University of Groningen

Simultaneous determination of warehouse layout and control policies

Roodbergen, Kees Jan; Vis, Iris F. A.; Taylor, G. Don

Published in:
International Journal of Production Research

DOI:
[10.1080/00207543.2014.978029](https://doi.org/10.1080/00207543.2014.978029)

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2015

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Roodbergen, K. J., Vis, I. F. A., & Taylor, G. D. (2015). Simultaneous determination of warehouse layout and control policies. *International Journal of Production Research*, 53(11), 3306-3326.
<https://doi.org/10.1080/00207543.2014.978029>

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Simultaneous determination of warehouse layout and control policies

Kees Jan Roodbergen^a, Iris F.A. Vis^a and G. Don Taylor Jr^{b*}

^a*Faculty of Economics and Business, University of Groningen, Groningen, The Netherlands;* ^b*Grado Department of Industrial and Systems Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA*

(Received 16 May 2013; accepted 7 October 2014)

In a supply chain's order fulfilment process, it is often the warehouse that plays a central role in making the right product available to the right customer at the right time. This paper aims to improve warehouse performance by deriving an effective design method for the simultaneous determination of warehouse layout and the warehouse's control policies. The authors consider layout variables for the warehouse such as the number of primary aisles utilised, the number of cross-aisles and the aisle length under several different design philosophies. Concurrently, control policies such as storage policies and routing are considered. Simulation is utilised to determine the performance of the various resulting scenarios. A screening and selection procedure is employed to reduce the required number of replications while achieving a predetermined precision in identifying the best configuration. The approach is applied for an industrial partner in this research and the results of experimentation are compared to a baseline scenario which describes a proposed new facility in the Netherlands. The results reveal a large potential for performance improvement.

Keywords: warehouse design methodology; warehouse layout; order picking; simulation

1. Introduction

Order fulfilment from warehouses is an important step in the order management cycle of any supply chain, i.e. the process from a customer's order to product delivery. Because competition for customers is increasing, having the right parts available at the right time is critical to the success or failure of a company. Thus, there is an incentive for companies to have stock available at well-placed, well-designed distribution centres and for these centres to operate efficiently. Clearly, this cannot be achieved free of charge. Warehousing and inventory carrying costs alone are a large part of total logistics costs. In the United States, it is estimated that 32.6% of total logistics costs are associated with inventory holding costs (Wilson 2012). In Europe, some estimates reveal that warehousing and inventory carrying costs are more than 40% of total logistics costs (Mayer et al. 2009). Any improvement in warehouses, functioning as the central hub in the flow of products from suppliers to customers, can therefore directly contribute to the overall cost reductions and improved customer service of supply chains.

In spite of the importance of warehousing activities, top supply chain management has had the tendency to ignore the order fulfilment process when optimising supply chains (Shapiro, Rangan, and Sviokla 1992), and it appears that warehousing design and operational procedures are rather historically inspired and that no design changes have been made to keep up with the requirements of a service economy (Cross 2006). The important role of warehouses is not only overlooked by some practitioners, but also by scientific researchers. The focus on warehousing forms a very small fraction of the total supply chain research literature, and cross fertilisation between practitioners and researchers appears to be very limited (Gu, Goetschalckx, and McGinnis 2010). To overcome these hurdles, we present new theoretical concepts in this paper and verify them at a warehouse in the Netherlands to come to the conclusion that our approach can be successfully used to reduce order fulfilment times.

The actual order fulfilment process starts the moment that the order is placed by the customer and is redirected electronically to the warehouse. After receiving the order, it is processed by the warehouse management system (WMS) and released to the work floor. The warehousing activity that retrieves products from storage in response to customer orders is commonly referred to as *order picking*. There are many other activities in warehouses, such as receiving of incoming goods or the replenishment of storage locations, but the order-picking process has by far the strongest link with customer order fulfilment. It is for this reason that the order-picking process is considered by many researchers and practitioners to be the most critical warehousing activity (Olson 1994; de Koster, Le-Duc, and Roodbergen 2007). At

*Corresponding author. Email: don.taylor@vt.edu

the same time, order picking is typically the most labour- and capital-intensive process in a warehouse (Tompkins et al. 2003). Therefore, it is vital that a design for the order-picking operation is carefully selected.

Important elements of a warehouse design include the choice of equipment, the layout and the control rules that govern the daily operations. Unfortunately, as observed by Goetschalckx, McGinnis, and Sharp (2008) 'No systematic functional design procedure for warehouse design has been proposed or demonstrated. Thus, teaching, research, and practice in warehouse design remain largely ad hoc in terms of approach'. Thorough evaluation of any design alternatives on all dimensions is therefore important. Typical performance measures include achievable throughput, response time and total cost over the lifetime of the facility. However, in the design phase of warehouses, more operational performance measures, such as travel distances or the retrieval time per order, should also be considered since the operational efficiency is strongly affected by the design decisions (Gu, Goetschalckx, and McGinnis 2010). And since order picking is very labour intensive, the operational efficiency has a major impact on life-time costs of the whole warehouse. Moreover, Hackman et al. (2001) concluded in a benchmarking analysis of 57 warehouses that warehouses that achieved a high efficiency focused on minimising the operational metric of worker travel times.

Analysing procedures from the warehouse design practice, Govindaraj et al. (2001) found a conceptual workflow being followed, but it appeared to be fairly informal, especially with regard to how and in what sequence decisions were made. Rouwenhorst et al. (2000) present the warehouse design process as a sequential approach going through three levels of analysis, namely strategic decisions (such as design of the process flow and selection of physical systems), tactical decisions (such as determination of storage capacity, number of employees, layout and storage assignment) and operational decisions (such as task scheduling, batching and routing). The authors state that 'since interfaces between different processes are typically handled within the design problems at the strategic and tactical level, this implies that at the operational level policies have less interaction and therefore can be analysed independently'. Other authors advocate a similar sequential procedure, but emphasise that it should be applied as an iterative procedure with multiple visits to every stage (Gray, Karmarkar, and Seidmann 1992; Goetschalckx, McGinnis, and Sharp 2008). However, examining the results of more than 2500 different realistic layout problems, Roodbergen and Vis (2006a) showed that sequential decisions do not necessarily give the same quality of results as a procedure that takes simultaneous decisions on layout and control policies.

It is in this space that we seek to work, by defining a design methodology that identifies the best warehouse design and layout while simultaneously selecting from a multitude of control policies. We examine at the same time the issues of strategic concern (type of warehouse and overall material storage/handling decisions), issues of tactical concern (establishing the layout and storage assignment) and issues of operational concern (routing policies). Thus, the approach taken herein is not the usual, myopic, analysis-based approach, but, in fact, ventures into developing an integrated methodology that considers warehouse layout and control simultaneously. The block diagram in Figure 1 shows the scope of our research within the framework of Rouwenhorst et al. (2000).

In Section 2, we discuss design and control of warehouse operations in more detail and we will describe relevant literature. Section 3 gives our design methodology that utilises simulation as both a solution tool and an evaluation system for the complex interrelated problems of both warehouse layout and warehouse control. The method presented in this paper can be applied in many alternative warehousing settings and for new or existing facilities. We demonstrate in Sections 4–7 how this method can be applied in practice by developing a design for a projected new warehouse in the Netherlands under several design philosophies. Section 8 presents conclusions.

Figure 1. Block diagram of approach overlaid on framework of Rouwenhorst et al. (2000).

2. Warehouse design and control

Warehouse design roughly consists of three interrelated aspects: the type of *systems* used, the *layout* and the *policies* to control all operational processes. The systems choice is largely driven by product characteristics and demand frequency, although some decision latitude may remain for the designer in certain cases. For example, larger and heavier products tend to be stored on pallets in pallet racks. Fast-moving high-volume products are typically stored in flow racks, in which the rack is constructed such that products naturally ‘flow’ to the front of the rack under the influence of gravity. Smaller, low-demand products may be stored on shelving racks, which have a lower storage capacity than flow racks and require less floor space. Obviously, numerous other variations exist, many of which are described in Frazelle (2001).

For all systems, a *layout* must be determined such that each system can achieve the required performance. Also, the consistency of the overall design requirements and layout consistency between areas must be taken into account. In Figure 2, we graphically depict a typical order-picking area. The layout of the area consists of a number of parallel pick aisles. Cross-aisles are positioned at the front and back of the warehouse and are used to break up aisles into sub-aisles. They do not contain or provide access to storage locations, but can be used to change aisles. A set of parallel sub-aisles forms a block. A depot functions as the central point where jobs are administered to employees, which may be shared between a number of areas if desired. Figures in Section 6 demonstrate how multiple areas can be positioned within a single warehouse. Other layouts featuring piecewise diagonal cross-aisles and picking aisles that are not parallel (such

Figure 2. Illustration of a typical order-picking area.

as ‘Fishbone’ or ‘Flying V’ layouts) appear in the literature (e.g. Gue and Meller 2009). It would be possible to include these types of layouts within the design methodology presented in this paper, but this paper focuses on more traditional design alternatives.

Control policies drive the daily operations of the warehouse. Incoming products need to have a storage location assigned, and employees need to receive instructions on the sequence of locations to visit to fulfil a specified customer’s order. Numerous variations exist for these policies and due to their mostly heuristic nature and the unpredictable interface with other policies, it cannot be said beforehand which combination of policies works best (de Koster, Le-Duc, and Roodbergen 2007).

The *random storage assignment policy* assigns an incoming product to a location that is randomly selected from all eligible empty locations with equal probability. The basic idea of *ABC storage assignment policies* is to allocate the most frequently requested products to the best accessible locations. Fast-moving products are referred to as A-items. The next fastest moving products go into the next class, and are called B-items. The slowest moving products are designated as C-items. Each class is assigned to a dedicated part of the order-picking area. Within such a dedicated part, products are randomly assigned to the storage locations. Several ABC storage policies are available. Within-aisle storage assumes that entire aisles are dedicated to one category (A, B or C). Across-aisle storage places an equal amount of A, B and C locations in each aisle. Nearest sub-aisle storage places A-items closest to the depot and C-items farthest from the depot, but allocates entire sub-aisles to a particular category. Finally, nearest location storage places all A-items

Figure 3. Example of four ABC-storage assignment policies.

closest to the depot and all C-items farthest from the depot without further restrictions. Figure 3 graphically depicts each of the four ABC storage assignment policies.

During a day, customers' orders arrive at the warehouse. An order picker starts at the depot, travels through the area (following the paper or electronic pick list) to collect all requested products and returns to the depot to drop off the picked products. Various *routing policies* are available from practice and literature to efficiently route order pickers through the area. This routing problem can be considered as a special case of the Travelling Salesman Problem (see e.g. Daniels, Rumel, and Schantz 1998). Order pickers are assumed to be able to traverse aisles in both directions and they can change directions in each aisle. The simple return routing policy traverses each aisle twice, once from the front to the back to retrieve products from the rack on the left, and once from the back to the front to retrieve products from the rack on the right; this is commonly referred to as one-sided picking. Often improvements can be obtained by implementing two-sided picking, provided the aisles are narrow enough (see Goetschalckx and Ratliff 1988). Roodbergen and de Koster (2001) discuss a number of routing policies in detail. The S-shape routing policy permits crossovers, so order pickers walk only one direction in each aisle. The largest-gap policy turns the order picker on each aisle when a large gap is encountered, with the rest of the items being picked from the other side from a cross-aisle. The aisle-by-aisle policy picks each aisle completely before moving to the next, with dynamic programming determining which cross-aisle to utilise when moving to the next aisle. Finally, the combined routing policy is basically an S-shape routing enriched with aspects of the largest gap heuristic. Examples of routes for each of the routing policies are given in Figure 4.

Scientific literature on the concurrent selection of layout and control policies for warehouses is very limited. Perhaps, the paper that comes closest to our approach is Rosenblatt and Roll (1984), and it seems that very little similar work has been done since the publication of that paper. The authors combine the problems of determining warehouse size, internal layout and storage policy, while minimising the costs associated with the initial investment (construction and handling costs for a given warehouse capacity and number of 'zones'), shortage costs (if the warehouse runs out of capacity) and storage policy costs. While these authors concurrently consider design and operational issues, they do so with only one type of storage (pallet only), and their experimental plan is much more limited in terms of layout options, storage options and routing policies.

Several papers utilise a combination of analytical optimisation and simulation to solve warehousing problems. Kim et al. (2003) examine a hybrid scheduling and control system for use in warehouse management. They utilise a mathematical model and a genetic algorithm, with simulation results to demonstrate the efficiency of the approach. Hwang and Cho (2006) evaluate the performance of order picking by focusing primarily on warehouse design (warehouse size, rack size, number of transporters, etc.) and use simulation to measure the system performance with probabilistic demand and picking frequency. Manzini, Gamberi, and Regattieri (2006) utilise simulation to solve a design and control problem in automated storage/retrieval systems (AS/RSs). They examine thousands of scenarios and suggest that visual interactive simulation is a key part of design and re-design of automated storage/retrieval systems. Chen et al. (2010) focus primarily on order-picking policies. They propose a framework in which they combine Data Envelopment Analysis and Monte Carlo simulation to study the performance of different combinations of order-picking policies. The authors consider the layout as fixed and only search for an optimal combination of policies by examining a very restricted set of options.

Summarising, in literature, most papers address either the layout problem in combination with pre-determined control rules or try to find an optimal control rule for the given layouts. We consider the simultaneous determination of warehouse layout and control policies in a general warehouse environment utilising, for example, pallet positions, shelving and/or flow racks. In the next section, we present a design methodology that can be used to simultaneously determine the warehouse layout and its control policies. The method presented can be used to set the values for various layout characteristics in combination with selecting the best routing and storage assignment policies.

3. Design methodology

In this section, we present a design methodology to determine the layout of a warehouse in concurrence with the selection of efficient control policies. We will first describe inputs, performance criteria and decision variables for the method. Next, reasons for the use of this approach as well as implications and applied guidelines are described. At the end of this section, we give a stepwise description of the method.

Input parameters, decision variables and the performance criterion

First, we will describe the input parameters that are needed. These fall into two categories: those input factors that can be collected from historical data, and those that can be decided on by the designer, but are not part of our design method. Collectable input data are:

Figure 4. Examples of the discussed routing policies.

- p_m , the probability that the number of picks in an order equals m (for all m)
- p_j , the probability that product j is included in an order (for all j)

Input data decided by the designer:

- Systems used (e.g. pallet positions, flow racks and/or shelving)
- Aisle width
- Cross-aisle width
- Required storage capacity
- Physical limitations (to the building)

As discussed in the previous section, the choice of systems depends strongly on the demand and product characteristics and is therefore not considered a decision variable here. Aisle width and cross-aisle width are a direct consequence of the chosen system and there is no decision latitude in these factors. The required storage capacity follows from

strategic considerations on required product availability and service level, as well as inventory policies. Physical limitations may, for example, reflect the available floor space, and thus induce a maximum for the aisle length or a maximum for the number of aisles to ensure that the design will fit into the available space.

Our design methodology aims to find a good layout in combination with strong control policies. Determination of the layout basically concerns defining a grid of locations where products can be stored (see Figure 2). The control policies determine which product is stored in which location, and in which sequence locations should be visited while picking orders. This gives us the following decision variables:

- a = number of pick aisles
- ℓ = length of aisles
- c = number of cross-aisles
- R = routing policy
- Z = storage assignment policy

To allow for alternative Fishbone or Flying V layouts (Gue and Meller 2009), we can simply add a binary decision variable that distinguishes between traditional and alternative layouts. Additional discussion regarding alternative layouts is provided in Section 5.2. The main objective of this procedure is to minimise travel times of order pickers. This performance criterion can be translated into labour costs, which together with direct investment costs makes up most of the total cost over the life span of the operation. In our method, we have five decision variables that influence total travel time T , so we can express T as a function of these five variables. The goal is now to minimise:

- $T(a, c, \ell, R, Z)$.

As may already be apparent, there is vast amount of possible solutions to choose the design from within the stated restrictions. Furthermore, as we will be using simulation, we also face uncertainties that are inherent to our approach. For these two aspects, we introduce the following configuration settings:

- δ = parameter that specifies the range of indifference
- α = parameter that specifies the required statistical precision

The desired statistical precision is determined by setting the statistical confidence level $1-\alpha$. That is, if we set α equal to 0.05, then we will be 95% certain that the final design is actually the best possible design. The parameter δ indicates the acceptable difference with the true (but unknown) optimal design. For example, if we set δ to one metre, then this means that we will be satisfied with the result if we find a design that gives an average travel distance that differs no more than one metre from the true optimum (at a $1-\alpha$ confidence level). Smaller values give a higher precision, but extremely small values may not be practically relevant. Very small values for δ will lead to unacceptable run times, because the method will try to distinguish between two designs that are both close to the optimum, but one of the two may be only marginally better than the other.

3.1. Concept of the approach

A central building block of our approach is a base simulation model that is capable of calculating the average travel time $T(a, c, \ell, R, Z)$ for any combination of layout parameters and control policies. This simulation model can be considered to be a Monte Carlo simulation where every order-picking route constitutes a replication. A random generator is included in the model for generating the picks and the pick locations for every replication based on the input parameters as specified above and the chosen storage policies. Multiple replications (i.e. routes) are to be simulated in order to decrease the confidence interval to the desired size.

The main difficulty of our project is that it is too time-consuming to use the base simulation model to simulate thousands of possible design alternatives at a sufficient statistical precision to make a reliable choice. We therefore seek to eliminate, obviously, the inferior system alternatives while securing the guarantee that a correct selection still can be made with the remaining alternatives. To this end, we employ a method with a screening and selection procedure at its core provided by Nelson et al. (2001).

In the screening stage, we take for each alternative a user-specified number of replications, n_0 . All alternatives are compared and the alternatives that are unlikely to be best are eliminated from consideration. Note that the value for n_0 has no impact on the quality of the final solution; it only influences computing times. A higher value for n_0 causes calculation times for the screening stage to go up, but more alternatives can be eliminated, which will reduce computational efforts in the selection stage.

Next, in the selection stage, a higher number of replications are taken for each of the remaining alternatives. The number of replications is determined per alternative by the procedure and is such that the user-specified precision will be obtained. The alternative with the lowest average total travel time $T(a, c, l, R, Z)$ is selected. The travel distance related to this alternative is within δ metres of the true (unknown) optimal solution with a probability of $1-\alpha$. For brevity, we will use the term ‘best’ to refer to this solution. The solution supplies the appropriate values for all layout variables and the control policies to be applied.

3.2. Justification of the approach

There are basically two methods found in the literature to estimate travel times in warehouses: closed-form expressions (e.g. Hall 1993, or Le-Duc and de Koster 2007) and simulation (e.g. Petersen and Schmenner 1999). A major drawback of closed-form expressions is that they first need to be developed before they can be used. A typical paper on closed-form expressions considers only a single type of configuration, whereas we wanted to consider all possible configurations, for most of which no closed-form expression exists. It is for this reason that we rely on simulation.

Past research has shown that the solution space in these types of problems is far from smooth, sometimes with many local minima that are just one unit apart (e.g. Roodbergen and Vis 2006a). Because of this, there is almost no hope for a search procedure that will consistently be able to identify the best solution. The typical approach is then to resort to some form of complete enumeration, which we do as well. The remaining challenge is to decide on how to identify the best configuration from among all eligible alternatives with a pre-specified precision and within reasonable computation times.

When comparing two alternatives by simulation, the well-known *t*-test can be applied (Law and Kelton 2000). Having k systems, we could do an all pairwise comparison of the systems, however, to achieve an overall confidence level of $1-\alpha$, we then would have to do $k(k-1)/2$ comparisons each at a confidence level of $1-\alpha/[k(k-1)/2]$. Thus to obtain a 95% confidence level in an analysis with 10,000 systems, alternatives would be required to work with a 99.9995% confidence level for each of the comparisons. It will then be very difficult to reject the null hypothesis of equal means. As a final result, we then would typically be left with a set of alternatives between which we cannot distinguish. In such an all pairwise comparison approach, there is no guideline to determine the number of replications, it cannot be determined beforehand how many alternatives will be left, and there is no possibility to distinguish between these remaining alternatives. The same line of reasoning holds for ANOVA. These approaches are therefore mostly appropriate if the number of replications is given and cannot easily be altered. For our design problem, on the other hand, we have the possibility to increase the number of replications until we can sufficiently distinguish between the alternatives, which brings us into the area of ranking and selection methods (Law and Kelton 2000).

If the number of alternatives grows larger, however, even the ranking and selection procedures require too much computing time. Recent literature provides the computationally more efficient approach of a screening and selection procedure as presented in Nelson et al. (2001). It is this procedure that we use as part of our design methodology. To our knowledge this procedure has never been used before in a warehousing design context for selecting the best simulated warehouse design among a large number of alternatives.

3.3. Stepwise description of the design method

In summary, the following design method can be followed by warehouse managers in general to design a warehouse regardless of the storage systems selected and the number of order-picking areas being designed. To assist the reader, pseudo-code for the software utilised for our study is provided in Appendix 1. In the next sections, we will show how this methodology can be used in practice for various design strategies.

- (1) Collect information on (anticipated) operational parameters for the (new) facility (e.g. desired total aisle length, pick list size, type and characteristics of storage area).
- (2) Specify for each variable (e.g. number of aisles, length of an aisle, number of cross-aisles, routing policies and storage assignment policies) all possible outcomes (e.g. the number of aisles varies between 2 and 50).
- (3) Set the constraints to exclude infeasible combinations from step 2 based on the parameters defined in step 1 (e.g. number of aisles multiplied with the aisle length must be larger than or equal to the storage requirement).
- (4) Construct a simulation model that can calculate the average total travel distance for each of the resulting system alternatives of steps 2 and 3.
- (5) Determine the desired statistical confidence level $1-\alpha$.

- (6) Select a value for the practically significant difference δ in metres. This value will be used in comparing layout alternatives as follows (see also step 7): a layout is considered to be the best if with probability $1-\alpha$, its solution differs at most δ with the real optimal value.
- (7) Apply a screening and selection procedure (Nelson et al. 2001). For more details, refer to Appendix 2.
 - (a) Screening phase: Calculate for each of the possible system alternatives the average total travel distances by simulating a small number of replications. Retain only those alternatives that are most likely to turn out to be the best alternative.
 - (b) Selection phase: Calculate for each of the remaining options of step 7a the average total travel distance by performing a sufficient number of replications. The number of replications is calculated by means of a procedure to achieve statistical significance and may vary per alternative.
- (8) Select the alternative with the lowest average total travel distance resulting from step 7.

4. Case study: company description

The implementation site to test the applicability of this method in practice is at a new distribution centre in the Netherlands. The distribution centre will support transshipment and distribution of electronic products. Although the new facility will support both unit-load handling and small-item picking, our paper deals only with small items in three different types of storage areas; pallets, shelves and flow racks. We will follow the steps of the design methodology as presented in Section 3.

An existing facility has over 14,000 m² of operational space including a total current order-picking area of 1250 m². It is anticipated that the new facility will utilise similar technologies and will have a similar distribution of products. The anticipated operational data for the new small-item order-picking area is provided in Table 1 (see step 1 in Section 3.3). The new facility will have a footprint of 2500 m² with two floors. The lower floor will be for unit-load handling and the upper floor will be for small-item picking. The exact design of the new facility will be finalised in the near future, using the results of this study.

In Table 1, the centre-to-centre distance between aisles and the width of cross-aisles is based on equipment needs and rack/shelving needs. For example, more space is needed for fork truck access in the pallet area than for people in the shelving area. The desired total aisle length assumes that there are no cross-aisles. The addition of cross-aisles increases this desired length. Given the millions of simulation replications required to find the warehouse design, the skewness of demand parameter provides a means of avoiding the use of individual demand curves in the replications. In practice, demand for products is always skewed, except in the unlikely event that all products of a company are sold in identical quantities. The partner company stocks approximately 11,700 unique parts. Based on the work of Caron, Marchet, and Perego (1998), the skewness of demand is described by an analytical function which predicts the percentage of parts that are classified as A (high demand), B (medium demand) or C (low demand). The function equals:

$$F(x) = \frac{(1+p) * x}{p+x}, \quad F(x) \geq 0, x \leq 1, p \geq 0, p+x \neq 0 \quad (1)$$

where x denotes the percentage of the storage space and $F(x)$ gives the percentage of picks resulting from this part of the storage space. p is a shape factor indicating the skewness of the ABC curve. For example, for $s = 0.067$, it holds that 80% of the picks are generated by 20% of the items. The p values in the table have been calculated based on current demand profiles at the partner company. When p is large, as in the pallet area, then the ABC curve approaches a straight line (uniform distribution). When it is small, as in the shelving area, a small number of parts account for a large amount of the demand. Although demand is expected to increase, it is assumed that the skewness of demand remains

Table 1. Operational data for small-item order-picking area in new facility.

Operational parameter	Pallets	Shelving	Flow racks
Centre-to-centre distance between aisles w_a (cm)	560	170	560
Width of the cross-aisles w_c (cm)	180	180	180
Desired total aisle length (m)	20	240	140
Skewness of demand p	0.907829	0.157427	0.231184
Pick list size m (empirical distribution with average m)	1.446	3.1089	3.0116
Maximum pick list size	12	89	57
Probability that items of an order need to be retrieved in that area	0.20514	0.57311	0.78343

constant. The average (maximum) pick list size indicates the average (maximum) number of picks per area, per customer order. Finally, the last row in Table 1 indicates the probability that an order contains items within each area. For example, there is a 20.514% chance that an order requires items from the pallet area. Given that a pick is required in the pallet area, an average order is for 1.446 items.

A simulation model has been programmed using the Delphi programming language, following the guidelines as described in Section 3. Numerical results have been extensively verified and validated. We have compared results for numerous scenarios with manual calculations for travel distances. Next to that, we have extensively checked the consistency of results by comparing simulation results to performance measures from observations in practice, both at the partner company as well as at other companies.

5. Case study: design strategies

In this section, we will first give two baseline scenarios using common rules of thumb for the layout and the control policies as currently applied by the partner company. Furthermore, we use our design method for two conceptually different options. First, we design the three systems simultaneously, which forces cross-aisle configuration and control policies to be the same across the systems. Secondly, we design each of the areas individually. All experiments assume that there is a single depot (located in the lower left of in the warehouse plan view) and that the three order-picking areas are placed with respect to the depot based on the highest number of picks per m^2 . Generally, this results in the pallet area being closest to the depot, followed by shelves and finally by flow racks. This differs from the current facility in that the shelving area is now farther from the depot than the flow rack area. We will now discuss the set-up of each of the approaches in more detail.

5.1. Establishing a baseline for comparison

Pertinent operational data for the current facility appears in Table 2. The table includes the number of aisles and cross-aisles by storage area, and the aisle length by area. Although the new facility will be larger and product demand higher, the new facility will be similar to the old one. To provide a baseline for comparison with alternative designs, we will utilise two different ‘baseline’ scenarios based on common ‘rule-of-thumb’ layouts comparable to the old warehouse. The first is a ‘square in time’ layout with two cross-aisles. The square in time layout is selected because it has been long ago demonstrated that for single pick situations in manual order-picking areas, the distance from the depot to the side wall should be equal to the distance from the depot to the back wall (see Francis 1967). Similarly, in the automated storage/retrieval systems literature, Bozer and White (1984) confirm this conclusion for single- and dual-command, unit-load cycles with random storage assignments (as utilised in the existing facility). A second baseline is established based on the results reported in Hall (1993). In this second baseline scenario, it is assumed that the facility is twice as wide as deep. Current control policies are return routing and random storage (see Section 2).

5.2. Simultaneous design for three areas

A second set of experiments determines the best configuration for each of the three picking areas concurrently. In this case, it is assumed that the number of cross-aisles and the aisle length must be uniform across the three areas. Also, the routing and storage assignment policies must be identical for all three areas. The advantage of this approach is that one route can be established for all three areas. This results in order integrity and simplicity of operation. Table 3 gives all possible system alternatives that will be taken into consideration.

In step 2 of the design method, we set appropriate values for the decision variables and in step 3, we set constraints to exclude any infeasible combinations. A wide range of a and c values are used to ensure that the best solution in terms of aisles and cross-aisles can truly be found, but minimum and maximum values are based partially on the value of s ,

Table 2. Current operational data for the small-item order-picking area.

Existing parameter	Pallets	Shelving	Flow racks
Number of aisles a	1	23	9
Number of cross-aisles c	2	2	2
Aisle length ℓ (m)	10.00	11.40	12.44

Table 3. Parameters for scenarios featuring concurrent operation.

Parameter	Values
$1-\alpha$	0.95
δ (m)	5
Number of aisles a	5 ... 50 (at least 1 aisle for the pallet area and 2 aisles for other areas)
Number of cross-aisles c	2 ... 11
%A and %B	From Roodbergen and Vis (2006b) formula
Routing policy	Return, S-shape, largest gap, aisle-by-aisle and combined
Storage assignment policy	Random, within aisle, across-aisle, nearest sub-aisle and nearest location

the desired total aisle length. The number of aisles, a , ranges from 5 to 50 in increments of 1 and the number of cross-aisles, c , ranges from 2 to 11.

Fishbone and Flying V layouts (Gue and Meller 2009) can be easily studied using the methods presented within this paper, but they have been excluded from consideration for reasons of computational tractability and also for reasons of expected performance. Celik and Süral (2014) show that shorter routes (orders with few picks and/or highly skewed demand in small picking areas) tend to become shorter when introducing a Fishbone structure. However, longer routes tend to become longer within a Fishbone structure. Since the demand scenarios utilised in our case study forms a mixture of both small and large orders (with an average of 7.6 picks per order), and since demand is not very skewed ($p = 0.191$), we expect that Fishbone layouts would likely not offer improvement in comparison to the traditional layouts utilised herein. This argument is further strengthened given that Celik and Süral (2014) only consider standard layouts with three cross-aisles while we consider any value between 2 and 11. Increased use of cross-aisles is known to be an influential factor on route length even within traditionally designed warehouses (Roodbergen and de Koster 2001). Furthermore, the determination of the best sizes for A- and B-item zones within Fishbone warehouse designs has not been well examined in the literature. Thus, a thorough analysis of Fishbone systems would require thousands of zone sizes to be considered, resulting in excessive calculation times.

In contrast, the best sizes for A- and B-item zones within traditional layouts can be determined by a formula resulting from a detailed regression analysis presented in Roodbergen and Vis (2006b). The formula specifies the percentage of A and B parts as a function of the number of aisles, the number of cross-aisles, the aisle length, the aisle width, the demand skewness and the number of picks per route. In the next steps, we need to define the desired reliability of the results. In Table 3, $1-\alpha$ represents a statistical confidence level of 95% as required for step 5. δ represents an arbitrary total difference between scenarios (in metres) deemed ‘practically significant’ (step 6). This value is used in the screening and selection processes to distinguish between scenarios. Although the value of δ in metres is arbitrary, smaller values tend to lead to unacceptable run times and larger values sacrifice the quality of solutions. The selected value is the lowest possible to still achieve reasonable computation times.

5.3. Separate design for the three areas

A third set of experiments determines the best configuration for each of the three picking areas independently. This decoupling means that each area can have a different shape, a different number of aisles and cross-aisles, and a different aisle length. Also, storage assignment policies and order-picking routes can differ amongst the three areas. The advantage of this type of approach is in maximising ‘within area’ efficiency. The values for each of the variables (step 2) utilised in these scenarios are provided in Table 4.

In Table 4, note that, the δ values specified total the δ values specified in Table 3. The number of aisles, a , varies depending upon practical, volume-based considerations for each of the three storage policies. The remainder of the table indicates parameters identical to those presented in Table 3.

6. Case study: results

A general discussion of computational results is now presented, along with an overview of the results of the screening and selection procedure employed in step 7 of the design methodology (for more details, see Appendix 2). Relevant design data and operational data used for these computations can be found in Tables 1, 3 and 4, and has been discussed in Section 5. We will follow this with an overview of the results of experimentation.

Table 4. Parameters for scenarios featuring separate design by area.

Parameter	Pallet	Shelves	Flow racks
$1-\alpha$	0.95	0.95	0.95
δ (m)	1	2	2
Number of aisles	2 ... 20	2 ... 40	2 ... 30
a			
Number of cross-aisles c	2 ... 11	2 ... 11	2 ... 11
%A and %B	From formula	From formula	From formula
Routing policy	Return, S-shape, largest gap, aisle-by aisle and combined	Return, S-shape, largest gap, aisle-by aisle and combined	Return, S-shape, largest gap, aisle-by aisle and combined
Storage assignment policy	Random, within aisle, across aisle, nearest sub-aisle and nearest location	Random, within aisle, across aisle, nearest sub-aisle and nearest location	Random, within aisle, across aisle, nearest sub-aisle and nearest location

6.1. Computational results

In comparison with the experiments dealing with separate and simultaneous designs of the three order-picking areas, the experimentation for the baseline scenarios is insignificant. Thus, we will focus on computational results associated with the former in this section. As part of the design methodology, the first step in the screening phase is to select an initial sample size $n_0 \geq 2$. Large n_0 values lead to more replications in the screening phase, but reduce the number of replications required subsequently in the selection phase (see Section 3). We utilised an n_0 value of 40 as a compromise. Recall that α and δ values are provided in Tables 3 and 4 for the separate and simultaneous design problems, respectively. Computational requirements for the screening phase (step 7a) appear in Table 5. It is notable that more than 1.3 million replications are performed in support of the screening phase.

Computational requirements for the selection phase (step 7b) are even more cumbersome. To illustrate, consider the simultaneous design experiments. As a result of the screening procedure, we eliminate 7361 of the 11,500 scenarios (64%). The remaining 4159 scenarios require an average of 27,979 total replications to ensure that we are within δ metres of an optimal solution with a probability of $1-\alpha$. This results in approximately 116 million additional replications. Computational requirements are summarised in Table 6. Additional replications to support selection of the best alternative for simultaneous and separate design problems total more than 332 million, taking approximately three days of run time on five desktop PCs. Each single replication takes an average 0.004 s. It is for this reason that the formula-based ABC allocation (see Tables 3 and 4) was a more acceptable alternative than adding several allocation rules to the experimental design.

6.2. Results of experimentation with baseline scenarios

The two baseline scenarios are developed to provide an initial basis for comparison. Because the return routing policy and the random storage policy are used in the current facility, it is assumed that they would be used in the new facility without compelling and quantifiable evidence that other policies are better. The results of experimentation (step 8) for the ‘square in time’ and ‘twice as wide as deep’ rules of thumb are provided in Table 7. The table provides the number

Table 5. Computational requirements for the screening phase of experimentation.

System parameter	Screening phase			
	Simultaneous design	Separate design: pallet only	Separate design: shelves only	Separate design: flow rack only
Number of aisles	46 (5 ... 50)	19 (2 ... 20)	39 (2 ... 40)	29 (2 ... 30)
Number of cross-aisles	10 (2 ... 11)	10 (2 ... 11)	10 (2 ... 11)	10 (2 ... 11)
Number of routing policies	5	5	5	5
Number storage policies	5	5	5	5
Total number of options	11,500	4750	9750	7250
Initial sample size	40	40	40	40
Number of replications	460,000	190,000	390,000	290,000

Table 6. Computational requirements for the selection phase of experimentation.

System parameter	Selection phase			
	Simultaneous design	Separate design: pallet only	Separate design: shelves only	Separate design: flow rack only
Number of alternatives	11,500	4750	9750	7250
Number left after screening	4159 (36%)	253 (5%)	5146 (53%)	2220 (31%)
Avg. reps./alternative	27,979	8180	25,183	37,879
Number of replications	116,364,661	2,069,540	129,591,718	84,091,380

of aisles (a), the number of cross-aisles (c , including ‘top’ and ‘bottom’ aisles at the edges of the facility) and the aisle length (ℓ) for both alternatives. Note that in the flow rack area, only ‘pick’ aisles are included in the aisle count. Additional ‘replenishment’ aisles access the back of the racks. The ‘twice as wide as deep’ layout performs better than the ‘square in time’ layout, resulting in a total average travel distance per order of 159.66 m, compared to 172.17 m for the square in time rule. This difference is due to the fact that the additional lanes provided in the wider layout leads to better routing possibilities. In this case, with the limitation that the facility is only twice as wide as deep, the benefits associated with routing outweigh the additional distance travelled to the depot. Drawings of the two layouts appear in Figure 5.

6.3. Results of experimentation with simultaneous design of the three areas

As stated previously, return routing and random storage policies are utilised in the existing facility upon which the two baseline scenarios are based. To quantify the potential improvements to the baseline scenarios that would be possible if the routing and storage policies were changed but route integrity were preserved (i.e. if we continue to utilise one person (one route) per order), we consider simultaneous design of all three picking areas under the various scenarios discussed in the ‘experimental plan’ section. That is, we determine layout and control policies concurrently. The results of this experimentation (step 8) are provided in Table 8.

The flexibility permitted in routing and storage, along with a change to ABC storage leads to drastic improvement in total average travel distance per order, even while maintaining the one person per order rule. The average distance of 118.80 m represents an improvement of 53.37 m (30.99%) compared to the ‘square in time’ baseline layout and 40.86 m (25.59%) compared to the more competitive ‘twice as wide as deep’ layout. A figure depicting the solution graphically is provided in Figure 6.

Noteworthy is the fact that the layout for this solution is almost square. So even though for the baseline scenarios ‘twice as wide as deep’ was better than ‘square in time’, an almost square picking areas appears best when simultaneously considering layout and control policies. Compared to results of previous research, the chosen control policies may come as a surprise. Roodbergen and de Koster (2001) show that aisle-by-aisle routing generally is outperformed by almost all other policies. Also, across aisle storage is often not the best choice (Petersen and Schmenner 1999). It must

Table 7. Results of experimentation with baseline scenarios.

Warehouse area	a	c	ℓ	Routing	Storage
Results for the ‘square in time’ rule of thumb					
Pallet	1	2	20	Return	Random
Shelves	6	2	40	Return	Random
Flow Racks	4	2	40	Return	Random
Total length = 43.60 m, total width = 38.2 m					
Total average travel distance: 172.17 m					
Results for the ‘twice as wide as deep’ rule of thumb					
Pallet	1	2	27	Return	Random
Shelves	9	2	27	Return	Random
Flow Racks	6	2	27	Return	Random
Total length = 30.6 m, total width = 54.5 m					
Total average travel distance: 159.66 m

Figure 5. Layouts for 'square in time' and 'twice as wide as deep' baseline scenarios.

be noted though that our study evaluates only one situation as a proof of concept. An interesting further research issue is to test whether an integrated method, in general, returns results that deviate from analyses on a single dimension.

6.4. Results of experimentation with separate design of the three areas

If we are now willing to make further policy relaxations to permit alternative shapes for the three order-picking areas, to permit the use of alternative routing policies and to permit different storage policies to be used in each order-picking area, additional performance improvements are possible. The results of experimentation with separate design for all three areas are provided in Table 9. The table reveals that each area has a unique number of aisles, cross-aisles and aisle lengths, and each has a different allocation of A, B and C parts. Nearest location storage is used for the pallet and shelving areas, while the nearest sub-aisle storage rule is best in the flow rack area. All three order-picking areas utilise combined routing. Since routes are no longer built for a single-order picker, it is necessary to calculate a value for the average distance per order that is equivalent for comparison with scenarios presented previously in this paper. This can

Figure 7. Best layout for separate design scenarios.

be achieved by adding the various component parts of a combined route under the assumption that not all three picking areas are visited in each order. This can be accomplished by multiplying the average travel distances per picking area by their probability of realisation per order as provided in Table 1. The resulting average travel distance per order is 74.99 m, an improvement of 43.81 m (36.88%) in comparison to the situation obtained with simultaneous design.

This improvement can only be obtained, however, if one assumes that the three areas share a common depot (see Section 2). Otherwise, additional distance would be required for transfers to the depot. Given that the three order-picking areas are no longer held to the same requirements in terms of shape and aisle configuration, the use of a common depot will likely lead to a facility that is not square and perhaps not 'regular' in shape. One possible configuration is provided in Figure 7. Even though the shape is irregular, the reduced cost of daily operation likely justifies the layout.

6.5. Results summary

The baseline scenarios, 'square in time' and 'twice as wide as deep', each employing the company's current operational policies, resulted in a total average travel distance per order of 172.17 m and 159.66 m, respectively. Simultaneous design of the three areas provided a significant improvement with an average distance of 118.80 m per order. The most efficient alternative is when all areas are allowed their own shape and operating policies, resulting in a travel distance of 74.99 m per order. Compared to the existing methods employed at the partner company, the results indicate savings of 53.0% in travel distances per order. From observations, we know that, on average, an order picker spends about 44% of his or her time on walking. The rest of the time is spent on activities such as administration and actual handling of products. Savings in total order-picking times, therefore, will be approximately 23.3%. Based on the company's 10 order pickers at an annual salary of approximately €30,000 each, the expected savings associated with this improvement would be approximately €69,900 per year.

7. Case study: implementation issues

With regard to the control policies, several changes in the WMS and physical layout of the warehouse are required to fully profit from the new policies. The WMS needs to be able to deal with having different control policies in the various areas. Instead of single-sided routing (return heuristic), order pickers will be routed two-sided (see Section 2). Therefore, it may be a good idea to change location numbering such that odd and even numbers are on opposite sides of the same aisle. This is not a requirement for the routing policies to function; however, it will reduce search time for the order pickers. Most standard WMS systems do not have a standard module containing good routing policies. Therefore, it will be necessary for implementing a new routing policy to add custom-written code to the WMS. This is typically not very complex since it is just a matter of adequately sorting the pick list already generated by the WMS.

The current system assigns products to fixed locations. Ideally, the system would determine the best storage location for every incoming load. This way, actual product locations are dynamically updated based on demand changes. This is not possible in the current WMS system of the partner company. However, it is fairly straightforward to perform a location analysis based on the policies presented in this paper outside of the WMS. This will need to be repeated several times per year. If a new WMS system is purchased in the future, it should be capable of dynamic product allocation.

8. Concluding remarks

We have developed a new design methodology that simultaneously determines the layout and most efficient control rules for order-picking areas in general warehouse environments. We show that in a realistically sized order-picking environment, the method needs to address over tens of thousands possible configurations with an enormous amount of replications to overcome the problem that route lengths are not always normally distributed and to prove statistical significance between the results.

With the use of a case study with a company in the Netherlands, we have demonstrated that this methodology easily can be applied in practice under various design strategies. When compared to the existing methods employed at the partner company, the results indicate savings in an average travel distance of 84.67 m per order (53%). These savings can be directly translated in significant, annually returning savings on labour costs.

The methods presented in this paper can be applied in many alternative warehousing settings. Whether a proposed warehouse utilises shelving, flow racks, pallet locations or other forms of storage, the simultaneous or separate design methods can be employed. The screening and selection procedure makes the problem tractable, and the results are easily implemented. In fact, no direct changes to WMSs are required. All that is required is a management philosophy that facilitates and encourages interaction between strategic planners and operational management, and that enables the use of flexible operating policies.

References

- Bechhofer, R. E., T. J. Santner, and D. Goldsman. 1995. *Design and Analysis of Experiments for Statistical Selection, Screening and Multiple Comparisons*. New York: Wiley.
- Bozer, Y. A., and J. A. White. 1984. "Travel-time Models for Automated Storage/Retrieval Systems." *IIE Transactions* 16: 329–338.
- Caron, F., G. Marchet, and A. Perego. 1998. "Routing Policies and COI-Based Storage Policies in Picker-to-part Systems." *International Journal of Production Research* 36: 713–732.
- Celik, M., and H. Süral. 2014. "Order Picking under Random and Turnover-based Storage Policies in Fishbone Aisle Warehouses." *IIE Transactions* 46: 283–300.
- Chen, C. M., Y. Gong, M. B. M. de Koster, and J. A. E. E. van Nunen. 2010. "A DEA-based Evaluative Framework for Order Picking Systems." *Production and Operations Management* 19: 70–82.
- Cross, C. S. 2006. "Warehouse Design Eliminates Slim Pickings." *Industrial Engineer* 38: 48–49.
- Daniels, R. L., J. L. Rummel, and R. Schantz. 1998. "A Model for Warehouse Order Picking." *European Journal of Operational Research* 105: 1–17.
- de Koster, R., T. Le-Duc, and K. J. Roodbergen. 2007. "Design and Control of Warehouse Order Picking: A Literature Review." *European Journal of Operational Research* 182: 481–501.
- Francis, R. L. 1967. "On Some Problems of Rectangular Warehouse Design and Layout." *Journal of Industrial Engineering* 18: 595–604.
- Frazelle, E. H. 2001. *World-class Warehousing and Material Handling*. New York: McGraw-Hill.
- Goetschalckx, M., and H. D. Ratliff. 1988. "Order Picking in an Aisle." *IIE Transactions* 20: 53–62.
- Goetschalckx, M., L. McGinnis, and G. Sharp. 2008. "Foundations for Formal Warehouse Design Workflow." In *Progress in Material Handling Research: 2008*, edited by K. Ellis, R. Meller, M. Ogle, B. Peters, G. Taylor, and J. Usher, 198–218. Charlotte, NC: Material Handling Institute.
- Govindaraj, T., N. F. Zerangue, D. A. Bodner, K. N. Karathur, L. F. McGinnis, M. Goetschalckx, and G. P. Sharp. 2001. "A Process Model of Expertise in the Design of Warehousing and Distribution Systems." *Proceedings of the IEEE Conference on Systems, Man, and Cybernetics* 5: 3403–3408. New York: IEEE Press.
- Gray, A. E., U. S. Karmarkar, and A. Seidmann. 1992. "Design and Operation of an Order-consolidation Warehouse: Models and Application." *European Journal of Operational Research* 58: 14–36.
- Gu, J., M. Goetschalckx, and L. F. McGinnis. 2010. "Research on Warehouse Design and Performance Evaluation: A Comprehensive Review." *European Journal of Operational Research* 203: 539–549.
- Gue, K. R., and R. D. Meller. 2009. "Aisle Configurations for Unit-load Warehouses." *IIE Transactions* 41: 171–182.
- Hackman, S. T., E. H. Frazelle, P. M. Griffin, S. O. Griffin, and D. A. Vlasta. 2001. "Benchmarking Warehousing and Distribution Operations- an Input-Output Approach." *Journal of Productivity Analysis* 16: 79–100.

- Hall, R. W. 1993. "Distance Approximations for Routing Manual Pickers in a warehouse." *IIE Transactions* 25: 76–87.
- Hwang, H. S., and G. S. Cho. 2006. "A Performance Evaluation Model for Order Picking Warehouse Design." *Computers & Industrial Engineering* 51: 335–342.
- Kim, B., S. S. Heragu, R. J. Graves, and A. St. Onge. 2003. "A Hybrid Scheduling and Control System Architecture for Warehouse Management." *IEEE Transactions on Robotics and Automation* 19: 991–1001.
- Law, A. M., and W. D. Kelton. 2000. *Simulation Modeling & Analysis*. New York: McGraw-Hill International Editions.
- Le-Duc, T., and R. de Koster. 2007. "Travel Time Estimation and Order Batching in a 2-Block Warehouse." *European Journal of Operational Research* 176: 374–388.
- Manzini, R., M. Gamberi, and A. Regattieri. 2006. "Design and Control of an aS/RS." *International Journal of Advanced Manufacturing Technology* 28: 1433–3015.
- Mayer, S., E. Thiry, C. Frank, G. Kara, and A. Menke. 2009. *A.T. Kearney and European Logistics Association Excellence in Logistics Study*. Chicago, IL: A.T. Kearney.
- Nelson, B. L., J. Swann, D. Goldsman, and W. Song. 2001. "Simple Procedures for Selecting the Best Simulated System When the Number of Alternatives is Large." *Operations Research* 49: 950–963.
- Olson, D. R. 1994. "Designing and Implementing Order Picking Systems." In *Perspectives on Material Handling Practice*, 1–24. Charlotte, NC: Material Handling Institute.
- Petersen, C. G., and R. W. Schmenner. 1999. "An Evaluation of Routing and Volume-based Storage Policies in an Order Picking Operation." *Decision Sciences* 30: 481–501.
- Roodbergen, K. J., and R. de Koster. 2001. "Routing Methods for Warehouses with Multiple Cross Aisles." *International Journal of Production Research* 39: 1865–1883.
- Roodbergen, K. J., and I. F. A. Vis. 2006a. "A Model for Warehouse Layout." *IIE Transactions* 38: 799–811.
- Roodbergen, K. J., and I. F. A. Vis. 2006b. "Zone Sizing for Class-based Storage in Manual Order Picking." In *Progress in Material Handling Research: 2006*, edited by R. Meller, M. K. Ogle, B. A. Peters, G. D. Taylor, and J. Usher, 365–386. Charlotte, NC: Material Handling Institute.
- Rosenblatt, M. J., and Y. Roll. 1984. "Warehouse Design with Storage Policy Considerations." *International Journal of Production Research* 22: 809–821.
- Rouwenhorst, B., B. Reuter, V. Stockrahm, G. J. van Houtum, R. J. Mantel, and W. H. M. Zijm. 2000. "Warehouse Design and Control: Framework and Literature Review." *European Journal of Operational Research* 122: 515–533.
- Shapiro, B. P., K. Rangan, and J. J. Sviokla. 1992. "Staple Yourself to an Order." *Harvard Business Review* Jul.–Aug.: 113–122.
- Tompkins, J. A., J. A. White, Y. A. Bozer, E. H. Frazelle, and J. M. A. Tanchoco. 2003. *Facilities Planning*. New York: Wiley.
- Wilson, R. 2012. *CSCMP's Annual State of Logistics Report*. <http://cscmp.org/member-benefits/state-of-logistics>.

Appendix 1. Pseudo-code for primary computer program

The pseudo-code below outlines the program's main structure. The content of Appendix 3 is not included to keep the description concise.

User input:

$a_{\max}$	the maximum number of aisles allowed
$c_{\max}$	the maximum number of cross-aisles allowed
S	the desired total aisle length
δ	the practically significant difference
α	the desired statistical confidence level equals $1-\alpha$
V_{route}	the set of allowed routing methods
V_{store}	the set of allowed storage assignment methods
V_{sys}	the set with system characteristics (aisle width, travel speed, etc.)
V_{order}	the set with order characteristics (demand frequencies and probabilities of pick list sizes)
n_0	sample size for the screening phase

Procedures used:

$Q(a, c, \ell, R, Z, V_{\text{sys}}, V_{\text{order}})$	A function that calculates the travel distance of one single, randomly generated order characterised by V_{order} in a layout defined by $\{a, c, \ell, V_{\text{sys}}\}$, using routing method R and storage assignment policy Z . See Roodbergen and de Koster (2001) and Petersen and Schmenner (1999).
$t(\beta, \nu)$	A function that calculates the β quantile of the t distribution with ν degrees of freedom.
$h(1-\alpha, n_0, k)$	A function that calculates Rinott's constant for a confidence level $1-\alpha$ with n_0 replications for each of k systems. See Bechhofer, Santner and Goldsman (1995).

// INITIALISATION

```

I = ∅ // start with an empty set.
for a = 1 to amax
  for c = 1 to cmax
    for each R in Vroute
      for each Z in Vstore
        ℓ = [S/a]
        I = I + {(a, c, ℓ, R, Z)} // add configuration (a, c, ℓ, R, Z) to set I
      endfor
    endfor
  endfor
endfor

```

// SCREENING PHASE

```

k = |I| // set k to cardinality of the set I
for i = 1 to k
  X[i] = 0;
  X2[i] = 0;
  for j = 1 to n0
    q = Q(I[i], Vsys, Vorder) // I(i) is the i-th element of set I containing values for a, c, ℓ, R, Z
    X[i] = X[i] + q
    X2[i] = X2[i] + q*q
  endfor
  X[i] = X[i]/n0 // the sample mean
  S2[i] = X2[i]/n0 - X[i]*X[i] // the sample variance
endfor

for i = 1 to k
  for j = 1 to k
    W[i,j] = t((1-α/2)^(1/(k-1)), n0-1)*(S2[i]/n0 + S2[j]/n0)^(1/2)
  endfor
endfor

```

```

for each i in I
  u = 1;
  for each j in I with j ≠ i
    if X[i] > X[j] + max(0; W[i,j]-δ) then u = 0;
  endfor
  if u = 0 then I = I - I[i] // remove configuration i from set I
endfor

```

// SELECTION PHASE

```

if |I| = 1 then Result_parameters = I[1]
else
  for each i in I
    n[i] = max(n0; ceil((h(1-α/2, n0, k)*S2[i]^(1/2)/δ)^2))
    //ceil(x) returns the smallest integer larger than or equal to x
  endfor
  k = |I|
  Result_routelength = infinity
  for i = 1 to k
    X[i] = X[i]*n0;
    for j = 1 to n[i] - n0
      X[i] = X[i] + Q(I[i], Vsys, Vorder)
    endfor
    X[i] = X[i]/(n[i]+n0)
    if X[i] < Result_routelength then
      Result_routelength = X[i]
      Result_parameters = I[i]
    endif
  endfor
endif
endif

```

Appendix 2. Screening and selection procedure (from Nelson et al. 2001)

Initialisation

- Select the overall confidence level $1-\alpha$ (we used 95% in our case study)
- Select the practically significant difference δ (specified in Tables 3 and 4 for our case study)
- Define set I to contain all system alternatives from which a choice must be made. Define $k = |I|$.

Screening phase

- Select initial sample size $n_0 \geq 2$ (we used 40 in our experiments)
- Set $t = t_{(1-\alpha/2)^{1/(k-1)}, n_0-1}$, where $t_{\beta, \nu}$ denotes the β quantile of the t distribution with ν degrees of freedom.
- Set $h = h(1-\alpha/2, n_0, k)$ where h is Rinott's constant (see Bechhofer, Santner, and Goldsman 1995)
- Generate replications X_{ij} , for each of the k system alternatives n_0 replications, where $i = 1, \dots, k; j = 1, \dots, n_0$. If required, use the averages of g_i replications instead of single replications to approach normality of the observations (see Appendix 3).
- Compute the sample means $\bar{X}_i^{(1)}$ and variances S_i^2 for $i = 1, 2, \dots, k$.
- Calculate $W_{ij} = t \sqrt{\frac{S_i^2}{n_0} + \frac{S_j^2}{n_0}}$ for all $i \neq j$.
- Keep system alternative i in set I if $\bar{X}_i^{(1)} \leq \bar{X}_j^{(1)} + \max\{0, W_{ij} - \delta\}$ for all $j \neq i$, otherwise remove system alternative i from set I .

Selection phase

- If set I only contains one system alternative, then stop.
- Otherwise, for all system alternatives i in set I , compute the second-stage sample size
- $n_i = \max\left\{n_0, \left\lceil \left(\frac{hS_i}{\delta}\right)^2 \right\rceil\right\}$
- Generate $n_i - n_0$ additional replications for each system alternative i in set I . If required, use averages of g_i replications instead of single replications, based on values determined during the screening phase.
- Compute the sample means $\bar{X}_i^{(2)}$.
- Select as the best system alternative, the system with the lowest value for $\bar{X}_i^{(2)}$.

For a proof that the system alternative with the lowest value for $\bar{X}_i^{(2)}$ that is selected at the end of the procedure, is indeed within δ from the true optimal solution (at a significance level of α), see Nelson et al. (2001).

Appendix 3. Averaging procedure

Based on the central limit theorem, we formulate the following procedure to determine the required number of replications g_i for system alternative i to ensure normality of the replications in the screening and selection procedure (or at least, to prevent large deviations from normality):

- (1) For system alternative i , generate n replications Y_λ for $\lambda = 1, \dots, n$.
- (2) Set $g_i = 1$.
- (3) Calculate averages $\bar{Y}_\ell = \frac{1}{g_i} \sum_{\lambda=g_i \cdot (\ell-1)+1}^{g_i \cdot \ell} Y_\lambda$, $\ell = 1, \dots, \lfloor n/g_i \rfloor$
- (4) Perform a χ^2 -test with $\alpha = 0.05$ to check for normality. If the null hypothesis of normality is rejected, set $g_i = g_i + 1$ and return to step 3. Otherwise, remember the value of g_i and consistently replace every single replication by the average of g_i replications in the screening and selection procedure for system alternative i .

The null hypothesis of the χ^2 -test is not rejected for $g_i = 1$ in the vast majority of the situations. Values for g_i of 2 or 3 are required in about 10% of the situations. Values for g_i larger than 3 rarely occur, but may go beyond 20.