WORKING PAPER SERIES

A New Mathematical Programming Formulation for the Single-Picker Routing Problem in a Single-Block Layout

Sebastian Henn/André Scholz/Meike Stuhlmann/Gerhard Wäscher

Working Paper No. 5/2015

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

FACULTY OF ECONOMICS AND MANAGEMENT

Impressum (§ 5 TMG)

Herausgeber:
Otto-von-Guericke-Universität Magdeburg
Fakultät für Wirtschaftswissenschaft
Der Dekan

Verantwortlich für diese Ausgabe:
S. Henn/A. Scholz/M. Stuhlmann/G. Wäscher
Otto-von-Guericke-Universität Magdeburg
Fakultät für Wirtschaftswissenschaft
Postfach 4120
39016 Magdeburg
Germany

http://www.fww.ovgu.de/femm

Bezug über den Herausgeber
ISSN 1615-4274

A New Mathematical Programming Formulation for the Single-Picker Routing Problem in a Single-Block Layout

S. Henn, A. Scholz, M. Stuhlmann, G. Wäscher

Abstract

The Single-Picker Routing Problem deals with the determination of sequences according to which items have to be picked in a distribution warehouse and the identification of the corresponding paths which have to be travelled by human operators (order pickers). The Single-Picker Routing Problem represents a special case of the classic Traveling Salesman Problem (TSP) and, therefore, can also be modeled as a TSP. However, the picking area of a warehouse typically possesses a block layout, i.e. the items are located in parallel picking aisles, and the order pickers can only change over to another picking aisle at certain positions by means of so-called cross aisles. In this paper, for the first time a mathematical programming formulation is proposed which takes into account this specific property. Based on extensive numerical experiments, it is shown that the proposed formulation is superior to standard TSP formulations.

Keywords: Order Picking, Picker Routing, Traveling Salesman Problem

Corresponding author:
André Scholz
Otto-von-Guericke University Magdeburg, Faculty of Economics and Management
Postbox 4120, 39016 Magdeburg, Germany
Phone: +49 391 67 11841
Fax: +49 391 67 18223
Email: andre.scholz@ovgu.de

1 Introduction

The Traveling Salesman Problem (TSP) is one of the most extensively studied problems in combinatorial optimization (Rego et al., 2011). It can be described as the problem of finding a least-weight Hamiltonian cycle in a complete edge-weighted graph (Glover & Punnen, 1997). Since the TSP is a NP-hard problem, there is no known exact polynomial-time algorithm. However, over a thousand publications exist dealing with model formulations and algorithms which solve the TSP with less computational effort (Rego et al., 2011).

This large number of publications can be explained by the fact that the TSP arises in many different contexts. In the basic situation, a starting point is given from which a product has to be distributed to other places in such a way that every place is visited and the distance to be travelled is minimal. Beyond this, the TSP is of prime importance for practical applications in engineering, management, health care and many other areas. For detailed reviews of applications of the TSP in practice we refer to Lenstra & Rinnooy Kan (1975), Matai et al. (2010) and Filip & Otakar (2011).

In this paper, we will focus on an application of the TSP that arises in distribution warehouses. Order picking deals with the retrieval of items from their storage location in such warehouses (Petersen & Schmenner, 1999; Wäscher, 2004). Human operators (order pickers) travel through the warehouse in order to collect items requested by customers. The respective Single-Picker Routing Problem (SPRP) includes the determination of the sequence according to which the items have to be picked and represents a special case of the TSP, too.

For the TSP several mathematical programming formulations have been proposed in the literature. These formulations seem not to be appropriate for the SPRP since they ignore the special structure of the latter. The aim of this paper is to provide a new mathematical programming formulation which takes into account the properties of optimal solutions for the SPRP and results in a substantial reduction of the number of variables and constraints. It is shown that the size of the formulation is independent of the number of items to be picked. Furthermore, numerical experiments demonstrate that the proposed formulation is superior to standard TSP formulations with respect to the computing time needed to solve the problems.

The remainder of this paper is organized as follows: In the following section we introduce the SPRP and review the related literature. In Section 3, we present some well known general mathematical programming formulations for the TSP. As central part of this paper, our new model formulation is introduced in Section 4. First, the construction of a graph representing the SPRP is described and after that the model is presented. In order to analyze the different formulations, we solved them with a commercial IP solver. The results of the experiments are depicted in Section 5. The paper concludes with a summary and an outlook on future research fields.

2 Picker Routing Problem

2.1 Problem description

In distribution warehouses typically a block layout is used (Roodbergen, 2001). It consists of a number of (vertical) picking aisles arranged parallel to each other and (horizontal) cross aisles which can be used to move from one picking aisle to another. The section between two adjacent cross aisles establishes a so-called block. Items are stored in and picked from racks on both sides of these picking aisles. Cross aisles do not contain any storage locations. Furthermore, the warehouse contains a depot where picked items are deposited. In the following, we will focus on a single-block layout, i.e. only two cross aisles exist, one in the front and one in the rear of the warehouse. (see Fig. 1).

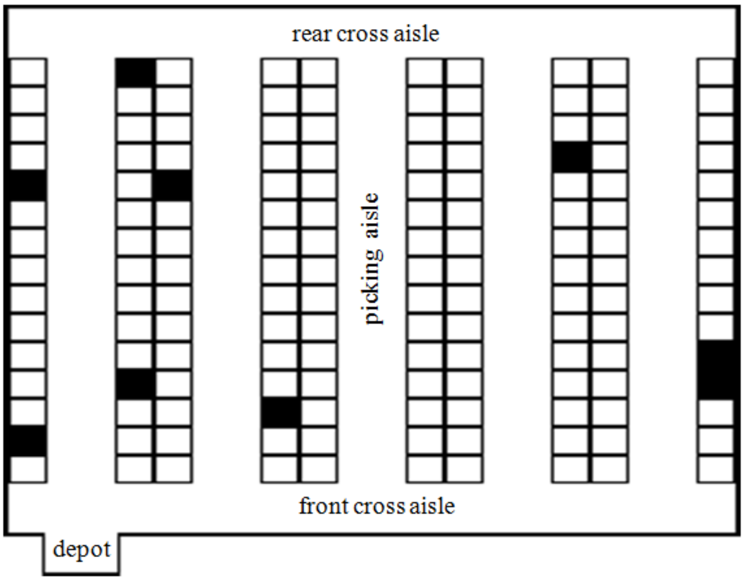

Fig. 1: Single-block layout

Let an order picker be available to collect items from the warehouse which are requested by (external or internal) customers. The black rectangles in Fig. 1 give an example for the locations from which items have to be collected (pick locations). The order picker starts at the depot, proceeds to the pick locations, and finally returns to the depot.

Due to the high proportion of time-consuming manual operations, order picking is considered as the most labor cost-intensive function in a warehouse (Tompkins et al., 2010). Consequently, the minimization of picking times is of vital importance for the efficient control of the picking process. The total order picking time (i.e. the time spent by an order picker to collect the items of a picking order) consists of the setup time for the route, the travel time needed to travel to, from, and between the locations of items to be picked, the search times for the identification of the items, and the times actually needed for retrieving the items (Tompkins et al., 2010). Among these components, the travel time consumes the major proportion of the total order picking time. While the other components can be looked upon to be constants (search times, pick times, setup times), the travel time represents the only variable part. It varies with the total length of the picker tour (Jarvis & McDowell, 1991) which, again, is dependent on the sequence according to which the items have to be picked. If we assume a constant picker travel velocity, the minimization of the total length of the picker tour becomes equivalent to the minimization of the travel time. Therefore, the SPRP can be defined as follows: Given a set of items to be picked from known storage locations, in which sequence should the locations be visited such that the total length of the corresponding picker tour is minimized?

2.2 Literature review

The SPRP can easily be recognized as a TSP in which the vertices of the corresponding graph are defined by the location of the depot and the locations of the items to be picked. Therefore, exact solution approaches as the solution of mathematical programming formulations for the TSP can be used to solve the SPRP. A standard TSP formulation was introduced by Dantzig et al. (1954) which includes one binary variable per edge indicating whether an edge is contained in the tour. However, this formulation requires an exponential number of constraints. Beyond this approach, also a variety of compact formulations exist which are characterized by requiring only a polynomial number of variables and constraints. In this paper, we consider three compact formulations for the TSP which are the formulations by Miller et al. (1960), Gavish & Graves (1978) and Claus (1984). These formulations will be explained in greater detail in Section 3. We further refer to Öncan et al. (2009) for a more general review of TSP formulations.

A more specific approach for the solution of the SPRP was proposed by Burkard et al. (1998). They formulated the SPRP as a Steiner TSP, which is defined as follows: Let $G = (V, E)$ be a graph with a set of vertices V and a set of edges E. Let P be a subset of V. The elements of $V \setminus P$ are called Steiner points. A Steiner tour is defined as a closed walk in which each vertex of P is visited at least once. The Steiner points do not have to be visited. Also, a Steiner tour may contain some vertices and edges more than once. In total, the Steiner TSP consists of finding a Steiner Tour with minimal length (Burkard et al., 1998).

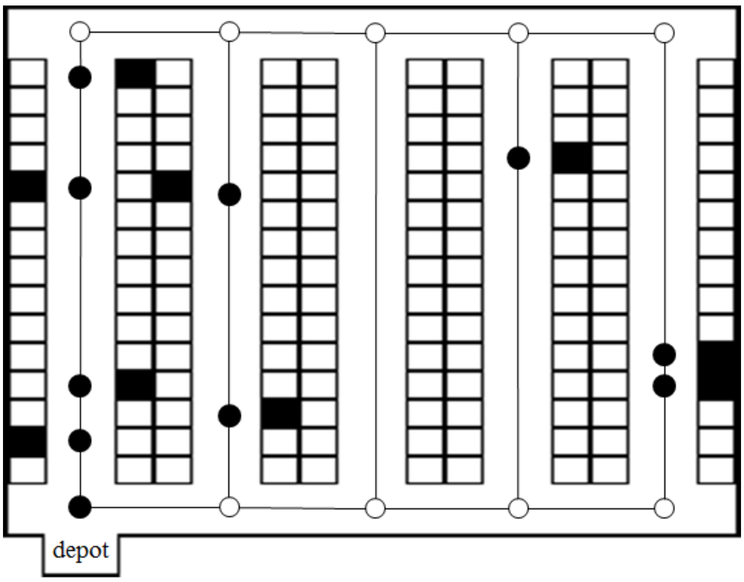

Fig. 2: Illustration of a Steiner TSP

As depicted in Fig. 2, when dealing with the SPRP, the set P is composed of the pick locations and the depot (black vertices), and the Steiner points are the intersections between the picking aisles and the cross aisle (white vertices). The distance between any pair of vertices is the length of the shortest path between the two vertices. By definition, a Steiner Tour then has to include the depot and the locations of all items to be picked, where visiting these locations more than once is permitted. This may occur, e.g., when a picking aisle is entered and left via the same cross aisle. In comparison to that, some of the white vertices may be skipped because two points for each picking aisle exist where an aisle can be entered or left and it is possible to use one vertex twice and skip the other one. Lechtford et al. (2013) developed some compact formulations for the Steiner TSP. These formulations are advantageous in comparison to general TSP formulations if the number of Steiner points is large compared to the total number of vertices (Lechtford et al., 2013). In case of the SPRP, the number of Steiner points is only dependent on the number of picking aisles and remains constant if the number of items to be picked is varied. Assuming the number of picking aisles to be constant, a large percentage of Steiner points implies a small number of items to be picked. However, in case of a small number of requested items, the corresponding general TSP formulation can be solved easily. Therefore, Steiner TSP formulations are not considered any further in this paper.

Under the assumption that the warehouse has a single- or a two-block layout, the SPRP can be solved efficiently (Ratliff & Rosenthal, 1983; Roodbergen & de Koster, 2001a). Ratliff & Rosenthal (1983) proposed an optimal algorithm for the SPRP with a single-block layout based on dynamic programming which solves the problem in $O(m+n)$ time, where m is the number of picking aisles and n is the number of requested items. However, the authors do not present any model formulation.

Since the model formulation presented in this paper will contain some elements of heuristic routing schemes, we also give a short overview of heuristic approaches for solving the SPRP. Heuristic routing schemes are fast to memorize and quite easy to follow and, therefore, prevalent in practice in order to solve the SPRP (Roodbergen, 2001). Their application helps to reduce the risk of missing an item to be picked. The simplest routing heuristics are the s-shape, the return and the largest gap strategies which are also described in de Koster et al. (2007) and Gu et al. (2007). The S-shape heuristic provides solutions in which the order picker enters and traverses a picking aisle completely if at least one required item is located in that aisle (an exception would be the last picking aisle in which an item has to be picked if the order picker is positioned on the front cross aisle). Afterwards, the order picker moves to the next aisle to be visited. The application of the return heuristic rises to a tour in which each picking aisle is entered from the front cross aisle. For each picking aisle, the picker walks to the farthest pick location and returns to the front cross aisle after picking the items. The largest gap heuristic gives a solution in which the order picker completely traverses the first and the last aisle containing a demanded item. All other aisles containing at least one required item are entered from the front and from the rear cross aisle in a way that the non-traversed distance between two adjacent pick locations or a pick location and the adjacent end of the aisle is maximal. The combined strategy (Roodbergen & de Koster, 2001b) integrates elements of the S-shape and return strategy. Aisles may be traversed entirely or may be entered and left from the same cross aisle. The respective solutions are provided by application of dynamic programming. The performance of the proposed heuristics is dependent on the problem characteristics (number of picking aisles, number of locations per aisle, position of the depot, number of requested items). Moreover, the policy according to which items are assigned to different locations has a significant impact on the tour lengths provided by the heuristics.

3 General mathematical model formulations for the TSP

Typically, the TSP is modeled on a complete graph $G^{TSP} = (V, E)$. In case of the SPRP, the set of vertices $V = \{0, \dots, n\}$ contains the depot (vertex 0) and all locations where an item has to be picked. The set of edges is defined as $E = \{(p,q) \mid p,q \in V, p \neq q\}$ and for each edge a distance c_{pq} is calculated. For the TSP a variety of mathematical formulations has been proposed so far (Öncan et al., 2009). In order to be able to solve the formulations without using cutting-plane methods, we focus on formulations with a polynomial number of both variables and constraints. The three following formulations have also been chosen by Lechtford et al. (2013) and differ in two characteristics which are the number of variables and constraints and the quality of the lower bound obtained by solving the LP relaxation.

3.1 Formulation by Miller, Tucker and Zemlin

A classic way to model the TSP has been proposed by Miller et al. (1960). This formulation uses the following variables:

$$x_{pq} = \begin{cases} 1, & \text{if edge } (p,q) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad ((p,q) \in E)$$

h_p: position of vertex p in the tour $(p \in V \backslash \{0\})$

Now, the TSP can be modeled as follows:

$$\min \sum_{(p,q) \in E} c_{pq} \cdot x_{pq} \tag{1}$$

$$\sum_{p \in V} x_{pq} = 1 \quad \forall q \in V \tag{2}$$

$$\sum_{q \in V} x_{pq} = 1 \quad \forall p \in V \tag{3}$$

$$h_p - h_q + (n+1)\, x_{pq} \leq n \quad \forall (p,q) \in E\colon\ p,q \neq 0 \tag{4}$$

$$x_{pq} \in \{0,1\} \quad \forall (p,q) \in E \tag{5}$$

$$h_p \geq 0 \quad \forall p \in V \backslash \{0\} \tag{6}$$

The objective function minimizes the travel distance. In (2) and (3) it is guaranteed that each vertex has to be visited exactly once. Furthermore, the conditions (4) exclude subcylces by ensuring that the position of vertex p is smaller than the position of vertex q if edge (p,q) is used. This formulation requires only $O(n^2)$ variables and constraints. However, the solution of its LP relaxation results in an extremely weak lower bound (Padberg & Sung, 1991).

3.2 Formulation by Gavish and Graves

A formulation based on single flow commodities has been introduced by Gavish & Graves (1978). Starting at the depot with n units of a commodity, a single unit of this commodity is delivered when a vertex is passed. For this model additional non-negative variables are introduced describing the flow on edge $(p,q) \in E$:

g_{pq}: amount of the commodity passing directly from vertex p to q $((p,q) \in E\colon\, q \neq 0)$

This results in the following model formulation:

$$\min \sum_{(p,q)\in E} c_{pq} \cdot x_{pq} \tag{7}$$

$$\sum_{p\in V} x_{pq} = 1 \quad \forall q \in V \tag{8}$$

$$\sum_{q\in V} x_{pq} = 1 \quad \forall p \in V \tag{9}$$

$$\sum_{q\in V} g_{qp} - \sum_{q\in V\setminus\{0\}} g_{pq} = 1 \quad \forall p \in V\setminus\{0\} \tag{10}$$

$$g_{pq} \leq n x_{pq} \quad \forall (p,q) \in E\colon\, q \neq 0 \tag{11}$$

$$x_{pq} \in \{0,1\} \quad \forall (p,q) \in E \tag{12}$$

$$g_{pq} \geq 0 \quad \forall (p,q) \in E\colon\, q \neq 0 \tag{13}$$

Constraints (10) ensure that exactly one unit of the commodity is delivered to each vertex $p \in V\setminus\{0\}$ while constraints (11) guarantee the flow to be zero along edges not included in the tour. The formulation by Gavish & Graves (1978) includes $O(n^2)$ variables and constraints and Padberg & Sung (1991) have shown that it leads to a stronger bound than the formulation by Miller et al. (1960).

3.3 Formulation by Claus

A further formulation, which uses multi-commodity flows in order to prohibit subcycles, was introduced by Claus (1984). Here, n commodities have to be delivered, one unit of a commodity to each customer, which results in the following additional variables:

w^k_{pq}: amount of commodity k passing directly from vertex p to q $((p,q) \in E,\, k \in V\setminus\{0\})$

The formulation for the TSP is as follows:

$$\min \sum_{(p,q)\in E} c_{pq} \cdot x_{pq} \tag{14}$$

$$\sum_{p\in V} x_{pq} = 1 \quad \forall q \in V \tag{15}$$

$$\sum_{q\in V} x_{pq} = 1 \quad \forall p \in V \tag{16}$$

$$\sum_{q\in V} w_{pq}^k - \sum_{q\in V} w_{qp}^k = 0 \quad \forall p, k \in V\backslash\{0\}\colon p \neq k \tag{17}$$

$$\sum_{q\in V\backslash\{0\}} w_{1q}^k - \sum_{q\in V\backslash\{0\}} w_{q1}^k = -1 \quad \forall k \in V\backslash\{0\} \tag{18}$$

$$\sum_{q\in V} w_{pq}^p - \sum_{q\in V} w_{qp}^p = 1 \quad \forall p \in V\backslash\{0\} \tag{19}$$

$$w_{pq}^k \leq x_{pq} \quad \forall (p,q) \in E,\, k \in V\backslash\{0\} \tag{20}$$

$$x_{pq} \in \{0,1\} \quad \forall (p,q) \in E \tag{21}$$

$$w_{pq}^k \geq 0 \quad \forall (p,q) \in E,\, k \in V\backslash\{0\} \tag{22}$$

Constraints (17) ensure that a commodity leaves a vertex which is not its final destination. In (18), it is guaranteed that each commodity leaves the depot and is delivered to a vertex and (19) ensure that each vertex gets exactly one commodity. $O\left(n^3\right)$ variables and constraints are used in this model formulation. The solution of its LP relaxation leads to the strongest lower bound of the three formulations considered here (Padberg & Sung, 1991).

Each of these three formulations can be used to model and solve the SPRP. However, the number of variables and constraints and, therefore, the computing time needed to solve the problem get quite large when the number of requested items increases.

4 An improved formulation for the SPRP

In routing problems encountered in an order picking warehouse, routes have a special structure which results from the fact that a cross aisle has to be used to move from one picking aisle to another. When considering optimal routes, movements within picking aisles are also quite restricted. Both properties are not considered by general TSP formulations.

In this section, we introduce a graph-theoretical formulation that takes into account these properties and we show that the respective number of vertices and edges is not dependent on the number or the locations of the requested items. Then, a TSP formulation is applied to this graph in order to model the SPRP.

4.1 Graph construction

When dealing with the SPRP in a single-block layout, it is not necessary to consider a complete graph with edges between each pair of vertices because only a few combinations of edges can be included in an optimal tour. Ratliff & Rosenthal (1983) have shown that only six different ways exist how items can be picked in a picking aisle. They are depicted in Fig. 3 and will be explained below.

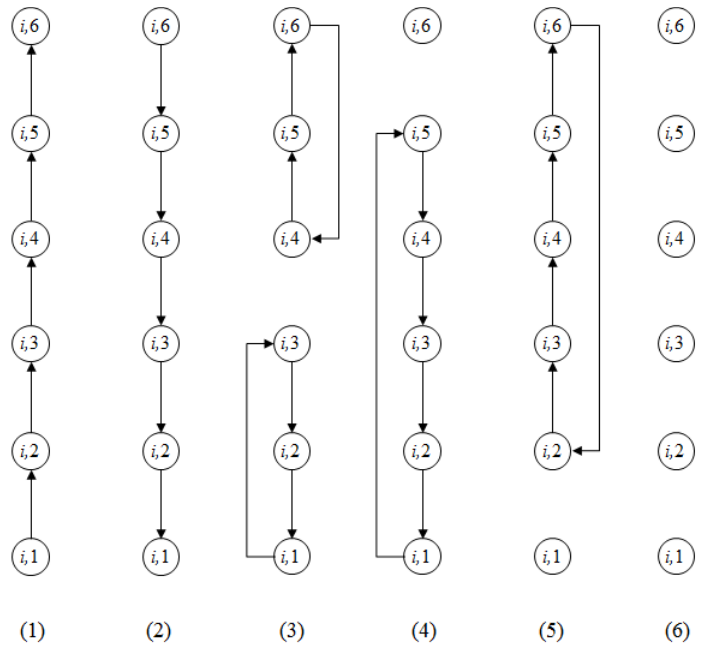

Fig. 3: Movements within a picking aisle permitted for an optimal solution

(1) The order picker enters the picking aisle from the front cross aisle, picks all requested items sequentially and exits the aisle at the rear cross aisle.

(2) The order picker enters the picking aisle from the rear cross aisle, picks all requested items and exits the aisle at the front cross aisle.

(3) In order to pick all items, the order picker enters and exits the picking aisle twice, once from and back to the front and once from and back to the rear cross aisle. In both cases he returns to the cross aisle from where the picking aisle was entered. The return point is defined by the "largest gap" which is the largest distance between two adjacent pick locations or a pick location and the adjacent cross aisle.

(4) In order to pick all items from the respective picking aisle, the order picker enters and leaves the aisle at the front cross aisle. The return point is defined by the pick location which corresponds to the largest distance from the front cross aisle.

(5) Likewise, the order picker enters and leaves the aisle at the rear cross aisle. The return point is defined by the pick location which corresponds to the largest distance from the rear cross aisle.

(6) The picking aisle is not entered at all since no requested item is located in that aisle.

With respect to these properties it is sufficient to consider only six points for a picking aisle i instead of all locations in which requested items are stored. In particular, these points are given by

(a) the intersections between a picking aisle i and the front / rear cross aisle (vertices $[i, 1]$ and $[i, 6]$),

(b) the two pick locations defining the largest gap (vertices $[i, 3]$ and $[i, 4]$) and

(c) the first and the last location where an item has to be picked (vertices $[i, 2]$ and $[i, 5]$).

In case that less than four items have to be picked from an aisle, points need to be duplicated in order to obtain the required number. In case that only one item is required, its location is represented by $[i, 2]$, $[i, 3]$, $[i, 4]$ and $[i, 5]$. If two items from different locations are required, then $[i, 2]$ and $[i, 3]$ ($[i, 4]$ and $[i, 5]$) are identical. In case of three items, the pair of locations defining the largest gap has to be determined. If the gap is between the two pick locations nearest to the front cross aisle, then $[i, 2]$ and $[i, 3]$ are identical, otherwise $[i, 4]$ and $[i, 5]$.

Based on these considerations, a graph representing the SPRP in a warehouse with a single-block layout can be constructed by introducing the six vertices for each picking aisle and choosing the edges that result from the options according to which items can be picked in a picking aisle (see Fig. 3). In order to represent moves of the order picker in the cross aisles, each pair of vertices $([i, 1], [i + 1])$ and $([i, 6], [i + 1, 6])$ $(i = 1, \ldots, m - 1$, where m denotes the number of picking aisles) is connected by two arcs. The depot is positioned in front of the leftmost picking aisle and identical to vertex $[1, 1]$ in the graph-theoretical representation. An example of a graph related to a (single-block) layout with five picking aisles is depicted in Fig. 4.

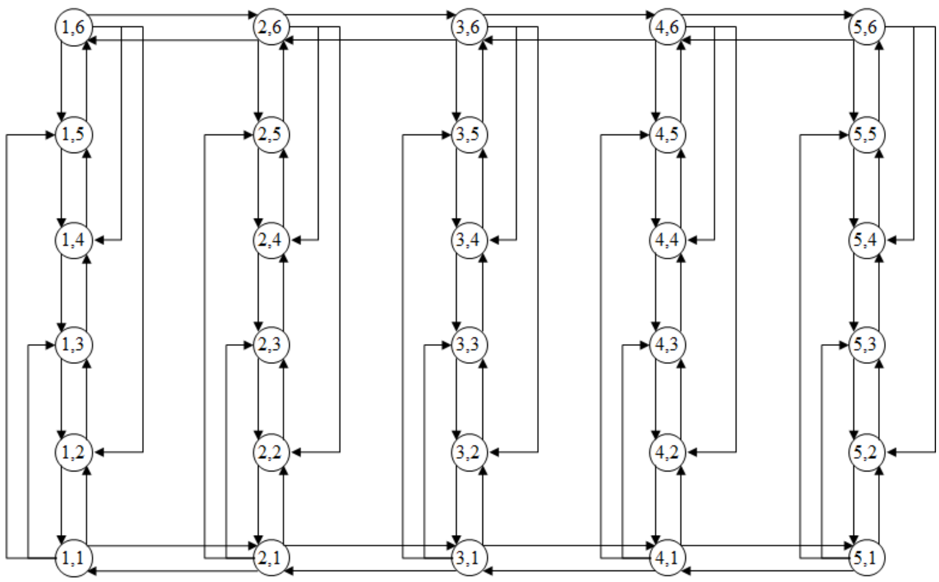

Fig. 4: A graph-theoretical representation of a SPRP in a single-block layout with five picking aisles

The weight c_e for an edge e in a picking aisle i can be determined as follows: The edge weight $c_{([i,3],[i,4])}$ is always equal to the largest gap between two adjacent requested items in picking aisle i or a requested item and the adjacent cross aisle. Let j_i^* be the location represented by vertex $[i, 3]$. Then, three different cases have to be distinguished for the determination of $c_{([i,1],[i,2])}$ and $c_{([i,2],[i,3])}$:

(1) If j_i^* does not correspond to the location of a requested item but to the point where picking aisle i can be entered via the front cross aisle, then $c_{([i,1],[i,2])} = c_{([i,2],[i,3])} = 0$.

(2) If j_i^* corresponds to the location of the requested item nearest to the front cross aisle, then $c_{([i,1],[i,2])}$ is equal to the distance between the front cross aisle and j_i^*, and $c_{([i,2],[i,3])} = 0$ holds.

(3) Otherwise, $c_{([i,1],[i,2])}$ is determined as in the second case, and $c_{([i,2],[i,3])}$ is the distance between the location of the requested item nearest to the front cross aisle and j_i^*.

The determination of $c_{([i,4],[i,5])}$ and $c_{([i,5],[i,6])}$ is performed analogously. If picking aisle i does not contain any requested item, then $c_{([i,3],[i,4])}$ is equal to the distance between the front and the rear cross aisle and the other edge weights are set to zero.

A feasible solution for the SPRP would be given by a tour which starts and ends at vertex $[1,1]$ and includes the vertices $[i,2]$, $[i,3]$, $[i,4]$ and $[i,5]$ for each picking aisle i that contains at least one requested item. When applying a model formulation for the TSP to this graph, we therefore need degree constraints for these vertices in order to ensure that they are contained in the tour. However, using this approach will result in two problems. First, even in an optimal solution for the SPRP, it is possible that vertices are visited more than once, which is not allowed in a standard TSP. Second, it is not sufficient to only guarantee that all vertices in a picking aisle are visited because this may lead to tours in which some requested items are skipped.

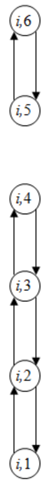

Fig. 5: Prohibited path in a solution

In Fig. 5, an infeasible combination of edges in a picking aisle is depicted. All vertices in this picking aisle are visited, however, it cannot be guaranteed that all items are included in the tour. This is caused by the fact that the number of vertices by which each picking aisle is defined, is neither dependent on the number nor on the location of the requested items. Because of this reason, it is possible that some requested items are situated between the locations that are represented by the vertices. Vertices $[i,3]$ and $[i,4]$ represent the two locations defining the largest gap and, therefore, no requested item can be situated between those locations. This is not true for vertices $[i,2]$ and $[i,3]$ as well as for $[i,4]$ and $[i,5]$. Since vertices $[i,2]$ and $[i,5]$ represent the location nearest and farthest from the front cross aisle, several requested items may be situated between these locations and the locations defining the largest gap. Therefore, guaranteeing that each vertex is included in the tour is not sufficient. In order to ensure feasibility of the solutions obtained by applying a TSP formulation based on this graph, additional predecessor and successor constraints for arcs would be needed. For example, we would have to ensure that arc $([i,4],[i,5])$ is used if arc $([i,3],[i,4])$ was chosen.

Another issue refers to the elimination of subtours. The general concept of the subtour elimination constraints in the three TSP formulations presented above consists of the enumeration of the vertices according to the sequence in which they appear in the tour. This approach cannot be successful when using this graph, because on the one hand, vertices are allowed to be visited more than once and, on the other hand, some cycles are allowed within the tour (e.g. if a largest gap strategy is used in an aisle).

In order to avoid visiting vertices more than once, vertices are split into several vertices in such a way that each generated vertex can only be visited one time. According to Ratliff & Rosenthal (1983) vertices corresponding to cross aisles can be visited up to three times, while the other vertices may be visited twice or less. Therefore, we replace each vertex $[i,1]$ and $[i,6]$ $(i=1,\ldots,m)$ by three vertices, where one vertex has to be used to enter a picking aisle and the other two vertices correspond to movements to the left and to the right in the cross aisles. The vertices $[i,2]$, $[i,3]$, $[i,4]$ and $[i,5]$ represent movements within a picking aisle i and are replaced by two vertices, where these vertices correspond to movements towards the rear cross aisle (up) and the front cross aisle (down), respectively. Furthermore, a vertex symbolizing the location of the depot is added. An example for the resulting graph is depicted in Fig. 6.

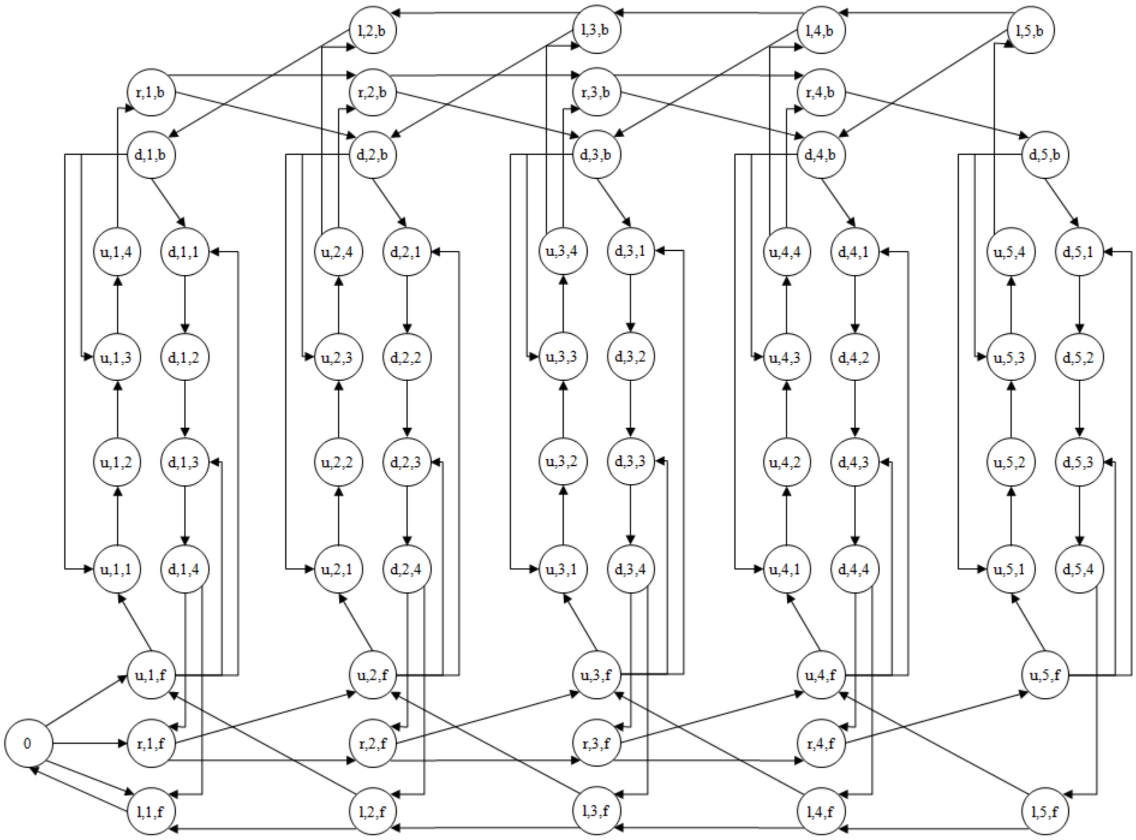

Fig. 6: Modified graph-theoretical representation of a SPRP in a single-block layout with five picking aisles

In general, the vertices of this graph can be described as follows. Vertex 0 symbolizes the location of the depot and the other vertices are characterized by a triple, where the first component represents the direction in which the tour can be continued. r and l indicate movements to the right and to the left, respectively. Movements towards the rear cross aisle and towards the front cross aisle are symbolized by u („up“) and d („down“). The second component characterizes the number of picking aisle i, where picking aisle 1 is the leftmost and aisle m the rightmost picking aisle. The last component of the triple represents the location of the vertex, where f and b mean that the vertex corresponds to the front and the rear cross aisle, respectively. The four locations in a particular picking aisle are enumerated from 1 to 4. Based on this denotation, the vertices $[l, 1, b]$, $[r, m, b]$ and $[r, m, f]$ do not exist, because at these points either moves to the left or to the right are possible. After having introduced the vertices of the graph, arcs are added according to the feasible options according to which requested items can be picked in a picking aisle (see Fig. 3).

These modifications result in a graph in which an optimal order picking tour can be constructed without visiting a vertex more than once. This graph includes more than twice the number of vertices as the previously presented graph does. However, the size of the improved model formulation for the SPRP will only be dependent on the number of arcs and not on the number of vertices. Furthermore, the size of the graph is completely independent of the number of requested items. Applying a TSP formulation to this modified graph, in which the number of subtour elimination constraints increases linearly to the number of arcs, will lead to a mathematical model whose size increases linearly with the number of picking aisles.

4.2 Model formulation based on the modified graph

In this section, it is shown how a TSP formulation is applied to the graph constructed in section 4.1. The complete mathematical model for the SPRP including the definition of all constants and variables is presented in the appendix. It includes the following classes of constraints:

- Degree constraints [(26) - (48)]: Each vertex visited has to be left afterwards.
- Subtour elimination constraints [(49) - (79)]: The resulting tour has to be connected.
- Depot inclusion constraint [(80)]: The depot has to be included in the tour.
- Item inclusion constraints [(81) - (82)]: Each requested item has to be included in the tour.

The degree and subtour elimination constraints [(26) - (79)] are very similar to the corresponding constraints [(2) - (4), (8) - (11) and (15) - (20)] used in the general TSP formulations and are only described briefly. As decision variables we introduce binary variables for each arc, indicating if the arc is contained in the tour (variable is equal to 1) or not (0). For a picking aisle i, the denotation of the variables for the different arc types is depicted in Fig. 7. For the sake of clarity, each arc type is only included for a single direction in this figure. Arcs corresponding to movements in the opposite direction are excluded. (For example, arc $([u, i, 3], [u, i, 4])$ is depicted and $([d, i, 1], [d, i, 2])$ is excluded.)

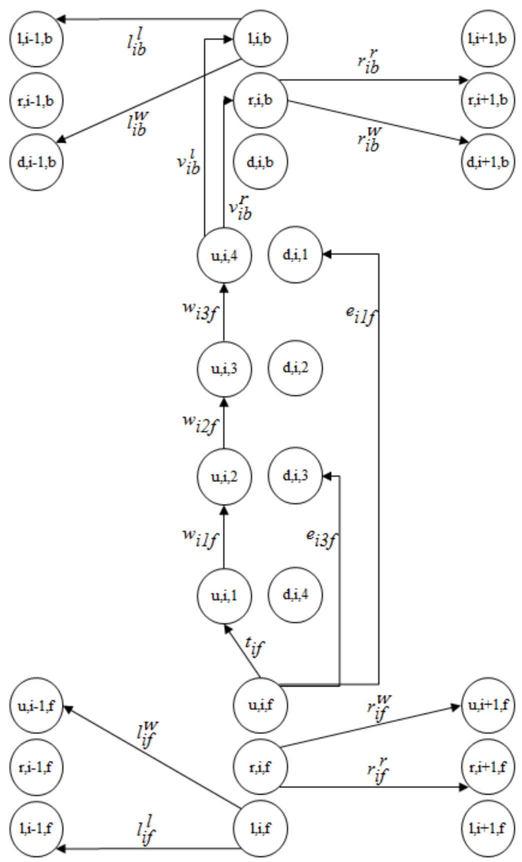

Fig. 7: Denotation of the variables in the model formulation

The degree constraints in general TSP formulations ensure that each vertex is visited exactly once, which means that the indegree and the outdegree of each vertex are equal to one, respectively. When dealing with a SPRP represented by the modified graph, each vertex is visited at most once. However, some vertices may exist which are not included in an optimal tour. Therefore, it has to be ensured that the indegree and the outdegree are equal to 1 if a vertex is visited and equal to 0 otherwise. This can be done by requiring that, for each vertex, the indegree is equal to the outdegree. In the degree constraints (26)-(48), the outegree is calculated on the left hand side of the equation while the right hand side represents the indegree.

The mathematical models presented in Section 3 only differ in the way how subtours are excluded. Based on pretests, we decided to apply the subtour elimination constraints by Gavish & Graves (1978) in which the arcs are enumerated according to the sequence they are used in the tour. Constraints (10) ensure that, for each vertex $i \in V \setminus \{0\}$, the sum of variables corresponding to arcs, which can be used to reach vertex i, has to be one greater than the sum of variables for leaving this vertex. Constraints (11) result in a solution where both sums contain exactly one variable greater than zero, respectively. The general principle of constraints (10) is also used in the model formulation based on the modified graph. The application of these constraints leads to constraints (49) to (69). The structure of the left hand side of these equations is equal to those of constraints (10). However, the right hand side of constraints (49) to (69) cannot be equal to one for each vertex, because this would lead to tours in which all vertices have to be visited although it is allowed to skip some vertices when using the modified graph. Therefore, the right hand side of constraints (49) to (69) has to be equal to zero if the corresponding vertex is not included in the tour and equal to 1, otherwise. This can be obtained by calculating the degree (here: outdegree) for each vertex. The second part of the subtour elimination constraints $[(70) - (79)]$ is equivalent to (11).

Constraint (80) ensures that the depot is included in the tour.

Since it is not necessary to visit all vertices, we have to introduce vertex-related criteria which have to be satisfied if it is permitted to skip this vertex. Each vertex not representing the depot or a location within a picking aisle can be skipped. However, in some cases it is also possible to skip vertices corresponding to a pick location. The pick locations of a picking aisle i are represented by four pairs of vertices. In order to guarantee that all requested items are contained in the tour, it is sufficient to visit the vertices nearest to the cross aisles because, in this case, the degree constraints ensure that the other locations are also visited. Let us consider the two vertices nearest to the front cross aisle which are denoted by $[u, i, 1]$ and $[d, i, 4]$. Both vertices represent the same pick location and, therefore, only one of them has to be included in the tour. If vertex $[u, i, 1]$ is visited, the next vertex to visit is $[u, i, 2]$, which means that the variable w_{i1f} has to be equal to one. If $[d, i, 4]$ is included in the tour, the vertex $[d, i, 3]$ has to be visited before, which implies $w_{i3b} = 1$. Since picking aisles which do not contain any requested items can also be skipped, the constraint $w_{i1f} + w_{i3b} \geq 1$ must hold for all picking aisles i containing at least one requested item. This is expressed by the constant b_i which is equal to 1 if picking aisle i has to be visited and 0 otherwise. Analogously, the constraints resulting from the pair of vertices nearest to the rear cross aisle can be constructed. This results in the following two constraints for a picking aisle i:

$$w_{i1f} + w_{i3b} \geq b_i \tag{23}$$

$$w_{i1b} + w_{i3f} \geq b_i \tag{24}$$

However, another special case exists in which one of these two pairs of vertices is allowed not to be contained in the tour. This case occurs if the largest gap is not defined by the location of two requested items but by an item and the adjacent cross aisle. If the corresponding cross aisle is the rear cross aisle, then the pair of vertices nearest to the rear cross aisle does not have to be visited. In this case, the vertices $[d, i, b]$, $[u, i, 4]$ and $[u, i, 3]$ represent the same location and, therefore, the distance between $[d, i, b]$ and $[u, i, 3]$, denoted by c^e_{i3b}, is equal to zero. This implies that constraint (24) must hold if and only if $c^e_{i3b} > 0$ which can be obtained by multiplying both sides of the constraint by c^e_{i3b} resulting in (82). The same line of argumentation holds if the front cross aisle is considered. In this case, multiplying constraint (23) by c^e_{i3f} leads to (81).

In total, we have $O(m)$ variables and constraints, i.e. the number of variables and constraints increases only linearly with the number of picking aisles m and is neither dependent on the amount nor on the location of the requested items. Therefore, it can be assumed that this formulation is far superior to general TSP formulations if many items are to be picked.

5 Numerical experiments

5.1 Test design

In order to analyze the proposed formulations, we compare the computing times needed by a commercial IP-solver for providing optimal solution for a variety of problem classes. In our experiments the number of picking aisles m determines the size of the warehouse and has been fixed to 5, 10, 15 and 20. Each picking aisle consists of 90 storage locations (45 on each side). The length of each storage location amounts to one length unit (LU). Whenever leaving an aisle, the order picker has to move one LU in the vertical direction from either the first or the last storage location in order to reach the cross aisle. The distance between two adjacent picking aisles is equal to 5 LU. The depot is assumed to be located in front of the leftmost picking aisle. For the size of a picking order (number of requested items) 30, 45, 60 and 75 items have been chosen. Moreover, we assume uniformly distributed demands, i.e. each item has the same probability to be included in a picking order.

Table 1: Size of mathematical programming formulations for the SPRP

$(\mathbf{m}, \mathbf{n})$	**MTZ**		**GG**		**C**		**HSSW**	
	#var	#cons	#var	#cons	#var	#cons	#var	#cons
(5, 30)	960	932	1830	992	28830	28892	220	254
(5, 45)	2115	2072	4095	2162	95220	95312	220	254
(5, 60)	3720	3662	7260	3782	223260	223382	220	254
(5, 75)	5775	5702	11325	5852	433200	433352	220	254
(10, 30)	960	932	1830	992	28830	28892	460	524
(10, 45)	2115	2072	4095	2162	95220	95312	460	524
(10, 60)	3720	3662	7260	3782	223260	223382	460	524
(10, 75)	5775	5702	11325	5852	433200	433352	460	524
(15, 30)	960	932	1830	992	28830	28892	700	794
(15, 45)	2115	2072	4095	2162	95220	95312	700	794
(15, 60)	3720	3662	7260	3782	223260	223382	700	794
(15, 75)	5775	5702	11325	5852	433200	433352	700	794
(20, 30)	960	932	1830	992	28830	28892	940	1064
(20, 45)	2115	2072	4095	2162	95220	95312	940	1064
(20, 60)	3720	3662	7260	3782	223260	223382	940	1064
(20, 75)	5775	5702	11325	5852	433200	433352	940	1064
	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(n^3)$	$O(n^3)$	$O(m)$	$O(m)$

Combination of these parameters gives rise to 16 problem classes. For each class, 30 instances have been generated, resulting in 480 instances in total. The size of the corresponding formulations is depicted in Table 1. In the model formulations by Miller et al. (1960) [MTZ], Gavish & Graves (1978) [GG] and Claus (1984) [C], the number of variables (#var) and constraints (#cons) depends on the number of requested items n, whereas the size of the newly-proposed formulation [HSSW] is determined by the number of picking aisles m only. Therefore, it can be assumed that the new formulation for the SPRP outperforms the other formulations if the ratio $\frac{n}{m}$ gets large. If we consider the chosen problem classes, the proposed formulation is superior to the general TSP formulations in terms of number of variables and constraints (see Table 1) because the size of the new model formulation increases linearly in the number of picking aisles. Even for a large number of picking aisles ($m = 20$) and a quite small number of requested items ($n = 30$) – resulting in a very small ratio $\frac{n}{m}$ – the number of variables in the HSSW formulation is smaller than the number of variables required for the general TSP formulations.

In order to analyze the time required to solve these model formulations, all formulations have been implemented and solved by CPLEX 12.6. The experiments have been carried out on a desktop PC with a 3.4 GHz Pentium processor with 8 GB RAM. The computing time for each instance and formulation has been limited to 30 minutes.

5.2 Results

In the following, the different mathematical programming formulations are evaluated with respect to the corresponding computing times. Table 2 depicts the number of optimal solutions which can be obtained within the predefined time interval. The results show that CPLEX – by application of the HSSW formulation – was able to identify an optimal solution for every instance. In case of the GG formulation, an optimal solution could be obtained for all instances with up to 45 items to be picked. Using the MTZ formulation, CPLEX was already unable to obtain optimal solutions for all instances with the smallest number of requested items. The C formulation leads to optimal solutions for all instances with $n = 30$. However, for instances with 60 requested items, the number of optimally solved instances was zero or close to zero. Furthermore, it should be noted that for the three approaches the number of identified optimal solutions decreases with an increasing number of picking aisles.

Table 2: Number of solved instances (out of 30) within 30 minutes of computing time

$(\mathbf{m}, \mathbf{n})$	MTZ	GG	C	HSSW
(5, 30)	28	**30**	**30**	**30**
(5, 45)	19	**30**	15	**30**
(5, 60)	3	28	0	**30**
(5, 75)	0	22	0	**30**
(10, 30)	26	**30**	**30**	**30**
(10, 45)	19	**30**	13	**30**
(10, 60)	1	**30**	0	**30**
(10, 75)	0	25	0	**30**
(15, 30)	26	**30**	**30**	**30**
(15, 45)	6	**30**	7	**30**
(15, 60)	0	29	1	**30**
(15, 75)	0	21	0	**30**
(20, 30)	24	**30**	**30**	**30**
(20, 45)	5	**30**	5	**30**
(20, 60)	0	26	0	**30**
(20, 75)	0	11	0	**30**

In Table 3 the average computing times for the four formulations are presented. Computing times have only been recorded if the instance has been solved to optimality and, therefore, no information about computing times is given for some problem classes when the MTZ or the C formulation has been applied.

Comparing the computing times to each other, the MTZ and the C formulations are outperformed by the GG and the HSSW formulations significantly. Solving the corresponding formulations based on GG and HSSW requires only a small fraction of the computing times for the MTZ and the C formulations. For those problem classes for which the application of the GG and the HSSW formulations were able to generate an optimal solution for every instance (problem classes with 30 or 45 requested items), application of the HSSW formulation requires the smallest computing times for most instances.

Table 3: Computing times [sec]

$(\mathbf{m}, \mathbf{n})$	**MTZ**	**GG**	**C**	**HSSW**
(5, 30)	109.18	2.65	25.67	**0.09**
(5, 45)	869.72	22.37	1169.04	**0.09**
(5, 60)	1666.20	453.94	-	**0.09**
(5, 75)	-	898.21	-	**0.09**
(10, 30)	310.62	1.94	114.14	**1.60**
(10, 45)	991.73	14.59	1408.99	**1.03**
(10, 60)	1750.94	90.74	-	**1.42**
(10, 75)	-	482.53	-	**1.36**
(15, 30)	372.77	3.40	89.14	**2.29**
(15, 45)	1564.07	20.20	1562.30	**5.28**
(15, 60)	-	395.01	1761.62	**10.64**
(15, 75)	-	1069.54	-	**15.10**
(20, 30)	555.07	**4.05**	104.96	10.57
(20, 45)	1649.94	36.64	1656.18	**27.32**
(20, 60)	-	524.44	-	**114.33**
(20, 75)	-	1551.92	-	**216.63**

As could be expected with respect to the number of constraints and variables, the computing times required by the three general TSP formulations increase with the number of requested items (i.e. the number of vertices in the problem), whereas the computing times required for solving the HSSW formulation increase with the number of picking aisles. If the number of aisles is small ($m = 5$), the application of the HSSW formulation decreases the computing times by a factor between 30 ($n = 30$) and 1000 ($n = 75$) compared to the best general TSP formulation. With an increasing number of aisles the computing times required for solving the HSSW formulation also increase. However, even in case of a small ratio $\frac{n}{m}$, only the formulation by GG outperforms the HSSW formulation. For a large number of aisles ($m = 20$) and requested items ($n = 75$) the computing times can be decreased by a factor of 7.

The results further show that computing times required for the solution of the HSSW formulation are larger when the number of requested items increases, even though the number of arcs and vertices of the modified graph is independent of the number and the location of requested items. This is due to two reasons: First, constraints (81) and (82) are redundant for a picking aisle i if $b_i = 0$, i.e. if picking aisle i does not contain any requested item. The less items have to be picked, the smaller the probability gets that a large number of picking aisles has to be visited. Second, a large number of requested items results in many solutions almost as good as an optimal solution and, therefore, proving optimality gets quite time consuming.

6 Conclusion and Outlook

In this article, a new mathematical programming formulation for the Single-Picker Routing Problem – a special variant of the Traveling Salesman Problem – was proposed. This formulation adopts the properties of optimal solutions for the Single-Picker Routing Problem in a single-block layout. It could be shown that the proposed formulation is advantageous with respect to the size (expressed in number of variables and number of constraints) of the resulting model which only depends on the number of aisles and not on the number of items to be picked.

In a series of numerical experiments this formulation is compared to more general state-of-the-art formulations for the Traveling Salesman Problem. The computing times necessary to solve the newly proposed formulation outperform those required for the more general formulations for the Traveling Salesman Problem. Moreover, we were able to find optimal solutions for instances which could not be solved by means of the general formulations. The numerical results clearly indicate that for special cases of the Traveling Salesman Problem which can be solved in polynomial time, the application of customized formulations is necessary in order to obtain optimal solutions within small computing times.

This work focused on a single-block layout. It is possible to extend the formulation for layouts with multiple blocks. In this case, it is necessary to extend the modified graph. For each block of the warehouse a copy of the modified graph has to be added (except for the vertex representing the depot). Furthermore, vertices and arcs representing a change of a block have to be added for each possible location where a picking aisle can be entered from a middle cross aisle.

For the proposed formulation it is assumed that order pickers can overtake each other (wide-aisle). Therefore, the tours through the warehouse can be determined independently from each other. In narrow-aisle warehouses aisles can be blocked by an order picker or traffic jams can occur caused by order pickers having to collect an item from the same storage location. For this purpose the proposed formulation can be combined with the optimization model for the Order Batching Problem suggested by Hong et al. (2012) which takes blocking considerations into account. However, the application of this model is restricted to warehouses in which one-way travel within picking aisles is allowed only and, therefore, the routing problem with consideration of picker blocking aspects would also be an interesting topic for further research.

References

Burkard, R.; Deĭneko, V. G; van der Veen, J. A. A. & Woeginger, G. J. (1998): Well-Solvable Special Cases of the Traveling Salesman Problem: A Survey. *SIAM Review* 40, 496-546.

Claus, A. (1984): A New Formulation for the Traveling Salesman Problem. *SIAM Journal on Algebraic and Discrete Methods* 5, 21-25.

Dantzig. G. B.; Fulkerson, D. R. & Johnson S. M. (1954): Solution of a Large-Scale Traveling Salesman Problem. *Journal of the Operations Research Society of America* 2, 363-410.

de Koster, R.; Le-Duc, T. & Roodbergen, K. J. (2007): Design and Control of Warehouse Order Picking: A Literature Review. *Science Direct* 182, 481-501.

Filip, E. & Otakar, M. (2011): The Travelling Salesman Problem and its Application in Logistic Practice. *WSEAS Transactions on Business and Economics* 8, 163-173.

Gavish, B. & Graves, S. C. (1978): The Traveling Salesman Problem and Related Problems. Working Paper GR-078-78, Operations Research Center, Massachusetts Institute of Technology.

Glover, F. & Punnen, A. P. (1997): The Travelling Salesman Problem: New Solvable Cases and Linkages with the Development of Approximation Algorithms. *Journal of the Operational Research Society* 48, 502-510.

Gu, J.; Goetschalckx, M. & McGinnis, L. F. (2007): Research on Warehouse Operation: A Comprehensive Review. *European Journal of Operational Research* 177, 1-21.

Hong. S.; Johnson, A. L. & Peters B. A. (2012): Batch Picking in Narrow-Aisle Order Picking Systems with Consideration for Picker Blocking. *European Journal of Operational Research* 221, 557-570.

Jarvis, J. M.; McDowell, E. D. (1991): Optimal Product Layout in an Order Picking Warehouse. *IIE Transactions* 23, 93-102.

Letchford, A. N.; Nasiri, S. D. & Theis, D. O. (2013): Compact Formulations of the Steiner Traveling Salesman Problem and Related Problems. *European Journal of Operational Research* 228, 83-92.

Lenstra, J. K.; & Rinnooy Kan, A. H. G. (1975): Some Simple Applications of the Travelling Salesman Problem. *Operational Research Quarterly* 26, 717-733.

Matai, R.; Singh, S. P.; Mittal, M. L. (2010): Traveling Salesman Problem: An Overview of Applications, Formulations and Solution Approaches. *Traveling Salesman Problem: Theory and Applications*, Davendra, D. (ed.), 1-24, InTech.

Miller, C. E.; Tucker, A. W. & Zemlin, R. A. (1960): Integer Programming Formulations and Traveling Salesman Problems. *Journal of the Association for Computing Machinery* 7, 326-329.

Öncan, T.; Altinel, K.; Laporte, G. (2009): A Comparative Analysis of Several Asymmetric Traveling Salesman Problem Formulations. *Computers & Operations Research* 36, 637-654.

Padberg, M.; & Sung, T. (1991): An Analytical Comparison of Different Formulations of the Traveling Salesman Problem. *Mathematical Programming* 52, 315-357.

Petersen, C. G. & Schmenner, R. W. (1999): An Evaluation of Routing and Volume-Based Storage Policies in an Order Picking Operation. *Decision Science* 30, 481-501.

Ratliff, H. D. & Rosenthal, A. R. (1983): Order-Picking in a Rectangular Warehouse: A Solvable Case of the Traveling Salesman Problem. *Operations Research* 31, 507-521.

Rego, C.; Gamboa, D.; Glover, F. & Osterman, C. (2011): Traveling Salesman Problem Heuristics: Leading Methods, Implementations and Latest Advances. *European Journal of Operational Research* 211, 427-441.

Roodbergen, K. J. (2001): Layout and Routing Methods for Warehouses. Trial: Rotterdam.

Roodbergen, K. J. & de Koster, R. (2001a): Routing Order Pickers in a Warehouse with a Middle Aisle. *European Journal of Operational Research* 133, 32-43.

Roodbergen, K. J. & de Koster, R. (2001b): Routing Methods for Warehouses with Multiple Cross Aisles. *International Journal of Production Research* 39, 1865-1883.

Tompkins, J. A.; White, J. A.; Bozer, Y. A. & Tanchoco, J. M. A. (2010): Facilities Planning. 4th edition, John Wiley & Sons, New Jersey.

Wäscher, G. (2004): Order Picking: A Survey of Planning Problems and Methods. *Supply Chain Management and Reverse Logistics*, Dyckhoff , H.; Lackes, R. & Reese, J. (eds.), 324-370, Springer: Berlin.

Appendix: Model formulation for the SPRP

Sets:

$I = \{1, \ldots, m\}$: set of picking aisles
$F = \{f, b\}$: set of cross aisles

Constants:

$$b_i = \begin{cases} 1, & \text{if picking aisle } i \in I \text{ contains at least one requested item} \\ 0, & \text{otherwise} \end{cases}$$

c^a: distance between two adjacent aisles

c^0: distance between the depot and the location on the front cross aisle where the first picking aisle can be entered

$c^e_{is\alpha}$: distance between front cross aisle $(\alpha = f)$ and vertex $[d, i, s]$ $(s \in \{1, 3\})$ or rear cross aisle $(\alpha = b)$ and vertex $[u, i, s]$ in picking aisle $i \in I$

$c^t_{i\alpha}$: distance between front cross aisle $(\alpha = f)$ and vertex $[u, i, 1]$ or back cross aisle $(\alpha = b)$ and vertex $[d, i, s]$ in picking aisle $i \in I$

c^w_{is}: distance between location $s \in \{1, 2, 3\}$ and location $s + 1$ in picking aisle $i \in I$

M: large number (e.g. number of vertices)

Binary variables indicating the edges included in the tour:

$$r^r_{i\alpha} = \begin{cases} 1, & \text{if edge } ([r, i, \alpha], [r, i+1, \alpha]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad (i, \alpha) \in (I \setminus \{m-1, m\}) \times F$$

$$r^w_{ib} = \begin{cases} 1, & \text{if edge } ([r, i, b], [d, i+1, b]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I \setminus \{m\}$$

$$r^w_{if} = \begin{cases} 1, & \text{if edge } ([r, i, f], [u, i+1, f]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I \setminus \{m\}$$

$$\ell^\ell_{i\alpha} = \begin{cases} 1, & \text{if edge } ([\ell, i, \alpha], [\ell, i-1, \alpha]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad (i, \alpha) \in ((I \setminus \{1, 2\}) \times F) \cup (2, f)$$

$$\ell^w_{ib} = \begin{cases} 1, & \text{if edge } ([\ell, i, b], [d, i-1, b]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I \setminus \{1\}$$

$$\ell_{if}^{w} = \begin{cases} 1, & \text{if edge } ([\ell, i, f], [u, i-1, f]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I \setminus \{1\}$$

$$e_{isb} = \begin{cases} 1, & \text{if edge } ([d, i, b], [u, i, s]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad (i, s) \in I \times \{1, 3\}$$

$$e_{isf} = \begin{cases} 1, & \text{if edge } ([u, i, f], [d, i, s]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad (i, s) \in I \times \{1, 3\}$$

$$t_{ib} = \begin{cases} 1, & \text{if edge } ([d, i, b], [d, i, 1]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I$$

$$t_{if} = \begin{cases} 1, & \text{if edge } ([u, i, f], [u, i, 1]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I$$

$$w_{isb} = \begin{cases} 1, & \text{if edge } ([d, i, s], [d, i, s+1]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad (i, s) \in I \times \{1, 2, 3\}$$

$$w_{isf} = \begin{cases} 1, & \text{if edge } ([u, i, s], [u, i, s+1]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad (i, s) \in I \times \{1, 2, 3\}$$

$$v_{ib}^{r} = \begin{cases} 1, & \text{if edge } ([u, i, 4], [r, i, b]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I \setminus \{m\}$$

$$v_{if}^{r} = \begin{cases} 1, & \text{if edge } ([d, i, 4], [r, i, f]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I \setminus \{m\}$$

$$v_{ib}^{\ell} = \begin{cases} 1, & \text{if edge } ([u, i, 4], [\ell, i, b]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I \setminus \{1\}$$

$$v_{if}^{\ell} = \begin{cases} 1, & \text{if edge } ([d, i, 4], [\ell, i, f]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad i \in I$$

$$y_{\alpha}^{0} = \begin{cases} 1, & \text{if edge } ([0], [\alpha, 1, f]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases} \qquad \alpha \in \{l, r, u\}$$

$$y_{0}^{\ell} = \begin{cases} 1, & \text{if edge } ([\ell, 1, f], [0]) \text{ is contained in the tour} \\ 0, & \text{otherwise} \end{cases}$$

Real-valued variables to exclude subcycles:

$\tilde{r}^r_{i\alpha}, \quad (i,\alpha) \in (I \setminus \{m-1, m\}) \times F$

$\tilde{r}^w_{i\alpha}, \quad (i,\alpha) \in (I \setminus \{m\}) \times F$

$\tilde{\ell}^\ell_{i\alpha}, \quad (i,\alpha) \in ((I \setminus \{1,2\}) \times F) \cup (2,f)$

$\tilde{\ell}^w_{i\alpha}, \quad (i,\alpha) \in (I \setminus \{1\}) \times F$

$\tilde{e}_{is\alpha}, \ (i,s,\alpha) \in I \times \{1,3\} \times F$

$\tilde{t}_{i\alpha}, \quad (i,\alpha) \in I \times F$

$\tilde{w}_{is\alpha}, \ (i,s,\alpha) \in I \times \{1,2,3\} \times F$

$\tilde{v}^r_{i\alpha}, \quad (i,\alpha) \in (I \setminus \{m\}) \times F$

$\tilde{v}^\ell_{i\alpha}, \quad (i,\alpha) \in ((I \setminus \{1\}) \times F) \cup (1,f)$

$\tilde{y}^0_\alpha, \quad \alpha \in \{l, r, u\}$

Objective function:

$$\begin{aligned} \min \ & \sum_{i=1}^{m-2} \sum_{\alpha \in F} c^a \cdot (r^r_{i\alpha} + r^w_{i\alpha}) + c^a \cdot \sum_{\alpha \in F} r^w_{m-1,\alpha} \\ & + \sum_{i=3}^{m} \sum_{\alpha \in F} c^a \cdot \left(\ell^\ell_{i\alpha} + \ell^w_{i\alpha}\right) + c^a \cdot \ell^\ell_{2f} + c^a \cdot \sum_{\alpha \in F} \ell^w_{2\alpha} \\ & + \sum_{i=1}^{m} \sum_{s \in \{1,3\}} \sum_{\alpha \in F} c^e_{is\alpha} \cdot e_{is\alpha} + \sum_{i=1}^{m} \sum_{\alpha \in F} c^t_{i\alpha} \cdot t_{i\alpha} \\ & + \sum_{i=1}^{m} \sum_{s=1}^{3} \sum_{\alpha \in F} c^w_{is\alpha} \cdot w_{is\alpha} + \sum_{i=1}^{m-1} \sum_{\alpha \in F} c^t_{i\alpha} \cdot v^r_{i\alpha} \\ & + \sum_{i=2}^{m} \sum_{\alpha \in F} c^t_{i\alpha} \cdot v^\ell_{i\alpha} + c^t_{1f} \cdot v^\ell_{1f} \\ & + c^0 \cdot \left(y^0_l + y^0_r + y^0_u + y^l_0\right) \end{aligned} \tag{25}$$

Degree constraints:

- constraints corresponding to the depot

$$y^0_l + y^0_r + y^0_u = y^l_0 \tag{26}$$

- constraints corresponding to vertices $[r, i, \alpha]$

$$r^r_{i\alpha} + r^w_{i\alpha} = v^r_{i\alpha} + r^r_{i-1,\alpha} \qquad \forall (i, \alpha) \in (I \setminus \{1, m-1, m\}) \times F \tag{27}$$

$$r^w_{m-1,\alpha} = v^r_{m-1,\alpha} + r^r_{m-2,\alpha} \qquad \forall \alpha \in F \tag{28}$$

$$r^r_{1f} + r^w_{1f} = v^r_{1f} + y^0_r \tag{29}$$

$$r^r_{1b} + r^w_{1b} = v^r_{1b} \tag{30}$$

- constraints corresponding to vertices $[\ell, i, \alpha]$

$$\ell^\ell_{i\alpha} + \ell^w_{i\alpha} = v^\ell_{i\alpha} + \ell^\ell_{i+1,\alpha} \qquad \forall (i, \alpha) \in (I \setminus \{1, 2, m\}) \times F \tag{31}$$

$$\ell^\ell_{m\alpha} + \ell^w_{m\alpha} = v^\ell_{m\alpha} \qquad \forall \alpha \in F \tag{32}$$

$$\ell^\ell_{2f} + \ell^w_{2f} = v^\ell_{2f} + \ell^\ell_{3f} \tag{33}$$

$$\ell^w_{2b} = v^\ell_{2f} + \ell^\ell_{3f} \tag{34}$$

$$y^\ell_0 = y^0_\ell + v^\ell_{1f} + \ell^\ell_{2f} \tag{35}$$

- constraints corresponding to vertices $[u, i, f]$ and $[d, i, b]$

$$t_{i\alpha} + e_{i1\alpha} + e_{i3\alpha} = r^w_{i-1,\alpha} + \ell^w_{i+1,\alpha} \qquad \forall (i, \alpha) \in (I \setminus \{2, m\}) \times F \tag{36}$$

$$t_{m\alpha} + e_{m1\alpha} + e_{m3\alpha} = r^w_{m-1,\alpha} \qquad \forall \alpha \in F \tag{37}$$

$$t_{1f} + e_{11f} + e_{13f} = y^0_u + \ell^w_{2f} \tag{38}$$

$$t_{1b} + e_{11b} + e_{13b} = \ell^w_{2b} \tag{39}$$

- constraints corresponding to vertices $[u, i, 4]$ and $[d, i, 4]$

$$v^r_{i\alpha} + v^\ell_{i\alpha} = w_{i3\alpha} \qquad \forall (i, \alpha) \in (I \setminus \{2, m\}) \times F \tag{40}$$

$$v^\ell_{m\alpha} = w_{m3\alpha} \qquad \forall \alpha \in F \tag{41}$$

$$v^r_{1f} + v^\ell_{1f} = w_{13f} \tag{42}$$

$$v^r_{1b} = w_{13b} \tag{43}$$

- constraints corresponding to vertices $[u, i, s]$ and $[d, i, s]$

$$w_{i1f} = t^{if} + e_{i1b} \qquad \forall i \in I \tag{44}$$

$$w_{i1b} = t^{ib} + e_{i1f} \qquad \forall i \in I \tag{45}$$

$$w_{i2\alpha} = w_{i1\alpha} \qquad \forall (i,\alpha) \in I \times F \tag{46}$$

$$w_{i3f} = w_{i2f} + e_{i3b} \qquad \forall i \in I \tag{47}$$

$$w_{i3b} = w_{i2b} + e_{i3f} \qquad \forall i \in I \tag{48}$$

Subtour elimination constraints:

- constraints corresponding to vertices $[r, i, \alpha]$

$$\tilde{v}^r_{i\alpha} + \tilde{r}^r_{i-1,\alpha} - (\tilde{r}^r_{i\alpha} + \tilde{r}^w_{i\alpha}) = r^r_{i\alpha} + r^w_{i\alpha} \qquad \forall (i,\alpha) \in (I \setminus \{1, m-1, m\}) \times F \tag{49}$$

$$\tilde{v}^r_{m-1,\alpha} + \tilde{r}^r_{m-2,\alpha} - \tilde{r}^w_{m-1,\alpha} = r^w_{m-1,\alpha} \qquad \forall \alpha \in F \tag{50}$$

$$\tilde{v}^r_{1f} + \tilde{y}^0_r - \left(\tilde{r}^r_{1f} + \tilde{r}^w_{1f}\right) = r^r_{1f} + r^w_{1f} \tag{51}$$

$$\tilde{v}^r_{1b} - (\tilde{r}^r_{1b} + \tilde{r}^w_{1b}) = r^r_{1b} + r^w_{1b} \tag{52}$$

- constraints corresponding to vertices $[\ell, i, \alpha]$

$$\tilde{v}^\ell_{i\alpha} + \tilde{\ell}^\ell_{i+1,\alpha} - \left(\tilde{\ell}^\ell_{i\alpha} + \tilde{\ell}^w_{i\alpha}\right) = \ell^\ell_{i\alpha} + \ell^w_{i\alpha} \qquad \forall (i,\alpha) \in (I \setminus \{1, 2, m\}) \times F \tag{53}$$

$$\tilde{v}^\ell_{m\alpha} - \left(\tilde{\ell}^\ell_{m\alpha} + \tilde{\ell}^w_{m\alpha}\right) = \ell^\ell_{m\alpha} + \ell^w_{m\alpha} \qquad \forall \alpha \in F \tag{54}$$

$$\tilde{v}^\ell_{2f} + \tilde{\ell}^\ell_{3f} - \left(\tilde{\ell}^\ell_{2f} + \tilde{\ell}^w_{2f}\right) = \ell^\ell_{2f} + \ell^w_{2f} \tag{55}$$

$$\tilde{v}^\ell_{2f} + \tilde{\ell}^\ell_{3f} - \tilde{\ell}^w_{2b} = \ell^w_{2b} \tag{56}$$

- constraints corresponding to vertices $[u, i, f]$ and $[d, i, b]$

$$\tilde{r}^w_{i-1,\alpha} + \tilde{\ell}^w_{i+1,\alpha} - \left(\tilde{t}_{i\alpha} + \tilde{e}_{i1\alpha} + \tilde{e}_{i3\alpha}\right) = t_{i\alpha} + e_{i1\alpha} + e_{i3\alpha} \qquad \forall (i,\alpha) \in (I \setminus \{2, m\}) \times F \tag{57}$$

$$\tilde{r}^w_{m-1,\alpha} - \left(\tilde{t}_{m\alpha} + \tilde{e}_{m1\alpha} + \tilde{e}_{m3\alpha}\right) = t_{m\alpha} + e_{m1\alpha} + e_{m3\alpha} \qquad \forall \alpha \in F \tag{58}$$

$$\tilde{y}^0_u + \tilde{\ell}^w_{2f} - \left(\tilde{t}_{1f} + \tilde{e}_{11f} + \tilde{e}_{13f}\right) = t_{1f} + e_{11f} + e_{13f} \tag{59}$$

$$\tilde{\ell}^w_{2b} - \left(\tilde{t}_{1b} + \tilde{e}_{11b} + \tilde{e}_{13b}\right) = t_{1b} + e_{11b} + e_{13b} \tag{60}$$

- constraints corresponding to vertices $[u, i, 4]$ and $[d, i, 4]$

$$\tilde{w}_{i3\alpha} - \left(\tilde{v}^r_{i\alpha} + \tilde{v}^\ell_{i\alpha}\right) = v^r_{i\alpha} + v^\ell_{i\alpha} \qquad \forall (i,\alpha) \in (I \setminus \{2, m\}) \times F \tag{61}$$

$$\tilde{w}_{m3\alpha} - \tilde{v}^\ell_{m\alpha} = v^\ell_{m\alpha} \qquad \forall \alpha \in F \tag{62}$$

$$\tilde{w}_{13f} - \left(\tilde{v}^r_{1f} + \tilde{v}^\ell_{1f}\right) = v^r_{1f} + v^\ell_{1f} \tag{63}$$

$$\tilde{w}_{13b} - \tilde{v}^r_{1b} = v^r_{1b} \tag{64}$$

- constraints corresponding to vertices $[u, i, s]$ and $[d, i, s]$

$$\tilde{t}^{if} + \tilde{e}_{i1b} - (\tilde{w}_{i1f}) = w_{i1f} \qquad \forall i \in I \tag{65}$$

$$\tilde{t}^{ib} + \tilde{e}_{i1f} - \tilde{w}_{i1b} = w_{i1b} \qquad \forall i \in I \tag{66}$$

$$\tilde{w}_{i1\alpha} - \tilde{w}_{i2\alpha} = w_{i2\alpha} \qquad \forall (i, \alpha) \in I \times F \tag{67}$$

$$\tilde{w}_{i2f} + \tilde{e}_{i3b} - \tilde{w}_{i3f} = w_{i3f} \qquad \forall i \in I \tag{68}$$

$$\tilde{w}_{i2b} + \tilde{e}_{i3f} - \tilde{w}_{i3b} = w_{i3b} \qquad \forall i \in I \tag{69}$$

- constraints to link variables

$$\tilde{r}^r_{i\alpha} \le M \cdot r^r_{i\alpha} \qquad \forall (i, \alpha) \in (I \setminus \{m-1, m\}) \times F \tag{70}$$

$$\tilde{r}^w_{i\alpha} \le M \cdot r^w_{i\alpha} \qquad \forall (i, \alpha) \in (I \setminus \{m\}) \times F \tag{71}$$

$$\tilde{\ell}^\ell_{i\alpha} \le M \cdot \ell^\ell_{i\alpha} \qquad \forall (i, \alpha) \in ((I \setminus \{1, 2\}) \times F) \cup (2, f) \tag{72}$$

$$\tilde{\ell}^w_{i\alpha} \le M \cdot \ell^w_{i\alpha} \qquad \forall (i, \alpha) \in (I \setminus \{1\}) \times F \tag{73}$$

$$\tilde{e}_{is\alpha} \le M \cdot e_{is\alpha} \qquad \forall (i, s, \alpha) \in I \times \{1, 3\} \times F \tag{74}$$

$$\tilde{t}_{i\alpha} \le M \cdot t_{i\alpha} \qquad \forall (i, \alpha) \in I \times F \tag{75}$$

$$\tilde{w}_{is\alpha} \le M \cdot w_{is\alpha} \qquad \forall (i, s, \alpha) \in I \times \{1, 2, 3\} \times F \tag{76}$$

$$\tilde{v}^r_{i\alpha} \le M \cdot v^r_{i\alpha} \qquad \forall (i, \alpha) \in (I \setminus \{m\}) \times F \tag{77}$$

$$\tilde{v}^\ell_{i\alpha} \le M \cdot v^\ell_{i\alpha} \qquad \forall (i, \alpha) \in ((I \setminus \{1\}) \times F) \cup (1, f) \tag{78}$$

$$\tilde{y}^0_\alpha \le M \cdot y^0_\alpha \qquad \forall \alpha \in \{l, r, u\} \tag{79}$$

Depot inclusion constraint:

$$y^0_l + y^0_r + y^0_u \ge 1 \tag{80}$$

Item inclusion constraints:

$$c^e_{i3f} \cdot (w_{i1f} + w_{i3b}) \ge b_i \cdot c^e_{i3f} \qquad \forall i \in I \tag{81}$$

$$c^e_{i3b} \cdot (w_{i1b} + w_{i3f}) \geq b_i \cdot c^e_{i3b} \qquad \forall i \in I \qquad (82)$$

Constraints for the domains of the variables:

$$r^r_{i\alpha} \in \{0,1\} \qquad \forall (i,\alpha) \in (I \setminus \{m-1,m\}) \times F \qquad (83)$$

$$r^w_{i\alpha} \in \{0,1\} \qquad \forall (i,\alpha) \in (I \setminus \{m\}) \times F \qquad (84)$$

$$\ell^\ell_{i\alpha} \in \{0,1\} \qquad \forall (i,\alpha) \in ((I \setminus \{1,2\}) \times F) \cup (2,f) \qquad (85)$$

$$\ell^w_{i\alpha} \in \{0,1\} \qquad \forall (i,\alpha) \in (I \setminus \{1\}) \times F \qquad (86)$$

$$e_{is\alpha} \in \{0,1\} \qquad \forall (i,s,\alpha) \in I \times \{1,3\} \times F \qquad (87)$$

$$t_{i\alpha} \in \{0,1\} \qquad \forall (i,\alpha) \in I \times F \qquad (88)$$

$$w_{is\alpha} \in \{0,1\} \qquad \forall (i,s,\alpha) \in I \times \{1,2,3\} \times F \qquad (89)$$

$$v^r_{i\alpha} \in \{0,1\} \qquad \forall (i,\alpha) \in (I \setminus \{m\}) \times F \qquad (90)$$

$$v^\ell_{i\alpha} \in \{0,1\} \qquad \forall (i,\alpha) \in ((I \setminus \{1\}) \times F) \cup (1,f) \qquad (91)$$

$$y^0_\alpha \in \{0,1\} \qquad \forall \alpha \in \{l,r,u\} \qquad (92)$$

$$y^\ell_0 \in \{0,1\} \qquad (93)$$

$$\tilde{r}^r_{i\alpha} \geq 0 \qquad \forall (i,\alpha) \in (I \setminus \{m-1,m\}) \times F \qquad (94)$$

$$\tilde{r}^w_{i\alpha} \geq 0 \qquad \forall (i,\alpha) \in (I \setminus \{m\}) \times F \qquad (95)$$

$$\tilde{\ell}^\ell_{i\alpha} \geq 0 \qquad \forall (i,\alpha) \in ((I \setminus \{1,2\}) \times F) \cup (2,f) \qquad (96)$$

$$\tilde{\ell}^w_{i\alpha} \geq 0 \qquad \forall (i,\alpha) \in (I \setminus \{1\}) \times F \qquad (97)$$

$$\tilde{e}_{is\alpha} \geq 0 \qquad \forall (i,s,\alpha) \in I \times \{1,3\} \times F \qquad (98)$$

$$\tilde{t}_{i\alpha} \geq 0 \qquad \forall (i,\alpha) \in I \times F \qquad (99)$$

$$\tilde{w}_{is\alpha} \geq 0 \qquad \forall (i,s,\alpha) \in I \times \{1,2,3\} \times F \qquad (100)$$

$$\tilde{v}^r_{i\alpha} \geq 0 \qquad \forall (i,\alpha) \in (I \setminus \{m\}) \times F \qquad (101)$$

$$\tilde{v}^\ell_{i\alpha} \geq 0 \qquad \forall (i,\alpha) \in ((I \setminus \{1\}) \times F) \cup (1,f) \qquad (102)$$

$$\tilde{y}^0_\alpha \geq 0 \qquad \forall \alpha \in \{l,r,u\} \qquad (103)$$

Otto von Guericke University Magdeburg
Faculty of Economics and Management
P.O. Box 4120 | 39016 Magdeburg | Germany

Tel.: +49 (0) 3 91/67-1 85 84
Fax: +49 (0) 3 91/67-1 21 20

www.fww.ovgu.de/femm

ISSN 1615-4274