Graph reduction for the planar Travelling Salesman Problem

Maastricht University

Farzaneh Rajabighamchi,
Stan van Hoesel, Christof Defryn

Graph reduction for the planar Travelling Salesman Problem. An application in order picking.

RM/23/004

ISSN: 2666-8807

GSBE

Maastricht University School of Business and Economics
Graduate School of Business and Economics

P.O Box 616
NL- 6200 MD Maastricht
The Netherlands

Graph reduction for the planar Travelling Salesman Problem. An application in order picking.

Farzaneh Rajabighamchi, Stan van Hoesel, and Christof Defryn

Maastricht University, School of Business and Economics, P.O.Box 616, 6200 MD Maastricht, The Netherlands

{*f.rajabighamchi; s.vanhoesel; c.defryn*}*@maastrichtuniversity.nl*

May 1, 2023

Abstract

This paper presents an improved exact algorithm for solving the order picking problem, a special case of the planar Travelling Salesperson Problem. The algorithm heavily relies on graph reduction techniques: it removes unnecessary vertices and edges from the planar graph that are not necessary in the optimal solution. As a result, we achieve a significant increase in calculation speed and reduction in the running time.

The order pickers routing problem entails collecting items from storage in response to customer requests. We use the Traveling Salesperson Problem (TSP) to optimize the routes taken by order pickers. In the literature, exact algorithms — typically based on dynamic programming — only exist for small warehouses with a small number of blocks (two), while for larger warehouse layouts mainly heuristic and metaheuristic methods are provided.

The presented graph reduction method allows us to adequately solve larger — more realistic — instances in a short amount of time. Our algorithm is tested on different problem instances from the literature and its performance is compared with the current state-of-the-art. We conclude that our algorithm outperforms existing algorithms in terms of simplicity, size and calculation time.

Keywords: Order Picking, Routing, Pre-processing, Graph Reduction, Warehouse management

Contents

1 Introduction and research context

For businesses operating with physical products, warehouses play an integral part in the efficiency of their supply chains. [24] highlight that — while modern supply chains have aimed at reducing inventory through initiatives such as “Just-In-Time” — warehouses are still present in most supply chain stages. The warehousing service is a very important component of the logistics system and plays a vital role in the supply chain process by balancing supply and demand. Thanks to the rapid growth of the e-commerce sector (accelerated by the COVID19 pandemic), the number of warehouses has even increased considerably over the last decade [17, 7]. As a result, supply chain sectors and specifically warehouses are forced to further streamline processes by increasing efficiency and cutting costs, while still ensuring high service levels to their customers Dembińska [16] and Gutelius, Theodore, et al. [25].

A warehouse process that provides significant cost-saving potential, is the *order picking process* as it is estimated to account for up to 55 percent of the total warehouse operating cost [46]. Efficient warehousing provides an important economic benefit to the business as well as to the customers. Due to the introduction of operating programs (such as cycle time reduction and quick response to orders) and new marketing strategies (e.g., micro marketing), the order picking process has become increasingly significant to manage. Moreover, catalyzed by the rapid technological advancements, the world of e-commerce is transforming fast. This significant growth in digital marketing together with the daily increase in the number of customers that buy online imposes challenges for warehouses to remain responsive as well as efficient. Any under-performance in order picking can result in high operational costs and an unsatisfactory service for the warehouse as well as the supply chain as a whole.

The order picking process is defined as *the retrieval of products from their storage locations based on customer orders.* Various activities comprise the order-picking process, including, e.g., traveling between items and packaging the order. Tompkins et al. [46] state that travelling can consume 50% of a picker’s time, which in manual-picking systems constitutes high (labour) costs. As a result, efficient routing algorithms are needed to optimize the pick tours for retrieving the products from storage.

The problem of sequencing and finding an optimal route for a picker, i.e., obtaining the shortest tour that starts and ends at the depot and visits all items included in an order list (each item is visited exactly once) resembles the Traveling Salesperson Problem (TSP). The TSP is an NP-hard optimization problem (see Garey, Graham, and Johnson [20], Liers, Martin, and Pape [31], and Arora [2]). Within the academic literature, therefore, exact algorithms are only available for very small instances with a standard warehouse lay-outs consisting of two and three ‘blocks’, separated by a cross aisle [45].

In this paper, we present an exact routing algorithm for general warehouse lay-outs using the TSP as the model to be solved. The TSP for order picking has a special structure in two ways. First, the warehouse lay-out provides an underlying planar graph. Second, all pick locations are found in aisles where they have degree 2 in the planar graph. Both properties are extensively used within our approach to reduce the size of the graph of the TSP: we remove a substantial amount of nodes and edges. Then, the Miller-Tucker-Zemlin formulation of the TSP is used to solve the problem to optimality.

The remainder of this paper is structured as follows. In Section 2, we review the related studies in the literature. In Section 3 we provide a formal problem description and briefly represent the warehouse layout with a mathematical explanation. The graph reduction methods are provided in section 4. The mathematical formulation of the problem is provided in section 5. Furthermore, in section 6, the computational results are presented. The conclusions of our work are presented in Section 7 along with some further research suggestions.

2 Related literature

The order picking problem has been the subject of significant research in the past few decades, with numerous solution approaches proposed, including Dynamic Programming, Integer Linear Programming (ILP), and various heuristics. Most research has focused on modeling the problem as either a Traveling Salesman Problem (TSP) or a Steiner TSP. Cases where the warehouse has one or two blocks have been shown to be solvable in polynomial time.

Ratliff and Rosenthal [37] and Cornuéjols, Fonlupt, and Naddef [12] were the first to propose a dynamic programming approach for a warehouse with one block, which was polynomial in the number of items and aisles. This approach was then extended by Roodbergen and De Koster [39] to the case of two blocks and later by Cambazard and Catusse [5] to warehouses with h cross-aisles (maximum h to be solved exactly is 8); the latter being solvable non-polynomially but exponential in h.

In some papers, the Steiner TSP is used for solving this problem since it is not necessary to visit all vertices in the warehouse graph. The Steiner TSP was first studied by Cornuéjols, Fonlupt, and Naddef [12] and Fleischmann [19]. Several other formulations for the compact Steiner TSP were proposed by Letchford, Nasiri, and Theis [29], Pansart, Catusse, and Cambazard [33], Scholz et al. [43], and Valle, Beasley, and Cunha [48].

The order picker routing problem has also been studied by modelling it as the capacitated vehicle routing problem (CVRP) in multiple studies, including Glock and Grosse [21] and Scholz and Wäscher [42]. In the CVRP formulation for this problem, each pick location is considered a node that should be visited only once by a vehicle (i.e., picker). The vehicle capacity is given by the picking trolley or forklift capacity. The objective is to find routes with the minimal total travel distance or time.

Given the limitations of the existing exact methods, in practice , the order picking problem is typically solved using heuristics. Common methods are the largest gap, return, midpoint, composite routing strategy, the combined routing strategy and finally, the S-shape method in which order pickers move in a S-shape curve along with the pick locations [22, 18, 26, 34]. A preliminary research on heuristic routing in multi-parallel aisle warehouses was done by Hall [26]. Some of the important and commonly used heuristic algorithms are illustrated in figure 1.

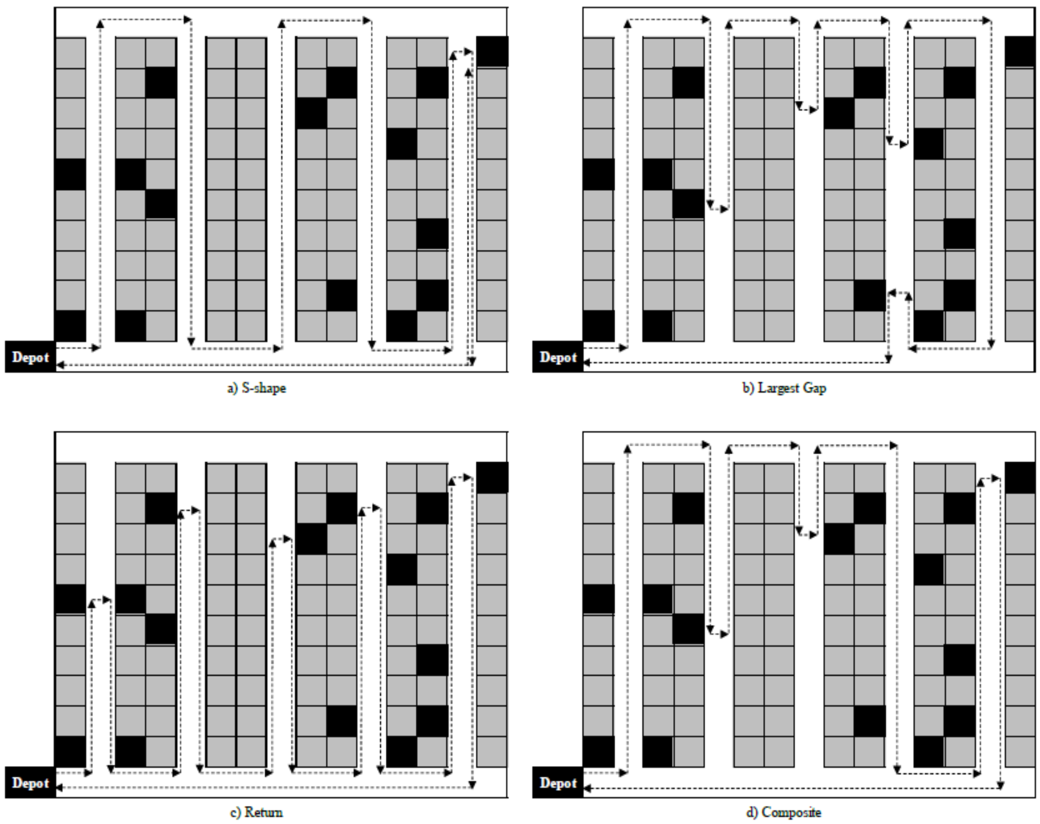


Figure 1: *Picker routing heuristics considered to retrieve items of a pick list (from Cano, Correa-Espinal, and Gómez-Montoya [6]).*

The *S-shape strategy* leads to a route in which each aisle containing a pick location to be visited is completely traversed, and aisles where nothing has to be picked are skipped. The picker enters an aisle from one end and leaves from the other end, starting at the left side of the warehouse. After picking the last item, the order picker returns to the front end of the aisle. This S-shape strategy is used frequently, because it is very simple to use and to understand.

The *Largest Gap strategy* has the picker entering an aisle as far as the largest gap within the aisle, with a gap representing the distance between any two adjacent picks, between the first pick and the front aisle, or between the last pick and the back aisle. The largest gap is the part of the aisle that the order picker does not visit, and if the largest gap is between two adjacent picks, the picker performs a return route from both ends of the aisle. If a return route is needed, it can be taken from either the front or back aisle. The largest gap in an aisle is the part the picker does not travel through. This method is particularly useful when switching aisles takes little time and there are not many picks per aisle. In their study, Ho and Tseng [27] present a new way to solve the picker routing problem by combining the largest gap heuristic with a simulated annealing heuristic. Their proposed method is more efficient than the largest gap heuristic alone.

The *combined heuristic* uses both the Largest Gap and S-Shape heuristics. This means an aisle is either fully traversed or entered and exited from the same side. The best option is chosen between these two methods, and then the next aisle is entered. This process is repeated until the last item is picked, and the best route between the two options is selected. The combined routing heuristic is one of the best heuristic methods available and is provided in Roodbergen [40] and Roodbergen and De Koster [39].

The S-shape and largest gap heuristics are the most commonly used routing policies in real warehouses. This is because order pickers prefer straightforward and easy-to-understand routing schemes.

In Petersen [35], more advanced heuristics are presented and their performance is compared to the optimal algorithm. De Koster and Van Der Poort [14] developed an algorithm for finding the shortest order picking routes in a warehouse with decentralized depositing. In the same year, Roodbergen and De Koster [41] provide three heuristics for different situations, including a narrow-aisle warehouse used by order picking trucks. Vaughan [49] present a routing heuristic that makes use of dynamic programming for warehouses with more than two cross aisles and studied the effect of warehouse cross aisles on order picking efficiency.

Petersen and Aase [36] evaluate several picking, routing, and storage policies to determine which policy or combination of policies would provide the biggest tour reduction in total, considering four factors: picking policy, routing policy, storage policy, and average order size.

In a recent study by Weidinger, Boysen, and Schneider [50] on the picker-routing problem for mixed shelves warehouses, a nearest neighborhood heuristic method is proposed. It considers a cart pushed by the picker that allows for the assembly of multiple orders concurrently and multiple access points to the central conveyor system where completed orders are handed over. Furthermore, Theys et al. [45] propose and compare the LKH (Lin–Kernighan–Helsgaun) TSP heuristic with some of the existing heuristics in the literature such as S-shaped and largest gap and concluded that the LKH heuristic provides better solution quality (closer to optimum), although its computation time is higher.

Various metaheuristic methods have been proposed in addition to the heuristic methods discussed in the literature, such as genetic algorithms [3, 47], Ant Colony Optimization [30, 11, 15], particle swarm optimization [23, 44, 32], and tabu search [13]. Chabot et al. [10] use an adaptive large neighborhood search (ALNS) to solve the order picker routing problem and compare their proposed heuristic solution with four other existing heuristics in the literature, namely S-shape, the largest gap, the mid-point, and the combined heuristics, showing that the ALNS outperforms the other four heuristics. Bódis and Botzheim [4] applied a bacterial memetic algorithm based on pick list characteristics and order picking system characteristics to solve the order picker routing problem. Recently, Zhou et al. [51] developed three routing metaheuristics, namely a genetic algorithm, an ant colony optimization, and a cuckoo search algorithm, to solve the order picker routing problem in non-conventional fishbone warehouses with narrow aisles and a single storage system. Ardjmand, Bajgiran, and Youssef [1] investigated the order picker routing problem using two genetic algorithms with a list-based simulated annealing. Metaheuristic methods improve the performance of the calculation method, provide a set of guidelines or strategies to find an approximate solution for the problem, and decrease the running time.

3 Problem description

In this section, we model the warehouse lay-out as a graph G_L and formulate the problem of picking orders as a shortest (closed) walk problem on G_L. Then, we define a smaller graph G_{PL} solely on the pick locations as vertices. On this graph the problem can be defined as a TSP. The TSP problem is easier to formulate and solve compared to the shortest walk problem. Afterwards, in Section 4, we develop ideas to reduce the number of edges and vertices in G_{PL} drastically to obtain good solution times.

3.1 Graph representation of the warehouse layout

Standard, multi-parallel-aisle warehouses consist of a number of longitudinal pick aisles, where product items can be picked, and intersecting cross aisles that connect these pick aisles. In practice, the cross aisles do not contain any items to pick but just allow the order picker to move efficiently from one pick aisle to another. The items are stored on both sides of the pick aisles. Order pickers are assumed to be able to traverse the aisles in both directions and to change direction within the aisles.

Each order consists of a number of items that are usually spread over multiple aisles. We assume that the items of an order can be picked in a single round. The task of a picker is to find a route (a closed walk) that starts at a depot, and then visits all picking locations, and ends at the depot again. This route should, of course, be as short as possible.

Each standard warehouse lay-out is divided in a number of blocks. A block is a row of pick aisles between two cross aisles. A detailed picture of the standard warehouse lay-out is given in figure 2.

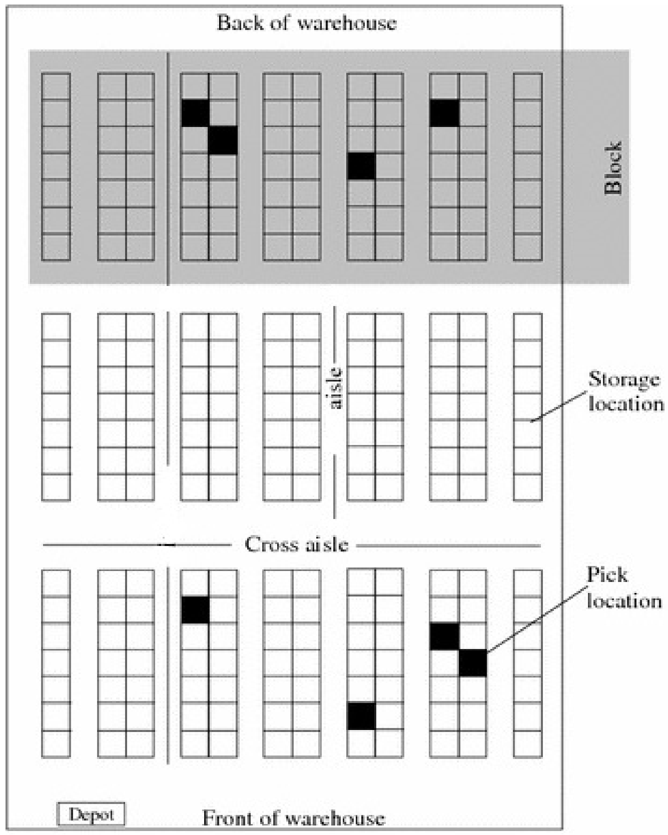


Figure 2: *A standard warehouse lay-out (from Roodbergen [38]).*

Though most warehouse lay-outs in the literature are indeed modelled as the standard lay-out above, this is not a limitation. Other lay-outs, such as the fish-bone lay-out, flying V, Chevron etc. have been researched too (see, e.g., Çelik and Süral [9] and Figure 3). For our research, the lay-out does not really matter: As long as it can be drawn in the plane (planar graph), our results apply.

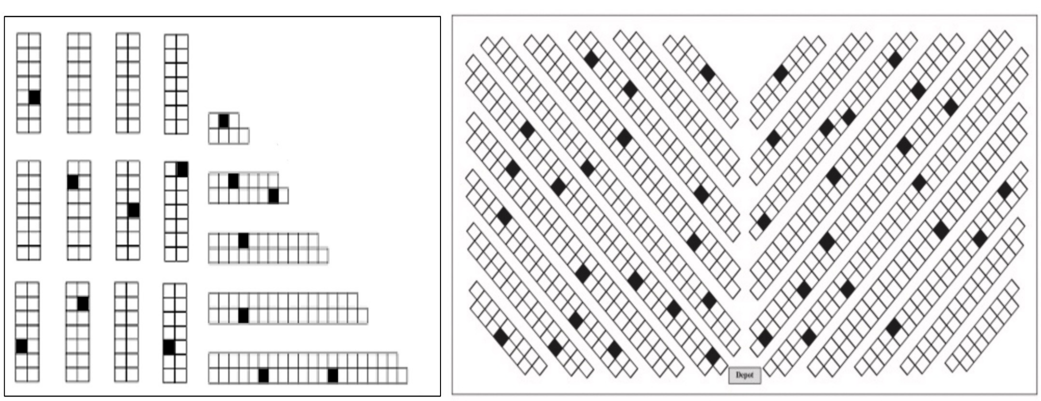


Figure 3: *Two other warehouse lay-outs (from Celik and Sural [8]).*

3.2 The pick location problem as a shortest walk problem

We model the lay-out of the warehouse with a graph $G_L = (V_L, E_L)$. Every point in the lay-out where two or more aisles meet is a vertex in V_L. Each aisle, connecting two such vertices, is represented by an edge in E_L. When picking locations are added as vertices, each edge in E_L is replaced by a path (see below). An edge in E_L may or may not contain picking locations. This brings us to the second part of the definition of G_L: a second set of vertices is defined by the pick locations.

For the standard lay-out described above, this results in a grid graph with symmetric distance. For the fish-bone and other warehouse lay-outs the graph is somewhat different. However, since all lay-outs are 2-dimensional physical structures, G_L is planar.

Now, every edge $\{v, w\} \in E_L$ containing pick locations is replaced by a path as follows. Suppose that the edge $\{v, w\}$ contains $j - 1$ pick locations $(p_1, \ldots, p_{j-1})$. Let $v = p_0$ and $w = p_j$. We now replace the edge $\{v, w\}$ with the path $v = p_0, p_1, \ldots, p_j = w$. So, besides adding the vertices $p_1, \ldots, p_{j-1}$, we also add the edges $\{p_{i-1}, p_i\}$, $(i = 1, \ldots, j)$ to E_L. Note that the vertices representing pick locations have degree 2 — which is important for the remainder of our analysis.

A special vertex in V_L is the depot, where we start and end the pick tour.

The graph G_L for two example lay-outs is illustrated in Figures 4 and 5. Here, black dots represent pick locations and white dots represent corner points of aisle.

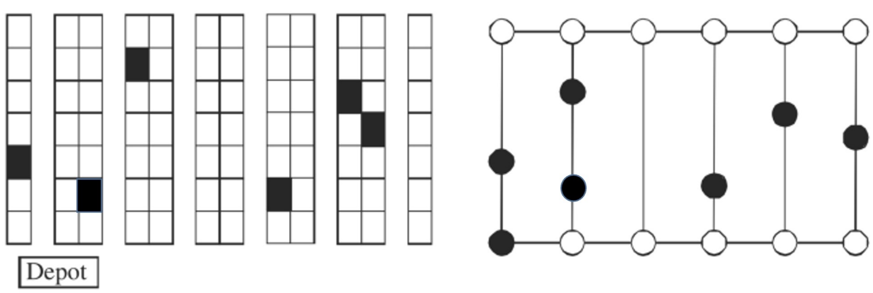


Figure 4: *Graph G_L for standard warehouse lay-out (from Roodbergen [38] and Çelik and Süral [9]).*

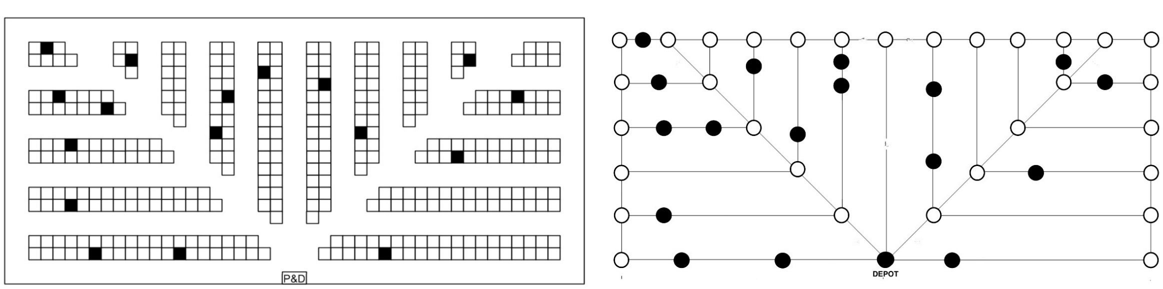


Figure 5: *Graph G_L for a fish-bone warehouse lay-out (from Çelik and Süral [9]).*

Each edge $\{v, w\} \in E_L$ has a length d_{vw}, representing the actual distance between the vertices v and w in the warehouse lay-out. Thus, this is simply the length of the edge in the lay-out connecting v and w.

We then define the order picking problem as follows: find a shortest walk in G_L that starts and ends at the depot, visiting each vertex, representing a pick location in V_L, at least once.

3.3 Modelling the pick location problem as a TSP

The problem can be modelled as a TSP on a graph, containing only the nodes of the pick locations. This graph $G_{PL} = (V_{PL}, E_{PL})$ only contains the vertices of G_L that represent pick locations and the depot. Each edge $\{v, w\} \in E_{PL}$ has a length that is the shortest distance between the pick locations v and w in the lay-out graph G_L. In G_{PL}, the problem is to find the shortest tour through all vertices, i.e., the standard TSP problem.

Distances in G_{PL} if the underlying graph G_L is from a standard lay-out

Consider the standard lay-out (see again Figure 4), where G_L is a grid graph. To calculate the length of each edge in G_{PL}, which is the shortest path between the two vertices (either a pick location or the depot) of the edge, we do the following:

1. If both nodes (product locations) are in different blocks (different row of aisles) the shortest distance between the two nodes is calculated as the Manhattan distance, which is the rectilinear route measured along parallels to the horizontal and vertical axes of the plane. The Manhattan distance between two points with coordinates (x_1, y_1) and (x_2, y_2) is
$$d_{12} = \mid x_1 - x_2 \mid + \mid y_1 - y_2 \mid .$$

2. If both nodes (product locations) are in the same block (same row of aisles), the shortest path has to go through one of the two cross aisles adjacent to the block. The length of both possible paths is then determined and the shorter one is kept. Therefore, for two points with coordinates (x_1, y_1) and (x_2, y_2) we will have
$$d_{12} = \{min\{\gamma_1 + \gamma_2, \beta_1 + \beta_2\} + \mid x_1 - x_2 \mid\} = \{\mid x_1 - x_2 \mid + \mid y_1 - y_2 \mid + 2 \times min\{\gamma_2, \beta_1\}\}.$$
Here, γ_i, $(i = 1, 2)$, is defined as the difference between the y_i-coordinate of each location with the cross node located above it, and β_i, $(i = 1, 2)$, is defined as the difference between the y_i-coordinate of each location with the cross node located below it. A graphical sketch of the warehouse lay-out for this case is illustrated in figure 6.

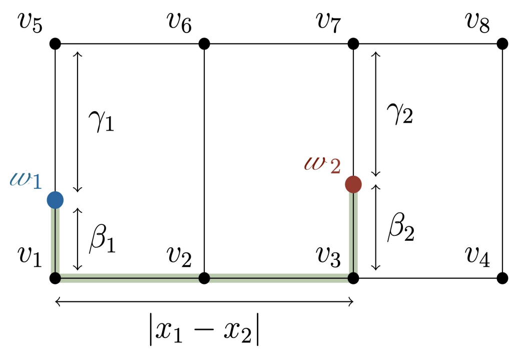


Figure 6: *Distance of vertices in the same block, but different aisles.*

4 Properties of the lay-out graph G_L and the pick locations graph G_{PL}

In this section we do two things: reduction of vertices and reduction of edges in G_{PL}. First, we show that some pick locations need not be present in G_{PL}. Second, we show that there is an optimal walk in G_L that can be translated into an optimal tour in G_{PL}, using only a very small portion of the edges in G_{PL}. Roughly speaking, we only need the edges of G_{PL} that correspond to shortest paths in G_L with at most one intermediate pick location.

4.1 Visiting pick locations in G_L

Following Pansart, Catusse, and Cambazard [33], there are three different ways by which items in each aisle in a warehouse can be picked, as illustrated in figure 7:

1. Visit the complete aisle in one direction (two ways). See 7a, and 7b.
2. Enter the aisle from one of the corner nodes until you reach the last product location on the aisle and then return to the same corner node (two ways). See 7c, and 7d.
3. Let the biggest gap between two pick locations be the one between p_{i-1} and p_i. Enter the aisle from corner node p_0 to p_{i-1} and return, and from corner node p_j to p_i and return. See 7e.

Consequently, there are at most 4 relevant pick locations in each aisle: the ones closest to the corner nodes and the ones that have the biggest gap between them. Note that there can be fewer than four relevant pick locations on an aisle, or some of the nodes could coincide. When removing irrelevant nodes one should pay attention to the situation where the biggest gap may move. In that case we take care of that in the model by adding constraints that force necessary edges to be in the walk. To do this, we implement following:

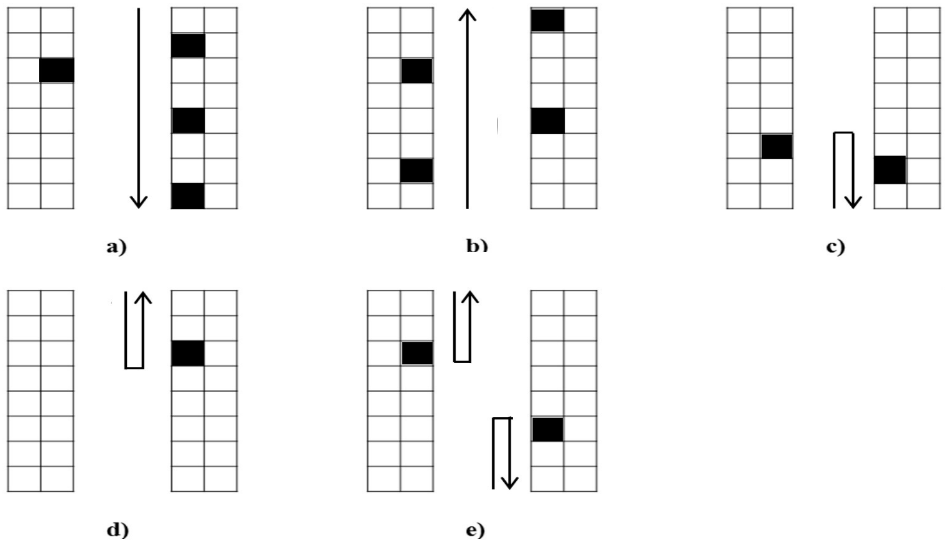


Figure 7: *the five different ways for aisle traversal (from Pansart, Catusse, and Cambazard [33]).*

for every sub-aisle **do**

- Compute a largest gap between two vertices

- Identify the two sets containing all products below and above the largest gap (call them set T and set S). These sets can be empty or singleton.

- In each subset, keep the two products that are the farthest apart ($t_, t_2 \in T$ and $s_1, s_2 \in S$)

- Add the constraints forcing the order picker to traverse each set once. (In the mathematical model it translates into the following constraints: ($\{s_1, s_2\} + \{s_2, s_1\} >= 1$ and $\{t_1, t_2\} + \{t_2, t_1\} >= 1$).

In these constraints, $\{s_1, s_2\}$ refers to the edges in set S connecting first and second farthest products.

end

Please note that the above mentioned constraints are added only when by removing one product, the place of Largest gap changes. Otherwise we do not need such constraints in our model.

4.2 Properties of an optimal walk in G_L

Lemma 4.1. *An optimal walk in $G_L = (V_L, E_L)$ that visits all pick locations at least once, will not use any edge in G_L twice, or more, in the same direction.*

Proof. Consider a walk W. Let the edge $e = \{v, w\} \in E_L$ be used twice (or more) in the direction $v \to w$ by W. Then, W contains two paths from w to v, say P_1 and P_2. The walk is now: $W = (v, w, P_1, v, w, P_2, v)$. Now, consider the walk $W' = (w, P_1, v, P_2^{-1}, w)$, where P_2^{-1} is the path P_2 traversed in backward direction. W' is shorter than W as it does not use the edge e anymore. Moreover, it visits every vertex that W visits, using edges from W. Concluding, W cannot be optimal. The proof is visualized in Figure 8. ☐

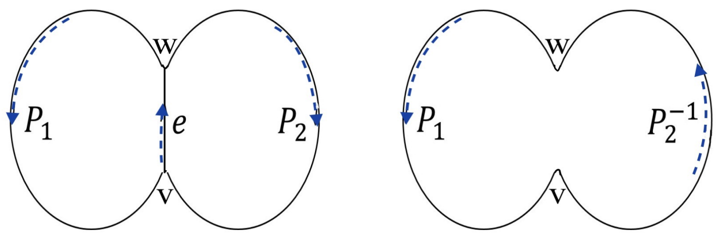


Figure 8: *Edge twice in same direction*

In Kai and Chuanhou [28], a somewhat more complicated proof is given of this lemma. From Lemma 4.1, we can conclude that in an optimal walk, every edge is traversed at most twice. And if twice, then in opposite directions.

Lemma 4.2. *If an optimal walk in $G_L = (V_L, E_L)$ uses the edge $e = \{v_e, w_e\}$ twice — in opposite direction, as a consequence of the previous lemma — (say, $W = (v_e, P_1, v_e, w_e, P_2, w_e, v_e)$), then the paths P_1 and P_2 have no vertices of G_L in common.*

Proof. Suppose that a walk W contains $e = \{v_e, w_e\}$ twice (in opposite direction), and that a vertex $v \neq v_e, w_e$ is visited by both paths P_1 and P_2. Then the walk can be described as

$$W = (v_e, P_1^{'}, v, P_1^{''}, v_e, w_e, P_2^{'}, v, P_2^{''}, w_e, v_e)$$

Here: $P_1 = (P_1^{'}, v, P_1^{''})$ and $P_2 = (P_2^{'}, v, P_2^{''})$, where

$P_1^{'}$ is a path from v_e to v.

$P_1^{''}$ is a path from v to v_e.

$P_2^{'}$ is a path from w_e to v.

$P_2^{''}$ is a path from v to w_e.

From W we create a better walk W' as follows:

$$W' = (v_e, P_1^{'}, v, P_2^{''}, w_e, P_2^{'}, v, P_1^{''}, v_e)$$

W' visits all vertices of W, but does not use the edge $e = \{v_e, w_e\}$ anymore and thus, it is shorter. See a visual of this proof in Figure 9. ☐

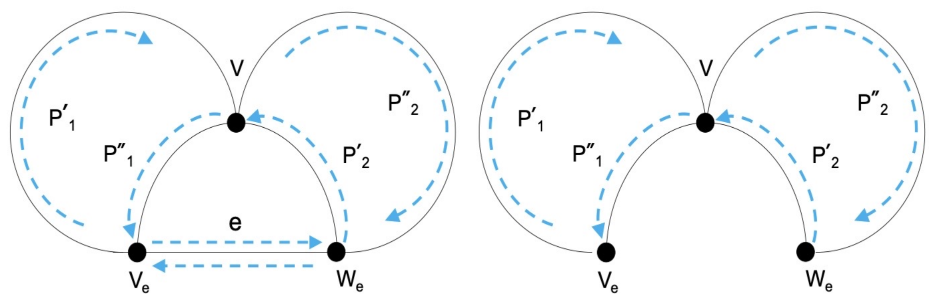


Figure 9: *No alternating edges*

As a corollary we now have the following Theorem.

Theorem 4.3. *Consider an optimal walk W in $G_L = (V_L, E_L)$ that uses the pick location v_1 twice. If there is a vertex v, also visited twice, then these locations are not visited in an alternating way, i.e., $W = (v_1, P_1, v, P_2, v_1, P_3, v, P_4, v_1)$ is not possible.*

Note that P_1 and P_3 denote paths from v_1 to v and P_2 and P_4 are paths from v to v_1.

The proof of this theorem leans on the fact that the pick location vertices have degree 2 in G_L. This applies for any practical lay-out, due to the fact that the pick location vertices lie on a path representing an aisle between the two corner vertices of the aisle.

Proof. If vertex v_1 is visited twice, then both incident edges must be used twice (each in opposite direction): If only one incident edge is visited twice, or both edges are visited once, then v_1 is visited only once.

Consider one of the two edges, say $e_1 = \{v_1, v_2\}$. Now, the optimal walk W can be described as $W = (v_1, P_1, v_1, e_1, v_2, P_2, v_2, e_1^{-1}, v_1)$. According to lemma 4.2, the paths P_1 and P_2 have no vertex in common. Thus, a vertex $v \neq v_1, v_2$ that occurs twice in W, does so either twice in P_1 or twice in P_2. □

This brings us to the conclusion that all vertices in the optimal walk that are visited twice are not alternating. For a specific vertex, say v, this means that, in the walk (v, P_1, v, P_2, v) the occurrence of the other vertices visited twice are either both in P_1 or both in P_2. Thus, the occurrence of the vertices has a nested structure. See the visualization in Figure 10. In this figure, the circle represents the complete walk W. Each piece between two connected vertices on the circle represents a subpath of W, with only vertices that are visited exactly once by W.

In case a vertex v has degree higher than 2, theorem 4.3 does not apply, since the theorem uses that if a node is visited twice then an edge is visited twice, and this need not be true when a vertex has degree higher than 2.

However, for a node with degree higher than two, we can take one of its incident edges and move it up that edge a small amount ϵ. Then, we can apply the previous theorem again

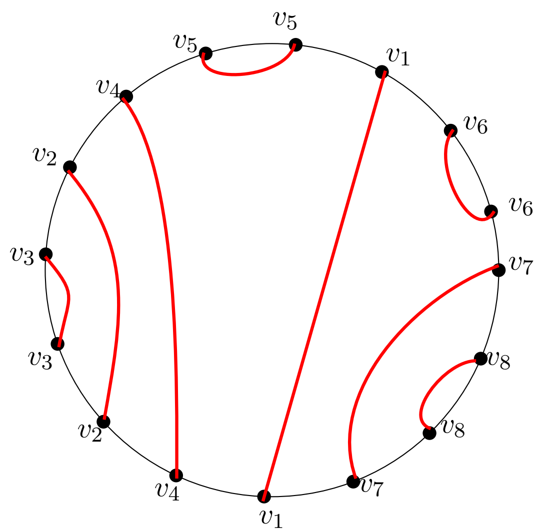


Figure 10: *No alternating vertices.*

(see figure 11), as all vertices of G_{PL} have degree 2 again in G_L. This may not be useful to the picker routing problem that we treat, but it is helpful for the TSP problem in general planar graphs.

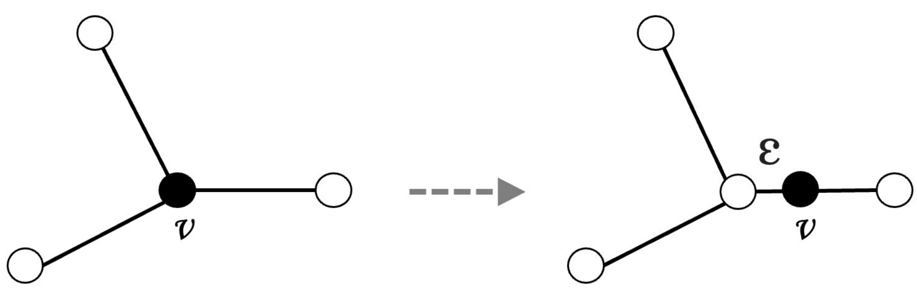


Figure 11: *From degree > 2 to degree = 2*

4.3 From an optimal walk in G_L to an optimal TSP tour in G_{PL}

Figure 10 illustrates an example of how pick locations appearing twice could be on the optimal walk W in G_L. Each connection between two such vertices is a path in G_L with possibly some vertices that occur exactly once. We shall transform the walk W into a tour on the pick locations only, e.g., the vertices in G_{PL}. In doing so we only use a limited set of edges in G_{PL}.

We start with removing the vertices of W that are not pick locations. Every sub-path that begins and ends with a pick location — with no other intermediate pick locations — is replaced by a single edge. This results in a walk W' in G_{PL}: the vertices are only pick locations, and the edges connect two pick locations. This walk is optimal as the "contracted" subpaths are shortest paths in G_L with the same length.

The pick locations that occur twice in W also appear twice in the new walk W'. Our next step is to remove one of the two occurrences of these vertices.

First, we number the occurrences of these doubly occurring vertices in the order that we visit them on the walk W'. See figure 12. Now note that due to theorem 4.3, the two occurrences have different parity: one has an odd number, the other an even number. Next, each occurrence of a vertex with an even number is removed by connecting the neighbor pick locations (independent of whether these occur once or twice in the walk) with a direct edge. Note that this edge is present in G_{PL}. Now, the new walk contains only edges of E_{PL}. Moreover, the edges that we use have at most one pick location on the path in G_L. Finally, each pick location is now visited exactly once. Thus, not only is the walk a tour in G_{PL}, but it only uses edges on which at most one intermediate pick location is present. Thus, in solving the TSP problem of G_{PL}, we can restrict ourselves to using such edges only.

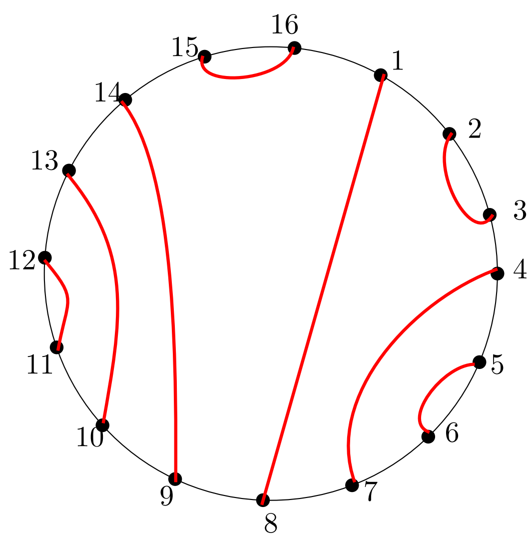
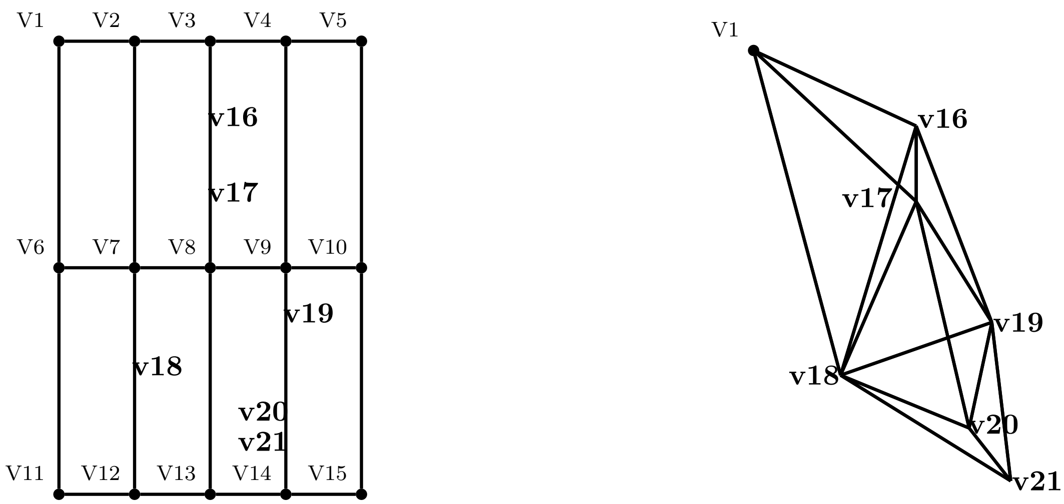


Figure 12: *No alternating vertices.*

In figure 13, we translated the lay-out graph G_L to the graph G_{PL} with only the edges needed to find an optimal TSP tour. The figure illustrates how the graph G_{PL} is reduced to a graph with a much smaller amount of edges than in the complete graph G_{PL}. The depot node V1, for example, is not connected to the product locations x_4, x_5 and x_6, since there exist more than one pick locations on the possible shortest paths between them.

5 Mathematical formulation

We now formulate the proposed exact picker routing model described above. In our algorithm, we are looking for the shortest tour in G_{PL} such that the order picker visits all the ordered



(a) *G_L warehouse layout.* ***(b)*** *G_{PL} with reduced set of edges.*

Figure 13: *Graph reduction example*

product locations only once, starting and ending from the depot. The problem is formulated as a traveling salesman problem (TSP). When solving the model, we find the optimal pick tour regardless of the warehouse layout or the number of items to be picked in each aisle.

Remark: Common practice is that nodes in mathematical formulations are described with indices i and j, rather than the descriptions v and w. In the sequel we shall do that as well. Nodes 0 and $n+1$ are considered to be the visiting order of the depot in the beginning and the end of the tour.

Table 1: *Notation for TSP model.*

Parameters	
d_{ij}	Distance between nodes indexed i and j.
Decision variables	
x_{ij}	Binary variable, 1 if arc (i,j) is traversed by the picker; 0 otherwise.
u_i	Position of node i in the pick tour.

The mathematical model for the general TSP problem is as follows.

$$\min \sum_{i,j} d_{i,j} x_{i,j} \tag{1}$$

$$s.t.$$

$$\sum_{i:i\neq j} x_{ij} = 1 \qquad \forall j \in V_{PL} \tag{2}$$

$$\sum_{j:j\neq i} x_{ij} = 1 \qquad \forall i \in V_{PL} \tag{3}$$

$$u_i - u_j + n * x_{ij} \leq n - 1 \qquad \forall i, j \in V_{PL} \tag{4}$$

$$u_0 = 0 \tag{5}$$

$$u_{n+1} = n + 1 \tag{6}$$

$$u \in \mathbb{Z}, x_{ij} \in \{0, 1\} \qquad \forall i, j \in V_{PL} \tag{7}$$

The objective function (1) minimizes the total distance travelled by the picker. Constraints (2) and (3) force the order pickers to enter and exit every location exactly once, respectively. Constraints (4) eliminates sub-tours by allocating an index to each visited location. In constraints (5) and (6) the depot is initiated as the start and the end of the tour, respectively. Finally, constraints (7) set the domain of the decision variables.

6 Model implementation and numerical results

To test the performance of our graph reduction approach, we perform a set of computation experiments on two different benchmark, single block instances from Scholz et al. [43] and multi-block instances from Theys et al. [45].

The mathematical model presented in this paper is implemented in GAMS 24.2 and solved using IBM CPLEX 12.9. All experiments are run on a Macbook Air with an Apple Silicon M1 chip and 16GB of RAM. The computing time for each instance is limited to 30 minutes.

6.1 Single block: comparison with the instances presented in Scholz et al. [43]

The numerical results using the first benchmark instances from Scholz et al. [43] containing only a single block (two cross aisles) are given in table 2. In this table, the two parameters given as m and n, are the number of aisles and the number of items in the order list respectively. We solved each instance 10 times and put the average results in the table 2. The numerical results of from Scholz et al. [43] are presented in the last column.

As shown in this table, the average graph reduction in comparison to the TSP model (complete graph without reduction) is 72.85 percent, which is significant. Furthermore, for all

Table 2: Numerical results of proposed model on different instances

m-n	Time (sec)	VarNum	ConNum	Initial edges	Reduced edges	Reduction(%)	Shcolz(sec)
5-30	0.031	68	93	435	21	95	0.09
5-45	0.046	76	101	990	29	97	0.09
5-60	0.063	84	109	1770	41	97	0.09
5-75	0.081	92	116	3492	50	98	0.09
5-90	0.109	100	125	4005	64	98	0.10
10-30	0.047	256	301	435	136	68	1.60
10-45	0.110	274	319	990	145	85	1.03
10-60	0.125	292	337	1770	191	89	1.42
10-75	0.281	360	413	2775	213	92	1.36
10-90	0.906	376	423	4005	220	94	0.62
15-30	0.479	456	504	435	167	61	2.29
15-45	1.187	576	631	990	247	75	5.28
15-60	2.297	620	685	1770	346	80	10.64
15-75	2.337	698	782	1914	391	79	15.10
15-90	2.508	730	813	2106	438	79	19.41
20-30	1.225	438	484	435	315	27	10.57
20-45	1.391	812	878	990	349	64	27.32
20-60	1.651	1016	1091	1770	476	73	114.33
20-75	1.981	1030	1108	2775	495	82	216.63
20-90	3.023	1046	1129	4005	679	83	485.71
25-30	0.911	618	672	435	206	52	54.46
25-45	2.160	1090	1178	990	377	62	85.46
25-60	2.425	1270	1353	1770	588	66	258.92
25-75	3.063	1346	1437	2775	703	74	527.39
25-90	4.469	2106	2213	4005	939	76	646.59
30-30	3.234	626	681	435	348	20	204.18
30-45	5.547	1176	1250	990	511	48	406.19
30-60	6.328	1906	2007	1770	627	64	508.80
30-75	7.042	2409	2521	2775	833	70	638.89
30-90	7.601	2862	2981	4005	997	75	786.29

of the instances, our model gives the optimal solution almost instantaneously. The maximum computation time for the biggest instance (with 30 aisles and 90 products) is 7.6 seconds, which is considerably less compared to 786.29 seconds by Scholz et al. [43].

Scholz et al. [43] are not able to solve the instances with more than 3 blocks. Also for 10% of the instances with only 3 blocks, no optimal solutions were reported. However, we will show in the next section that our model with the help of the pre-processing phase is able to solve much larger instances in a very short time.

6.2 Multi-block: Comparison with the instances presented in Pansart, Catusse, and Cambazard [33]

The second benchmark is from Pansart, Catusse, and Cambazard [33] and consists of 9 different scenarios: three different numbers of aisles (5,15,60), three different numbers of cross-aisles (3,6,11) and three different numbers of products in the order (15,60,240). We solve these instances on the multi-block warehouse. As it is shown in the table 3, our method is able to solve all the multi-block instances optimally in less than one minute.

Table 4 illustrates the number of unsolved instances within the 30 minutes time limit both for Pansart, Catusse, and Cambazard [33] and our proposed method. In this table,

Table 3: *Numerical results of the proposed model on different instances*

Aisles	**Cross aisles**	**Products**	**CPLEX Time(sec)**
5	3	15	0.42
5	3	60	0.89
5	3	240	1.62
5	6	15	0.60
5	6	60	1.43
5	6	240	3.13
5	11	15	1.02
5	11	60	4.30
5	11	240	8.65
15	3	15	1.42
15	3	60	4.07
15	3	240	11.05
15	6	15	4.32
15	6	60	17.64
15	6	240	30.77
15	11	15	17.81
15	11	60	43.68
15	11	240	48.98
60	3	15	1.23
60	3	60	6.33
60	3	240	14.09
60	6	15	5.32
60	6	60	21.52
60	6	240	34.08
60	11	15	22.13
60	11	60	50.25
60	11	240	59.69

SCFS+ stands for the standard single commodity flow formulation with pre-processing and the additional valid inequalities and PDYN refers to a dynamic program, both proposed by Pansart, Catusse, and Cambazard [33].

As it is shown in this table, Pansart, Catusse, and Cambazard [33] are not able to solve larger instances with 240 products or more than 6 cross-aisles (15-11-240, 5-11-240, 60-11-240, 60-3-240, 60-6-240). Furthermore, their proposed dynamic program is very efficient for the small instances, however, it is not scalable and it becomes intractable for instances with more than 6 cross-aisles. Our model, however, is able to solve all the instances to optimality within the time limit.

Moreover, it is shown that the average running time for the largest instance solved in less than 30 minutes for them is 111.64 seconds which is 4.7 times larger than our results. Furthermore, the average computation time for our model is 15.34 seconds for all the instances which is 58% lower than their proposed single commodity flow formulation. It is worth mentioning that the lower total average time for the dynamic program model for Pansart, Catusse, and Cambazard [33] is due to the fact that this model is only able to solve the smaller instances.

Table 4: *Number of unsolved instances after 30 minutes. Benchmark against Pansart, Catusse, and Cambazard [33]*

	Total	# aisles			# cross-aisles			# products		
		5	15	60	3	6	11	15	60	240
SCFS+	18	1	4	13	1	2	15	0	0	18
PDYN	90	30	30	30	0	0	90	30	30	30
Our method	0	0	0	0	0	0	0	0	0	0
# instances	270	90	90	90	90	90	90	90	90	90

Table 5: *Average computation time (seconds) for instances solved in 30 minutes. Benchmark against Pansart, Catusse, and Cambazard [33].*

	Total	# aisles			# cross-aisles			# products		
		5	15	60	3	6	11	15	60	240
SCFS+	36.07	23.89	56.96	27.12	3.44	35.88	71.69	0.07	4.05	111.64
PDYN	0.27	0.05	0.16	0.61	0	0.54	-	0.24	0.27	0.30
Our method	15.34	2.34	19.82	23.84	4.56	13.20	28.50	6.03	16.67	23.56

7 Concluding Remarks

This study aimed to develop an efficient algorithm for the planar TSP, specifically for the order pickers routing problem in a multi-block warehouse layout. While exact algorithms for small warehouses exist in the literature, solution algorithms for larger warehouse layouts often rely on (meta)heuristic approaches.

The proposed algorithm utilizes graph reduction to eliminate unnecessary vertices and edges from the graph (The network size reduction achieved by our algorithm was, on average, 72.85%), resulting in a significant reduction in computation time. This approach is applicable to any warehouse layout presented as a planar graph, making it practical for real-world applications.

To solve the routing problem, a general TSP model was used. The algorithm was implemented on various problem instances from the literature, and its performance was compared with existing methods. The results showed that the proposed algorithm outperformed existing formulations in terms of simplicity, size, and calculation time. Overall, the results indicate that the proposed algorithm is a promising approach for solving the order pickers routing problem in multi-block warehouse layouts.

References

[1] Ehsan Ardjmand, Omid Sanei Bajgiran, and Eyad Youssef. "Using list-based simulated annealing and genetic algorithm for order batching and picker routing in put wall based picking systems". In: *Applied Soft Computing* 75 (2019), pp. 106–119.

[2] Sanjeev Arora. "Approximation schemes for NP-hard geometric optimization problems: A survey". In: *Mathematical Programming* 97 (July 2003), pp. 43–69. DOI: 10.1007/s10107-003-0438-y.

[3] Amir Hossein Azadnia et al. "Order batching in warehouses by minimizing total tardiness: a hybrid approach of weighted association rule mining and genetic algorithms". In: *The Scientific World Journal* 2013 (2013).

[4] Tamás Bódis and János Botzheim. "Bacterial memetic algorithms for order picking routing problem with loading constraints". In: *Expert Systems with Applications* 105 (2018), pp. 196–220.

[5] Hadrien Cambazard and Nicolas Catusse. "Fixed-parameter algorithms for rectilinear steiner tree and rectilinear traveling salesman problem in the plane". In: *European Journal of Operational Research* 270.2 (2018), pp. 419–429.

[6] Jose Alejandro Cano, Alexander Alberto Correa-Espinal, and Rodrigo Andrés Gómez-Montoya. "An evaluation of picking routing policies to improve warehouse efficiency". In: *International Journal of Industrial Engineering and Management* 8.4 (2017), pp. 229–238.

[7] Angelo Castelda. *Understanding The Impacts of eCommerce On Warehouse Operations.* https://www.floship.com/blog/_ecommerce-warehouse-operations/. Accessed: 2020-05-27.

[8] Melih Celik and Haldun Sural. "The order picking problem in fishbone aisle warehouses". In: (2012).

[9] Melih Çelik and H Süral. "Order picking in a parallel-aisle warehouse with turn penalties". In: *International Journal of Production Research* 54.14 (2016), pp. 4340–4355.

[10] Thomas Chabot et al. "Order picking problems under weight, fragility and category constraints". In: *International Journal of Production Research* 55.21 (2017), pp. 6361–6379.

[11] Fangyu Chen et al. "An ACO-based online routing method for multiple order pickers with congestion consideration in warehouse". In: *Journal of Intelligent Manufacturing* 27.2 (2016), pp. 389–408.

[12] Gérard Cornuéjols, Jean Fonlupt, and Denis Naddef. "The traveling salesman problem on a graph and some related integer polyhedra". In: *Mathematical programming* 33.1 (1985), pp. 1–27.

[13] Pablo Cortés et al. "A tabu search approach to solving the picking routing problem for large-and medium-size distribution centres considering the availability of inventory and K heterogeneous material handling equipment". In: *Applied Soft Computing* 53 (2017), pp. 61–73.

[14] René De Koster and Edo Van Der Poort. "Routing orderpickers in a warehouse: a comparison between optimal and heuristic solutions". In: *IIE transactions* 30.5 (1998), pp. 469–480.

[15] Roberta De Santis et al. "An adapted ant colony optimization algorithm for the minimization of the travel distance of pickers in manual warehouses". In: *European Journal of Operational Research* 267.1 (2018), pp. 120–137.

[16] Izabela Dembińska. "The Impact of E-Commerce Development on the Warehouse Space Market in Poland". In: *Economics and culture* 13.2 (2016), pp. 5–13.

[17] M Devaraj. *Impact of eCommerce Growth on the Logistics Sector*. https://www.linkedin.com/pulse/impact-ecommerce-growth-logistics-sector-devaraj-m/?trk=pulse-article_more-articles_related-content-card. Accessed: 2022-06-23.

[18] Goran Dukic and Cedomir Oluic. "Order-picking methods: improving order-picking efficiency". In: *International Journal of Logistics Systems and Management* 3.4 (2007), pp. 451–460.

[19] Bernhard Fleischmann. "A cutting plane procedure for the travelling salesman problem on road networks". In: *European Journal of Operational Research* 21.3 (1985), pp. 307–317.

[20] M. R. Garey, R. L. Graham, and D. S. Johnson. "Some NP-Complete Geometric Problems". In: *Proceedings of the Eighth Annual ACM Symposium on Theory of Computing*. STOC '76. Hershey, Pennsylvania, USA: Association for Computing Machinery, 1976, pp. 10–22. ISBN: 9781450374149. DOI: 10.1145/800113.803626. URL: https://doi.org/10.1145/800113.803626.

[21] Christoph H Glock and Eric H Grosse. "Storage policies and order picking strategies in U-shaped order-picking systems with a movable base". In: *International Journal of Production Research* 50.16 (2012), pp. 4344–4357.

[22] Marc Goetschalckx and H Donald Ratliff. "Order picking in an aisle". In: *IIE transactions* 20.1 (1988), pp. 53–62.

[23] Rodrigo Andrés Gómez-Montoya, Alexander Alberto Correa-Espinal, and José Daniel Hernández-Vahos. "Picking Routing Problem with K homogenous material handling equipment for a refrigerated warehouse". In: *Revista Facultad de Ingeniería Universidad de Antioquia* 80 (2016), pp. 9–20.

[24] David B Grant, Chee Yew Wong, and Alexander Trautrims. *Sustainable logistics and supply chain management: principles and practices for sustainable operations and management*. Kogan Page Publishers, 2017.

[25] Beth Gutelius, Nik Theodore, et al. "The future of warehouse work: Technological change in the US logistics industry". In: *UC Berkeley Labor Center* (2019).

[26] Randolph W Hall. "Distance approximations for routing manual pickers in a warehouse". In: *IIE transactions* 25.4 (1993), pp. 76–87.

[27] Y-C Ho and Y-Y Tseng. "A study on order-batching methods of order-picking in a distribution centre with two cross-aisles". In: *International Journal of Production Research* 44.17 (2006), pp. 3391–3417.

[28] Zhang Kai and Gao Chuanhou. "Improved formulations of the joint order batching and picker routing problem". In: *arXiv preprint arXiv:2207.05305* (2022).

[29] Adam N Letchford, Saeideh D Nasiri, and Dirk Oliver Theis. "Compact formulations of the Steiner traveling salesman problem and related problems". In: *European Journal of Operational Research* 228.1 (2013), pp. 83–92.

[30] Jianbin Li, Rihuan Huang, and James B Dai. "Joint optimisation of order batching and picker routing in the online retailer's warehouse in China". In: *International Journal of Production Research* 55.2 (2017), pp. 447–461.

[31] Frauke Liers, Alexander Martin, and Susanne Pape. *Steiner trees with degree constraints: Structural results and an exact solution approach*. Tech. rep. Technical report, Department Mathematik, 2014.

[32] Chun-Cheng Lin et al. "Joint order batching and picker Manhattan routing problem". In: *Computers & Industrial Engineering* 95 (2016), pp. 164–174.

[33] Lucie Pansart, Nicolas Catusse, and Hadrien Cambazard. "Exact algorithms for the order picking problem". In: *Computers & Operations Research* 100 (2018), pp. 117–127.

[34] CG Petersen. "Routeing and storage policy interaction in order picking operations". In: *Decision Sciences Institute Proceedings*. Vol. 3. 1995, pp. 1614–1616.

[35] Charles G Petersen. "An evaluation of order picking routeing policies". In: *International Journal of Operations & Production Management* 17.11 (1997), pp. 1098–1111.

[36] Charles G Petersen and Gerald Aase. "A comparison of picking, storage, and routing policies in manual order picking". In: *International Journal of Production Economics* 92.1 (2004), pp. 11–19.

[37] H Donald Ratliff and Arnon S Rosenthal. "Order-picking in a rectangular warehouse: a solvable case of the traveling salesman problem". In: *Operations Research* 31.3 (1983), pp. 507–521.

[38] Kees Jan Roodbergen. "Storage assignment for order picking in multiple-block warehouses". In: *Warehousing in the global supply chain*. Springer, 2012, pp. 139–155.

[39] Kees Jan Roodbergen and René De Koster. "Routing order pickers in a warehouse with a middle aisle". In: *European Journal of Operational Research* 133.1 (2001), pp. 32–43.

[40] Kees-Jan Roodbergen. *Layout and routing methods for warehouses*. EPS-2001-004-LIS. 2001.

[41] KJ Roodbergen and R De Koster. "Routing order pickers in a warehouse with multiple cross aisles". In: *Progress in Material Handling Research* 1998 (1998), pp. 451–467.

[42] André Scholz and Gerhard Wäscher. "Order batching and picker routing in manual order picking systems: the benefits of integrated routing". In: *Central European Journal of Operations Research* 25.2 (2017), pp. 491–520.

[43] André Scholz et al. "A new mathematical programming formulation for the single-picker routing problem". In: *European Journal of Operational Research* 253.1 (2016), pp. 68–84.

[44] A Immanuel Selvakumar and K Thanushkodi. "A new particle swarm optimization solution to nonconvex economic dispatch problems". In: *IEEE transactions on power systems* 22.1 (2007), pp. 42–51.

[45] Christophe Theys et al. "Using a TSP heuristic for routing order pickers in warehouses". In: *European Journal of Operational Research* 200.3 (2010), pp. 755–763.

[46] James A Tompkins et al. *Facilities planning*. John Wiley & Sons, 2010.

[47] C-Y Tsai, James JH Liou, and T-M Huang. "Using a multiple-GA method to solve the batch picking problem: considering travel distance and order due time". In: *International Journal of Production Research* 46.22 (2008), pp. 6533–6555.

[48] Cristiano Arbex Valle, John E Beasley, and Alexandre Salles da Cunha. "Modelling and solving the joint order batching and picker routing problem in inventories". In: *International symposium on combinatorial optimization*. Springer. 2016, pp. 81–97.

[49] TS Vaughan. "The effect of warehouse cross aisles on order picking efficiency". In: *International Journal of Production Research* 37.4 (1999), pp. 881–897.

[50] Felix Weidinger, Nils Boysen, and Michael Schneider. "Picker routing in the mixed-shelves warehouses of e-commerce retailers". In: *European Journal of Operational Research* 274.2 (2019), pp. 501–515.

[51] Li Zhou et al. "Performance Analysis of Three Intelligent Algorithms on Route Selection of Fishbone Layout". In: *Sustainability* 11.4 (2019), p. 1148.