021 / Accepted: 8 August 2022
© The Author(s) 2022

Abstract

The rapid and severe outbreak of COVID-19 caused by SARS-CoV-2 has heavily impacted warehouse operations around the world. In particular, picker-to-parts warehousing systems, in which human pickers collect requested items by moving from picking location to picking location, are very susceptible to the spread of infection among pickers because the latter generally work close to each other. This paper aims to mitigate the risk of infection in manual order picking. Given multiple pickers, each associated with a given sequence of picking tours for collecting the items specified by a picking order, we aim to execute the tours in a way that minimizes the time pickers simultaneously spend in the same picking aisles, but without changing the distance traveled by the pickers. To achieve this, we exploit the degrees of freedom induced by the fact that picking tours contain cycles which can be traversed in both directions, i.e., at the entry to each of these cycles, the decision makers can decide between the two possible directions. We formulate the resulting picking tour execution problem as a mixed integer program and propose an efficient iterated local search heuristic to solve it. In extensive numerical studies, we show that an average reduction of 50% of the total temporal overlap between pickers can be achieved compared to randomly executing the picking tours. Moreover, we compare our approach to a zone picking approach, in which infection risk between pickers can be almost eliminated. However, compared to our approach, the results show that the zone picking approach increases the makespan by up to 1066%.

Keywords Routing · Iterated local search · Picker routing · COVID-19 · Pandemics

✉ Ivan Žulj
zulj@uni-hohenheim.de

Maximilian Löffler
loeffler@dpo.rwth-aachen.de

Michael Schneider
schneider@dpo.rwth-aachen.de

[1] Deutsche Post Chair – Optimization of Distribution Networks, RWTH Aachen University, Aachen, Germany

[2] Department of Procurement and Production, University of Hohenheim, Stuttgart, Germany

Published online: 23 November 2022

1 Introduction

Warehousing is an essential component of many supply chains (see, e.g., Boysen et al. 2021) and has a strong influence on on-time deliveries and customer satisfaction. High cost pressure, steadily increasing global retail sales volumes (Statista 2019), and next- or same-day deliveries force warehouse managers to improve the performance of their warehousing processes to meet the customer expectations for fast delivery and to gain advantages over competitors (see, e.g., Weidinger 2018; Boysen et al. 2019).

The most labor-intensive warehousing activity is order picking, which is the process of retrieving items from their storage locations to fulfill customer orders. Studies indicate that a large share of all order picking systems in Western Europe follow the traditional picker-to-parts warehousing setup, in which pickers move through the warehouse to retrieve the requested items from their storage locations (Napolitano 2012). The major drawback of such systems is the large fraction of unproductive picker walking time (de Koster et al. 2007; Tompkins et al. 2010). While there are technologies to automate order picking (see, e.g., Azadeh et al. 2019), warehouse managers rely on manual order picking because human pickers are flexible and can adapt to changes in real time compared to automated systems (Grosse et al. 2014).

The rapid spread of SARS-CoV-2 forces warehouse managers to reduce the risk of spreading the virus to avoid a complete stop of the workflow due to infected pickers. Because SARS-CoV-2 is mainly spread person-to-person through close contact, and pickers work close to each other, the risk of virus transmission in manual order picking is very high. As a response to COVID-19, warehouse managers initially aimed at maintaining physical distance between pickers by restricting the number of pickers allowed in a picking area at one time or by implementing one-way picking aisles to avoid congestion or picker blocking (DC Velocity 2020). Because such safety measures make it hard to meet pre-pandemic picking performance due to longer picking routes, Amazon, for example, launched a social distancing monitor based on artificial intelligence, which visually alerts pickers when physical distance is not maintained (Amazon 2020). Softeon, a global supply chain software vendor, offers a software that alerts a picker if the picker's next pick is in a picking aisle which is already occupied by another picker (DC Velocity 2020).

This study aims at mitigating the risk of becoming infected or spreading SARS-CoV-2 (or other infectious diseases) in manual order picking by minimizing the time pickers simultaneously spend in the same picking aisles, which supports maintaining social distancing practices. We assume that the risk of infection is proportional to the time that pickers spend in close proximity to each other, which is in accordance with the current state of knowledge (see, e.g., World Health Organization 2020). We consider a rectangular single-block warehouse layout (see Fig. 1) with a central depot and with parallel picking aisles that are connected by a cross aisle at the front and at the rear of the picking aisles. Items are stored in storage locations arranged along both sides of the picking aisles.

We assume that each picker is given a sequence, in which the picking orders are to be executed. A picking order comprises a single or multiple customer

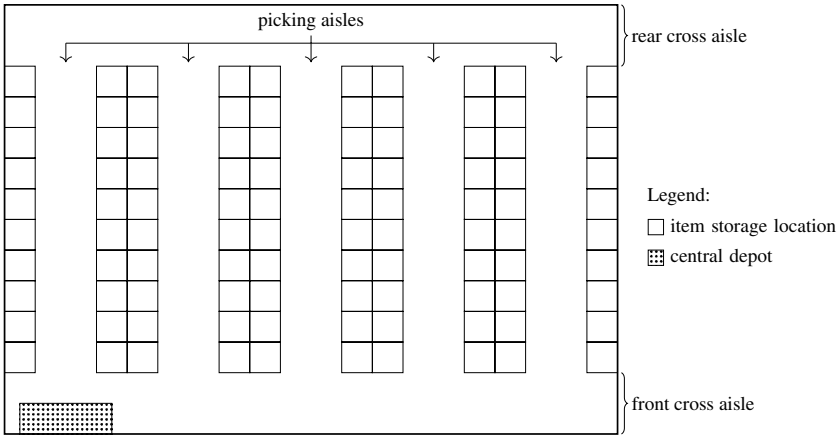


Fig. 1 An example of a rectangular single-block warehouse layout

orders. For each picking order, the picking tour along the storage locations is given. A common feature of picking tours is that they contain cycles which can be traversed in both directions, i.e., the decision maker can choose between the two possible directions at the entry to each of these cycles. Figure 2 gives an example of different execution possibilities (see Fig. 2(b)–(e)) of a picking tour (see Fig. 2(a)). The double circle marked by d represents the central depot, the other double circles indicate the front/rear locations of each picking aisle, and the standard circles denote the picking positions. The picking tour is represented by the solid lines. The front/rear end locations of picking aisles at which a degree of freedom occurs are marked red. Here, the picker can decide whether to enter the respective picking aisle (if so, we indicate this by a red vertical arc) or to proceed along the front/rear cross aisle (if so, we indicate this by a red horizontal arc). For each of the four depicted execution possibilities, the time periods in which the picker occupies a certain picking aisle differ. Consequently, to reduce virus transmission among pickers, we seek to execute the tours such that the time that pickers spend simultaneously in the same picking aisles is minimized. Note that the decision on how the picking tours are executed does not affect the distance traveled by the pickers.

The risk of infection in the cross aisles is not considered because cross aisles are generally significantly wider than the picking aisles, and thus, wide enough to allow pickers to pass each other while maintaining a social distance. Moreover, the risk of virus transmission via goods is assumed to be negligible for the following reason: when a picker moves to the storage location of a requested item, she[1] does not touch the surrounding items. The risk of infection via goods can be further decreased by regularly disinfecting hands.

[1] For the sake of readability, we have decided to speak exclusively of female pickers. Of course, a picker can also be of any other gender.

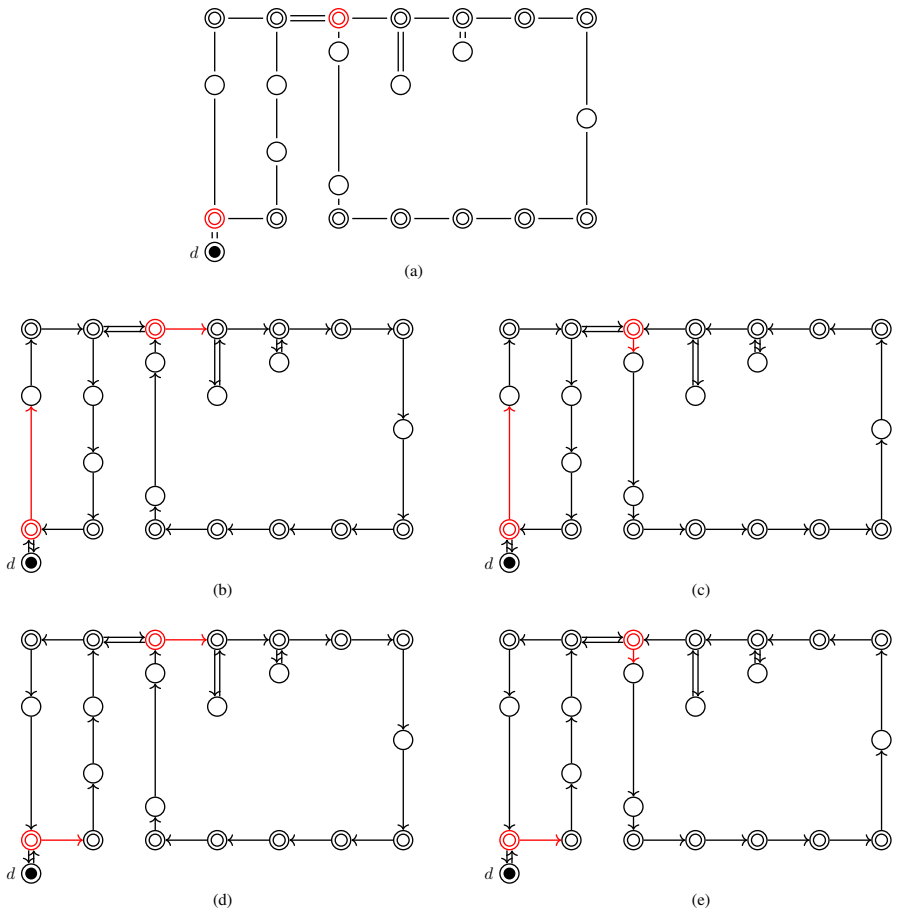


Fig. 2 Example of different execution possibilities (b)–(e) of the picking tour given in (a)

In Fig. 3, we show the impact of different executions of picking tours on the total time pickers spend together in the same picking aisles. To this end, we consider an instance with four picking tours, each associated with one of four pickers. In the upper part of Fig. 3(a), the picking tours are illustrated by the sequence in which the picking aisles are visited by the respective picker. The number in a rectangle indicates the picking aisle, and the length of a rectangle represents the time period the respective picker spends in this picking aisle. For example, picker 1 starts in picking aisle 1 (from $t = 0$ to $t = 6$) and from there proceeds to picking aisles 4 (from $t = 12$ to $t = 16$), 5 (from $t = 18$ to $t = 20$), 6 (from $t = 22$ to $t = 28$), and so on. In the lower part of Fig. 3(a), we report the time that the pickers simultaneously spend in the same picking aisles resulting from the given execution of the picking tours. We assume that in the case in which n pickers occupy a picking aisle at a point in time, $0.5 \cdot n \cdot (n-1)$ overlaps occur. Executing the picking tours given in Fig. 3(a) leads to a total temporal overlap of $Z = 102$. Modifying the execution of the picking tours

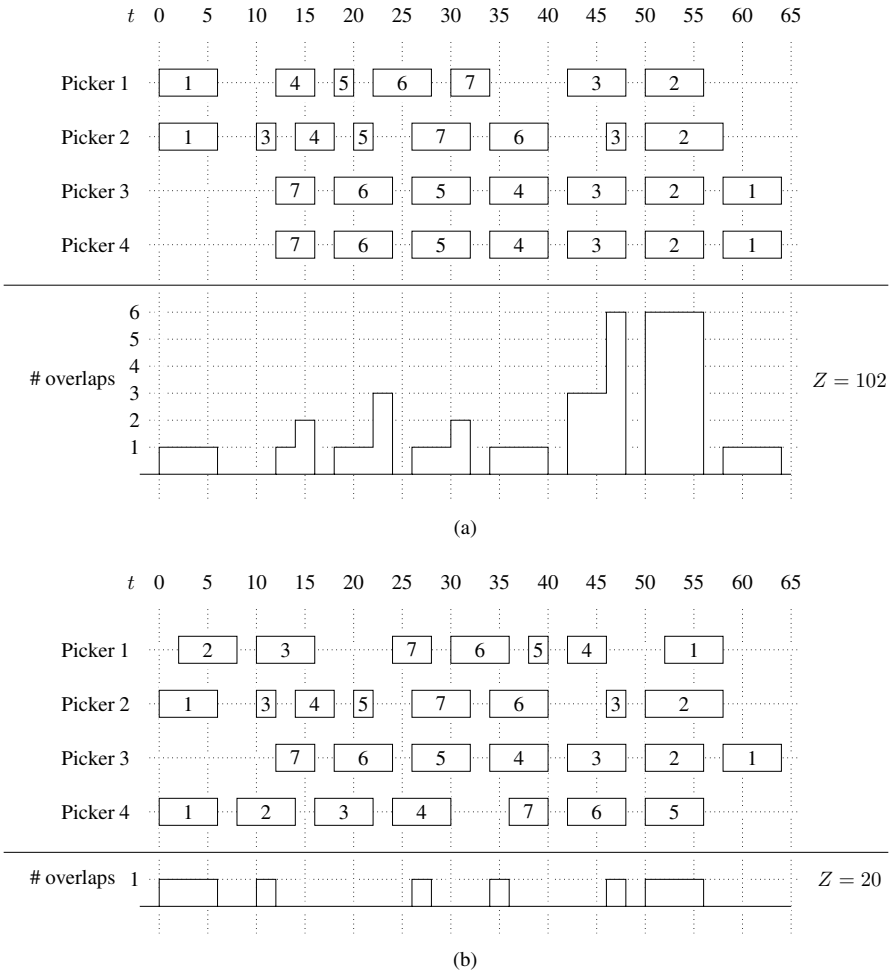


Fig. 3 Total temporal overlap between pickers (Z) for different picking tours

of pickers 1 and 4 as shown in the upper part of Fig. 3(b) leads to a reduction in the temporal overlap by approximately 80%. The example also shows that the length of the rectangles in Fig. 3(a) is the same as in Fig. 3(b), i.e., the distance traveled (or time required) by each picker is not affected.

To the best of our knowledge, the resulting picking tour execution problem (PTEP) constitutes a novel setting, which has not been studied in the literature so far but is of high relevance in real-world warehouses. The contributions of our paper are the following:

- We formally describe the PTEP as a mixed integer program and propose an efficient iterated local search heuristic (ILS) to provide solutions for large-sized problem instances.

- In extensive numerical studies, we test the performance of our PTEP model using the optimization software Gurobi. Our formulation is able to solve small and medium instances but is not able to consistently solve large instances within a given time limit. Moreover, we compare the performance of ILS to those of Gurobi solving PTEP. The results clearly demonstrate the ability of ILS to find high-quality solutions within reasonable runtimes.
- Additional experiments show that reductions between 17% and 100% of the total temporal overlap between pickers can be achieved compared to randomly executing the picking tours without increasing the distance traveled by pickers.
- To provide managerial insights, we compare our PTEP to a zone picking approach, in which the risk of infection spread is almost eliminated. The results show that the zone picking approach strongly increases the makespan compared to our approach.

The remainder of the paper is structured as follows: In Sect. 2, we discuss the related literature. In Sect. 3, we introduce the PTEP and present a mixed integer formulation. Our ILS is detailed in Sect. 4. Section 5 is devoted to the numerical studies. In Sect. 6, we compare the PTEP to a zone picking approach, followed by a conclusion in Sect. 7.

2 Literature review

Because traveling is considered the most time-consuming warehousing activity, research mainly focuses on reducing the average distance traveled (or time required) by pickers for collecting the items of a given set of picking orders. Travel distance depends on the design of the following four planning problems: warehouse layout (configuration of the warehouse, including the number of blocks and the number, length, and width of the picking aisles and cross aisles in each block), picker routing (determining the picking tour through the warehouse and the retrieval sequence of the items), order batching (grouping or splitting of customer orders), and storage assignment (assignment of items to storage locations).

The PTEP is closely related to picker routing problems (PRPs), which, in general, aim at determining a cost-minimal picking tour along the storage locations defined by a picking order (see, e.g., Ratliff and Rosenthal 1983). The most well-studied PRP in the literature is the standard single picker routing problem (standard SPRP), which can be defined as follows: Given a single-block parallel-aisle warehouse with a central depot and dedicated storage, i.e., each item is available from one storage location, the standard SPRP seeks the cost-minimal picking tour for collecting the items defined by a picking order. For an extensive review on PRPs, we refer the reader to Masae et al. (2019). Contrary to PRPs, in our PTEP, the picking tour is already given as input, and it can be generated by an arbitrary solution method for the PRP. Moreover, in the PTEP, picking tours are executed such that the temporal overlap between pickers is minimized, whereas previous research on PRPs has mainly concentrated on determining picking tours such that the distance traveled (or time required) by pickers is minimized.

The risk of infection spread in warehousing operations is only considered in Ardjmand et al. (2021). The authors study an order batching problem which aims at grouping customer orders into (larger) picking orders such that the total order picking time, the makespan, and the time in which pickers are picking closer than at a specified physical distance are minimized. To solve the resulting problem, they propose three multi-objective metaheuristics. Ardjmand et al. (2021) do not aim to reduce the risk of infection by optimizing the routing of pickers but assume that pickers are fixedly routed according to an S-shape routing policy.

Another closely related problem concerns the blocking of pickers. The studies that exist on picker blocking consider additional travel distance (or time) that occurs when, for example, pickers meet and passing one another is not possible due to narrow picking aisles. Gue et al. (2006) investigate the effect of pick density on picker blocking in warehousing systems with high space utilization, in which pickers move in an S-shape fashion along unidirectional picking aisles, and passing by other pickers is not allowed. To avoid blocking, the authors discuss different routing methods, e.g., forcing a blocked picker to leave the picking aisle and proceed along a picker-free picking aisle to continue order picking. Pan and Shih (2008) and Parikh and Meller (2009) investigate a multiple-picker setting with congestion considerations from a queuing theory perspective: Pan and Shih (2008) present a throughput model for a picker-to-parts warehouse involving multiple pickers and narrow picking aisles, which simultaneously considers the total travel time and the congestion effect. A picker is routed according to the S-shape routing policy, and in case that an aisle is already occupied by another picker, she has to wait in a buffer zone until the other picker leaves the picking aisle. Parikh and Meller (2009) develop analytical models to estimate picker idle time in a wide-aisle distribution center, in which picker blocking occurs if a picker blocks access to a picking position of another picker. Zhang et al. (2009) consider workflow congestion in the context of material handling equipment interruptions in a manufacturing or warehousing facility. The authors combine a probabilistic and physics-based model to evaluate the expected link travel time when interruptions occur. Two heuristic algorithms are developed to solve the combined optimization model. Hong et al. (2012) introduce an integrated batching and batch sequencing problem for a narrow-aisle order picking system that aims at minimizing the sum of travel time, pick time, and congestion delays. To address realistically sized instances, the authors present a simulated annealing heuristic. Chen et al. (2013, 2014) consider a warehousing system with narrow picking aisles, in which picker congestion has to be avoided. Chen et al. (2013) propose a routing algorithm based on ant colony optimization for two pickers and Chen et al. (2014) for multiple pickers. Schrotenboer et al. (2017) develop a genetic algorithm to evaluate the tradeoff between delays caused by so-called picker interactions and the time required to avoid an interaction. An interaction delay occurs, for example, when a picker blocks access to a storage location from which another picker needs to collect an item. To conclude, methods on picker blocking cannot be used to solve the PTEP because picker blocking problems assume that passing one another is not possible.

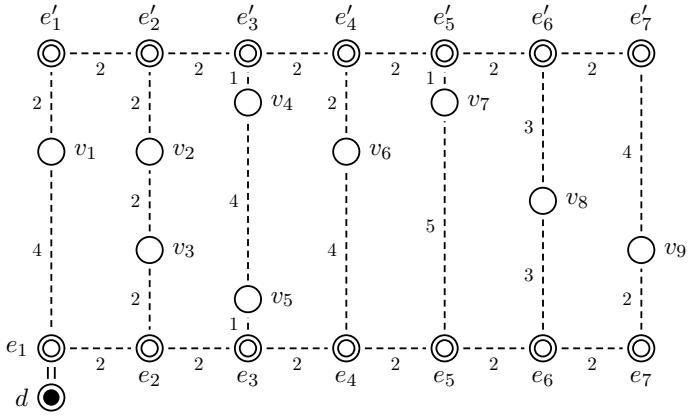


Fig. 4 An example of a warehouse graph with $|A| = 7$ picking aisles and $|I| = 9$ picking positions

3 The picking tour execution problem

In this section, we provide a detailed description of the PTEP (see Sect. 3.1) and present a mathematical formulation (see Sect. 3.2).

3.1 Problem description

The PTEP considers a rectangular single-block warehouse with parallel picking aisles (see Fig. 1). Let A denote the ordered set of picking aisles, which are numbered in ascending order from the leftmost to the rightmost picking aisle. We use a weighted graph $G = (V, E)$ to describe a picking order within a warehouse. Set V indicates the vertices including the front (rear) locations e_a (e'_a) of each picking aisle $a \in A$, the depot d, and the picking positions v_i of each requested item $i \in I$, where I denotes the set of requested items of the picking order. Set E consists of an unlimited number of parallel edges between every pair of adjacent vertices. The weight of each edge represents the travel distance (or time) between two adjacent vertices. An example of a warehouse graph with $|A| = 7$ picking aisles and $|I| = 9$ picking positions is given in Fig. 4. For simplicity, the figure represents the parallel edges connecting adjacent vertices by a single dashed edge.

The items of a given set of picking orders have to be collected. Let K denote the set of pickers, where each picker $k \in K$ is associated with an ordered set O^k of so-called tour subgraphs. A tour subgraph o is a subgraph of G from which a directed picking tour can be constructed such that every arc of the directed picking tour corresponds to exactly one edge in o. Note that the PTEP works independent of the underlying warehouse layout and routing policy because any routing policy used in any warehouse layout can be described such that the output is a tour subgraph, which is used as input to the PTEP and ILS. In Fig. 2(a), a tour subgraph returned by the optimal routing policy is depicted, and in Fig. 2(b)–(e), we show the four directed picking tours that can be constructed from the tour

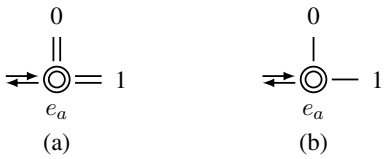


Fig. 5 Degrees of freedom resulting from Theorem 2

subgraph in Fig. 2(a). Theorem 1 states the formal properties of a tour subgraph and has been proved by Ratliff and Rosenthal (1983).

Theorem 1 (Properties of a tour subgraph) *A tour subgraph has the following properties*:

1. *Vertices v_i and vertex d have nonzero even degree in o.*
2. *Excluding vertices with zero degree, o is connected.*
3. *Every vertex in o has even or zero degree.*

The PTEP aims to construct directed picking tours from given tour subgraphs such that the total temporal overlap between pickers is minimized. Therefore, we describe the procedure for constructing directed picking tours (see Theorem 2, proved by Ratliff and Rosenthal (1983)), and we show that picking tours generally contain cycles which can be traversed in both directions.

Theorem 2 (Procedure for constructing directed picking tours) *Given a valid tour subgraph, the directed picking tour can be constructed as follows*:

1. *Start the tour at the depot vertex d.*
2. *If there is a pair of unused parallel arcs in the tour subgraph incident to the current vertex, use an arbitrary one of them to get to the next vertex. Continue with step* 2.
3. *If there are any unused single arcs in the tour subgraph (i.e., not one of a pair of parallel arcs), use an arbitrary one of them to get to the next vertex. Continue with step* 2.
4. *If there is a pair of parallel arcs in the tour subgraph with one arc used and one arc still unused, use the unused arc to get to the next vertex. Continue with step* 2.
5. *Stop. The directed picking tour is complete.*

In Theorem 2, a degree of freedom occurs in steps 2 and 3, respectively, i.e., each of these steps identifies a vertex associated with a degree of freedom. Figure 5 shows the two edge configurations for which degrees of freedom occur in a single-block parallel-aisle warehouse layout. For both configurations, there are two ways for selecting the next arc, and thus, at the entry e_a to the respective cycle, the decision maker can decide between the two possible directions, represented by 0 and 1. Note that these degrees of freedom only occur at the front or rear end of a picking aisle.

Table 1 Overview of the notation used in the PTEP model

Sets	
A	Ordered set of picking aisles (index: a)
K	Set of pickers (index: k)
N	Set of possible numbers of pickers that simultaneously occupy a picking aisle (index: n)
O^k	Ordered set of tour subgraphs associated with picker $k \in K$ (index: o)
T	Set of discrete points in time (index: t)
$T_{a\pi}$	Set of discrete points in time at which a picker is in picking aisle $a \in A$ if execution possibility $\pi \in \Pi_o^k$ is selected, where $T_{a\pi} \subseteq T$ (index: t)
Π_o^k	Set of all possibilities for executing tour subgraph $o \in O^k$ of picker $k \in K$ (index: π)
Binary decision variables	
x_{at}^n	1, if n pickers are in picking aisle $a \in A$ at time $t \in T$; 0, otherwise
σ_π	1, if execution possibility $\pi \in \Pi_o^k$ is selected; 0, otherwise

In Fig. 5(a), the picker reaches vertex e_a via the upper arc. At e_a, the picker can decide which of the edge pairs marked 0 or 1 she traverses first. Note that the picker traverses both edges of a parallel edge pair before she traverses the edges of the other edge pair. Such a configuration occurs in tour subgraphs in which the picking aisles are entered and left from the same cross aisle. In Fig. 5(b), the picker reaches vertex e_a via the upper arc. At e_a, both of the edges belong to the same cycle. Here, the picker returns to vertex e_a via the unselected edge, e.g., if edge 0 is selected first, she returns to vertex e_a via edge 1.

As described, the picker's decision influences the time periods in which she occupies a certain picking aisle without changing the distance traveled by the picker. We use this observation to minimize the time that pickers simultaneously work in the same picking aisles.

3.2 Mathematical model formulation

This section introduces the notation used in the PTEP model and presents a mathematical model formulation for the PTEP.

Set Π_o^k contains all possibilities for executing tour subgraph $o \in O^k$ by picker $k \in K$, where $\pi \in \Pi_o^k$. Binary variable σ_π denotes whether execution possibility $\pi \in \Pi_o^k$ is selected ($\sigma_\pi = 1$) or not ($\sigma_\pi = 0$). We assume discrete points in time $t \in T$. Because each execution possibility $\pi \in \Pi_o^k$ represents a picking tour, the points in time at which picker $k \in K$ is in picking aisle $a \in A$ executing tour subgraph $o \in O^k$ can be determined before solving the PTEP model and are represented by set $T_{a\pi}$, where $T_{a\pi} \subseteq T$.

Finally, binary variable x_{at}^n indicates whether $n \in N$ pickers are in picking aisle $a \in A$ at time $t \in T$ ($x_{at}^n = 1$) or not ($x_{at}^n = 0$), where N is the set of possible numbers of pickers that simultaneously occupy a picking aisle.

Using the notation summarized in Table 1, the PTEP can be formulated as the following mixed integer program.

$$\min Z = \sum_{a \in A} \sum_{t \in T} \sum_{n \in N : n \geq 2} \frac{n \cdot (n-1)}{2} \cdot x_{at}^{n} \tag{1}$$

subject to

$$\sum_{n \in N : n \geq 2} (n-1) \cdot x_{at}^{n} + 1 \geq \sum_{k \in K} \sum_{o \in O^k} \sum_{\pi \in \Pi_o^k : t \in T_{a\pi}} \sigma_{\pi} \quad \forall t \in T; a \in A \tag{2}$$

$$\sum_{n \in N : n \geq 2} x_{at}^{n} \leq 1 \quad \forall t \in T; a \in A \tag{3}$$

$$\sum_{\pi \in \Pi_o^k} \sigma_{\pi} = 1 \quad \forall k \in K; o \in O^k \tag{4}$$

$$x_{at}^{n} \in \{0,1\} \quad \forall n \in N : n \geq 2; t \in T; a \in A \tag{5}$$

$$\sigma_{\pi} \in \{0,1\} \quad \forall k \in K; o \in O^k; \pi \in \Pi_o^k \tag{6}$$

Objective function (1) minimizes the total temporal overlap between pickers. Constraints (2) ensure that binary variable x_{at}^{n} is correctly defined by linking x_{at}^{n} (left side) to the number of pickers at each point in time $t \in T$ and in each picking aisle $a \in A$ (right side). To determine the number of pickers, we sum over all selected decision variables σ_{π} for which a picker occupies picking aisle $a \in A$ at point in time $t \in T$. Constraints (3) ensure that the binary decision variable x_{at}^{n} is chosen at most once for each picking aisle $a \in A$ and point in time $t \in T$, i.e., only n pickers (or none) can simultaneously occupy a picking aisle. Constraints (4) guarantee that exactly one decision variable σ_{π} is selected for each tour subgraph $o \in O^k$ assigned to picker $k \in K$, i.e., exactly one directed picking tour is executed from the execution possibilities of the respective tour subgraph. Finally, the binary decision variables are defined in constraints (5) and (6), respectively.

4 An iterated local search for the picking tour execution problem

We define binary variable $\pi_j^k \in \{0,1\}$, where index $j = 1, \ldots, n_o^k$ denotes the j-th degree of freedom of tour subgraph $o \in O^k$ of picker $k \in K$. Index j is defined in ascending order based on its occurrence in the tour subgraph from left to right. Note that the j-th degree of freedom is associated with the entry at e_a (or e'_a) to a cycle at which picker $k \in K$ can decide whether to traverse picking aisle $a \in A$ that is adjacent to vertex e_a (or e'_a) or to proceed along the front (or rear) cross aisle. Binary variable π_j^k takes value 0 if picker $k \in K$ enters picking aisle $a \in A$ that is adjacent to vertex e_a (or e'_a) associated with the j-th degree of freedom, and 0 otherwise. Note that there are $|\Pi_o^k| = 2^{n_o^k}$ possibilities for executing tour subgraph $o \in O^k$ of picker $k \in K$.

generate initial solution s by randomly setting the binary variable π_j^k for each picker k and degree of freedom j
$s \leftarrow LocalSearch(s)$
$\kappa \leftarrow 1$
while $\kappa \leq \kappa_{\max}$ **do**
  $s' \leftarrow Perturbation(s, \kappa)$
  $s'' \leftarrow LocalSearch(s')$
  if $f(s'') < f(s)$ **then**
    $s \leftarrow s''$
    $\kappa \leftarrow 1$
  end if
  $\kappa \leftarrow \kappa + 1$
end while
return s

Fig. 6 Overview of the iterated local search algorithm

In Fig. 6, a pseudocode overview of the basic ILS framework is given, which can be described as follows: First, the initial solution is determined by randomly setting the binary variable π_j^k for each picker k and degree of freedom j, i.e., we randomly define whether the picker first traverses the picking aisle associated with j (i.e., $\pi_j^k = 0$) or proceeds along the respective cross aisle (i.e., $\pi_j^k = 1$). Second, the local search is applied to an initial solution s. The neighborhood s' used in the local search step is defined by all possible inversions of one of the binary decision variables π_j^k. Third, an incumbent solution s' is generated by perturbating s. The strength of the perturbation is defined dynamically as follows: At iteration κ, $\frac{\kappa}{\kappa_{\max}} \sum_{k \in K} \sum_{o \in O^k} |\Pi_o^k|$ randomly chosen decision variables are inverted, where $\kappa_{\max}$ denotes the number of iterations without improving the best found solution. Fourth, the local search is applied to s'. The generated solution s'' is accepted as the new current solution if it is better than the current solution s, i.e., if $f(s'') < f(s)$, where $f(\cdot)$ denotes the total temporal overlap between pickers. To achieve a good tradeoff between the solution quality and the runtime of our algorithm, the ILS terminates after $\kappa_{\max} = 10$. Higher numbers of iterations slightly improve the average solution quality but significantly increase the runtimes.

5 Numerical studies

This section presents the numerical studies (i) to assess the performance of our PTEP formulation using Gurobi, (ii) to compare the performance of ILS to that of Gurobi solving PTEP, and (iii) to give managerial insights with respect to the potential avoidance of temporal overlaps between pickers. In Sect. 5.1, we describe the test instances, and in Sect. 5.2, we present the results of our computational experiments.

5.1 Test instances

Our experiments are based on the instances of Henn and Wäscher (2012), originally designed for the order batching problem. The instances consider a single-block parallel-aisle warehouse with ten parallel picking aisles, each containing 90 different

items (45 on the left and 45 on the right). The depot is located below the entry of the leftmost picking aisle in the front cross aisle. The physical dimensions of the warehouse are as follows: An order picker has to cover 1 length unit (LU) (or 1 time unit) to get from the depot to the front cross aisle. The distance between the front cross aisle and the first storage location of a picking aisle is 1 LU, and the distance between two neighboring picking aisles is 5 LUs. The authors assume two different demand scenarios, i.e., uniformly distributed demand (UDD) and class-based demand (CBD). For UDD, items are randomly assigned to storage locations. For CBD, they define three classes with high (A), medium (B), and low (C) demand frequencies. Class A contains 10% of the items that account for 52% of the demand, class B 30% of the items that account for 36% of the demand, and class C 60% of the items that account for 12% of the demand. Items of class A are stored in picking aisle $a = 1$, items of class B in picking aisles $a = 2$ to $a = 4$, and items of class C in picking aisles $a = 5$ to $a = 10$. The authors consider groups of 40 instances, which are identified by the demand scenario and the number of picking orders (20, 40, 60, 80, and 100). The number of items per picking order is randomly drawn from the interval [2, 25].

In our experiments, we consider $|K| = \{2, 5, 10, 20, 40\}$ pickers and $|O^k| = \{2, 5, 10, 20\}$ tour subgraphs per picker $k \in K$. To use the instances of Henn and Wäscher (2012) for our experiments, we make the following adjustments: We select the first 2, 5, 10, or 20 picking orders from an instance of Henn and Wäscher (2012), who originally assume 20, 40, 60, 80, or 100 picking orders, to generate $|O^k| = \{2, 5, 10, 20\}$ tour subgraphs. Moreover, their instances consider a single picker and thus cannot be used directly for our experiments. However, given that 40 instances were generated per instance group, we select the first 2, 5, 10, 20, or 40 instances of a group, and we use, e.g., the two selected instances for the case of $|K| = 2$ pickers, one for each of the pickers, the 5 selected for $|K| = 5$ pickers, and so on.

As described before, any routing policy can be used to generate a tour subgraph. In the numerical experiments, tour subgraphs are generated by the exact algorithm of Ratliff and Rosenthal (1983), which is called "optimal" routing policy in the following, the largest gap routing policy by Hall (1993), the S-shape routing policy by Goetschalckx and Ratliff (1988), and a newly introduced S-shape+ routing policy. Figure 7 illustrates the tour subgraphs that result from the optimal (a), the largest gap (b), the S-shape (c), and the S-shape+ (d) routing policy for a given picking order. For each tour subgraph, we indicate the degrees of freedom by π_j^k. The routing policies are described as follows:

- *Largest gap:* According to the largest gap policy, the leftmost and the rightmost picking aisle are completely traversed if picks are required in there. The other picking aisles that contain a picking position are entered and left from the same cross aisle such that the largest gap is not traversed. A gap represents the distance between (i) any two adjacent picking positions, (ii) the first picking position and the front cross aisle, or (iii) the last picking position and the rear cross aisle. In the case of (i), a picking aisle is accessed via the front (or rear) cross aisle,

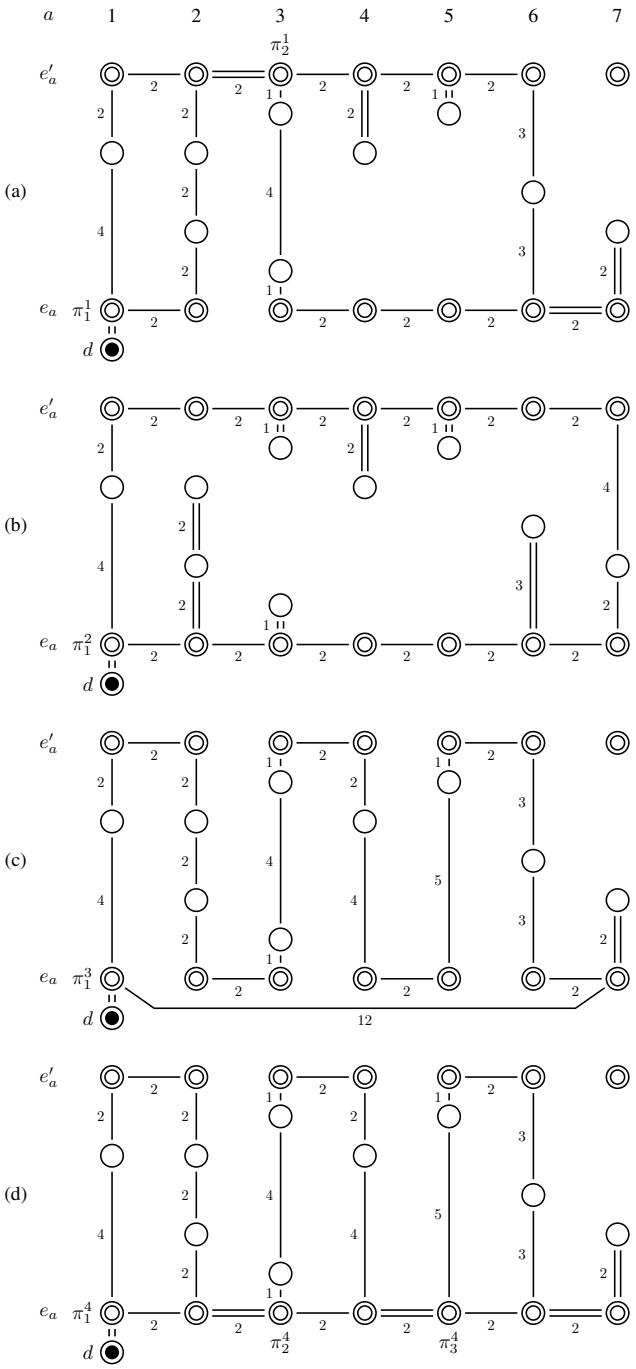


Fig. 7 Example of tour subgraphs based on (a) optimal, (b) largest gap, (c) S-shape, and (d) S-shape+ routing policies

whereas in the case of (ii) or (iii), a picking aisle is accessed from the rear (ii) or the front cross aisle (iii). The largest gap policy provides a single degree of freedom at the beginning of the tour.

- *S-shape:* In the S-shape routing policy, a picker enters picking aisles in an alternating manner from the front and the rear cross aisle and traverses them completely (if picks are required in there) or not at all (if no pick is required in there).
- *S-shape+:* As in the S-shape routing policy, in S-shape+, picking aisles are entered in an alternating manner from the front and the rear cross aisle and traversed completely (if picks are required). The difference between these two routing policies is the following: In the standard S-shape routing policy, the picker directly travels from the front location of the rightmost picking aisle to be visited to the depot without accessing the other front locations of the picking aisles (see Fig. 7(c)). Thus, all picks are executed on the way to the rightmost picking aisle that contains a picking position. In the S-Shape+ routing policy, the picker successively accesses each front location of the picking aisles on the way back to the depot. As shown in Fig. 7(d), this leads to a tour subgraph which allows the picker to skip required picking aisles on the way from the depot to the rightmost picking aisle to be visited and then to access the skipped picking aisles on the way back to the depot. Both routing policies lead to the same distance traveled by the picker but differ in the number of degrees of freedom they offer. The S-shape routing policy contains only one degree of freedom at the beginning of the tour, whereas the S-shape+ routing policy offers multiple degrees of freedom and therefore provides greater flexibility in designing the picker routes.

Combining the above described parameter values leads to 160 instance groups, which are identified by the demand scenario (UDD or CBD), the number of pickers $|K|$, the number of tour subgraphs per picker $|O^k|$, and the routing policy used to generate the respective tour subgraphs. For each instance group, we generate five instances, i.e., $160 \cdot 5 = 800$ instances in total.

All experiments are conducted on a computing cluster running CentOS 7 with $2 \times$ Intel Xeon E5-2430v2 Processors at 2.50 GHz and 64 GB of memory per compute node. Gurobi is used to solve the mixed integer program presented in Sect. 3.2, and each process runs multithreaded on 6 CPU cores. We restrict the solution time of Gurobi to 3600 s for all experiments. ILS is implemented in single-threaded C++ and compiled using GCC 8.2 with full optimizations enabled.

5.2 Computational results

In this section, we assess the performance of the PTEP model and of our ILS. Moreover, we provide managerial insights with respect to the potential avoidance of temporal overlaps between pickers in the warehousing system under study.

Tables 2 and 3 present aggregate results for different routing policies on the UDD instances and Tables 4 and 5 on the CBD instances. Each table reports averages for groups of instances defined by the number of pickers (column K) and the number of tour subgraphs per picker (column $|O^k|$). The remaining columns are

Table 2 Results on the UDD instances grouped by number of pickers $|K|$ and number of tour subgraphs per picker $|O^k|$ using the optimal and largest gap routing policy

		Optimal routing policy									Largest gap routing policy								
				Gurobi				ILS					Gurobi				ILS		
$\|K\|$	$\|O^k\|$	f^{rand}	td	ub (%)	lb (%)	#opt	t_a (s)	$best$ (%)	avg (%)	t_a (s)	f^{rand}	td	ub (%)	lb (%)	#opt	t_a (s)	$best$ (%)	avg (%)	t_a (s)
2	2	39	1344	–100	–100	5/5	0.13	–100	–100	0.01	55	1465	–89	–89	5/5	0.12	–89	–89	0.01
2	5	94	3391	–99	–99	5/5	0.22	–99	–97	0.02	104	3662	–84	–84	5/5	0.21	–84	–83	0.02
2	10	179	6842	–99	–99	5/5	0.41	–99	–98	0.05	192	7321	–86	–86	5/5	0.43	–85	–84	0.04
2	20	331	13576	–100	–100	5/5	0.69	–99	–98	0.14	440	14526	–88	–88	5/5	0.71	–84	–83	0.09
2	Avg.	161	6288	–100	–100	20/20	0.36	–99	–98	0.06	198	6791	–87	–87	20/20	0.37	–85	–85	0.04
5	2	414	3413	–73	–73	5/5	0.62	–69	–69	0.02	563	3686	–48	–48	5/5	0.28	–48	–45	0.02
5	5	978	8679	–74	–74	5/5	2.25	–69	–67	0.08	1144	9373	–50	–50	5/5	0.60	–49	–47	0.05
5	10	1897	17355	–77	–77	5/5	7.00	–76	–75	0.84	2220	18570	–52	–52	5/5	1.06	–52	–52	0.43
5	20	3746	34224	–75	–75	5/5	15.32	–72	–72	2.54	4421	36620	–52	–52	5/5	2.49	–52	–52	1.38
5	Avg.	1759	15918	–75	–75	20/20	6.29	–71	–71	0.87	2146	17191	–51	–51	20/20	1.11	–50	–49	0.47
10	2	1712	6708	–52	–52	5/5	22.15	–49	–48	0.06	2243	7245	–33	–33	5/5	1.56	–33	–32	0.04
10	5	4034	16890	–53	–53	5/5	881.99	–52	–52	1.21	4881	18072	–35	–35	5/5	5.28	–35	–35	0.64
10	10	7949	33773	–53	–58	0/5	3600.00	–51	–51	4.35	9300	36137	–34	–34	5/5	16.55	–34	–34	2.38
10	20	15891	67633	–52	–58	0/5	3600.00	–50	–50	18.79	19070	72367	–34	–34	5/5	44.69	–34	–34	11.38
10	Avg.	7397	31251	–52	–55	10/20	2026.73	–51	–50	6.10	9024	33439	–34	–34	20/20	17.02	–34	–34	3.61
20	2	8012	13322	–35	–35	5/5	884.96	–34	–33	0.17	9934	14255	–24	–24	5/5	27.47	–24	–23	0.13
20	5	17658	33488	–34	–39	0/5	3600.00	–34	–34	9.16	21190	35832	–23	–23	4/5	2941.38	–23	–23	5.84
20	10	34590	67297	–31	–41	0/5	3600.00	–34	–34	42.94	40816	72008	–23	–25	0/5	3600.00	–23	–23	27.38
20	20	67836	134620	–23	–41	0/5	3600.00	–33	–33	215.49	80724	144043	–23	–26	0/5	3600.00	–23	–23	134.20
20	Avg.	32024	62182	–31	–39	5/20	2922.73	–34	–33	66.94	38429	66535	–23	–25	9/20	2543.27	–23	–23	41.89
40	2	34279	26766	–27	–28	0/5	3600.00	–27	–27	13.86	42506	28640	–20	–20	5/5	258.63	–20	–20	8.38
40	5	74749	67302	–22	–26	0/5	3600.00	–23	–23	112.71	89699	72013	–17	–19	0/5	3600.00	–17	–17	73.37
40	10	143115	134534	–10	–26	0/5	3600.00	–22	–22	556.21	168875	143951	0	–18	0/5	3600.00	–16	–16	352.10
40	20	280723	269418	–3	–25	0/5	3600.00	–21	–21	2998.16	331253	288277	0	–100	0/5	3600.00	–15	–15	1953.74
40	Avg.	133216	124505	–16	–26	0/20	3600.00	–23	–23	920.24	159860	133220	–9	–39	5/20	2766.94	–17	–17	596.90

Table 3 Results on the UDD instances grouped by number of pickers $|K|$ and number of tour subgraphs per picker $|O^k|$ using the S-shape and S-shape+ routing policy

		S-shape routing policy									S-shape+ routing policy								
				Gurobi				ILS					Gurobi				ILS		
$\|K\|$	$\|O^k\|$	f^{rand}	td	ub (%)	lb (%)	#opt	t_a (s)	$best$ (%)	avg (%)	t_a (s)	f^{rand}	td	ub (%)	lb (%)	#opt	t_a (s)	$best$ (%)	avg (%)	t_a (s)
2	2	109	1801	–100	–100	5/5	0.14	–100	–98	0.00	94	1801	–100	–100	5/5	0.17	–100	–100	0.01
2	5	175	4476	–94	–94	5/5	0.24	–94	–91	0.01	159	4476	–100	–100	5/5	0.30	–100	–100	0.03
2	10	270	8758	–95	–95	5/5	0.43	–91	–89	0.02	270	8758	–100	–100	5/5	0.55	–100	–100	0.07
2	20	539	17513	–96	–96	5/5	0.83	–93	–91	0.06	552	17513	–100	–100	5/5	1.12	–100	–99	0.17
2	Avg.	315	8237	–96	–96	20/20	0.41	–94	–92	0.02	294	8237	–100	–100	20/20	0.54	–100	–100	0.07
5	2	948	4505	–61	–61	5/5	0.33	–61	–60	0.01	750	4505	–95	–95	5/5	1.12	–83	–78	0.03
5	5	1770	11196	–62	–62	5/5	0.73	–62	–59	0.03	1564	11196	–95	–95	5/5	4.66	–79	–77	0.09
5	10	3244	22214	–64	–64	5/5	1.63	–63	–63	0.31	2998	22214	–95	–95	5/5	29.50	–85	–85	0.84
5	20	5919	43807	–61	–61	5/5	2.68	–60	–60	1.03	5694	43807	–95	–95	5/5	583.85	–84	–83	3.21
5	Avg.	3254	20693	–62	–62	20/20	1.34	–62	–61	0.34	2867	20693	–95	–95	20/20	154.78	–83	–81	1.04
10	2	3528	8720	–40	–40	5/5	1.63	–39	–39	0.03	2963	8720	–69	–69	5/5	31.02	–60	–58	0.07
10	5	7059	21788	–40	–40	5/5	4.99	–40	–40	0.47	6535	21788	–67	–73	0/5	3600.00	–61	–61	1.21
10	10	12957	43229	–41	–41	5/5	12.31	–41	–40	1.98	12480	43229	–66	–77	0/5	3600.00	–59	–59	4.71
10	20	24791	86570	–41	–41	4/5	29.38	–40	–40	8.74	24473	86570	–61	–79	0/5	3600.00	–58	–58	20.98
10	Avg.	12944	40314	–40	–40	19/20	12.08	–40	–40	2.80	11983	40314	–66	–74	5/20	2708.87	–60	–59	6.74
20	2	15542	17319	–30	–30	5/5	11.35	–29	–29	0.09	12979	17319	–46	–47	4/5	2299.72	–42	–41	0.18
20	5	30548	43200	–28	–28	5/5	284.85	–27	–27	4.67	28076	43200	–41	–47	0/5	3600.00	–40	–39	10.48
20	10	56382	86813	–27	–28	0/5	3600.00	–26	–26	26.98	53614	86813	–29	–47	0/5	3600.00	–38	–38	44.63
20	20	106502	172314	–26	–28	0/5	3600.00	–25	–25	108.33	104467	172314	–27	–47	0/5	3600.00	–36	–36	220.56
20	Avg.	55081	80215	–27	–28	10/20	1875.41	–27	–27	35.02	50598	80215	–36	–47	4/20	3276.73	–39	–39	68.97
40	2	66158	34528	–22	–22	5/5	56.06	–22	–22	6.19	54846	34528	–32	–33	0/5	3600.00	–31	–31	12.33
40	5	129316	86820	–20	–20	0/5	3600.00	–19	–19	55.25	117356	86820	–22	–29	0/5	3600.00	–26	–26	108.78
40	10	233277	173549	–3	–19	0/5	3600.00	–18	–18	284.65	221828	173549	–19	–28	0/5	3600.00	–24	–24	534.06
40	20	440734	344855	1	–100	0/5	3600.00	–16	–16	1448.24	432313	344855	–17	–27	0/5	3600.00	–22	–22	2860.94
40	Avg.	227800	160611	–11	–40	5/20	2716.91	–19	–19	448.58	209150	160611	–22	–29	0/20	3600.00	–26	–26	879.03

divided into two blocks, where each block reports the results for one of the underlying routing policies.

In column f^{rand}, we give the average temporal overlap of 1000 randomly generated solutions, i.e., we randomly construct directed picking tours from the respective tour subgraphs. In addition, we report the average of the total travel distance of pickers (column *td*). All other values that are reported are provided as percentage deviation from the objective function value reported in column f^{rand}. Column *ub* (%) (*lb* (%)) denotes the percentage deviation of the best objective function (lower bound) values

Table 4 Results on the CBD instances grouped by number of pickers $|K|$ and number of tour subgraphs per picker $|O^k|$ using the optimal and largest gap routing policy

		Optimal routing policy									Largest gap routing policy								
				Gurobi				ILS					Gurobi				ILS		
$\|K\|$	$\|O^k\|$	f^{rand}	td	ub (%)	lb (%)	#opt	t_a (s)	$best$ (%)	avg (%)	t_a (s)	f^{rand}	td	ub (%)	lb (%)	#opt	t_a (s)	$best$ (%)	avg (%)	t_a (s)
2	2	43	1000	–92	–92	5/5	0.08	–92	–90	0.01	43	1050	–97	–97	5/5	0.08	–95	–92	0.01
2	5	86	2632	–96	–96	5/5	0.15	–96	–93	0.01	100	2764	–96	–96	5/5	0.16	–96	–95	0.01
2	10	178	5282	–97	–97	5/5	0.26	–96	–94	0.03	199	5546	–95	–95	5/5	0.28	–91	–91	0.03
2	20	348	10650	–97	–97	5/5	0.52	–96	–95	0.08	383	11183	–90	–90	5/5	0.54	–84	–83	0.07
2	Avg.	164	4891	–95	–95	20/20	0.25	–95	–93	0.03	179	5136	–94	–94	20/20	0.27	–92	–90	0.03
5	2	427	2562	–56	–56	5/5	0.18	–54	–53	0.01	474	2690	–50	–50	5/5	0.19	–49	–47	0.01
5	5	950	6580	–61	–61	5/5	0.52	–58	–56	0.04	1074	6843	–54	–54	5/5	0.44	–53	–52	0.04
5	10	1869	13312	–64	–64	5/5	1.08	–64	–63	0.36	2131	13978	–53	–53	5/5	1.04	–53	–53	0.30
5	20	3657	26767	–63	–63	5/5	2.47	–61	–61	1.26	3950	28105	–54	–54	5/5	1.98	–53	–53	1.01
5	Avg.	1726	12305	–61	–61	20/20	1.06	–59	–58	0.42	1916	12920	–53	–53	20/20	0.91	–52	–51	0.34
10	2	2027	5145	–41	–41	5/5	1.08	–40	–39	0.03	2270	5402	–37	–37	5/5	0.87	–36	–36	0.03
10	5	4264	13091	–41	–41	5/5	5.39	–40	–40	0.55	4691	13746	–35	–35	5/5	3.91	–35	–35	0.45
10	10	8394	26445	–41	–41	5/5	47.24	–41	–41	2.12	8982	27767	–35	–35	5/5	11.26	–35	–35	1.77
10	20	16329	53164	–40	–40	5/5	193.34	–40	–39	8.71	17472	55822	–35	–35	4/5	45.27	–34	–34	7.64
10	Avg.	7754	24461	–41	–41	20/20	61.76	–40	–40	2.85	8451	25684	–36	–36	19/20	15.33	–35	–35	2.47
20	2	8392	10403	–32	–32	5/5	15.36	–31	–31	0.10	9231	10923	–29	–29	5/5	8.03	–29	–29	0.10
20	5	17904	26131	–29	–29	4/5	1548.49	–29	–29	4.74	19516	27438	–26	–26	5/5	133.13	–26	–26	4.10
20	10	34662	52963	–27	–29	0/5	3600.00	–27	–27	21.17	37435	55611	–24	–24	0/5	3600.00	–24	–24	17.03
20	20	67718	105943	–26	–28	0/5	3600.00	–25	–25	101.52	73135	111240	–22	–24	0/5	3600.00	–22	–22	86.60
20	Avg.	32169	48860	–28	–29	9/20	2191.92	–28	–28	31.88	34743	51303	–25	–26	10/20	1836.29	–25	–25	26.95
40	2	34431	20992	–27	–27	5/5	98.35	–27	–27	6.92	37874	22042	–24	–24	5/5	53.11	–24	–24	5.58
40	5	73471	52414	–21	–22	0/5	3600.00	–21	–21	63.18	80083	55035	–19	–19	0/5	3600.00	–19	–19	46.49
40	10	141015	105497	–17	–19	0/5	3600.00	–18	–18	271.33	153707	110772	–16	–17	0/5	3600.00	–16	–16	223.08
40	20	276106	211074	0	–18	0/5	3600.00	–16	–16	1434.55	300955	221628	0	–16	0/5	3600.00	–15	–15	1172.02
40	Avg.	131256	97494	–16	–22	5/20	2729.82	–21	–21	443.99	143069	102369	–15	–19	5/20	2716.22	–19	–19	361.79

Table 5 Results on the CBD instances grouped by number of pickers $|K|$ and number of tour subgraphs per picker $|O^k|$ using the S-shape and S-shape+ routing policy

		S-shape routing policy									S-shape+ routing policy								
				Gurobi				ILS					Gurobi				ILS		
$\|K\|$	$\|O^k\|$	f^{rand}	td	ub (%)	lb (%)	#opt	t_a (s)	$best$ (%)	avg (%)	t_a (s)	f^{rand}	td	ub (%)	lb (%)	#opt	t_a (s)	$best$ (%)	avg (%)	t_a (s)
2	2	56	1270	–90	–90	5/5	0.10	–90	–88	0.00	55	1270	–100	–100	5/5	0.10	–100	–100	0.01
2	5	135	3290	–82	–82	5/5	0.20	–81	–78	0.01	125	3290	–100	–100	5/5	0.20	–99	–98	0.02
2	10	238	6603	–88	–88	5/5	0.34	–83	–80	0.02	236	6603	–100	–100	5/5	0.36	–100	–99	0.04
2	20	446	13313	–92	–92	5/5	0.64	–86	–85	0.05	463	13313	–100	–100	5/5	0.70	–100	–99	0.11
2	Avg.	225	6114	–88	–88	20/20	0.32	–85	–83	0.02	220	6114	–100	–100	20/20	0.34	–100	–99	0.04
5	2	666	3177	–57	–57	5/5	0.21	–53	–53	0.01	572	3177	–81	–81	5/5	0.45	–74	–72	0.02
5	5	1330	8225	–56	–56	5/5	0.49	–53	–51	0.02	1254	8225	–84	–84	5/5	1.61	–72	–70	0.06
5	10	2542	16640	–57	–57	5/5	1.03	–56	–56	0.24	2467	16640	–83	–83	5/5	5.29	–77	–76	0.55
5	20	4901	33459	–57	–57	5/5	2.31	–57	–56	0.84	4901	33459	–84	–84	5/5	29.90	–76	–76	1.78
5	Avg.	2451	15381	–57	–57	20/20	1.01	–55	–54	0.28	2295	15381	–83	–83	20/20	9.31	–75	–74	0.60
10	2	3041	6483	–37	–37	5/5	0.95	–35	–34	0.02	2696	6483	–56	–56	5/5	4.93	–50	–50	0.04
10	5	6013	16364	–37	–37	5/5	3.70	–37	–37	0.37	5714	16364	–56	–57	4/5	979.18	–54	–53	0.72
10	10	11332	33056	–38	–38	5/5	9.94	–37	–37	1.34	11164	33056	–55	–58	0/5	3600.00	–52	–52	2.60
10	20	21880	66455	–38	–38	5/5	31.12	–37	–37	6.53	21880	66455	–55	–61	0/5	3600.00	–50	–50	11.72
10	Avg.	10855	30576	–37	–37	20/20	11.43	–36	–36	2.07	10390	30576	–56	–58	9/20	2046.73	–52	–51	3.77
20	2	12504	13004	–29	–29	4/5	7.87	–28	–28	0.08	11245	13004	–41	–41	5/5	55.66	–39	–38	0.13
20	5	25424	32664	–26	–26	4/5	128.05	–26	–26	3.51	24171	32664	–37	–39	0/5	3600.00	–36	–35	5.38
20	10	47487	66204	–25	–25	0/5	3600.00	–25	–25	14.94	46794	66204	–34	–39	0/5	3600.00	–33	–33	24.86
20	20	92774	132429	–24	–25	0/5	3600.00	–24	–24	73.76	92097	132429	–20	–38	0/5	3600.00	–32	–32	124.67
20	Avg.	45359	61075	–26	–27	8/20	1835.17	–26	–26	23.07	43428	61075	–33	–39	5/20	2715.27	–35	–35	38.76
40	2	51991	26240	–22	–22	3/5	31.37	–22	–22	4.65	46827	26240	–31	–31	3/5	960.09	–30	–30	7.35
40	5	105063	65518	–18	–19	1/5	3249.68	–18	–18	39.39	99920	65518	–24	–26	0/5	3600.00	–24	–24	64.14
40	10	196011	131871	–16	–17	0/5	3600.00	–16	–16	188.46	191781	131871	–18	–23	0/5	3600.00	–21	–21	300.67
40	20	378265	263843	–1	–100	0/5	3600.00	–15	–15	989.20	372743	263843	–13	–22	0/5	3600.00	–19	–19	1568.09
40	Avg.	186383	121868	–14	–40	4/20	2621.95	–18	–18	305.43	178508	121868	–21	–25	3/20	2942.55	–23	–23	485.06

given by Gurobi. Column *#opt* indicates the number of instances solved to proven optimality. For ILS, the reported results are based on five runs on each instance. In column *best* (%) (*avg* (%)), we show the percentage deviation of the best (average) solution value over the five runs. Column t_a (s) gives the average runtimes in seconds.

In the following, we discuss the results of our experiments:

Comparison of ILS to Gurobi The results show that the number of pickers, the number of tour subgraphs per picker, and the routing policy significantly influence the performance of Gurobi. The main findings are as follows:

- *Optimal:* Using the optimal routing policy, Gurobi is able to solve 55 UDD instances and 74 CBD instances to optimality (UDD instances with $|K| = \{2, 5\}$ and those with $|K| = \{10\}$ assuming $|O^k| = \{2, 5\}$ as well as all CBD instances with $|K| = \{2, 5, 10\}$). On the UDD and CBD instances with $|K| = 20$, all instances with $|O^k| = 2$ are solved, and on the CBD instances with $|O^k| = 5$, four out of five instances are solved. On the instances with $|K| = 40$, Gurobi is not able to solve any of the UDD and CBD instances within the given runtime limit.
- *Largest gap and S-shape:* Most of the instances can be solved to optimality using the largest gap policy (148 out of 200 instances), i.e., all UDD and CBD instances with $|K| = \{2, 5, 10\}$, except a single CBD instance with $|K| = 10$ and $|O^k| = 20$. On the CBD instances with $|K| = 20$ and $|O^k| = \{2, 5\}$, Gurobi provides optimal solutions for all tested instances and on the UDD instances, for all except for a single instance. Even on the larger UDD and CBD instances with $|K| = 40$ and $|O^k| = 2$, Gurobi finds optimal solutions for all tested instances. Similar results can be observed for the case of the S-shape routing policy (146 out of 200 instances are solved to optimality).
- *S-shape+:* The fewest of the instances can be solved using the S-shape+ routing policy (106 out of 200 instances), i.e., all UDD and CBD instances with $|K| = \{2, 5\}$ and $|O^k| = \{2, 5, 10, 20\}$. On the larger UDD instances, Gurobi finds optimal solutions for all instances with $|K| = 10$ and $|O^k| = 2$, and for 4 out of 5 instances with $|K| = 20$ and $|O^k| = 2$. In the case of larger CBD instances, Gurobi provides optimal solutions for all instances with $|K| = \{10, 20\}$ and $|O^k| = 2$, and for 4 out of 5 instances with $|K| = 10$ and $|O^k| = 5$, and 3 out of 5 instances with $|K| = 40$ and $|O^k| = 2$.

The results show that Gurobi succeeds in solving small and medium instances within reasonable runtimes but is not able to consistently solve larger instances to optimality within the given time limit. The tables show that ILS finds the same or slightly worse solutions as Gurobi on the smaller and medium instances but clearly outperforms Gurobi on the larger instances: It is worth noting that the solution quality of ILS on the largest instances is close to the lower bound of Gurobi (in the cases in which the lower bound is not equal to zero). In addition, the results indicate a very good robustness of our ILS, i.e., the best solutions found by ILS deviate from the average solutions by only 0.5% for the UDD case and 0.6% for the CBD case. The runtimes of ILS are reasonable and stay below 10 minutes for all but the largest instances with $|K| = 40$ and $|O^k| = 20$. The results of ILS clearly demonstrate that ILS is able to solve the largest instances within reasonable runtimes.

Performance of ILS for different routing policies In the following, we investigate the performance of ILS for different routing policies. To this end, we compare the results of the arbitrary solution (see column "f^{rand}") with those obtained by our ILS (see column "*best*") for the same picker routing policy. We use the reduction in temporal overlap between pickers to compare the results of an arbitrary solution to the PTEP with that obtained by our ILS for the same picker routing policy. This can be an insightful measure for managers if a fixed routing policy is used in their warehouse.

The results show that ILS is able to significantly reduce the temporal overlap between pickers independent of the picker routing policy at hand. For example, on the smallest UDD and CBD instances with $|K| = 2$ and S-shape+ routing policy, ILS generates solutions that are completely free of temporal overlaps. Even on the largest UDD (CBD) instances with $|K| = \{20, 40\}$, a reduction of 39% (35%) and 26% (23%) is achieved although the instances assume a warehouse with only 10 picking aisles. ILS achieves the smallest reduction in the temporal overlap between pickers in the case of S-shape and largest gap routing policies. For instance, even on the smallest UDD instances with $|K| = 2$, temporal overlaps between pickers cannot be completely avoided. This is because both routing policies provide only one degree of freedom in the routing: the picker performs the S-shape routing either via the leftmost or via the rightmost picking aisle to be visited, and the largest gap routing starts either along the front or the rear cross aisle.

In the following, we investigate the effect of (i) the number of pickers, (ii) the number of tour subgraphs per picker, and (iii) the demand scenario on the total temporal overlap between pickers. Moreover, we briefly discuss the results with respect to the total travel distance of the pickers.

- For all routing policies, we observe that the larger the number of pickers $|K|$, the smaller the average reduction in the total temporal overlap between pickers. Obviously, in the case of a large number of pickers, there is less potential for reduction in comparison with a small number of pickers because the probability of temporal overlaps increases with the number of pickers. For instance, on the UDD instances using the S-shape+ routing policy, the average reduction ranges between 26% in the case of $|K| = 40$ and 100% for $|K| = 2$.
- The influence of the number of tour subgraphs per picker is hardly visible. For example, on the UDD instances with $|K| = 20$ using largest gap routing policy, the reductions achieved for $|O^k| = \{2, 5, 10, 20\}$ are 24%, 23%, 23%, and 23%, respectively.
- For CBD, ILS generates solutions that are slightly worse compared to those for UDD. For example, for CBD, $|K| = 40$, and the S-shape+ routing policy, ILS yields a reduction of 23% on average compared to 26% for UDD.
- As expected, the optimal routing leads to the shortest picking tours, and the S-shape and S-shape+ routing policies result in the longest routes. This is because all picking aisles containing even a single item to be picked must be completely traversed.

Our numerical results show that large reductions in the total temporal overlap between pickers are achieved independent of the underlying routing policy. The smallest temporal overlap is observed for the optimal routing policy (see Tables 7, 8, 9, and 10). With respect to picking performance, the optimal routing policy is obviously superior to the other routing policies because it leads to shorter picking tours. However, a shortcoming of using the optimal routing policy is that pickers may get confused by the complex routes, and therefore, tend to deviate from optimal routing patterns (see, e.g., Elbert et al. 2017). Deviating from given routes can have a negative impact on picking efficiency and also increase the risk of

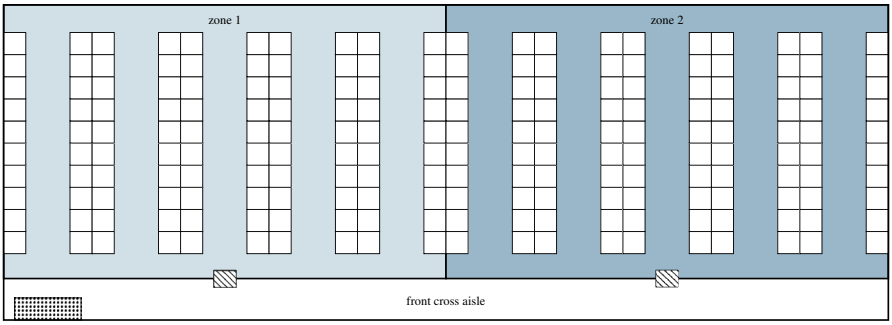
infection spread. This effect is particularly strong if no modern technologies such as tablets, pick-by-light, or pick-by-voice are used to assist pickers.

To reduce infection risk in picker-to-parts warehousing systems through our PTEP, it is paramount that pickers follow prescribed routes. While pickers will not always follow prescribed routes in practice in non-pandemic periods for several reasons (e.g., social interactions), in pandemic periods, pickers are more likely to do so to avoid infection. Moreover, although our PTEP is not able to completely prevent possibilities for social interaction, it significantly reduces them through exploiting the degrees of freedom that the routing policies offer.

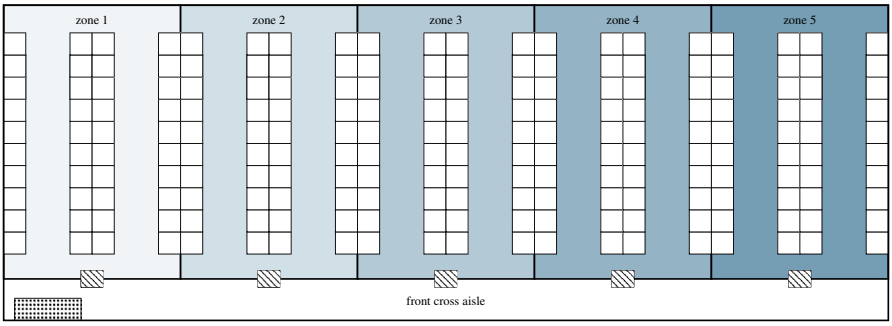
6 Zone picking

In this section, we compare our approach to a zone picking approach, which is another possible way to reduce infection risk in picker-to-parts warehouses. To make a fair comparison possible, we choose the following setup for the zone picking approach:

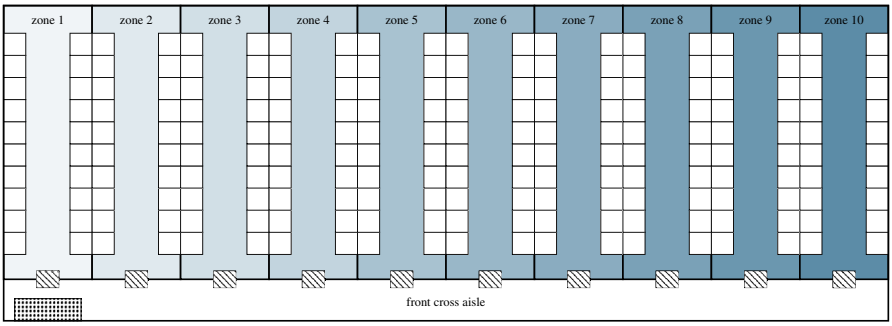
- *Warehouse layout:* The picking area follows the single-block parallel-aisle warehouse layout described in Sect. 5.1, but is divided into smaller zones, each with a single picker assigned to it. A zone comprises a fixed number of picking aisles, and each zone is associated with a handover location located in the front cross aisle. Figure 8 illustrates the zone picking system with different numbers of zones. Obviously, physical distancing practices are easy to implement, and infection risk between pickers can be almost eliminated in such systems.
- *Parallel zone picking:* We assume parallel zoning, i.e., the items of a picking order are simultaneously retrieved by multiple zone pickers in multiple zones as described in the following: Each zone picker is initially positioned at her handover location. Starting from the handover location, the zone pickers collect the picking order items from their zone according to an optimal routing pattern (see Ratliff and Rosenthal 1983). Once a zone picker has collected all picking order items from her zone, she returns to her handover location to deposit the items. Subsequently, she executes the next picking order from her picking list, starting from her handover location. We assume that each zone picker uses a bin with sufficient capacity for temporarily storing the picking order items and that each handover location has infinite storage capacity (buffer) for depositing items.
- *Sequence for picking orders:* The sequence according to which the picking orders have to be collected by the zone pickers is given.
- *Consolidator:* A consolidator collects the items of a single picking order from the relevant handover locations (i.e., those from which the picking order items are to be collected) on a single tour as follows: The consolidator starts at the depot, where she is equipped with a trolley with sufficient capacity for the items of a single picking order, proceeds along the front cross aisle to the relevant handover locations, and from the last visited handover location to the depot. The sequence according to which a consolidator visits the relevant handover loca-



(a) Picking area divided into 2 zones each with 5 picking aisles.



(b) Picking area divided into 5 zones each with 2 picking aisles.



(c) Picking area divided into 10 zones each with a single picking aisle.

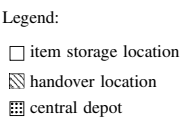


Fig. 8 Warehouse layout assumed in the zone picking system

tions is determined in optimal fashion (concerning the objective of minimizing the completion time of a picking order). We assume that no effort for consolidating the items occurs at the depot. The consolidator collects the picking orders in the given picking sequence.

- *Test instances:* Our experiment is based on the instances described in Sect. 5.1. For both approaches, we use the exact algorithm of Ratliff and Rosenthal (1983) to determine a picking tour. For our PTEP, we consider $|K| = \{3, 6, 11\}$ pickers and $|O^k| = \{2, 5, 10, 20\}$ tour subgraphs per picker $k \in K$. For the zoning approach, the picking area is divided into 2, 5, and 10 zones comprising 5, 2, and 1 picking aisles, respectively. We assume $|K| = \{3, 6, 11\}$ pickers, where the 3rd, 6th, and 11th picker represents the consolidator. The number of picking orders in the zone picking system is defined as $|O| = |O^k| \cdot |K|$. Combing the above described parameter values leads to 12 instance groups, which are identified by the number of pickers $|K|$ and the number of picking orders in the system $|O|$. For each instance group, we generate 5 instances, i.e., $12 \cdot 5 = 60$ instances in total.

In the following, we use the makespan, i.e., the time when all picking orders are completed, as performance measure for comparing the PTEP and the zone picking approach. In Table 6, we present aggregate results obtained by our ILS for the PTEP and by the zone picking approach on the UDD instances and the CBD instances. The table reports averages for groups of instances defined by the number of pickers (column $|K|$) and the number of picking orders in the system (column $|O|$). The remaining columns are divided into two blocks, where the first block represents the results on the UDD instances and the second block those on the CBD instances. In column *ATO*, we give the average temporal overlap. Column M_{PTEP} (M_{ZONE}) reports the average makespan required for the PTEP (zone picking approach). Column Δ (%) denotes the average percentage deviation between M_{ZONE} and M_{PTEP}.

The results show that our PTEP approach strongly outperforms the zone picking approach with respect to makespan. For example, on the smallest instances with $|K| = 3$ pickers and $|O| = 6$ picking orders in the system, the average percentage deviation between the PTEP and the zone picking approach is approximately 40% in the UDD case (49% in the CBD case). Moreover, on these instances, our ILS finds solutions that are almost free of temporal overlaps between pickers. The larger the number of pickers $|K|$, the greater the deviation in makespan between the PTEP and the zone picking approach. For example, on the largest CBD instances with $|K| = 11$ pickers and $|O| = 220$ picking orders in the system, the makespan obtained by the zone picking approach is on average 1066% higher than that obtained by our ILS for the PTEP.

To sum up, the zone picking approach presented above significantly increases the makespan although many favorable assumptions are made (e.g., no effort for consolidating the collected items at the depot, infinite storage capacity for depositing items at the handover locations). Clearly, the zone picking approach is able to prevent pickers from operating in close proximity to each other, and thus, to almost eliminate the infection risk between pickers. Consequently, the tradeoff between basically no temporal overlap but excessive makespan (zone picking approach) and some temporal overlap without raising costs (PTEP) must be evaluated by warehouse managers. Given medical tests and personal protective measures (e.g., vaccinations and/or mouth-nose protection masks), we believe that our PTEP is an interesting approach for warehouse managers to reduce infection risk between pickers without compromising order picking performance.

Table 6 Results on the UDD and CBD instances

		UDD				CBD			
$\|K\|$	$\|O\|$	ATO	M_{PTEP}	M_{ZONE}	Δ (%)	ATO	M_{PTEP}	M_{ZONE}	Δ (%)
3	6	16	770	1081	40.22	43	579	856	48.49
3	15	54	1873	2723	45.28	80	1416	2115	49.19
3	30	103	3647	5247	43.91	134	2821	4206	49.11
3	60	108	7109	10146	42.84	200	5533	8315	50.34
6	12	213	800	2265	184.45	284	591	2235	282.08
6	30	516	1942	5561	186.75	654	1462	5515	278.80
6	60	878	3670	11089	202.33	1182	2837	10951	286.41
6	120	1833	7302	22141	203.23	2502	5698	21871	283.94
11	22	1105	857	6910	706.72	1483	639	6768	966.51
11	55	2439	1988	17208	767.53	3158	1549	16974	998.10
11	110	5000	3708	34366	827.12	6247	2946	33946	1054.39
11	220	10231	7331	68708	837.22	12470	5836	67994	1066.03

7 Conclusion

This paper aims to reduce the risk of infection in a picker-to-parts warehousing system in which multiple pickers operate in the same picking area. To this end, we introduce the PTEP that generates directed picking tours from given tour subgraphs such that the time period in which pickers simultaneously occupy the same picking aisles is minimized. We formally describe the PTEP as a mixed integer program and provide an efficient ILS.

Our main finding is that significant reductions in the total temporal overlap between pickers can be achieved by exploiting the degrees of freedom that routing policies offer. On average, a reduction in approximately 52% is achieved in the UDD scenario, while the reduction in the CBD scenario is slightly lower with 49%. The smallest temporal overlap between pickers is achieved in the case of the optimal routing policy. The S-shape routing policy tends to perform worst.

We believe that our results are quite interesting for warehouse managers in times of pandemics: Avoiding temporal overlaps between pickers does not only reduce the infection risk, but also increases the order picking efficiency (e.g., by reducing in-aisle congestion). Moreover, the reductions are achieved without compromising order picking performance, i.e., without changing the distance traveled (or time required) by the pickers. The comparison of our approach to a zone picking approach reveals that the zone picking approach results in excessive makespan: the deviation between the PTEP and the zone picking approach amounts to 396% on average.

To further reduce temporal overlaps between pickers, our problem could be extended to incorporate modern warehousing concepts, such as scattered storage or multiple end depots. For example, with scattered storage, an item type can be available from several storage locations (see Goeke and Schneider 2018). Assigning the stock keeping units of a frequently requested item type to multiple storage locations

could reduce the probability of pickers working close to each other. The PTEP could be extended to incorporate congestion in picking aisles and picker blocking, which is particularly relevant for warehouses with narrow picking aisles. Moreover, future research could investigate alternative objective functions for reducing the infection risk, e.g., one could minimize infection risk that results from pickers operating within a given physical distance for more than a critical period of time.

Pandemics or epidemics will certainly be given more focus in future research. Incorporating infection risks into well-known order picking problems will pose new challenges in the research on warehousing to ensure a safe order picking environment during pandemics.

Appendix: Additional results

Tables 7 and 8 present aggregate results for different routing policies on the UDD instances and Tables 9 and 10 on the CBD instances. Each table reports averages for

Table 7 Results on the UDD instances grouped by number of pickers $|K|$ and number of tour subgraphs per picker $|O^k|$ using the optimal and largest gap routing policy

		Optimal routing policy					Largest gap routing policy				
			Gurobi		ILS			Gurobi		ILS	
$\|K\|$	$\|O^k\|$	f^{rand}	ub	lb	$best$	avg	f^{rand}	ub	lb	$best$	avg
2	2	39	0	0	0	0	55	6	6	6	6
2	5	94	1	1	1	3	104	17	17	17	18
2	10	179	2	2	2	4	192	27	27	29	31
2	20	331	0	0	3	7	440	53	53	70	75
2	Avg.	161	0	0	2	3	198	26	26	30	30
5	2	414	112	112	128	128	563	293	293	293	310
5	5	978	254	254	303	323	1144	572	572	583	606
5	10	1897	436	436	455	474	2220	1066	1066	1066	1066
5	20	3746	937	937	1049	1049	4421	2122	2122	2122	2122
5	Avg.	1759	440	440	510	510	2146	1052	1052	1073	1094
10	2	1712	822	822	873	890	2243	1503	1503	1503	1525
10	5	4034	1896	1896	1936	1936	4881	3173	3173	3173	3173
10	10	7949	3736	3339	3895	3895	9300	6138	6138	6138	6138
10	20	15891	7628	6674	7946	7946	19070	12586	12586	12586	12586
10	Avg.	7397	3551	3329	3625	3699	9024	5956	5956	5956	5956
20	2	8012	5208	5208	5288	5368	9934	7550	7550	7550	7649
20	5	17658	11654	10771	11654	11654	21190	16316	16316	16316	16316
20	10	34590	23867	20408	22829	22829	40816	31428	30612	31428	31428
20	20	67836	52234	40023	45450	45450	80724	62157	59736	62157	62157
20	Avg.	32024	22097	19535	21136	21456	38429	29590	28822	29590	29590
40	2	34279	25024	24681	25024	25024	42506	34005	34005	34005	34005
40	5	74749	58304	55314	57557	57557	89699	74450	72656	74450	74450
40	10	143115	128804	105905	111630	111630	168875	168875	138478	141855	141855
40	20	280723	272301	210542	221771	221771	331253	331253	0	281565	281565
40	Avg.	133216	111901	98580	102576	102576	159860	145473	97515	132684	132684

Table 8 Results on the UDD instances grouped by number of pickers $|K|$ and number of tour subgraphs per picker $|O^k|$ using the S-shape and S-shape+ routing policy

		S-shape routing policy					S-shape+ routing policy				
			Gurobi		ILS			Gurobi		ILS	
$\|K\|$	$\|O^k\|$	f^{rand}	*ub*	*lb*	*best*	*avg*	f^{rand}	*ub*	*lb*	*best*	*avg*
2	2	109	0	0	0	2	94	0	0	0	0
2	5	175	11	11	11	16	159	0	0	0	0
2	10	270	14	14	24	30	270	0	0	0	0
2	20	539	22	22	38	49	552	0	0	0	6
2	Avg.	315	13	13	19	25	294	0	0	0	0
5	2	948	370	370	370	379	750	38	38	128	165
5	5	1770	673	673	673	726	1564	78	78	328	360
5	10	3244	1168	1168	1200	1200	2998	150	150	450	450
5	20	5919	2308	2308	2368	2368	5694	285	285	911	968
5	Avg.	3254	1237	1237	1237	1269	2867	143	143	487	545
10	2	3528	2117	2117	2152	2152	2963	919	919	1185	1244
10	5	7059	4235	4235	4235	4235	6535	2157	1764	2549	2549
10	10	12957	7645	7645	7645	7774	12480	4243	2870	5117	5117
10	20	24791	14627	14627	14875	14875	24473	9544	5139	10279	10279
10	Avg.	12944	7766	7766	7766	7766	11983	4074	3116	4793	4913
20	2	15542	10879	10879	11035	11035	12979	7009	6879	7528	7658
20	5	30548	21995	21995	22300	22300	28076	16565	14880	16846	17126
20	10	56382	41159	40595	41723	41723	53614	38066	28415	33241	33241
20	20	106502	78811	76681	79877	79877	104467	76261	55368	66859	66859
20	Avg.	55081	40209	39658	40209	40209	50598	32383	26817	30865	30865
40	2	66158	51603	51603	51603	51603	54846	37295	36747	37844	37844
40	5	129316	103453	103453	104746	104746	117356	91538	83323	86843	86843
40	10	233277	226279	188954	191287	191287	221828	179681	159716	168589	168589
40	20	440734	436327	0	370217	370217	432313	358820	315588	337204	337204
40	Avg.	227800	202742	136680	184518	184518	209150	163137	148497	154771	154771

groups of instances defined by the number of pickers (column K) and the number of tour subgraphs per picker (column $|O^k|$). The remaining columns are divided into two blocks, where each block reports the results for one of the underlying routing policies.

In column f^{rand}, we give the average temporal overlap of 1000 randomly generated solutions, i.e., we randomly construct directed picking tours from the respective tour subgraphs. Column *ub* (*lb*) indicates the average temporal overlap of the best objective function (lower bound) values given by Gurobi. For ILS, the reported results are based on five runs on each instance. In column *best* (*avg*), we show the average temporal overlap of the best (average) solution value over the five runs.

Table 9 Results on the CBD instances grouped by number of pickers $|K|$ and number of tour subgraphs per picker $|O^k|$ using the optimal and largest gap routing policy

		Optimal routing policy					Largest gap routing policy				
$\|K\|$	$\|O^k\|$	f^{rand}	Gurobi		ILS		f^{rand}	Gurobi		ILS	
			ub	lb	$best$	avg		ub	lb	$best$	avg
2	2	43	3	3	3	4	43	1	1	2	3
2	5	86	3	3	3	6	100	4	4	4	5
2	10	178	5	5	7	11	199	10	10	18	18
2	20	348	10	10	14	17	383	38	38	61	65
2	Avg.	164	8	8	8	11	179	11	11	14	18
5	2	427	188	188	196	201	474	237	237	242	251
5	5	950	371	371	399	418	1074	494	494	505	516
5	10	1869	673	673	673	692	2131	1002	1002	1002	1002
5	20	3657	1353	1353	1426	1426	3950	1817	1817	1857	1857
5	Avg.	1726	673	673	708	725	1916	901	901	920	939
10	2	2027	1196	1196	1216	1236	2270	1430	1430	1453	1453
10	5	4264	2516	2516	2558	2558	4691	3049	3049	3049	3049
10	10	8394	4952	4952	4952	4952	8982	5838	5838	5838	5838
10	20	16329	9797	9797	9797	9961	17472	11357	11357	11532	11532
10	Avg.	7754	4575	4575	4652	4652	8451	5409	5409	5493	5493
20	2	8392	5707	5707	5790	5790	9231	6554	6554	6554	6554
20	5	17904	12712	12712	12712	12712	19516	14442	14442	14442	14442
20	10	34662	25303	24610	25303	25303	37435	28451	28451	28451	28451
20	20	67718	50111	48757	50789	50789	73135	57045	55583	57045	57045
20	Avg.	32169	23162	22840	23162	23162	34743	26057	25710	26057	26057
40	2	34431	25135	25135	25135	25135	37874	28784	28784	28784	28784
40	5	73471	58042	57307	58042	58042	80083	64867	64867	64867	64867
40	10	141015	117042	114222	115632	115632	153707	129114	127577	129114	129114
40	20	276106	276106	226407	231929	231929	300955	300955	252802	255812	255812
40	Avg.	131256	110255	102380	103692	103692	143069	121609	115886	115886	115886

Table 10 Results on the CBD instances grouped by number of pickers $|K|$ and number of tour subgraphs per picker $|O^k|$ using the S-shape and S-shape+ routing policy

		S-shape routing policy					S-shape+ routing policy				
			Gurobi		ILS			Gurobi		ILS	
$\|K\|$	$\|O^k\|$	f^{rand}	ub	lb	$best$	avg	f^{rand}	ub	lb	$best$	avg
2	2	56	6	6	6	7	55	0	0	0	0
2	5	135	24	24	26	30	125	0	0	1	3
2	10	238	29	29	40	48	236	0	0	0	2
2	20	446	36	36	62	67	463	0	0	0	5
2	Avg.	225	27	27	34	38	220	0	0	0	2
5	2	666	286	286	313	313	572	109	109	149	160
5	5	1330	585	585	625	652	1254	201	201	351	376
5	10	2542	1093	1093	1118	1118	2467	419	419	567	592
5	20	4901	2107	2107	2107	2156	4901	784	784	1176	1176
5	Avg.	2451	1054	1054	1103	1127	2295	390	390	574	597
10	2	3041	1916	1916	1977	2007	2696	1186	1186	1348	1348
10	5	6013	3788	3788	3788	3788	5714	2514	2457	2628	2686
10	10	11332	7026	7026	7139	7139	11164	5024	4689	5359	5359
10	20	21880	13566	13566	13784	13784	21880	9846	8533	10940	10940
10	Avg.	10855	6839	6839	6947	6947	10390	4572	4364	4987	5091
20	2	12504	8878	8878	9003	9003	11245	6635	6635	6859	6972
20	5	25424	18814	18814	18814	18814	24171	15228	14744	15469	15711
20	10	47487	35615	35615	35615	35615	46794	30884	28544	31352	31352
20	20	92774	70508	69581	70508	70508	92097	73678	57100	62626	62626
20	Avg.	45359	33566	33112	33566	33566	43428	29097	26491	28228	28228
40	2	51991	40553	40553	40553	40553	46827	32311	32311	32779	32779
40	5	105063	86152	85101	86152	86152	99920	75939	73941	75939	75939
40	10	196011	164649	162689	164649	164649	191781	157260	147671	151507	151507
40	20	378265	374482	0	321525	321525	372743	324286	290740	301922	301922
40	Avg.	186383	160289	111830	152834	152834	178508	141021	133881	137451	137451

Acknowledgements We gratefully acknowledge the support of the Computational Science Lab (CSL) Hohenheim through the CSL Seed Grant Program.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability The data that support the findings of this study are available from the corresponding author upon request.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Amazon (2020) The company's latest innovation provides real-time social distancing feedback and we plan to open source the technology, 2020. Available online at https://www.aboutamazon.com/news/operations/amazon-introduces-distance-assistant. Accessed 8 Feb 2022

Ardjmand E, Singh M, Shakeri H, Tavasoli A, Young WA II (2021) Mitigating the risk of infection spread in manual order picking operations: a multi-objective approach. Appl Soft Comput 100:1–21

Azadeh K, de Koster RBM, Roy D (2019) Robotized and automated warehouse systems: review and recent developments. Transp Sci 53(4):917–945

Boysen N, de Koster RBM, Weidinger F (2019) Warehousing in the e-commerce era: a survey. Eur J Oper Res 277(2):396–411

Boysen N, de Koster RBM, Füßler D (2021) The forgotten sons: warehousing systems for brick-and-mortar retail chains. Eur J Oper Res 288(2):361–381

Chen F, Wang H, Qi C, Xie Y (2013) An ant colony optimization routing algorithm for two order pickers with congestion consideration. Comput Ind Eng 66(1):77–85

Chen F, Wang H, Xie Y, Qi C (2014) An ACO-based online routing method for multiple order pickers with congestion consideration in warehouse. J Intell Manuf 27(2):389–408

DC Velocity (2020) In pandemic times, DC managers rethink their labor strategies, 2020. Available online at https://www.dcvelocity.com/articles/48024-in-pandemic-times-dc-managers-rethink-their-labor-strategies. Accessed 8 Feb 2022

de Koster RBM, Le-Duc T, Roodbergen KJ (2007) Design and control of warehouse order picking: a literature review. Eur J Oper Res 182(2):481–501

Elbert RM, Franzke T, Glock CH, Grosse EH (2017) The effects of human behavior on the efficiency of routing policies in order picking: the case of route deviations. Comput Ind Eng 111:537–551

Goeke D, Schneider M (2018) Modeling single picker routing problems in classical and modern warehouses. Working paper, DPO-2018-11 (version 1, 04.11.2018), Deutsche Post Chair—Optimization of Distribution Networks, RWTH Aachen University

Goetschalckx M, Ratliff HD (1988) Order picking in an aisle. IIE Trans 20(1):53–62

Grosse EH, Glock CH, Ballester-Ripoll R (2014) A simulated annealing approach for the joint order batching and order picker routing problem with weight restrictions. Int J Oper Quant Manag 20(2):65–83

Gue KR, Meller RD, Skufca JD (2006) The effects of pick density on order picking areas with narrow aisles. IIE Trans 38(10):859–868

Hall RW (1993) Distance approximations for routing manual pickers in a warehouse. IIE Trans 25(4):76–87

Henn S, Wäscher G (2012) Tabu search heuristics for the order batching problem in manual order picking systems. Eur J Oper Res 222(3):484–494

Hong S, Johnson AL, Peters BA (2012) Batch picking in narrow-aisle order picking systems with consideration for picker blocking. Eur J Oper Res 221(3):557–570

Masae M, Glock CH, Grosse EH (2019) Order picker routing in warehouses: a systematic literature review. Int J Prod Econ. https://doi.org/10.1016/j.ijpe.2019.107564

Napolitano M (2012) 2012 warehouse/DC operations survey: mixed signals. Mod Mater Handl 51(11):48–56

Pan JC-H, Shih P-H (2008) Evaluation of the throughput of a multiple-picker order picking system with congestion consideration. Comput Ind Eng 55(2):379–389

Parikh PJ, Meller RD (2009) Estimating picker blocking in wide-aisle order picking systems. IIE Trans 41(3):232–246

Ratliff HD, Rosenthal AS (1983) Order picking in a rectangular warehouse: a solvable case of the traveling salesman problem. Oper Res 31(3):507–521

Schrotenboer AH, Wruck S, Roodbergen KJ, Veenstra M, Dijkstra AS (2017) Order picker routing with product returns and interaction delays. Int J Prod Res 55(21):6394–6406

Statista (2020) Total retail sales worldwide from 2017 to 2023, 2019. Available online at https://www.statista.com/statistics/443522/global-retail-sales. Accessed 1 Mar 2021

Tompkins JA, White JA, Bozer YA, Tanchoco JMA (2010) Facilities planning, 4th edn. Wiley

Weidinger F (2018) Picker routing in rectangular mixed shelves warehouses. Comput Oper Res 95:139–150

World Health Organization. Transmission of SARS-CoV-2, 2020. Available online at https://www.https://www.who.int/news-room/commentaries/detail/transmission-of-sars-cov-2-implications-for-infection-prevention-precautions. Accessed 6 April 2021

Zhang M, Batta R, Nagi R (2009) Modeling of workflow congestion and optimization of flow routing in a manufacturing/warehouse facility. Manage Sci 55(2):267–280

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.