Data-driven decision support system for managing item allocation in an ASRS: A framework development and a case study



(Article begins on next page)

# Data-driven decision support system for managing item allocation in an ASRS: a framework development and a case study

Sara Antomarioni*[a], Laura Lucantoni[a], Filippo Emanuele Ciarapica[a], Maurizio Bevilacqua[a]
(s.antomarioni@univpm.it, l.lucantoni@pm.univpm.it, f.ciarapica@univpm.it, m.bevilacqua@univpm.it)

[1]Department of Industrial Engineering and Mathematical Science – Università Politecnica delle Marche, via Brecce Bianche 12, 60131, Ancona (Italy)

*Corresponding author

## Abstract

When dealing with Automated Storage and Retrieval Systems (ASRS), the allocation of items to the most convenient storage location depends on the vast amount of data produced internally (e.g., Enterprise Resource Planning, Manufacturing Enterprise Systems) and externally (e.g. Supply Chain Management). Moreover, a proper item allocation in the warehouse has a strong influence on the warehouse saturation levels and picking times.

In this perspective, the present work proposes the application of data-driven algorithms for managing items in an Automated Storage and Retrieval System (ASRS) in order to reduce the picking times and storage space. Specifically, a four-layer framework is adopted for collecting data produced by different information sources and analyzing them through a data-driven approach. The analytics layer is performed by combining the Association Rule Mining (ARM) technique, to investigate the network of influences among data collected, and a simulation approach for assessing the feasibility of the proposed implementation. The Association Rule Mining allows company managers to identify the components that should be located on the same tray in the ASRS, defining the couples of items frequently picked together in order to reduce the total picking time. The proposed approach is applied to the case study of a shoe manufacturing company to explain the research approach and show how the implementation of the data-driven methodology can provide valuable support in defining item allocation and picking rules. The proposed Association Rule Mining method is new in this context and it has shown a positive impact in comparison to traditional solutions of warehouse management, providing a complete overview of the items' interactions and identifying communities of items that define local and global patterns and locate influential entities.

KEYWORDS: Warehouse Management; Automated Storage and Retrieval System (ASRS); Data-driven techniques; Association Rule Mining

# 1. Introduction

In the current industrial scenario, the growing amount of data produced by different information sources represents an opportunity in warehouse management, as well as a criticality to turn the information into useful knowledge in order to enhance the decision-making process (Bevilacqua, Ciarapica, Diamantini, & Potena, 2017). These aspects highlight the need to support the organizations in extracting reliable information from the vast amount of data produced internally (e.g., Enterprise Resource Planning, Manufacturing Enterprise Systems) and externally (e.g., Supply Chain Management). Automated Storage and Retrieval Systems (ASRS) are not exempt from such criticism: indeed, several data have to be managed when organizing items in an ASRS, like, for example, stock level, picking lists, production schedules, order policies, and schedules. Item allocation in an ASRS should be carried out consciously to facilitate the picking process and achieve consistent performance in terms of picking time. Indeed, since the picking process order is a repetitive activity in a warehouse, it has a relevant impact on the overall efficiency (Yener & Yazgan, 2019). Additionally, a well-managed ASRS enables the limitation of wastes in terms of time and space (Bevilacqua, Ciarapica, & Antomarioni, 2019).

When dealing with ASRS, the allocation of items to the most convenient storage location represents the key to reducing the picking times. In this sense, there is a need for decision support systems capable of (a) guiding organizations using the data daily produced by the information sources and (b) defining the management policy for item allocation. Dukic & Oluic (2007) underlined that warehouse operators spent almost 90% of total time on order-picking activities, and 55% of all operating costs are attributed to order-picking. In recent years, several works on order picking in warehouse management systems have been developed for both traditional and ASRS warehouses (Anđelković & Radosavljević, 2018; Hofmann, 2018; Petersen & Aase, 2017). All these studies highlighted that the picking process is one of the most laborious and crucial factors for efficient warehouse management.

In this context, this work aims at developing a data-driven decision support system providing a roadmap for companies in allocating items in an ASRS and improving the picking process, capitalizing on the opportunity offered by data analytics techniques in view of intelligent internal logistics.

Many techniques and solutions have been proposed in the literature for strategic warehouse management (e.g., Shiau & Lee, 2010; Peixoto et al., 2016; Zhang et al., 2019). Some of these models are used to implement storage and picking solutions for manually retrieved systems (Caron et al., 2010; Yener& Yazgan, 2019); others are used for modeled or simulated ASRS (Calzavara et al., 2018; Bevilacqua et al., 2019). In the ASRS research field, the research focus is mainly on defining storage assignment of the items in order to reduce travel distances and the consequent time for the composition of orders. Less attention has been paid to the development of a framework for both the organization of items in an ASRS and the picking process using data-driven techniques. Although the existing research is valuable, a framework based on data-driven algorithms that structurally analyses and predicts relationships between components picked in an ASRS and identifies communities of items to be closely stored in order to optimize picking time and warehouse space, to the best of the authors' knowledge, is not present in literature.

For this reason, the framework proposed in this work aims to address this gap by introducing an innovative decision-making tool in this critical activity through a data-driven methodology. In particular, the proposed approach is based on the simultaneous adoption of the Association Rule Mining (ARM), for identifying the hidden interactions between picked items and simulation models for supporting company managers in defining the feasibility of the item allocation, as well as in developing what-if scenarios.

The Association Rule Mining (ARM) technique, for the data analytics step, allows warehouse managers to identify the components that should be located on the same tray in the ASRS, defining the pairs of items frequently picked together. This approach aims at reducing the picking time since the warehouse operators have to take the components from a smaller number of trays during the total picking time, thus reducing its idle time and the number of single and dual command cycles of the order picking process.

In order to present the proposed approach, the remainder of the paper is organized as follows: Section 2 is dedicated to a summary of the existing literature dealing with data-driven methods for warehouse management and the picking process; in Section 3, the framework is described, while in Section 4 an application to a case study is proposed. Section 5 discusses the results proposed by the study, highlighting the theoretical and practical contributions; lastly, Section 6 draws the conclusions and remarks, as well as making future research directions.

# 2. Literature Review

The interaction between warehouse storage policies and efficiency has emerged in several studies emphasizing the importance of a comprehensive overview of warehouse management (Petersen & Aase, 2017; Hui et al., 2015).

In Figure 1, a classification of the main literary contributions focusing on the item allocation is reported. As immediately noticeable, the majority of the contributions involve the storage allocation of the items in traditional picker-to-part warehouses (Manual, in the legend). Automated warehouses are less analyzed by researchers, regardless of the methodology implemented. Among the strategies for item allocation, the Cube-per-Order Index (COI) can be mentioned: it was first proposed by Heskeet in 1963, and it is used to assign product classes to storage location defined as the ratio of the item storage space requirement. Some authors use the COI-based storage assignment to improve the subsequent picking activities (Caron et al., 2010; Hwang et al., 2004) through the development of simulated scenarios guiding the decision-making process. Yang and Nguyen (2016) propose a heuristic approach in which the COI-based storage assignment is integrated with the Principal Component Analysis in order to create more precise clusters of items.

The COI assignment approach can also be compared with the Class-Based Storage Assignment (CBS), as showed by Chand and Chan (2011). In CBS, SKUs are ranked according to their total picking volume (Petersen & Aase, 2017) and partitioned into several classes according to their total popularity. SKUs are then randomly assigned to storage locations within their respective class storage area (Petersen et al., 2004). Observing Figure 1, it can be noticed that the CBS approach is the one presenting the highest number of applications to automated warehouses. For example, Bortolini et al. (2015) propose three different CBS approaches and simulate some scenarios to determine the best storage assignment in order to make the picking process efficient, while Popović et al. (2014), based on a CBS strategy, implement different heuristics to improve the performance of an ASRS. Similarly, Li et al. (2015) test a class-based storage assignment heuristic aiming to store similar items close together in the ASRS, comparing the traditional ABC allocation and a product affinity-based strategy developed through data mining analysis.

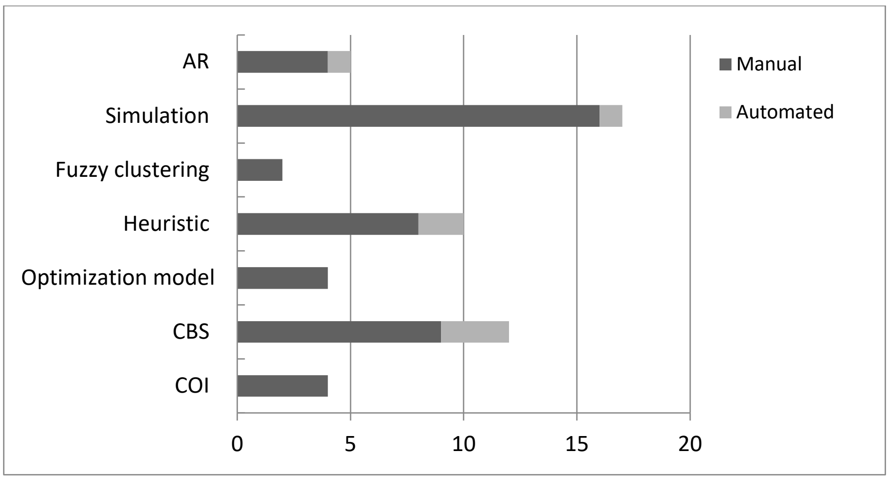


*Figure 1 Storage policies classification of the existing contributions.*

In recent literature, more efficient storage assignment methods have been proposed using the fuzzy clustering technique. For instance, Hui et al. (2015) propose an approach based on fuzzy clustering and association rule mining, aiming to closely store similar products or products frequently retrieved together: their case study is based on a food warehouse. Hence the storage assignment should take into account the retrieval of items frequently picked together in order to reduce their stay out of the warehouse and, thus, avoid the damaging and spoilage of products.
Researchers like Accorsi et al. (2014) and Quintanilla et al. (2015) have developed heuristic algorithms for storage system design and item allocation, while Chen et al. (2005) rely on the definition of the association rules among items frequently picked together to place them in the warehouse.

As shown in Figure 1, several studies in the literature are referred to a traditional warehouse. For an automated warehouse design, the simulation approach is one of the main methods used to develop a data-driven decision support system for the picking process. Peixoto et al. (2016) highlighted that real warehouse systems are often too complex to be evaluated analytically. In contrast, simulation models are more precise and naturally able to integrate the stochastic and dynamic nature of the decisions (Yener &Yazgan, 2019).
Since only individual item attributes are considered in different storage allocation methods, Zhang et al. (2019) developed a heuristic and simulated algorithm based on demand association patterns without considering the batching problem. Hui et al. (2015) proposed a tri-modular intelligent fuzzy-based storage assignment system, integrating fuzzy logic and Association Rule Mining to analyze the hidden relationships of product sales. In order to solve the assignment problems and determine which items should be closely allocated in an ASRS, more data are needed to improve picking efficiency. In this sense, Association Rule (AR), an algorithm for variable interaction and pattern recognition of a set of frequently collected articles concurrently, is considered a valid solution for reducing the number and the duration of picking processes (Bevilacqua et al., 2019). Additionally, this approach avoids the need to formulate a research hypothesis for factor combination before formal evaluation, which may become practically infeasible even for a moderately sized set of variables (Ciarapica et al., 2019). Besides, the 'support' of the Association Rule is used to select the items more frequently picked, and the 'confidence' for the definition of the ones having a strong relationship. This choice contrasts with Chen et al. (2005), who, in their application to a manual warehouse, only considered the support in allocating the items closely.
In light of the existing literary contributions, it can be noticed how the current research on items allocation is mainly focused on manual warehouses. Moreover, in the ASRS research field, many works primarily paid attention to the items storage assignment to decrease travel distances and the consequent time for the order composition. A framework for both the items refilling in an ASRS and the picking route using data-driven techniques has not been proposed in literature. Hence, considering the benefits harbored by the introduction of the Association Rule Mining (ARM), this work proposes an innovative approach for managing the allocation of the items in an ASRS. Specifically, ARM is used to predict the couples of items frequently picked together, so that they can be placed closely inside the ASRS and, consequently, the future pickings can be made more efficiently, as verified in the proposed simulations.

# 3. Research approach

In this work, a four-layer framework (Figure 2) is proposed for managing external and internal data related to an automated warehouse and picking processes. Specifically, the framework defines a data-driven strategy for identifying the best item allocation in an ASRS in terms of warehouse saturation levels and picking time.
Each layer of the framework is dedicated to a specific bundle of activities. The first layer regards data gathering, the second layer the management and organization of this data; then the data analytics is performed in the third layer, and, finally, the allocation is implemented and the results obtained are controlled in the Allocation Layer.

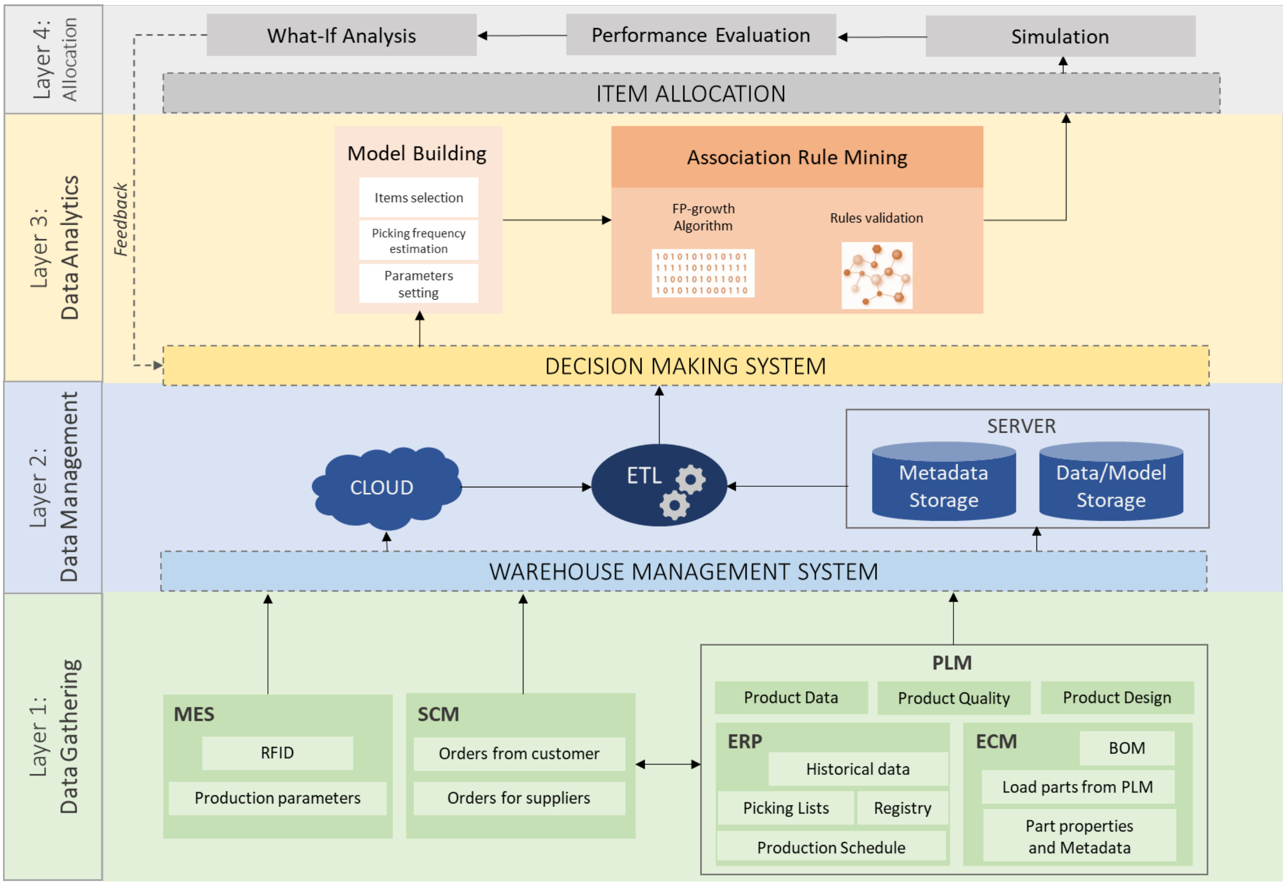


*Figure 2 Four-layer framework for managing an ASRS and the picking process*

## 3.1 Data gathering layer

The data gathering layer represents the foundation of the whole framework since the aim of the work is to define a data-driven roadmap to facilitate the warehouse management and picking process. Indeed, a robust database (Warehouse Management System) is needed in order to perform an accurate and meaningful analysis. Different data sources have to be considered and integrated into the warehouse management system to achieve this objective. In the following bullet list, the main data contained in and extracted from the information systems are exemplified:

- Product Lifecycle Management (PLM): information on product design, general data, product quality, Bill of Materials (BOM), production processes, and part properties are extracted from the PLM. In particular, information concerning item volume, weight, and storage conditions is necessary for defining the item allocation rules.
- Manufacturing Execution System (MES): data regarding the production processes (productivity, capacity, raw material availability, work in progress, and final product currently available) are collected from the MES. Additionally, the parameters tracked by the RFID systems, as well as machinery parameters, are useful to integrate production information and define the picking rules.
- Supply Chain Management (SCM) covers a variety of company logistics activities and provides data regarding the definition of the service levels to the end customer, the operating costs and the capital employed, the demand forecasting and planning, order processing, and optimal resource utilization (materials, production plants, and workers). This information is used both for picking planning and item allocation.

It should be pointed out that, in this work, a general framework is proposed. Hence, each company may use only some of the previously mentioned data sources, depending on their information system structure and on the available data. In any case, this first layer represents the basis for building a warehouse management system to have optimal information management, which will be described in the following section.

## 3.2 Data management layer

The second layer of the proposed framework regards the management of the collected data. It is necessary to integrate them opportunely in the warehouse management system and, considering the amount of data, this is not a trivial step since there are many data sources to take into consideration. Data can be stored into cloud applications or local servers. To carry out a meaningful analysis, a set of consistent data has to be analyzed. Hence, measurement errors (e.g., misreading and repetitions) must be eliminated or substituted by valid measurements. Besides, the heterogeneities generated by duplicate data from different sources using different terminology have to be standardized. Lastly, only the data relevant for the aim of the analysis should be considered, thus leaving only the relevant ones. Through the Extraction, Transformation, and Loading (ETL) process, a unique dataset to support the decision-making process is created, that is, a single information source from which to carry out the actual analysis.

## 3.3 Data analytics layer

The third layer is dedicated to data analytics, and it consists of a preliminary data processing step and Association Rule Mining step integrated into a general allocation process. During the first step, the preliminary data processing and items received from suppliers are collected from the ETL system.

The second step, Association Rule Mining (ARM), regards the identification of pairs of items frequently picked concurrently. Indeed, such items should be allocated closely in the ASRS (if possible, on the same tray) in order to reduce the picking time. According to the proposed approach, the warehouse operators will be able to pick up components from a smaller number of trays during the total picking time, thus reducing its idle time and the number of single and dual command cycles.

### 3.3.1 Association Rule Mining

The Association Rule Mining is an analytic technique aimed at finding valuable relations among attributes from large datasets (Buddhakulsomsiri et al., 2006). Specifically, this methodology ensures the identification of attribute-value conditions frequently occurring contextually in order to extract knowledge from the big data produced daily.

A formal definition of an Association Rule (AR) can be the following: let $A = \{a_1, a_2, \ldots, a_n\}$ be a set of boolean data hereafter called *items* and $V = \{v_1, v_2, \ldots, v_m\}$ be the set of *transactions;* each transaction vi is composed of a set of items, namely an *itemset*, taken from A. An AR is an implication, $I \rightarrow J$, where $I$ and $J$ are itemsets taken from A (i.e., $I, J \subseteq A$), and they have no items in common ($I \cap J = \emptyset$).

In the proposed application, $a_1$ represents the picking of item $i_1$, while the transaction $v_1$ represents the set of components belonging to the same picking list of $i_1$. The quality of an AR can be evaluated through several metrics, among whom the support (Supp) and confidence (Conf) should be recalled:

- $Supp\ (I \rightarrow J) = \frac{\#\{I\ , J\ \}}{\#\{V\})}$: this metric measures the statistical significance of a rule, hence it is calculated as the probability of finding both $I$ and $J$ in a transaction; thus, it can also be seen as a joint probability: $Supp\ (I,J) = P(I,J)$
- $Conf\ (I \rightarrow J) = \frac{Supp\ (I,J)}{Supp\ (I)}$: this metric measures the strength of a rule; hence it is calculated as the probability of finding item J in a transaction containing I. In other words, it is the conditional probability $P(J \mid I) = Conf(I,J)$.

In this work, the algorithm selected to perform the frequent itemset mining is the FP-growth (Han, Cheng, & Xin, 2007). The procedure to mine the ARs can be summarized as follows:

1. The minimum support and minimum confidence thresholds are defined by the decision maker;
2. FP-growth algorithm is executed to generate the Frequent Itemsets (FI);
3. Combining the items included in each FI, the ARs with a confidence higher than the minimum confidence threshold are mined.

In comparison with other applicable algorithms (e.g., Apriori or ECLAT), the FP-growth is more efficient in terms of memory requirement and time to conduct tests. Indeed, the itemset generation follows a “divide and rule" path: only the ones respecting the minimum support threshold are generated and considered in the following computation, leading to a lower growth of the itemsets to analyze.

### 3.3.2 Application of Association Rule Mining for item allocation in an ASRS

In order to explain the proposed approach, based on ARM for item allocation in an ASRS, we can use a simplified example. Let "A", "B", "C", "D", and "E" be the items managed by a company. At first, we analyze the Picking Lists that involved these items "A", "B", "C", "D", and "E" in the last year.
Supposing we have five picking lists in the last year, as represented in Figure 3 by five trolleys, each trolley represents a transaction and is, therefore, composed of itemsets.


| Rules | Support (I → J) | Confidence (I → J) |
|---|---|---|
| A → D | 2/5 | 2/3 |
| A → C | 2/5 | 2/3 |
| A → B | 1/5 | 1/3 |
| C → A | 2/5 | 1/2 |
| C → D | 2/5 | 1/2 |

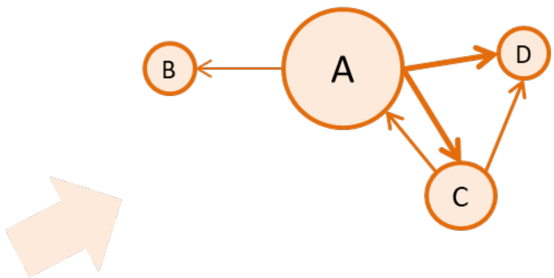


*Figure 3 Example of the application of Association Rule Mining for item allocation in ASRS*

The table reported in Figure 3 contains some of the ARs mined from the exemplified picking lists. Considering the first rule in the table of Figure 3, the "Support" index provides an indication of how frequent the combination of itemsets (e.g., $A \rightarrow D$) appears in the dataset, while the "Confidence" indicates the conditional probability, i.e., the probability of finding the consequent "D" in transactions under the condition that these transactions also contain the head of the rule "A".
In the network of Figure 3, each node represents an item to be stored in the ASRS; an edge among two nodes only exists if an association rule among the two items exists. The thickness of the edges is proportional to the confidence of the association rule among the nodes, i.e., the weight of the edge; the dimension of the nodes represents the out-degree of the node, that is, the weighted sum of the outgoing edges from it.
The warehouse management system repeats this analysis every time a supplier delivers a new item, and the company has to allocate this item in the ASRS.
If a supplier delivers a new lot of items "A" in the future, this lot of "A" must be placed in the same tray as "D", because, from historical data, when the warehouseman picks "A", he usually picks "D" as well.
By following this procedure, repetitively, it is possible to reduce the number of trays to be retrieved to complete a picking list. Picking time is reduced as more and more similar orders from customers are received. For example, if a picking order is composed of four items, all located in the same tray, only one tray should be retrieved. In the worst case, instead, it would be necessary to move all four trays in order to collect items.

## 3.4 Allocation layer

The last layer of the framework is dedicated to the allocation of the items to a specific tray and to the simulation and performance evaluation of the proposed item allocation. Indeed, the coupling proposed through the ARM is used to define the item allocation to the trays. Simulation is used to perform different tests on the trays partitioning (e.g., modifying the number of stock keeping units per tray). So, this step supports company managers in defining whether the configuration recommended in the algorithm is feasible (e.g., all the items can be stored in the ASRS because the maximum number of trays required is lower than the available ones) or if some adjustments are needed. In this sense, simulating the average stock level per day and dividing the items into SKUs and trays allows one to see the quality of the solution provided in the first instance.

### 3.4.1 Item allocation algorithm

Figure 4 shows the general procedure for item allocation during the regime functioning of the ASRS. When a new lot ($Q_i$) of an item $i$ has to be stored there is a ***preliminary scanning*** among all the ASRS trays ($k$), in order to check if the item "$i$" is already contained in one of the trays of the ASRS. If there is a tray $k$ that already contains item $i$, it is verified whether there is available space in the same tray to hold $Q_i$.
If the total quantity storable in tray $k$ ($Q_k$) exceeds the $Q_i$, $Q_i$ is completely stored in tray $k$; otherwise, a further tray, if available, is recalled ($k = k + 1$), and the procedure restarts.
If the item $i$ is not present in any tray of the ASRS or if the available space is not sufficient to store the total amount of item $i$, then the remaining amount has to be stored in another tray.
In this case, the ***ARM*** method can be applied. The ARs containing $i$ (e.g., $i \rightarrow t$ and $t \rightarrow i\ \forall\ item\ t$) are scanned, and the one with the highest confidence is selected. If there is a tray containing item $t$ with enough space to store item $i$, then item i is allocated to this tray. Otherwise, the following rule in terms of confidence is selected, and the same procedure is applied. As previously stated, the aim of the ARM is to define the couples of items frequently picked together so that they can be placed near to each other in order to reduce the picking time. Indeed, when the probability of picking two items together is high, having them on the same tray has a relevant impact in reducing the picking time.
If there are no associated items in the ASRS, or if the trays containing them are saturated, then item $i$ is assigned to an empty tray closest to the picking point.
When there are no more free trays, the items will be stored out of the ASRS in a traditional shelving warehouse.
The procedure is repeated until all items (for the items $i$ ranging from 1 to $n$) have been stored.

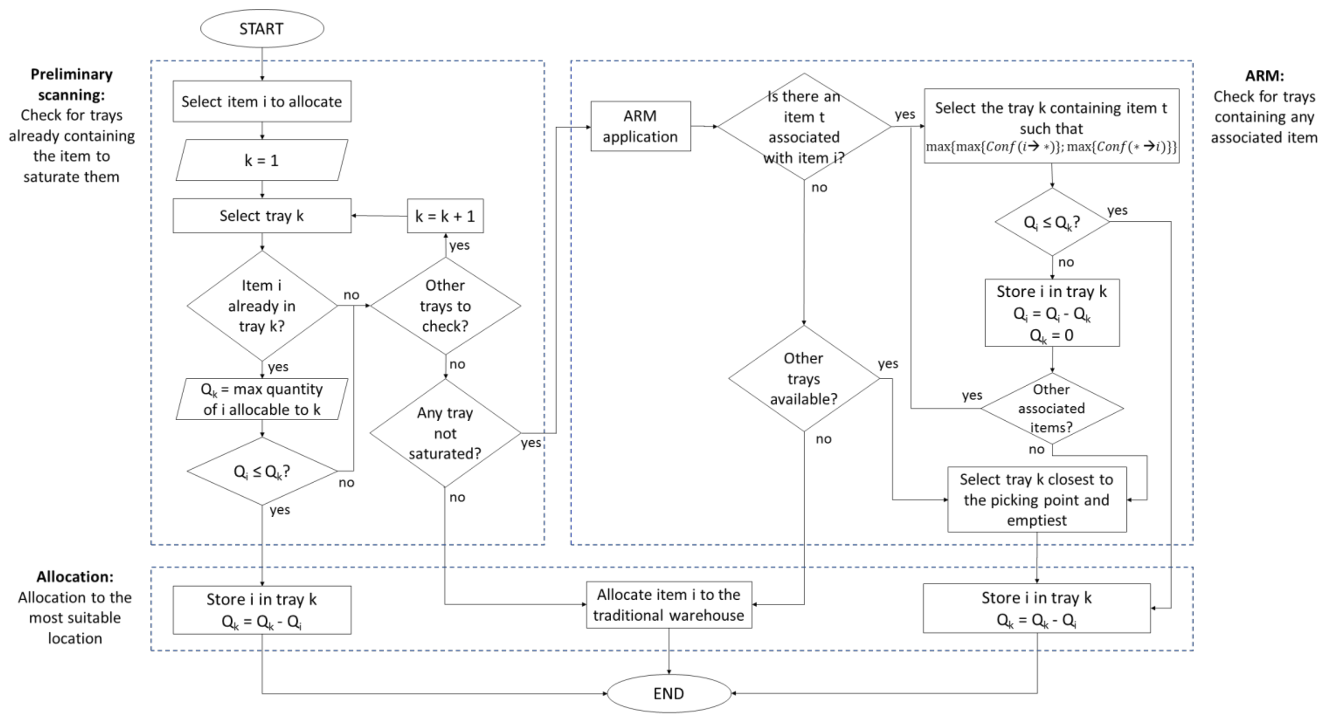

*Figure 4 General procedure for item allocation during the regime functioning of the ASRS.*

# 4. Research approach application

The framework proposed in figure 2 has been implemented to define the warehouse management policy of a shoe manufacturing company during its transition from a standard picker-to-part approach to an automated part-to-picker one. The company sells high-value footwear in the fashion sector and has a turnover of about 100 million €.

The ASRS of the company being studied is a vertical warehouse, divided into three modules. The machine handling unit consists of a vertical trolley equipped with motorization, which activates the upward and downward movement of the trolley above which the horizontal trolley moves. On the horizontal trolley, in turn, the gripping tray system is inserted. Absolute encoders control all the axes of the handling unit mentioned above. Trays are the storage unit managed by the warehouse , and the maximum number of trays storable in the ASRS is equal to 550. Trays are equipped with sliding pads that allow them to slide on the storage shelves. They have the fixing arrangements for the insertion of the perimeter lifting edges and for the introduction of the dividers and separators that determine their division into compartments (hereafter considered as Stock Keeping Unit - SKU). Perimeter elevations, longitudinal dividers, and separators are optional. An example of tray dimensions and a possible partition into six SKUs are reported in Figure 5.

Worth noting is that the current study addresses the allocation of the items to the ASRS trays while the optimal positioning of trays is updated daily on the basis of a class-based storage algorithm included in the software of the ASRS. Specifically, in this application, the trays more frequently recalled for the picking and refilling practices are situated closer to the picking point.

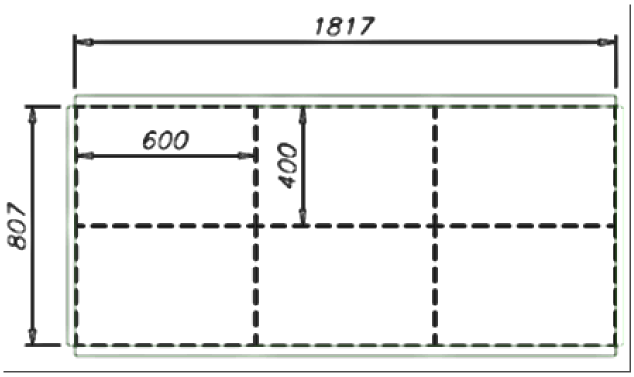


*Figure 5 Tray's dimension and example of a possible partition into six SKUs.*

## 4.1 Layers 1 and 2: data gathering and management

In order to define the picking orders to get information on the items frequently retrieved together, some information is needed: the customer orders have to be accessed from the Supply Chain Management system in order to be aware of the products that have to be produced; consequently, it is necessary to access the bill of materials - stored in the PLM - of such products in order to identify the items to be picked and, then, build the picking list.

SCM and MES data have been extracted from a cloud application, while the PLM from the company's server. Data collected from the different information systems employed in the company being studied have been cleaned, integrated, and standardized. In particular, the cleaning activity mainly regarded negative stock values that have been identified in the data collected from the PLM. They have been compared to the corresponding production lists of the MES in order to understand the origin of the inconsistencies and correct them, as well as some errors in terms of dates. Furthermore, a comparison among the orders for the suppliers and the items entered in the warehouse has been performed in order to control data correctness.

A single dataset to perform the analysis has been created to support the decision-making process. The analysis described by the last two layers is performed considering the data of the orders delivered in the last year. An excerpt of the data collected is shown in Table 2: each row represents a picking order (P.O.), while the items are reported in the columns. The numbers inside the table represent the quantity of each item picked during each order. In total, during 2019, the company managed 24,320 different items and 11,283 picking orders.

Table 2. Dataset analyzed

| Picking orders | ITEM 1 | ITEM 2 | ITEM 3 | ITEM 4 | ITEM 5 | ITEM 6 | ITEM 7 | … | ITEM N |
|---|---|---|---|---|---|---|---|---|---|
| **P.O. 1** | 1 | 0 | 1 | 3 | 0 | 5 | 5 | … | 0 |
| **P.O. 2** | 1 | 3 | 0 | 3 | 1 | 0 | 0 | … | 0 |
| **P.O. 3** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | … | 1 |
| **P.O. 4** | 0 | 0 | 1 | 0 | 2 | 0 | 0 | … | 1 |
| **P.O. 5** | 0 | 0 | 0 | 0 | 4 | 4 | 0 | … | 6 |
| **P.O. 6** | 3 | 3 | 0 | 0 | 0 | 0 | 0 | … | 0 |
| **P.O. 7** | 0 | 0 | 5 | 0 | 0 | 0 | 1 | … | 0 |
| **P.O. 8** | 0 | 0 | 0 | 1 | 0 | 0 | 3 | … | 0 |
| **…** | … | … | … | … | … | … | … | … | … |

The company initially decided to divide each tray into 12 stock keeping units (SKUs). Subsequently, this value will be varied to make a what-if analysis. To facilitate the picking for the workers, each SKU can contain a unique item, while the quantity of each item that can be allocated to each SKU is known and depends on item volume.

In Table 3, an excerpt of the table that reports the values of the picking frequency for each item (Supp) is shown, together with the maximum number of items that can be stored in each SKU (Q/SKU). The picking frequency is calculated for each item using the dataset reported in Table 2. It is defined by determining the support of the itemsets composed of a single item, i.e., the number of times an item appears in the picking orders over the total pickings. The analysis is carried out through the RapidMiner software, selecting 0 as minimum support and confidence thresholds.

Table 3. Excerpt of the item support and quantity of each item allocable to an SKU

| ITEMS | Supp | Q/SKU |
|---|---|---|
| Item 1 | 0.107 | 80 |
| Item 2 | 0.101 | 110 |
| Item 3 | 0.096 | 30 |
| Item 4 | 0.095 | 80 |
| Item 5 | 0.093 | 80 |
| Item 6 | 0.090 | 80 |
| Item 7 | 0.081 | 110 |
| Item 8 | 0.078 | 40 |
| Item 9 | 0.074 | 30 |
| Item 10 | 0.056 | 35 |

## 4.2 Layer 3: data analytics

In total, considering a minimum support threshold of 0.05 and minimum confidence of 0.10, 33,970 rules are obtained. Since there are no univocal strategies to select the thresholds, they are kept low in order to be able to associate the components among them as much as possible. In Figure 6, an excerpt of the network describing the associations among the items is presented. It is worth noting that, for the sake of clarity, Figure 6 only reports 20 nodes (i.e., the association among 20 items) and the related 370 association rules. Each node represents an item to be stored in the ASRS; an edge among two nodes only exists if an association rule among the two items exists. The thickness of the edges is proportional to the confidence of the association rule among the nodes, i.e., the weight of the edge; the dimension of the nodes represents the out-degree of the node, that is, the weighted sum of the outgoing edges from it. The color is used to emphasize the information provided by the dimension of the nodes and the thickness of the arcs: green color indicates high values (specifically, the darker the green, the higher the respective values), while pink indicates a low value.

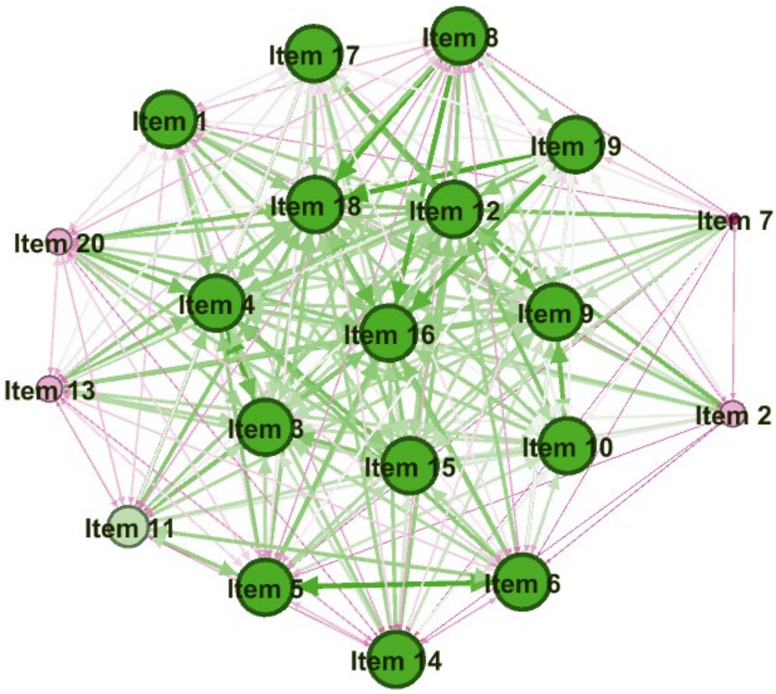


*Figure 5 Excerpt of the network describing the association among items.*

Only as an example, we report in Table 4 an excerpt of the association rules containing Item 1.

Table 4. Excerpt of the association rules containing Item 1.

| I | J | Conf |
|---|---|---|
| Item 1 | Item 3 | 0.331 |
| Item 1 | Item 4 | 0.330 |
| Item 1 | Item 2 | 0.297 |
| Item 1 | Item 5 | 0.266 |
| Item 1 | Item 11 | 0.238 |
| Item 1 | Item 12 | 0.238 |
| Item 1 | Item 7 | 0.232 |
| Item 3 | Item 1 | 0.210 |
| Item 4 | Item 1 | 0.205 |
| Item 2 | Item 1 | 0.197 |
| Item 5 | Item 1 | 0.174 |
| Item 11 | Item 1 | 0.154 |
| Item 12 | Item 1 | 0.151 |

## 4.3 Layer 4: Allocation layer

Once the ASRS is installed, the company needs to fill the automated warehouse with the products it had previously stored in a traditional warehouse. As previously mentioned, the company needs to move from a standard picker-to-part approach to an automated part-to-picker one.
Following the general procedure presented in Figure 4, the flow chart reported in Figure 7 details the specific procedure followed by the company in allocating the items to the ASRS trays. The following notation is adopted in order to schematize the policy:

- $i$, $m$, and $t$ are items that have to be stored in the ASRS;
- $i \rightarrow t$ represents the association rule among the items $i$ and $t$.
- The warehouse is composed of trays ($k$), each of which can be fractioned.
- Each fractioned area of the tray is referred to as Stock Keeping Unit (SKU);
- $Q_i$ is the quantity of item $i$ that has to be allocated;
- $Q_j$ is the maximum quantity allocable to the SKU $j$.

When an item $i$ has to be stored in the ASRS, all trays and SKUs are scanned in order to check whether the item $i$ is already in the ASRS. If any, all the SKUs already containing the item $i$ are saved since they are the possible candidates to receive the $Q_i$ to fully saturate the trays and, therefore, the warehouse. If there is at least one SKU that already contains $i$, it is assigned to it: if $Q_i \leq Q_j$, then all the $Q_i$ is stored in the SKU $j$ and the cycle ends; otherwise, the SKU $j$ is saturated, and the search for another SKU is performed with the same criteria. When in doubt regarding the SKU selection, e.g., more than one is eligible for the allocation, the one belonging to the tray that is closer to the picking bay is selected.

If item $i$ is not present in the ASRS, then it is analyzed with the ARs. All the ARs in which $i$ appears (rules in the form $i \rightarrow t$ and $t \rightarrow i$) are selected and sorted by descending confidence. Assuming $t$ is the item belonging to the rule having the highest confidence, in the same trays where $t$ is located, it is checked whether there is a spare SKU to contain $i$. If so, then $i$ is stored in the same tray as $t$ (totally or partially). Otherwise, other associated items are then searched for.
As an example, let's assume we have to allocate $Item\ 1$ and consider the ARs reported in Table 4. $Item\ 1$ is the item $i$ described in the current procedure, while $Item\ 3$ is item $t$; if in the trays containing $Item\ 3$ there is a spare SKU, then $Item\ 1$ is assigned to it. If the quantity of $Item\ 1$ exceeds the capacity of the SKU or if there is no available space on this SKU, then the trays containing $Item\ 4$ are checked, and so on.
If the item $i$ has no associated item or the trays containing the associated ones have no more available space, then i is allocated to a brand-new tray. In the case where there are no empty trays, it is assigned to a partially full one, as long as the totality of $Q_i$ can be stored on it (eventually, in separate SKUs), choosing the closest to the picking point. In the event that there is no available space for storing the totality of $Q_i$ in the ASRS, the remaining is stored on manual shelves.

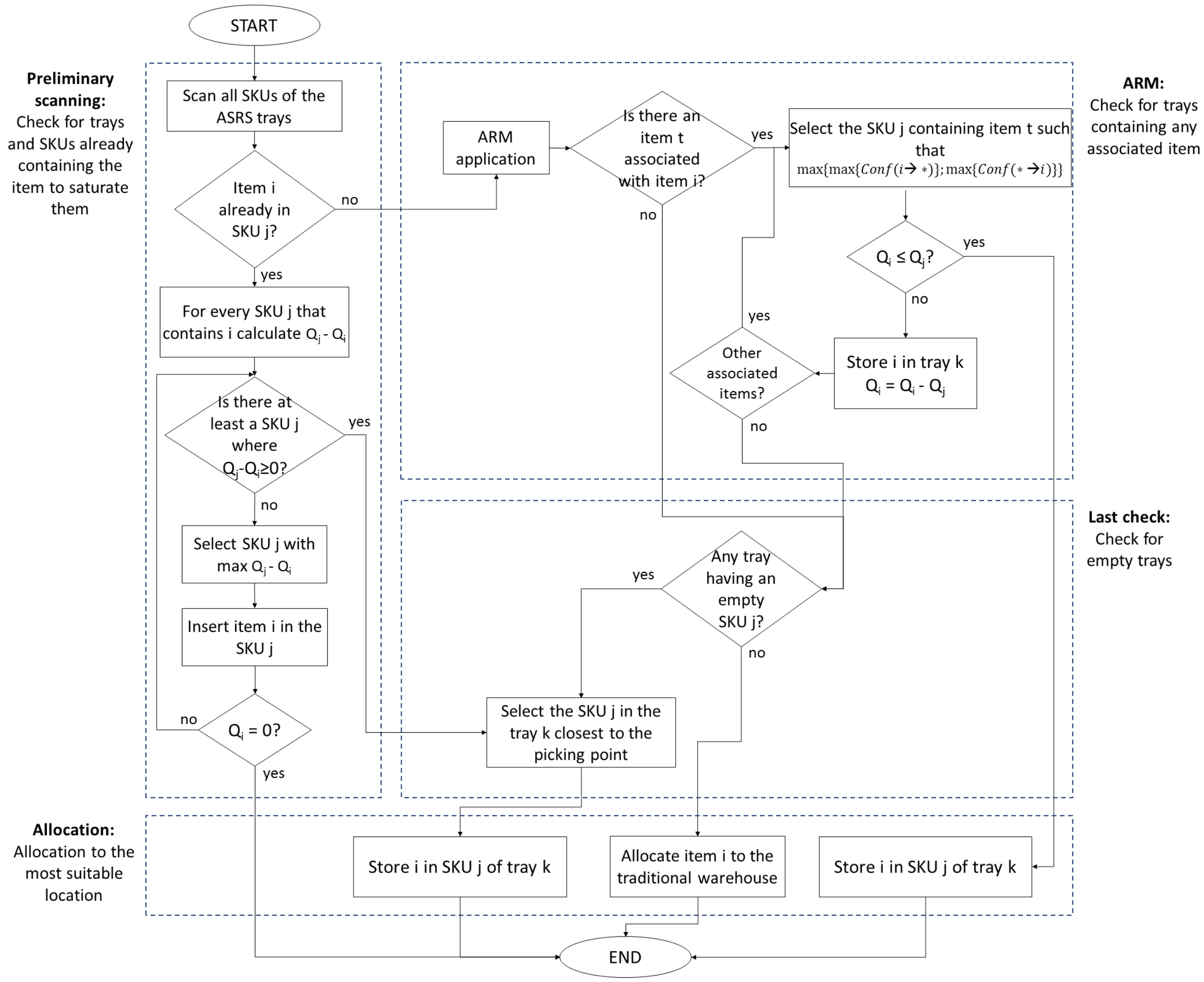


*Figure 6 Item allocation to the SKUs detailed for the case study*

## 4.3.1 Simulation results

On the basis of the allocation deployed in this layer, the number of trays required daily is simulated during a one-year interval. Specifically, the aim of this analysis is to check how many trays, organized according to the ARs mined, are necessary to stock all the items. Existing items at the beginning of the year and the inbound and outbound daily quantities are considered in the test. In Figure 7, the data series represented by a rhomboidal indicator shows the number of trays required daily: it ranges between 684 and 785 trays.

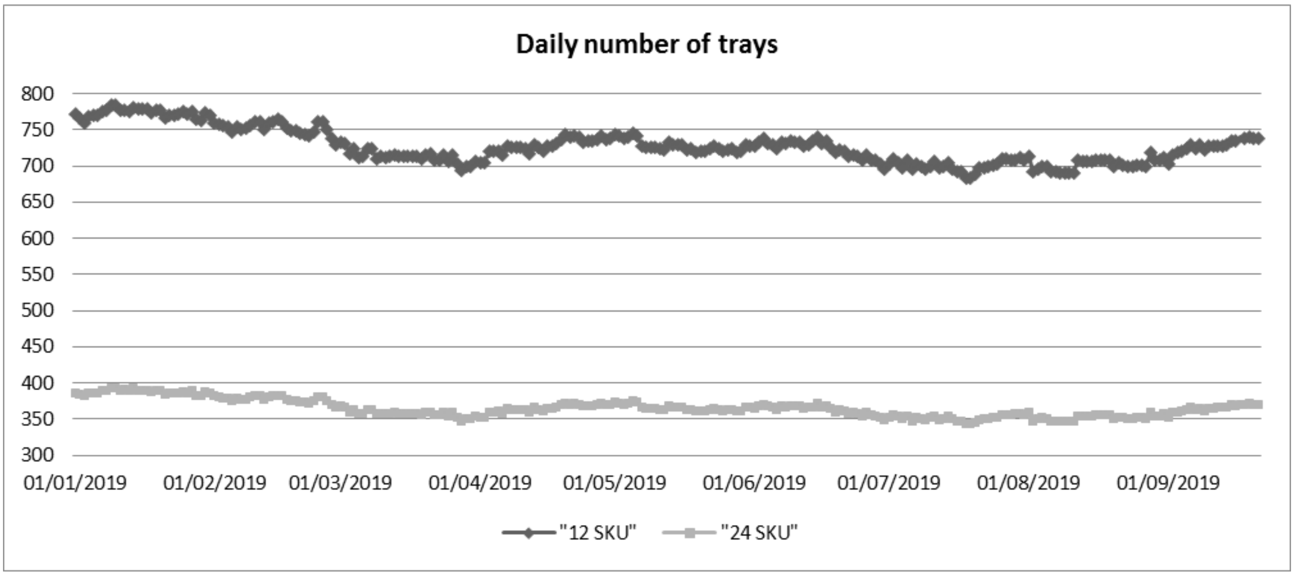


*Figure 7 Comparison between the daily number of trays required to stock the items organizing each tray in 12 and 24 SKUs.*

The allocation of the items to the trays is further studied by monitoring the saturation level of the SKUs. On average, only 19.38% of the SKU is saturated during the year, with a minimum average value of 6.39% and a maximum average value of 46.24%. These results are essentially due to the fact that the company handles small batches of many codes that have a low rotation index.
Hence, the size of the SKU has been halved so that each tray can store 24 SKUs instead of 12. This attempt aims to improve the performance of the ASRS in terms of saturation, and it considers the dimensional constraints of the items to be stored. Thus, the average saturation of the SKU increases by up to 38.8%. Under this hypothesis, the daily requirement of trays (Figure 7, square indicator) ranges between 343 and 394, increasing the number of items storable in the ASRS. Any other splitting of the tray into smaller SKUs is not allowed due to the size of the items. Indeed, the dimension of the SKU currently tested is the minimum one capable of containing all the items. Hence, this solution is the one implemented by the company since it allows acceptable saturation levels.

### 4.3.2 What-if analysis

The performance achieved through the ARM-based methodology proposed in this article has been compared with two other item allocation methodologies. The first one (called “ARM+Shoe shape”) is a variant of the ARM approach in which an additional constraint on shoe shape has been introduced. According to this approach, priority is given, in the allocation to the tray, to those items belonging to the same shape of shoes. The other method of comparison is Random Storage Assignment, a traditional method of allocating items in an ASRS.
The ARM+Shoe shape allocation approach has been proven because historically, the footwear company organizes the warehouse by placing the items belonging to the same shoe shape close together. In this sense, an additional constraint is considering in the allocation of items to trays. Specifically, a tray can be composed only of items belonging to the same shoe shape. Hence, the first modification is carried out during the ARM phase. Indeed, only rules describing the relations among items referring to the same shape have to be taken into consideration. The procedure to allocate items to the trays is the same presented in the previous sections and reported in Figures 3 and 6: according to the association rule mined, items that refer to the same shoe shape are allocated based on the decreasing confidence values. Under this hypothesis, the average saturation of the SKU decreases (29.68%). In parallel, there is an increase in the requirement of the trays, that range between 341 and 550 (401 trays, on average). Hence this solution would not be very flexible in the case where the demand increases. The estimation of the picking time duration is compared through the simulation of typical daily picking processes for both solutions.

According to historical data collected during the ASRS management, the assumptions made before the execution of the simulation are as follows:

- The average cycle time provided in the technical specifications of the ASRS is considered: 27 s (SD = 5,5 s) for the single and 33 s (SD = 7,2 s) for the dual command cycles;
- A single operator is responsible for the picking process and works using two picking points;
- While the worker picks a set of items from a tray, the ASRS brings another tray to the second picking point;
- 45 real picking lists processed during the previous year are randomly selected from the historical picking lists.

Table 5. Comparison of the results achieved with the possible solutions.

| | | ARM | ARM + Shoe shape | Random Storage Assignment |
|---|---|---|---|---|
| Group statistics | Mean number of trays needed | 365 | 401 | 327 |
| | Mean picking time [s] | 31243.50 | 27803.02 | 33128.13 |
| | Standard Deviation | 1230.82 | 1121.23 | 1335.54 |
| | Standard Error Mean | 181.47 | 165.32 | 196.91 |
| Independent Samples Test | t-test for Equality of Means | t = 14.015; df = 88; sig.: <0.001 | | t = -7.038; df = 88; sig.: <0.001 |
| | Mean difference | 3440.48 | | -1884.63 |
| | Std. Error Difference | 245.48 | | 267.68 |
| | 95% Confidence Interval of the Difference | [2952.78 ; 3928.18] | | [-2416.63 ; -1352.63] |

In Table 5, the ARM solution is compared to the ARM+Shoe Shape solution and Random Storage Assignment in terms of number of trays and picking time. Specifically, the ARM+Shoe Shape solution allows a reduction in terms of picking times if compared with the ones related to the allocation only following the association rules. Such a reduction is, on average, 12.89%.
The different mean values obtained by the two options have been compared through a t-test aiming at verifying the significance of the results with a 95% confidence interval. Remarkably, the mean difference is significant ($p< 0.001$). Hence, even if the addition of the constraint on the shoe shape as a driver for the allocation of the items to the trays is penalizing in terms of saturation level, it allows better performance in terms of picking time. The higher number of trays required to store the items in case of the adoption of the approach analyzed in the ARM+Shoe Shape solution is not critical, on average. However, in periods characterized by a higher number of stocks, the peaks could not be sustained by such an approach, requiring the item to be stored outside the ASRS.
The ARM solution is also compared to a traditional technique for item allocation, the Random Storage Assignment. Dividing the items on a random basis surely provides benefits in terms of the number of trays required daily (327 versus 365). However, the difference between the picking times in the two cases is statistically significant t = -7.038; df = 88; sig.: <0.001) and is favorable to the approach proposed in this work. Indeed, the time required to perform the picking on a typical day's work is of about 9 and a half hours (33128.13 s). Neither of the solutions can be considered as the global optimum since the results are contrasting, even though the proposed approach allows a company a good trade-off between ASRS saturation and picking times.

# 5. Discussion

Some theoretical contributions and implications for management can be highlighted from the implementation of the research approach proposed in this work. In the next two sub-sections, a discussion about practical and theoretical aspects has been inserted.

## 5.1 Practical implication

The novel integration of the Association Rule Mining and simulation within the same data-driven framework represents a promising approach to managing the items in an ASRS.

From a practical point of view, the data-driven decision support system for ASRS management can provide some benefits: firstly, as presented in the results, the picking process is smoother and faster without compromising the ASRS saturation. Moreover, since the items are contained in single-product SKUs, the chance of committing errors is lower. Indeed, the interface of the ASRS indicates the SKU from which the item has to be picked, supporting the operator in executing the correct picking process.

The proposed approach represents a novelty in comparison with the ones already adopted in the literature. On the one hand, this is because of its application to the automated storage and retrieval system, while other works (e.g., Hui et al., 2015; Chen et al., 2005) focus on the traditional manual warehouses; on the other hand, because it takes into account both the support of the items and the confidence of the rules. Moreover, the approach proposed in this work takes into account data coming from different sources and additionally relies on a data-driven analysis: indeed, items are not clustered on the basis of their category but by considering the historical data. In this way, not only will similar items be stored closely, but also a further knowledge dimension is added since previously unknown patterns are derived through the association rule mining.

The proposed approach will provide better results if customers tend to request the same products frequently. In this case, in fact, the allocation of the items foreseen through the association rules will allow warehouse operators to minimize the number of trays to be recalled for daily picking activities. Indeed, the rationale behind the adoption of such a technique for defining the storage allocation is based on the assumption that if one item has been frequently asked together with another one in the past, most likely, the two items will be asked in the same picking list also in the future. Additionally, this approach overcomes the formulation of a research hypothesis for factor combination: this aspect, in fact, may become time- and resource-consuming even for a moderately sized set of variables (Ciarapica et al., 2019).

A specific comment on the item allocation order is needed: indeed, this step may have an impact on the whole item allocation. In the specific case study, it has been decided to allocate the items according to the arrival date of the shipments from the supplier. This is linked to the need not to take up too much space by stacking material already delivered. In addition, this avoids taking material that has not yet been registered and entered into the ASRS and, consequently, avoids a loss of information. Alternatively, it could be chosen to allocate the highest turnover material first or the largest quantities delivered by the suppliers. Adopting these solutions requires the introduction of additional constraints in the storage processes and a delay in entering materials into the ASRS. This means increasing the space in the acceptance and control area for incoming goods from suppliers, making in many cases impossible to implement these two approaches in practice. In fact, the acceptance and control area has a limited size and in many cases it is not possible to leave goods on pallets for more than 24 hours.

During the implementation of the proposed research approach in the shoe company, some problems connected to human behavior have been observed. Indeed, the developed data-driven decision support system requires several data to be taken into account. In this regard, one of the main criticisms is guaranteeing the accuracy in data recording and standardizing them in order to avoid duplicates. In the case study proposed in this work, the different data sources were initially used separately and not integrated. In order to solve this issue, standardized procedures are needed to ensure that the operators record the right information at the right time, avoiding forgetfulness and inaccuracy in the data management process. A substantial cultural change is necessary to convey to all members of the company the feeling that accurate data collection and recording is the basic step in moving towards an effective data-driven decision support system and correctly predicting the picking activities.

## 5.2 Theoretical aspects

Considering the algorithmic aspect, the selection of the FP-Growth algorithm, in spite of the widely used Apriori, is related to its higher efficiency in extracting the frequent patterns in large databases (Agrawal & Srikant, 1994). Even the setting of the minimum support and confidence levels has an impact on the association rules mined. Indeed, indicating a threshold higher than 0 implies the loss of some rules, the ones characterized by a low probability of occurrence. An example of the items picked contextually only a few times could be the ones used to create sample shoes or because of special customer requirements. The larger the dataset, the more important this setting is: indeed, the number of association rules to be mined affects the running time of the algorithm.

On the other hand, the big data perspective has to be taken into account. Massive amounts of heterogeneous data are continuously produced, possibly with different velocities. Dealing with them in real-time requires the adoption of adequate technologies and, of course, techniques and expertise. Thus, it can be said that the datasets to be analyzed grow both in terms of volume and complexity. Such complexity regards the variety of data, e.g., heterogeneous sources, types, and structures. Even variability and veridicality is a critical issue in terms of increasing complexity: indeed, fast-changing information may involve the use of data that are no longer coherent with the previously collected ones. In the fashion industry, for example, this issue can be represented by trend changes or brand reputation modifications. The application context of the proposed approach has to be taken into account when defining the parameters setting, like, for example, the time interval for the association rule mining. In the proposed case study, data refer to a one-year time period. Enlarging the used dataset could provide a broader set of rules, but, in this case, the fast-changing trends characterizing the fashion industry have to be taken into account. In this perspective, in fact, having a wider set of association rules is not a synonym for better management policy. In any case, a periodic review of the allocation and stock level is necessary in order to limit the propagation of possible errors committed, as well as a periodic re-mining of the association rules, considering the possibility of extending the dataset currently used or introducing a new one, based on the characteristics of the company being studied. During the reiteration of the approach (e.g., the update of the dataset), the simulation and sensitivity tests help in defining whether the proposed solution is feasible or not, taking into account possible physical constraints and policies specific to the company.

# 6. Conclusions and future development

Data-driven approaches can support the decision-makers in understanding how to deal with warehouse operations and how to manage the wide amount of data nowadays, characterizing operations profitably.

Existing contributions present the successful implementation of decision support systems for the improvement of the warehouse management policies, the majority of which are based on simulation approaches.

In this work, an innovative perspective is provided by the combination of data mining techniques, like the Association Rule Mining and simulation. In both cases, a fundamental role is played by the quality of data, opportunely analyzed through data-driven approaches. The framework proposed for the management of Automated Storage and Retrieval Systems and the picking process is organized into four steps, respectively dedicated to data collection, data management, data analytics, and understanding of results. The aim of the proposed approach is twofold: it provides a roadmap to guide the organization in understanding data produced every day and to benefit from them. Secondly, it supports the prediction of picking activities and, as a result, the definition of the item assignment to the stock-keeping units through association rule mining. In this way, both the saturation level of the stock-keeping units and the picking time can be taken into account.

A practical application of the framework is presented, highlighting how companies, in their daily practice, have to find trade-offs in pursuing their objectives. Indeed, solutions allowing a higher level in terms of saturation of the stock-keeping units is penalizing in terms of picking duration. On the contrary, solutions ensuring shorter picking times is characterized by a lower saturation level. This aspect highlights the need for further research on this topic. Further development of this study, indeed, involves the introduction of a multi-criteria decision-making approach to support the decision-makers in defining the best alternative. Undoubtedly, the decision is subjective since it is strictly related to organizational objectives and modus operandi, but some general guidelines could be developed.

## References


Accorsi, R., Manzini, R., & Maranesi, F. (2014). A decision-support system for the design and management of warehousing systems. *Computers in Industry*, *65*(1), 175–186. https://doi.org/10.1016/j.compind.2013.08.007

Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. In *In Proc. 20th int. conf. very large data bases, VLDB* (pp. 487–499).

Anđelković, Aleksandra Radosavljević, M. (2018). Improving order-picking process through implementation of warehouse management system. *Strategic Management*, *23*(1), 3–10.

Bevilacqua, M., Ciarapica, F. E., & Antomarioni, S. (2019). Lean principles for organizing items in an Automated Storage and Retrieval System: An Association Rule mining-based approach, *10*(1), 29–36. https://doi.org/10.24425/mper.2019.128241

Bevilacqua, M., Ciarapica, F. E., Diamantini, C., & Potena, D. (2017). Big data analytics methodologies applied at energy management in industrial sector: A case study. *International Journal of RF Technologies Research and Applications*, (December). https://doi.org/10.3233/RFT-171671

Bortolini, M., Accorsi, R., Gamberi, M., & Regattieri, Riccardo Manzini, A. (2015). Optimal design of AS/RS storage systems with three-class-based assignment strategy under single and dual command operations. *The International Journal of Advanced Manufacturing Technology*, *79*(9–12), 1747–1759.

Boysen, N., & Stephan, K. (2016). A survey on single crane scheduling in automated storage / retrieval systems. *European Journal of Operational Research*, *254*(3), 691–704. https://doi.org/10.1016/j.ejor.2016.04.008

Buddhakulsomsiri, J. Siradeghyan, Y., Zakarian, A., & Li, X. (2006). Association rule-generation algorithm for mining automotive warranty data. *International Journal of Production Research*, *44*(14), 2749–2770.

Burinskiene, A. (2010). Order picking process at warehouses. *Int. J. Logistics Systems and Management, 6*(2), 162–178.

Calzavara, M., Persona, A., & Sgarbossa, F. (2018). Modelling of Rail Guided Vehicles serving an automated parts-to-picker system. *IFAC-PapersOnLine*, *51*(11), 1476–1481. https://doi.org/10.1016/j.ifacol.2018.08.295

Caron, F., Marchet, G., & Perego, A. (2010). Routing policies and COI-based storage policies in picker-to-part systems. *International Journal of Production Research*, *36*(3), 713–732.

Chan, F. T. S., & Chan, H. K. (2011). Improving the productivity of order picking of a manual-pick and multi-level rack distribution warehouse through the implementation of class-based storage. *Expert Systems With Applications*, *38*(3), 2686–2700. https://doi.org/10.1016/j.eswa.2010.08.058

Chen, M., Huang, C., Chen, K., & Wu, H. (2005). Aggregation of orders in distribution centers using data mining. *Expert System with Applications*, *28*(3), 453–460. https://doi.org/10.1016/j.eswa.2004.12.006

Chen, M., & Wu, H. (2005). An association-based clustering approach to order batching considering customer demand patterns. *Omega*, *33*, 333–343. https://doi.org/10.1016/j.omega.2004.05.003

Ciarapica, F., Bevilacqua, M., & Antomarioni, S. (2019). An approach based on association rules and social network analysis for managing environmental risk : A case study from a process industry. *Process Safety and Environmental Protection*, *128*, 50–64. https://doi.org/10.1016/j.psep.2019.05.037

Dukic, G., & Oluic, C. (2007). Order-picking methods : improving order-picking efficiency. *Int. J. Logistics Systems and Management*, *3*(4).

Elsayed, E. A., & Stern, R. G. (1983). Computerized algorithms for order processing in automated warehousing systems. *International Journal of Production Research*, *21*(4), 579–586. https://doi.org/10.1080/00207548308942392

Ene, S., & Öztürk, N. (2012). Storage location assignment and order picking optimization in the automotive industry. *The International Journal of Advanced Manufacturing Technology*, *60*(5–8), 787–797.

Frazelle, E., & Sharp, G. (1989). Correlated assignment strategy can improve order-picking operation. *Infustrial Engeneering*, *4*, 33–37.

Giannikas, V., Lu, W., Robertson, B., & Mcfarlane, D. (2017). An interventionist strategy for warehouse order picking : Evidence from two case studies. *Intern. Journal of Production Economics*, *189*, 63–76. https://doi.org/10.1016/j.ijpe.2017.04.002

Gils, T. Van, Ramaekers, K., Braekers, K., Depaire, B., & Caris, A. (2018). Increasing order picking efficiency by integrating storage , batching , zone picking , and routing policy decisions. *International Journal of Production Economics*, *197*, 243–261. https://doi.org/10.1016/j.ijpe.2017.11.021

Hadi-vencheh, A., & Mohamadghasemi, A. (2011). A fuzzy AHP-DEA approach for multiple criteria ABC inventory classification. *Expert Systems With Applications*, *38*(4), 3346–3352. https://doi.org/10.1016/j.eswa.2010.08.119

Han, J., Cheng, H., & Xin, D. (2007). Frequent pattern mining: current status and future directions. *Data Mining and Knowledge Discovery*, *15*(1), 55–86. https://doi.org/10.1007/s10618-006-0059-1

Hofmann, C. (2018). Storage and order-picking system and method for providing articles in a particular order.

Hsu, C., Chen, K., & Chen, M. (2005). Batching orders in warehouses by minimizing travel distance with genetic algorithms. *Computers in Indusrty*, *56*, 169–178. https://doi.org/10.1016/j.compind.2004.06.001

Hui, Y. Y. Y., Choy, K. L., Ho, G. T. S., Lam, C. H. Y., Lee, C. K. H., & Cheng, S. W. Y. (2015). An Intelligent Fuzzy-based Storage Assignment System for Packaged Food Warehousing. In *Management of the Technology Age* (pp. 1869–1878).

Hwang, H., Oh, Y. H., & Lee, Y. K. (2004). An evaluation of routing policies for order-picking operations in low-level picker-to-part system. *International Journal of Production Research*, *42*(18), 3873–3889.

Hwang, H. S., & Cho, J. S. (2006). A performance evaluation model for order picking warehouse design. *Computers & Industrial Engineering*, *51*, 335–342. https://doi.org/10.1016/j.cie.2005.10.002

Kulak, O., Sahin, Y., & Taner, M. E. (2012). Joint order batching and picker routing in single and multiple-cross-aisle warehouses using cluster-based tabu search algorithms. *Flexible Services and Manufacturing Journal*, *24*(1), 52–80.

Lee, C. K. M., Lv, Y., Ng, K. K. H., Ho, W., & Choy, K. L. (2017). Design and application of Internet of things-based warehouse management system for smart logistics. *International Journal of Production Research*, *56*(8), 2753–2768. https://doi.org/10.1080/00207543.2017.1394592

Li, Jianbin, Huang, R., & Dai, J. B. (2016). Joint optimisation of order batching and picker routing in the online retailer ’ s warehouse in China. *International Journal of Production Research*. https://doi.org/10.1080/00207543.2016.1187313

Li, Jiaxi, Moghaddam, M., & Nof, S. Y. (2015). Dynamic storage assignment with product affinity and ABC classification — a case study. *Int J Adv Manuf Technol*. https://doi.org/10.1007/s00170-015-7806-7

Macro, J. G., & Salmi, R. E. (2002). A SIMULATION TOOL TO DETERMINE WAREHOUSE EFFICIENCIES AND STORAGE ALLOCATIONS. In *Winter Simulation Conference*.

Matusiak, M., Koster, R. De, Kroon, L., & Saarinen, J. (2014). A fast simulated annealing method for batching precedence-constrained customer orders in a warehouse. *European Journal of Operational Research*, *236*(3), 968–977. https://doi.org/10.1016/j.ejor.2013.06.001

Pan, J. C., & Wu, M. (2009). A study of storage assignment problem for an order picking line in a pick-and-pass warehousing system. *Computers & Industrial Engineering*, *57*(1), 261–268. https://doi.org/10.1016/j.cie.2008.11.026

Peixoto, R., Dias, L., Carvalho, M. S., Pereira, G., & Geraldes, C. A. S. (2016). An automated warehouse design validation using discrete simulation. In *IEEE*. https://doi.org/10.1109/ITSC.2016.7795554

Petersen, C. G., & Aase, G. R. (2017). Improving Order Picking Efficiency with the Use of Cross Aisles and Storage Policies. *Open Journal of Business and Management*, *5*(1).

Petersen, C. G., Aase, G. R., & Heiser, D. R. (2004). Improving order-picking performance through the implementation of class-based storage. *IJPDLM*, 534–544. https://doi.org/10.1108/09600030410552230

Popović, D., Vidović, M., & Bjelić, N. F. S. M. J. (2014). Application of genetic algorithms for sequencing of AS/RS with a triple-shuttle module in class-based storage. *Flexible Services and Manufacturing Journal*, *26*(3), 432–453.

Quintanilla, S., Pérez, Á., Ballestín, F., & Lino, P. (2015). Heuristic algorithms for a storage location assignment problem in a chaotic warehouse. *Engineering Optimization*, *47*(10), 1405–1422. https://doi.org/10.1080/0305215X.2014.969727

Shiau, J., & Lee, M. (2010). A warehouse management system with sequential picking for multi-container deliveries. *Computers & Industrial Engineering*, *58*, 382–392. https://doi.org/10.1016/j.cie.2009.04.017

Sooksaksun, N., & Kachitvichyanukul, V. (2012). A class-based storage warehouse design using a particle swarm optimisation algorithm. *Int. J. Operational Research*, *13*(2), 219–237.

Tarczyński, G. (2015). Estimating order-picking times for return heuristic - equations and simulations. *Scientific Journal of Logistics*, *11*(3), 295–303.

Taylor, P., Kung, Y., Kobayashi, Y., Higashi, T., Sugi, M., & Ota, J. (2014). Order scheduling of multiple stacker cranes on common rails in an automated storage / retrieval system. *International Journal of Production Research*, *52*(4), 1171–1187. https://doi.org/10.1080/00207543.2013.848040

Yang, C. L., & Nguyen, T. P. Q. (2016). Constrained clustering method for class-based storage location assignment in warehouse. *Industrial Management & Data Systems*.

Yang, M. F. (2008). Using simulation to object-oriented order picking system. *Information Technology Journal*, *7*(1), 224–227.

Yener, F., & Yazgan, H. R. (2019). Optimal warehouse design : Literature review and case study application. *Computers & Industrial Engineering*, *129*, 1–13. https://doi.org/10.1016/j.cie.2019.01.006

Zhang, R., Wang, M., & Pan, X. (2019). New model of the storage location assignment problem considering demand correlation pattern. *Computers & Industrial Engineering*, *129*, 210–219. https://doi.org/10.1016/j.cie.2019.01.027