

Roger KSIAŻEK, Katarzyna GDOWSKA, Radosław KAPŁAN
AGH University of Krakow

ROZMIESZCZENIE PRODUKTÓW W MAGAZYNIE KOMPLETACYJNYM MINIMALIZUJĄCE CZAS KOMPLETACJI ZAMÓWIEŃ PRZY LOSOWYM POPYCIE

Streszczenie. Centra kompletacyjne muszą z jednej strony utrzymywać zapas pozwalający na szybką reakcję na potrzeby rynku, a z drugiej – minimalizować koszty swojego funkcjonowania. Pogodzenie tych dwóch aspektów jest zagadnieniem trudnym i wielopoziomowym. Obejmuje ono nie tylko tak oczywiste kwestie, jak wielkość magazynu czy poziomy zapasów poszczególnych pozycji asortymentowych, lecz także zagadnienia związane z właściwą kolejnością pobrań wynikającą z fizycznych własności kompletowanych produktów. W pracy przedstawiono problem optymalizacyjny równoważenia systemu kompletacji zamówień w celu minimalizacji czasu kompletacji zamówienia (Balancing Product Allocation for Minimising Order-Picking Time, BPAMOPT) oraz dedykowany dla niego model programowania całkowitoliczbowego (MIP). Ze względu na praktyczne zastosowanie zdefiniowanego problemu, tj. konieczność uzyskiwania rozwiązań w akceptowalnym czasie obliczeń, zarekomendowano dedykowaną metodykę postępowania zapewniającą jak największe uproszczenie problemu optymalizacyjnego.

PRODUCT ALLOCATION IN AN ORDER-PICKING WAREHOUSE MINIMIZING ORDER-PICKING TIME UNDER RANDOM DEMAND

Summary. Order-picking centres must, on the one hand, maintain sufficient inventory to respond rapidly to market demand and, on the other hand, minimise their operating costs. Reconciling these two objectives is a complex and multi-level challenge. It encompasses not only evident issues such as warehouse capacity and inventory levels for individual stock-keeping units, but also matters related to determining an appropriate picking sequence based on the physical characteristics of the products being handled. This paper introduces the Balancing Product Allocation for Minimising Order-Picking Time (BPAMOPT) optimisation problem and proposes a dedicated mixed-integer programming (MIP) model to solve it. Given the practical application of the problem, particularly the need to obtain solutions within an acceptable computation time, a dedicated solution methodology is recommended to simplify the optimisation problem as far as possible.

1. Introduction

The fourth industrial revolution is becoming a reality. Dynamic changes and innovations are taking place throughout the value chain, with order-picking centres among its fundamental links. These units are the basis of on-demand logistics, and their continuous development is driven by ever-faster economic change, particularly the rapid growth of e-commerce and rising customer expectations regarding order lead times.

It should be remembered that order-picking centres must, on the one hand, maintain a stock allowing for a quick response to market needs and, on the other hand, minimise their maintenance costs. Reconciling these two aspects is a very difficult and multi-level issue. We are dealing here not only with such obvious questions as the choice of warehouse size or the stock levels of particular items, but also with issues related to the proper picking sequence resulting from the physical properties of the items being picked, as well as with the minimisation of bottlenecks arising during the picking process itself. At the same time, demand at most order-picking centres is a random phenomenon, further complicating the problem.

This paper attempts to solve the problem of allocating products in a warehouse with fixed slots to minimise order-picking time under random demand. The remainder of the paper is organised as follows. Section 1.1. situates the paper within the related literature. Section 2. describes the proposed procedure for balancing an order-picking system and formulates the Balancing Product Allocation for Minimising Order-Picking Time together with a dedicated MIP model. Section 3. reports the computational experiments, and Section 4. concludes the paper and outlines directions for further research.

1.1. Paper positioning

Order picking is commonly regarded as the most labour- and cost-intensive warehouse operation, estimated to account for more than half of a typical warehouse's total operating costs [4]. Comprehensive overviews of the research on the design and control of order-picking systems are provided in [4] and [6], whereas [22] classify the literature from the perspective of combining interdependent planning problems – storage assignment, order batching, zoning, and picker routing – and show that solving these problems jointly, rather than in isolation, offers considerable efficiency gains. The rapid growth of e-commerce has further intensified interest in this area, giving rise to new warehousing systems and new variants of classical decision problems [1].

Within this research field, most studies focus on picker routing and picking policies. The seminal work [17] showed that, for a rectangular single-block warehouse, the picker-routing problem is a solvable case of the travelling salesman problem. Since then, routing research has been extended to numerous layouts and operating conditions. An efficient method for routing order pickers in a warehouse with a non-standard, leaf-shaped layout was proposed in [14]. The problem of dynamic order picking, in which orders arrive while the picking process is underway, was analysed in [13], where a dynamic routing algorithm that allows on-line route updates was developed. In turn, [19] showed that the picker-routing problem can be effectively modelled as a variant of the vehicle routing problem (VRP) and that the resulting solutions can be verified via simulation. It is also worth noting that, despite the progressing automation of picking processes, the human factor remains essential – empirical evidence on human learning in the transition to automated order picking was presented in [12], while [11] demonstrated that stora-

ge assignment decisions can be used to jointly manage picking efficiency and workers' discomfort.

The second major research stream, to which this paper belongs, is the Storage Location Assignment Problem (SLAP), i.e., the problem of assigning products to storage locations so as to optimise a measure of picking performance. Classical storage policies include the cube-per-order index rule [8] and class-based (ABC) storage, introduced in [7] for automated storage/retrieval systems; a simulation-based comparison of storage, picking, and routing policies in manual picker-to-parts warehouses was presented in [16]. A systematic review of the SLAP literature, classifying contributions by solution methods, objectives, and practical considerations, is presented in [18]. Beyond frequency-based policies, several authors exploit information about the structure of orders: [2] used the product structure to reduce picking times, and [23] formulated an SLAP model that accounts for demand correlation patterns. Another group of studies integrates storage assignment with subsequent operational decisions: [5] derived exact route-length formulas used within a storage assignment heuristic, [20] integrated storage location and order-picking decisions in warehouse planning, and [10] proposed an iterative method for the joint dynamic storage location assignment, order batching, and picker-routing problem.

Relatively few papers address precedence constraints in picking, i.e., situations in which some products must be picked before others because of their weight, fragility, stackability, or a preferred unloading sequence. A rich order-picking problem under weight, fragility, and category constraints, motivated by grocery retail practice, was introduced in [3]. Precedence-constrained picker routing combined with an analysis of storage assignment strategies was studied in [24], and a fast simulated annealing method for batching precedence-constrained customer orders was proposed in [15]. From the storage assignment perspective, [21] developed a two-phase heuristic for the SLAP with precedence constraints and validated it in a real retail warehouse. Balancing the workload among pickers or warehouse zones – one of the objectives of the model proposed in this paper – was addressed, among other approaches, in [9], where items were assigned to zones of a synchronised zone-picking system to balance pickers' workloads.

In contrast to the above-mentioned studies, this paper focuses not on routing but on the preceding decision stage – the storage location assignment problem in a warehouse with fixed slots. To the best of our knowledge, the simultaneous consideration of (i) the random nature of demand, expressed by the probabilities of products occurring in orders, (ii) the precedence constraints on pick-ups resulting from the physical properties of the products, and (iii) the balancing of workloads across slot groups within a single MIP formulation has not been addressed in the literature so far. Moreover, the proposed approach explicitly takes into account the requirement of practical applicability of the method, i.e., obtaining solutions within an acceptable computation time. The proposed procedure combines exact optimisation based on the Balancing Product Allocation for Minimising Order-Picking Time, formulated as an MIP problem and applied to aggregated data, with a subsequent decomposition stage. While the decomposition may be performed using fast heuristic algorithms, the design of such algorithms is beyond the scope of this paper.

2. Balancing an Order-Picking System

Allocating products in a warehouse with fixed slots to minimise order-picking time under random demand is computationally complex due to the size, which scales with the number of products in the warehouse. This problem is to be solved using an IT tool that will eventually support decision-making processes related to warehouse management. Taking into consideration the practical use of the defined problem, i.e., obtaining solutions in an acceptable computation time, we recommend the procedure presented in Fig. 1. In this procedure, the optimisation occurs in Step 3, where a mixed-integer programming model is used to solve the Balancing Product Allocation for Minimising Order-Picking Time, ie., to determine the best possible solution. A feature of exact optimisation algorithms is their significant computational complexity and the consequent demand for computing power. Therefore, the optimisation model used in Step 3 operates on suitably aggregated data and finds a solution for them. The resulting solution cannot be applied directly due to data aggregation, so it must be decomposed to obtain a solution at the detailed level of individual products. Only such a comprehensive approach can guarantee the practical use of the model by combining the benefits of combinatorial optimisation with fast, effective heuristic algorithms.

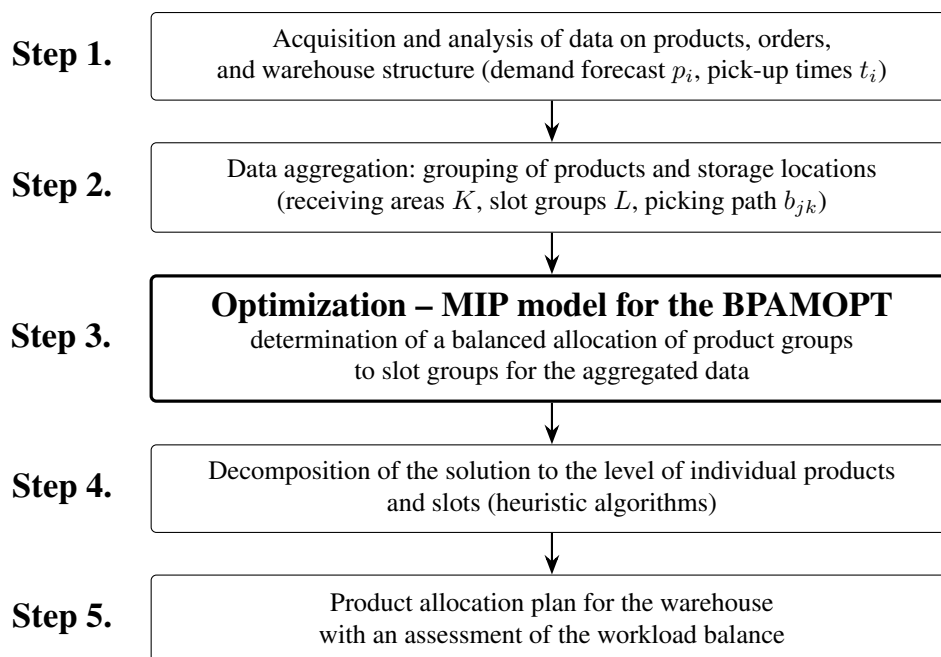


Fig. 1. Procedure for determining the allocation of products in an order-picking warehouse

2.1. Balancing Product Allocation for Minimising Order-Picking Time

In the Balancing Product Allocation for Minimising Order-Picking Time problem, a set of products I to be deployed in a warehouse during a certain planning period can be grouped and assigned to a known set of receiving areas K . In each receiving area, groups of slots can be distinguished, e.g., aisles, racks, or parts of racks. Let the set J contain pairs (j, k) assigning particular slot groups $j \in L$ to a given receiving area k .

During picking, the order in which products are picked from the warehouse must be preserved, so a set R can be determined that contains pairs (i, r) of products such that

any picking order requires picking product i before product r . In the system, the number of units of product i (counted, e.g., in basic packages of the product) which should be placed in the warehouse (d_i^L) in the considered planning period is known. Based on the orders, the number of units of product i scheduled for pickup (d_i^G) in the considered planning period is also known. Products are placed in their respective available storage locations (e.g., slots or groups of slots), from which they will be picked during order picking. If the warehouse practice of continuously replenishing storage locations (with respect to the planning period) by reloading from reserve storage is applied, the number of product units scheduled for pick-up can exceed the number of units of this product allocated to a given slot during the planning period. The pick-up time of one unit of product i is t_i . For each product i , the probability that it appears in an order (p_i) is also known. Each product i can be deployed only in the allowed receiving area k , as indicated by the incidence matrix a_{ik} . Also known is the size of the slot groups (e.g., individual slots, aisles, racks), i.e., the available number of space units g_{jk} in a given slot group j from the receiving area k . Since order picking requires that the products be picked according to the order determined by the elements of the set R , it is possible to determine the sequence numbers b_{jk} for a given slot group in the receiving area k , which determine the order in which the runs are performed during order picking (picking may take place according to some defined spanning tree on a graph representing the warehouse structure).

To determine the final plan for product allocation in the warehouse, for the BPA-MOPT, the MIP model (1a)–(1n) is used. The notation used in the model is presented in table 1.

$$\min: \quad f = \alpha \cdot z + \beta \cdot \sum_{i \in I} \sum_{(j,k) \in J} y_{i,(jk)} \quad (1a)$$

$$\sum_{i \in I} t_i \cdot x_{i,(jk)}^G \cdot \left(1 + \frac{p_i}{1 - p_i}\right) \leq z, \quad (j, k) \in J; \quad (1b)$$

$$\frac{x_{i,(jk)}^G}{d_i^G} = \frac{x_{i,(jk)}^L}{d_i^L}, \quad i \in I, (j, k) \in J; \quad (1c)$$

$$\sum_{i \in I} x_{i,(jk)}^L \leq g_{(jk)}, \quad (j, k) \in J; \quad (1d)$$

$$\sum_{(j,k) \in J} x_{i,(jk)}^G = d_i^G, \quad i \in I; \quad (1e)$$

$$\sum_{(j,k) \in J} x_{i,(jk)}^L = d_i^L, \quad i \in I; \quad (1f)$$

$$x_{i,(jk)}^G \leq M \cdot y_{i,(jk)}, \quad i \in I, (j, k) \in J; \quad (1g)$$

$$x_{i,(jk)}^L \leq M \cdot a_{ik}, \quad i \in I, (j, k) \in J; \quad (1h)$$

$$b_{(jk)} y_{i,(jk)} - b_{(j',k')} y_{r,(j'k')} \leq M(1 - y_{r,(j'k')}), \quad (i, r) \in R, (j, k), (j', k') \in J; \quad (1i)$$

$$x_{i,(jk)}^L \leq M \cdot y_{i,(jk)}, \quad i \in I, (j, k) \in J; \quad (1j)$$

$$x_{i,(jk)}^G \geq 0, \quad i \in I, (j, k) \in J; \quad (1k)$$

Tabela 1

Notation used in the BPAMOPT

Sets	
I	set of products to be placed in the warehouse
K	set of receiving areas
L	set of available slot groups (e.g., slots, aisles, racks, parts of racks)
J	set of slot groups assigned to the receiving areas, i.e., a set of pairs (j, k) for slot groups j in receiving area k
R	set of product pairs (i, r) such that order picking requires picking product i before product r
Parameters	
d_i^L	number of units of product i to be deployed in the warehouse in a given planning period
d_i^G	number of units of product i scheduled to be picked in a given planning period
p_i	probability of occurrence of product i in an order (forecast based on historical data according to the method adopted by the company)
t_i	time to pick up one unit of product i
g_{jk}	available storage space (measured by the number of units of space) in slot group j in receiving area k
a_{ik}	$= 1$ if product i belongs to receiving area k ; 0 otherwise
b_{jk}	precedence priority of slot group j in receiving area k on the picking path, expressed by a sequence number for a given location (j, k) that determines the order in which order picking is performed, $b_{jk} \in \{1, 2, \dots\}$
α, β	weights of the objective function components
M	a sufficiently large constant
Decision variables	
$x_{i,(jk)}^L$	number of units of product i stored at location j in receiving area k
$x_{i,(jk)}^G$	number of units of product i scheduled for pick-up at location j in receiving area k
$y_{i,(jk)}$	$= 1$ if product i has been allocated to storage location j in receiving area k ; 0 otherwise
z	the highest workload found for the slot groups of the warehouse (i.e., the elements of set L) in the analysed system

$$x_{i,(jk)}^L \geq 0 \quad , i \in I, (j, k) \in J; \quad (1l)$$

$$y_{i,(jk)} \in \{0, 1\} \quad , i \in I, (j, k) \in J; \quad (1m)$$

$$z \geq 0 \quad (1n)$$

The objective function (1a) minimises the maximum workload time for the solved problem with a fixed weight α and minimises the dispersion of product allocation in the warehouse with a fixed weight β . Constraint (1b) determines the maximum workload time in the solution – all remaining pick-up locations will have workloads less than or equal to z . Constraint (1c) allows scheduling the pick-up of products in proportion to the number of products allocated to the given location (j, k) . Constraint (1d) ensures

that, for each location j in receiving area k , no more products will be scheduled for allocation than the available storage space g_{jk} . Constraint (1f) ensures that the planned number of units of a given product i will be allocated in the warehouse, while constraint (1e) assigns to all the planned locations of a given product i the planned volume of its pick-up (proportional to the number of allocated products in accordance with (1c)). Additionally, constraint (1g) allows pick-up to be scheduled only at locations where product i has been allocated. Constraint (1h) allows a product to be allocated only in the receiving area k to which the given product i belongs. In turn, the correct arrangement of products on the defined picking path is enforced by constraint (1i): if a given product i must be picked before product r , then product i must be located at a location (j, k) whose sequence number b_{jk} is smaller than or equal to the sequence number $b_{j'k'}$ of the location of product r . Constraint (1j) enforces the correct values of the binary variables $y_{i,(jk)}$. Constraints (1k)–(1n) ensure that the individual decision variables take the correct values.

3. Computational Experiments

The computational experiments were conducted to evaluate the scalability of the proposed MIP model for BPAMOPT. A series of eight test instances of increasing size was generated based on the structure of a real order-picking warehouse. The experiments were performed on a PC equipped with an Intel Core i7 processor and 32 GB RAM, using Python 3.12, Pyomo 6.x and the CBC solver. The computation time limit was set to 300 seconds.

Each aggregated product may represent a group of similar individual products; therefore, the actual warehouse may contain several thousand products while the optimisation model operates on a much smaller number of aggregated entities. The test dataset parameters and the obtained results are presented in table 2.

Tabela 2

Computational results for the test instances							
Instance	$ I $	$ L $	Variables	Constraints	Time [s]	Gap [%]	Status
S1	3	5	46	80	0.14	0.000	Optimal
S2	6	10	181	391	0.12	0.000	Optimal
S3	10	15	451	1136	0.61	0.000	Optimal
S4	15	20	901	2910	0.17	0.000	Optimal
S5	20	25	1501	5259	8.74	0.000	Optimal
S6	30	40	3601	21190	300.74	0.001	Time limit
S7	50	60	9001	73720	302.35	0.012	Time limit
S8	75	80	18001	165013	304.19	0.006	Time limit

The number of variables and constraints increases rapidly with the problem size, reflecting the combinatorial nature of the proposed formulation. As expected, the computation time increased with the number of products and slot groups. The first five instances were solved to proven optimality, whereas the remaining instances reached the imposed time limit of 300 s. Nevertheless, the optimality gaps obtained remained below 0.02%, indicating that the reported solutions are very close to optimal. The results obta-

ined confirm that the proposed aggregation approach enables the solution of practically relevant instances using exact optimisation techniques. The proposed MIP model solved instances containing up to 20 aggregated products to proven optimality within a few seconds. Larger instances reached the imposed 300 s time limit; however, the resulting optimality gaps remained below 0.02%, indicating that the solutions are very close to optimal.

4. Conclusions

This paper presented the optimisation problem of the Balancing Product Allocation for Minimising Order-Picking Time. A mixed-integer programming model was formulated that simultaneously accounts for the random nature of demand (through the probabilities of products occurring in orders), the precedence constraints on pick-ups arising from the physical properties of the products, and the balancing of workloads across warehouse slot groups. Due to the significant computational complexity of the problem, a procedure was recommended which combines exact optimisation on aggregated data with a heuristic decomposition of the solution to the level of individual products. Such an approach enables the practical use of the model in an IT tool supporting decision-making processes related to warehouse management, while maintaining an acceptable computation time. The computational experiments confirmed that the proposed model can solve aggregated problem instances within acceptable computation times, which makes it suitable as the optimisation core of the proposed decision-support procedure.

Further work will include extending the model with dynamic stock replenishment during the planning period, developing dedicated heuristic algorithms for large-scale instances, and verifying the proposed methodology in an industrial setting.

The research was financed by a subsidy and the program „Excellence Initiative – Research University” for the AGH University of Krakow.

REFERENCES

1. Boysen N., de Koster R., Weidinger F.: Warehousing in the e-commerce era: A survey. *European Journal of Operational Research*, 2019, Vol.277(2), p. 396–411.
2. Brynzér H., Johansson M. I.: Storage location assignment: Using the product structure to reduce order picking times. *International Journal of Production Economics*, 1996, Vol.46–47(1), p. 595–603.
3. Chabot T., Lahyani R., Coelho L. C., Renaud J.: Order picking problems under weight, fragility and category constraints. *International Journal of Production Research*, 2017, Vol.55(21), p. 6361–6379.
4. de Koster R., Le-Duc T., Roodbergen K. J.: Design and control of warehouse order picking: A literature review. *European Journal of Operational Research*, 2007, Vol.182(2), p. 481–501.
5. Dijkstra A. S., Roodbergen K. J.: Exact route-length formulas and a storage location assignment heuristic for picker-to-parts warehouses. *Transportation Research Part E: Logistics and Transportation Review*, 2017, Vol.102, p. 38–59.

6. Gu J., Goetschalckx M., McGinnis L. F.: Research on warehouse operation: A comprehensive review. *European Journal of Operational Research*, 2007, Vol.177(1), p. 1–21.
7. Hausman W. H., Schwarz L. B., Graves S. C.: Optimal storage assignment in automatic warehousing systems. *Management Science*, 1976, Vol.22(6), p. 629–638.
8. Heskett J. L.: Cube-per-order index – a key to warehouse stock location. *Transportation and Distribution Management*, 1963, Vol.3, p. 27–31.
9. Jane C.-C., Lai H. Y.-W.: A clustering algorithm for item assignment in a synchronized zone order picking system. *European Journal of Operational Research*, 2005, Vol.166(2), p. 489–496.
10. Kübler P., Glock C. H., Bauernhansl T.: A new iterative method for solving the joint dynamic storage location assignment, order batching and picker routing problem in manual picker-to-parts warehouses. *Computers & Industrial Engineering*, 2020, Vol.147, p. 106645.
11. Larco J. A., de Koster R., Roodbergen K. J., Dul J.: Managing warehouse efficiency and worker discomfort through enhanced storage assignment decisions. *International Journal of Production Research*, 2017, Vol.55(21), p. 6407–6422.
12. Loske D.: Empirical evidence on human learning and work characteristics in the transition to automated order picking. *Journal of Business Logistics*, 2022, Vol.43(3), p. 302–342.
13. Lu W., McFarlane D., Giannikas V., Zhang Q.: An algorithm for dynamic order-picking in warehouse operations. *European Journal of Operational Research*, 2016, Vol.248(1), p. 107–122.
14. Masae M., Glock C. H., Vichitkunakorn P.: A method for efficiently routing order pickers in the leaf warehouse. *International Journal of Production Economics*, 2021, Vol.234, p. 108069.
15. Matusiak M., de Koster R., Kroon L., Saarinen J.: A fast simulated annealing method for batching precedence-constrained customer orders in a warehouse. *European Journal of Operational Research*, 2014, Vol.236(3), p. 968–977.
16. Petersen C. G., Aase G.: A comparison of picking, storage, and routing policies in manual order picking. *International Journal of Production Economics*, 2004, Vol.92(1), p. 11–19.
17. Ratliff H. D., Rosenthal A. S.: Order-picking in a rectangular warehouse: A solvable case of the traveling salesman problem. *Operations Research*, 1983, Vol.31(3), p. 507–521.
18. Reyes J. J. R., Solano-Charris E. L., Montoya-Torres J. R.: The storage location assignment problem: A literature review. *International Journal of Industrial Engineering Computations*, 2019, Vol.10(2), p. 199–224.
19. Shetty N., Sah B., Chung S. H.: Route optimization for warehouse order picking operations via vehicle routing and simulation. *SN Applied Sciences*, 2020, Vol.2(2), p. 311.

20. Silva A., Coelho L. C., Darvish M., Renaud J.: Integrating storage location and order picking problems in warehouse planning. *Transportation Research Part E: Logistics and Transportation Review*, 2020, Vol.140, p. 102003.
21. Trindade M. A. M., Sousa P. S. A., Moreira M. R. A.: Ramping up a heuristic procedure for storage location assignment problem with precedence constraints. *Flexible Services and Manufacturing Journal*, 2022, Vol.34(3), p. 646–669.
22. van Gils T., Ramaekers K., Caris A., de Koster R. B. M.: Designing efficient order picking systems by combining planning problems: State-of-the-art classification and review. *European Journal of Operational Research*, 2018, Vol.267(1), p. 1–15.
23. Zhang R.-Q., Wang M., Pan X.: New model of the storage location assignment problem considering demand correlation pattern. *Computers & Industrial Engineering*, 2019, Vol.129, p. 210–219.
24. Žulj I., Glock C. H., Grosse E. H., Schneider M.: Picker routing and storage-assignment strategies for precedence-constrained order picking. *Computers & Industrial Engineering*, 2018, Vol.123, p. 338–347.