Argumenta Oeconomica
Year 2024, Vol. 52, No. 1
ISSN 2720-5088

A linearization approach in solving a non-linear shelf space allocation problem with vertical categorisation of sales potential levels

Kateryna Czerniachowska
Department of Process Management, Wroclaw University of Economics and Business, Poland
e-mail: kateryna.czerniachowska@ue.wroc.pl
ORCID: 0000-0002-1808-6020

Marcin Hernes
Department of Process Management, Wroclaw University of Economics and Business, Poland
e-mail: marcin.hernes@ue.wroc.pl
ORCID: 0000-0002-3832-8154

Krzysztof Lutosławski
Department of Process Management, Wroclaw University of Economics and Business, Poland
e-mail: krzysztof.lutoslawski@ue.wroc.pl
ORCID: 0000-0002-2978-1184

Quote as: Czerniachowska, K., Hernes, M., & Lutosławski, K. (2024). A linearization approach in solving a non-linear shelf space allocation problem with vertical categorisation of sales potential levels. *Argumenta Oeconomica*, 1(52), 37-55.

DOI: 10.15611/aoe.2024.1.03

JEL Classification: C61, C63

Abstract: Retail is a profit-driven, highly competitive industry. Customers expect retailers to provide appropriate assortment sets and excellent product visibility. The aim of this study was to develop and examine two models for optimising category-level shelf space management that maximise a retailer's profit. The authors developed two shelf space allocation problem models. The first model combines three sets of constraints: shelf, product, and product group constraints, while the second model enlarges the first with the multi-shelves constraints. The study showed that this approach gives an optimal solution in a very short time (approximately 3 seconds on average and less than a second in 93 of the 134 instances for the first, and in 84 of the 146 instances in the second problem) for large-scale instances, which in most of the test cases is even less than a second. First, non-linear formulations of both problems were presented. Next, the authors proposed to use and adjust linearization techniques which allow transforming both problems into linear ones, thus obtaining the optimal solutions. Finally, both problems were solved using the CPLEX solver, the computational results were provided.

Keywords: linear programming, merchandising, retailing, shelf space allocation, shelf space allocation problems (SSAP)

1. Introduction

The final aim of shelf space planning is to maximise profit by assigning products to shelves while maintaining shelf space limitations and allocation limits. The goal of this planning assignment is to distribute the limited shelf space in a retail store among the many products to be displayed.

Real-world shelf space allocation problems (SSAP) come to light due to the conflict between scarce shelf space and the large number of products that need to be placed there. Empirical studies and shelf space optimisation models have attracted attention in recent years because of the continuous competition for scarce shelf space among different manufacturers. In consequence, the increased amount of products and wider customer demand should be thoroughly comprehended, which leads to the need for a complete understanding of customer requirements and suggesting appropriate merchandising tactics by the retailers.

A growing stream of research papers, models and solves the SSAP, considering space and cross-space elasticity effects. Yet, due to the combinatorial complexity caused by them, the majority of solution approaches cannot be used for extensive practical implementation. A large body of literature points the space elasticity as a measurement of increased responsiveness of sales if more space is given to a product (Curhan, 1972; Chen and Lin, 2007; Chandon et al., 2009). Cross-space elasticity is a measurement of the dependency between neighbouring products; it is assumed to be positive for complementary or similar products and negative for substitutable products (Corstjens and Doyle, 1981; Chen and Lin, 2007). Schaal and Hübner (2017) concluded that cross-space effects have a minor impact on the allocation of products on shelves, therefore they can be disregarded in SSAP empirical and modelling research. Furthermore, the empirical measurement of cross-space elasticity is costly and very complex (Bianchi-Aguiar, 2015; Schaal and Hübner, 2017).

Some scientists have examined product grouping and categorisation effects. Bianchi-Aguiar et al. (2017) categorised products into families and studied their allocation into blocks in a multi-level hierarchical planogram structure. Anic, Radas and Lim (2010), Desrochers and Nelson (2006), Elbers (2016) studied the dependencies between the product on shelf location and visual merchandising, the customers buying behaviour and product categorisation. Czerniachowska and Hernes (2021) proposed horizontal and vertical product grouping based on product sales potential, so that the products could be placed on the shelf not lower than their sales potential.

Other authors have studied customers' purchasing behaviour. Inside stores, in-store elements that increase product awareness and interest have a significant impact on client demand. Parket al. (2015) proposed a measure of merchandising awareness while studying the impact of customer visual perception on customer's purchase intention and brand recognition. Desrochers and Nelson (2006) concluded that customer behaviour is a significant factor in the category management process. Furthermore, they showed that customers' choice of the product is determined by the product category to which it is assigned and the store location. Gabrielli and Cavazza (2014) studied brand recognition depending on product location on low, medium and high shelf levels as well as the location on the end of the aisle. They suggested that the store and shelf location is the strong factor in brand awareness and customers' buying decisions. Czerniachowska (2021) enriched the known SSAP models with local and regional, convenience and complementary products that could be placed in the appropriate shelf segment. The shelf segments were of flexible sizes.

One of the key roles of merchandising in performing commercial decisions is considering product brands, colours, sizes, price, and the movement which attracts customers and generates profit for the retailer. To the best of the authors' knowledge, the recent literature neglects such merchandising tactics, and one of its key limitations is that it does not consider the different vertical shelf levels with regard to product movement, which is reflected in the current research.

Another limitation of the SSAP literature is neglecting the possibility of the product being placed above the main facings. The so-called cappings and nestings were first presented in the studies by Czerniachowska and Hernes (2020, 2021) and Czerniachowska (2021). Despite the popularity of planograms, where products remaining before shelf replenishment are placed above the main facings in another orientation (capping), which then allows putting more products on the shelves, other authors did not formulate the model which investigates such an allocation method. The same applies to the nesting effect.

Therefore the contribution of this research is the following:

- formulation of SSAPs involving merchandising rules based on simultaneous product categorisation of their sales potentials or movement,
- consideration of the allocation above the facings on the top, such as cereal boxes (i.e. capping), and allocation above the facings inside them, such as cups and plates (i.e. nesting),
- deepening the knowledge related to current SSAP with new ways of the formulation of cappings and nestings parameters in non-linear and linear SSAPs.

The aim of this paper was to propose two practical approaches for the SSAP, including merchandising rules regulated by the sales potentials shelf levels and product movement, different orientation possibilities, as well as cappings and nestings parameters. The first problem (SSAP-1) includes shelf, product, and product groups constraints. The second problem (SSAP-2) includes all the constraints from the first one and adds the multi-shelves constraints. In both models, the product item is represented by the facings, cappings and nestings. Next, the authors modelled non-linear integer SSAPs, and using the linearization technique, transformed some constraints into linear integer SSAPs and found an optimal solution to these problems. The experiments in the CPLEX solver were performed on different problem sizes, inspired by retail practice. The proposed methods may help the retailers while performing complex shelf space operations and to reduce the amount of time spent on manual work. The research was motivated by the large impact of shelf space allocation on store efficiency.

The remainder of this paper is structured as follows. The problem definition is presented in Section 2. In Section 3, the non-linear problem formulation and the linearization technique, are given. Section 4 presents the results of computational experiments, and the article is concluded in Section 5.

2. Problem definition

Retailers feel the necessity of efficient modelling approaches to satisfy customer requirements and perform effective decision-making operations. Retailers must assign the shelf space to the items included in the assortment that must be placed on planogram shelves. Usually, in real stores, planograms help retailers to perform visual shelf monitoring, which results in better planning of the scarce shelf space, out-of-stock monitoring, brand visibility, promotion efficiency and also customer satisfaction. Furthermore, planograms make it easier for retailers to gather data for correct merchandising decisions based on more accurate business calculations that drive in-store sales and retailers' profit. Practical, relevant shelf space optimisation models help retailers to find the optimal planogram.

The problem can be formulated as follows. There are a given number of products P which must be allocated on S shelves of a planogram in a retail store. Products P are assigned to K vertical category levels depending on their sales potential. Shelves S are also assigned to the vertical category levels. The sales potential category levels are horizontal. The more expensive or branded the product is (i.e. the higher the sales potential), the higher must be the category level of the shelf where it will be placed. It is allowed to place the products on the shelves with higher or equal category levels compared to their category level, but not on the shelves with lower category levels. This illustrated the retail case when the more expensive or branded products which have higher sales potential are placed on better shelves (higher shelves, eye-level shelves), the cheaper or more frequently bought products can be placed on lower shelves and also on the higher ones, whereas branded products cannot be placed on lower shelves.

Fig. 1 shows the possible allocations of product sales potential categories on different shelf levels. Table 1 describes the retailers' products on shelves with allocating rules. There is a planogram with shelves indicated as 10, 20, 30 sales potentials (white) and products also assigned to 10, 20, 30 sales potentials (coloured). Products with a sales potential 10 can be placed on the shelves indicated as 10, 20, 30. Otherwise, products with a sales potential of 30 cannot be placed on the shelves for other sales potential category levels. Next, products with a sales potential of 20 can be placed on shelves marked as 20 and 30. In this example, interval 10 in sales potentials categories is chosen for practical reasons because frequently, temporary or seasonal assortment (e.g. "Christmas" or "Back to School") must be sold during a short period of time. Hence, there is no reason to reassign all general assortments to other categories, it is easier to assign temporary products to the sales potential category, e.g. 15 or any other between the left 10-number interval.

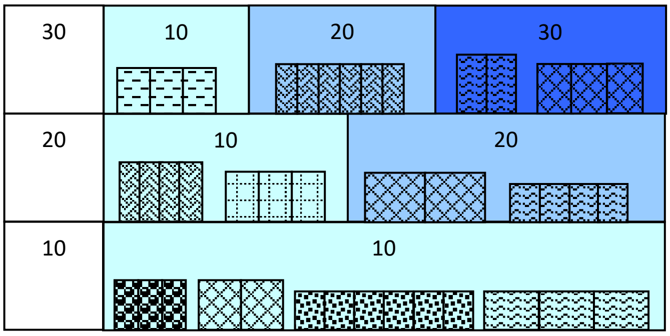

Fig. 1. Planogram with the allocation of sales potential categories: white numbers – sales potential of shelves; coloured numbers – sales potential of products which are placed on the appropriate shelf levels – the darker the colour, the higher the sales potential of the product; different products have a different background.

Source: authors' work.

Table 1. Sales potential category allocation rules

		Product sales potential category level		
		10	20	30
Shelf sales potential category level	30			•
	20		•	•
	10	•	•	•

Source: authors' work.

The product can be placed on the shelf in two orientations – front or side. Generally, each product can be placed on the shelf in its main front orientation, but some products depending on their packaging, can be rotated 90 degrees and can be placed on the shelf in its secondary side orientation. Obviously, for front-oriented products, their width is taken as a linear parameter, whereas for side-oriented products, their depth is a linear parameter. Similar substitutable products with the same features such as type, purpose, taste, etc. can be grouped into clusters and must be placed on the same shelf in order to make it easier for the customers to switch to a similar product if their preferred product is out-of-stock or if there is a promotion of a similar product. Generally, the retailer can control out-of-stock situations by setting the supply limit parameter of the product p_j^s, which determines the maximum possible number of product items.

According to Czerniachowska and Hernes (2020, 2021) and Czerniachowska (2021), depending on the product packaging or its physical characteristics, the product item should be referred not only to one facing of the product but also to the possible capping or nesting of the product item. The capped unit is typically the top unit placed on top of the main units (e.g. for a box of cereal or tea). The nested unit is the unit above and inside the main unit (e.g. for coffee cups, plates, saucers). Parameters c_j^{max} and n_j^{max} restrict the maximum possible cappings and nestings on vertical dimension, which can be placed above the main facings without destroying it, and to ensure the stable vertical placement of cappings and nestings. Obviously, the total number of items of the product equals the sum of facings, cappings and nestings. Czerniachowska and Hernes (2020, 2021) and Czerniachowska (2021) shows the cappings and nestings allocation methods.

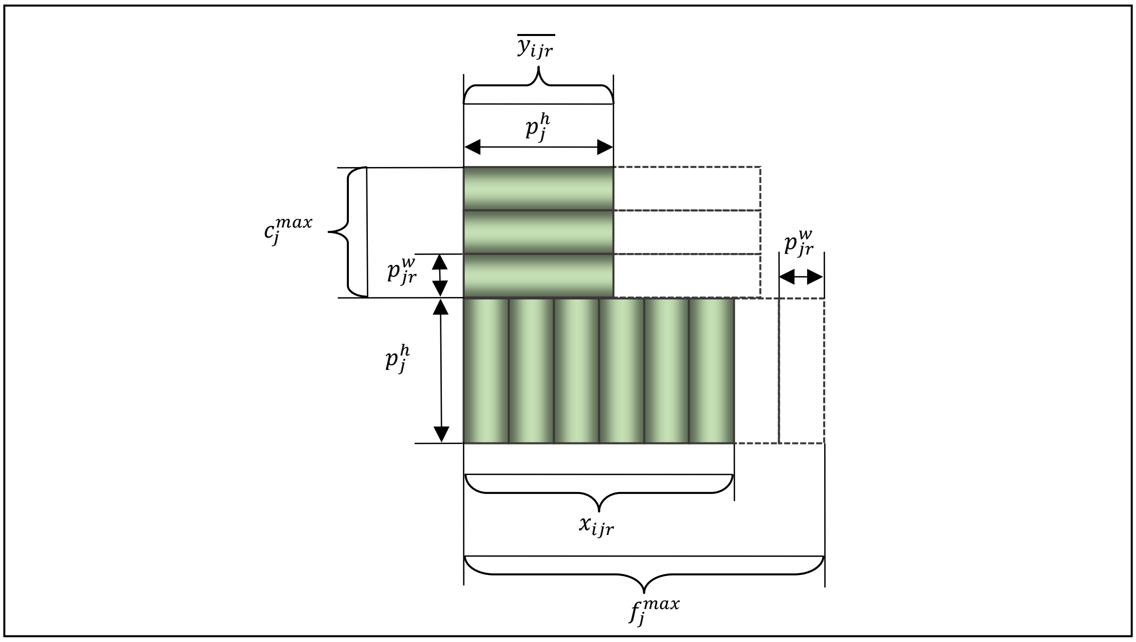

Fig. 2. Cappings allocation

Source: authors' work based on Czerniachowska (2021).

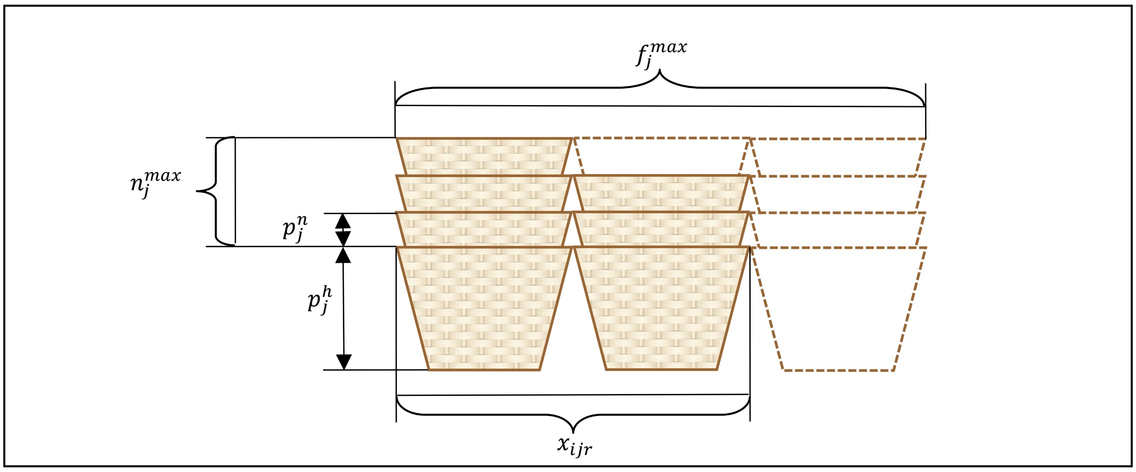

Fig. 3. Nestings allocation

Source: authors' work based on Czerniachowska (2021).

The authors used the variables and parameters listed below. The subscripts indicate a variable's indices, represent a variable's description or mnemonics, and should not be read as indices. M – m

Parameters and indices:

K – total number of category levels,
S – total number of shelves,
P – total number of products,
k – category level index, $k = 1,\ldots,K$,
i – shelf index, $i = 1,\ldots,S$,
j – product index, $j = 1,\ldots,P$,
$r = \begin{cases} 0, \text{ for front orientation} \\ 1, \text{ for side orientation} \end{cases}$ – orientation index.

Shelf parameters:

s_i^w – width of shelf i,
s_i^d – depth of shelf i,
s_i^h – height of shelf i,
s_i^k – category of shelf i.

Product parameters:

p_{jr}^w – width or depth of product j on orientation r,
$p_{jr}^w = \begin{cases} p_{j0}^w, \text{ if } r = 0, \text{ width for front orientation} \\ p_{j1}^w, \text{ if } r = 1, \text{ depth for side orientation} \end{cases}$,
p_j^h – height of product j,
p_j^s – supply limit of product j,
p_j^u – unit profit of product j,
p_j^k – sales potential category level of product j,
p_j^n – nesting height of product j, $p_j^n \le p_j^h$, or $p_j^n = 0$ if product cannot be nested,
$p_j^o = \begin{cases} 1, \text{ if side orientation is available for product } j \\ 0, \text{ otherwise} \end{cases}$ – orientation binary parameter,
p_j^c – cluster of product j,
f_j^{min} – minimum number of facings of product j,
f_j^{max} – maximum number of facings of product j,
c_j^{max} – maximum number of cappings per facings group of product j,
n_j^{max} – maximum number of nestings of one facing of product j.

Decision variables:

$\alpha_j = \begin{cases} 0, \text{ if product } j \text{ is on front orientation} \\ 1, \text{ if product } j \text{ is on side orientation} \end{cases}$ – orientation of product j,
$\alpha_j \in \{0,1\}$ for all $j = 1,\ldots,P$,
$u_{ijr} = \begin{cases} 1, \text{ if product } j \text{ is placed on shelf } i \text{ on orientation } r \\ 0, \text{ otherwise} \end{cases}$ – product placement binary variable,
$u_{ijr} \in \{0,1\}$ for all $i = 1,\ldots,S$, $j = 1,\ldots,P$, $r \in \{0,1\}$,
a_{jr} – lower shelf number where the product is placed,
b_{jr} – upper shelf number where the product is placed,
x_{ijr} – number of facings of product j on shelf i on orientation r,
$x_{ijr} = \{f_j^{min} \ldots f_j^{max}\}$ for all $i = 1,\ldots,S$, $j = 1,\ldots,P$, $r \in \{0,1\}$,
y_{ijr} – number of cappings of product j on shelf i on orientation r,

$y_{ijr} = \left\{ 0 \ldots c_j^{\max} \cdot \left\lfloor \frac{p_{jr}^w \cdot f_j^{\max}}{p_j^h} \right\rfloor \right\}$ for all $i = 1, \ldots, S, j = 1, \ldots, P, r \in \{0,1\}$),

$\overline{y_{ijr}}$ – the number of horizontal cappings of product j on shelf i on orientation r,

$\overline{y_{ijr}} = \left\{ 0 \ldots \left\lfloor \frac{p_{jr}^w \cdot f_j^{\max}}{p_j^h} \right\rfloor \right\}$ for all $i = 1, \ldots, S, j = 1, \ldots, P, r \in \{0,1\}$,

z_{ijr} – the number of nestings of product j on shelf i on orientation r,

$z_{ijr} = \{0 \ldots n_j^{max} \cdot f_j^{max}\}$ for all $i = 1, \ldots, S, j = 1, \ldots, P, r \in \{0,1\}$.

In the this research, the authors took into consideration one (top) facings row with cappings or nestings above it. The facings in a vertical and in a depth-dimension filling the possible shelf space in height and depth were not considered. Shelf height s_i^h and shelf depth capacity s_i^d was taken with regard to the first visible facings row. The task was to define the quantity number of facings x_{ijr}, cappings y_{ijr} and nestings z_{ijr} of product j on orientation r which was placed on shelf i with regard to the shelf, product, orientation and product group constraints, with the goal of maximising the retailers' profit. The horizontal cappings decision variable $\overline{y_{ijr}}$ ensures the correct calculation of the total number of cappings. Product placement binary decision variable u_{ijr} shows if the product physically exists on the shelf. Lower a_{jr} and upper b_{jr} shelf numbers are used for products that are placed on multiple shelves.

3. Problem formulation

The linear integer model can then be formulated as follows:

$$max \sum_{j=1}^{P} \sum_{i=1}^{S} \sum_{r=0}^{1} p_j^u (x_{ijr} + y_{ijr} + z_{ijr}) \tag{1}$$

subject to:

3.1. Constraints in SSAP-1

3.1.1. Shelf constraints

shelf width $\sum_{j=1}^{P} \sum_{r=0}^{1} p_{jr}^w x_{ijr} \leq s_i^w$ for all $i = 1, \ldots, S$ (2)

shelf depth:

for front orientation $x_{ij0} = 0$ for all $i = 1, \ldots, S, j = 1, \ldots, P, p_{j1}^w > s_i^d$ (3)

for side orientation $x_{ij1} = 0$ for all $i = 1, \ldots, S, j = 1, \ldots, P, p_{j0}^w > s_i^d$ (4)

product height $x_{ijr} = 0$ for all $i = 1, \ldots, S, j = 1, \ldots, P, r \in \{0,1\}, p_j^h > s_i^h$ (5)

3.1.2. Product constraints

supply limit $\sum_{i=1}^{S} \sum_{r=0}^{1} x_{ijr} + y_{ijr} + z_{ijr} \leq p_j^s$ for all $j = 1, \ldots, P$ (6)

minimum and maximum number of facings

$$f_j^{\min} \leq \sum_{i=1}^{S} \sum_{r=0}^{1} x_{ijr} \leq f_j^{\max} \text{ for all } j = 1, \ldots, P \tag{7}$$

maximum number of cappings

$$\begin{cases} y_{ijr} \leq c_j^{max} \cdot \overline{y_{ijr}} \\ y_{ijr} \leq \left\lfloor \frac{s_i^h - p_j^h}{p_{jr}^w} \right\rfloor \cdot \overline{y_{ijr}} \end{cases} \text{for all } i = 1, \ldots, S,\ j = 1, \ldots, P,\ r \in \{0,1\} \tag{8}$$

maximum number of horizontal cappings

$$p_j^h \overline{y_{ijr}} \leq p_{jr}^w x_{ijr} \text{ for all } i = 1,\ldots,S, j = 1,\ldots,P, r \in \{0,1\} \quad (9)$$

maximum number of nestings

$$\begin{cases} z_{ijr} \leq n_j^{max_{ijr}} \cdot x_{ijr} \\ z_{ijr} \leq \left\lfloor \frac{s_i^h - p_j^h}{p_j^n} \right\rfloor \cdot x_{ijr} \end{cases} \text{for all } i = 1,\ldots,S, j = 1,\ldots,P \quad (10)$$

3.1.3. Orientation constraints

side orientation is possible

$$\alpha_j \leq p_j^o \text{ for all } j = 1,\ldots,P \quad (11)$$

only one orientation is possible:

$$\begin{cases} \alpha_j x_{ij0} = 0 \\ (1-\alpha_j) x_{ij1} = 0 \end{cases} \text{for all } i = 1,\ldots,S, j = 1,\ldots,P \quad (12)$$

$$\begin{cases} \alpha_j y_{ij0} = 0 \\ (1-\alpha_j) y_{ij1} = 0 \end{cases} \text{for all } i = 1,\ldots,S, j = 1,\ldots,P \quad (13)$$

$$\begin{cases} \alpha_j \overline{y_{ij0}} = 0 \\ (1-\alpha_j) \overline{y_{ij1}} = 0 \end{cases} \text{for all } i = 1,\ldots,S, j = 1,\ldots,P \quad (14)$$

$$\begin{cases} \alpha_j z_{ij0} = 0 \\ (1-\alpha_j) z_{ij1} = 0 \end{cases} \text{for all } i = 1,\ldots,S, j = 1,\ldots,P \quad (15)$$

3.1.4. Product groups constraint

category level $\quad x_{ijr} = 0 \text{ for all } i = 1,\ldots,S, j = 1,\ldots,P, r \in \{0,1\}, p_j^k > s_i^k \quad$ (16)

3.2. Constraints in SSAP-2

SSAP-2 uses all the constraints from SSAP-1 with the addition of the following.

3.2.1. Multi-shelves constraints

product is placed on the shelf

$$\begin{cases} (1-u_{ijr}) x_{ijr} = 0 \\ x_{ijr} \geq u_{ijr} \\ \sum_{r=0}^{1} u_{ijr} \leq 1 \end{cases} \text{for all } i = 1,\ldots,S, j = 1,\ldots,P, r \in \{0,1\} \quad (17)$$

the same shelf for clusters

$$\sum_{r=0}^{1} u_{ier} = \sum_{r=0}^{1} u_{igr} \text{ for all } i = 1,\ldots,S, e,g: p_e^c = p_g^c, \quad e,g = 1,\ldots,P \quad (18)$$

the next shelf only

$$\sum_{i=1}^{S} \sum_{r=0}^{1} u_{ijr} = \sum_{r=0}^{1} b_{jr} - \sum_{r=0}^{1} a_{jr} + 1 \text{ for all } j = 1,\ldots,P \quad (19)$$

lower and upper shelf number where the product is placed

$$\begin{cases} (i - b_{jr}) u_{ijr} \leq 0 \\ (a_{jr} - i) u_{ijr} \leq 0 \end{cases} \text{for all } i = 1,\ldots,S, j = 1,\ldots,P, r \in \{0,1\} \quad (20)$$

3.3. Linearization technique in SSAP-1

Constraints (12) to (15) are not linear, hence the study used the following linearization technique proposed by Dantzig (1961). The method is based on using binary variables to outline various kinds of non-linear and logical conditions. Let z be a 0/1 variable and M be a big number. The linearization below uses the following assumptions:

$r = zy$ can be linearized as $y-(1-z)M \le r \le y+(1-z)M, -zM \le r \le zM$.

In this case

$$M = \left\{\begin{array}{ll} f_j^{\max}, & \text{if equation is with } x_{ijr} \\ c_j^{\max} \cdot \left\lfloor \frac{\max_{r=\{0,1\}}(p_{jr}^w)\cdot f_j^{\max}}{p_j^h} \right\rfloor, & \text{if equation is with } y_{ijr} \\ \left\lfloor \frac{\max_{r=\{0,1\}}(p_{jr}^w)\cdot f_j^{\max}}{p_j^h} \right\rfloor, & \text{if equation is with } \overline{y_{ijr}} \\ n_j^{\max} f_j^{\max}, & \text{if equation is with } z_{ijr} \end{array}\right\} \text{for all } j = 1,\dots,P$$

From constraint (12)

$$\alpha_j x_{ij0} = 0 \Rightarrow \begin{cases} r = zy \\ y = x_{ij0}, y \in [0, M] \\ z = \alpha_j, z = \{0,1\} \\ r = 0 \end{cases}$$

$$(1-\alpha_j)x_{ij1} = 0 \Rightarrow \begin{cases} r = zy \\ y = x_{ij1}, y \in [0, M] \\ z = 1-\alpha_j, z = \{0,1\} \\ r = 0 \end{cases}$$

$$y-(1-z)M \le r \le y+(1-z)M$$

$$\begin{cases} z = 0 \Rightarrow y - M \le 0 \le y + M \Rightarrow y \le M \\ z = 1 \Rightarrow y \le r \le y \Rightarrow y = 0 \end{cases}$$

$$\alpha_j x_{ij0} = 0 \Rightarrow \begin{cases} \alpha_j = 0 \Rightarrow x_{ij0} \le M \\ \alpha_j = 1 \Rightarrow \ \ x_{ij0} = 0 \end{cases}$$

$$(1-\alpha_j)x_{ij1} = 0 \Rightarrow \begin{cases} 1-\alpha_j = 0 \Rightarrow x_{ij1} \le M \\ 1-\alpha_j = 1 \Rightarrow \ \ x_{ij1} = 0 \end{cases} \Rightarrow \begin{cases} \alpha_j = 1 \Rightarrow x_{ij1} \le M \\ \alpha_j = 0 \Rightarrow \ \ x_{ij1} = 0 \end{cases}$$

$$\begin{cases} \alpha_j = 0 \Rightarrow x_{ij0} \le M, \ \ x_{ij1} = 0 \\ \alpha_j = 1 \Rightarrow \ \ x_{ij0} = 0, \ \ \ x_{ij1} \le M \end{cases} \Leftrightarrow \begin{cases} x_{ij0} \le (1-\alpha_j)M \\ x_{ij1} \le \alpha_j M \end{cases}$$

for all $i = 1,\dots,S, j = 1,\dots,P$

Constraints (12) to (15) could be rewritten as follows:

$$\begin{cases} x_{ij0} \le (1-\alpha_j) f_j^{max} \\ x_{ij1} \le \alpha_j f_j^{max} \{ \end{cases} \text{for all } i = 1,\dots,S, j = 1,\dots,P \tag{21}$$

$$\begin{cases} y_{ij0} \le (1-\alpha_j) c_j^{\max} \cdot \left\lfloor \frac{\max_{r=\{0,1\}}(p_{jr}^w)\cdot f_j^{\max}}{p_j^h} \right\rfloor \\ y_{ij1} \le \alpha_j c_j^{\max} \cdot \left\lfloor \frac{\max_{r=\{0,1\}}(p_{jr}^w)\cdot f_j^{\max}}{p_j^h} \right\rfloor \end{cases} \text{for all } i = 1,\dots,S, j = 1,\dots,P \tag{22}$$

$$\begin{cases} \overline{y_{ij0}} \leq (1-\alpha_j) \cdot \left\lfloor \frac{\max\limits_{r=\{0,1\}}(p_{jr}^{w}) \cdot f_j^{\max}}{p_j^h} \right\rfloor \\ \overline{y_{ij1}} \leq \alpha_j \cdot \left\lfloor \frac{\max\limits_{r=\{0,1\}}(p_{jr}^{w}) \cdot f_j^{\max}}{p_j^h} \right\rfloor \end{cases} \text{for all } i = 1,\dots,S,\ j = 1,\dots,P \quad (23)$$

$$\begin{cases} z_{ij0} \leq (1-\alpha_j) n_j^{\max} f_j^{\max} \\ z_{ij1} \leq \alpha_j n_j^{\max} f_j^{\max} \end{cases} \text{for all } i = 1,\dots,S,\ j = 1,\dots,P \quad (24)$$

3.4. Linearization technique in SSAP-2

In SSAP-2, constraints (17) are not linear, therefore the authors used the same linearization technique as in SSAP-1.

$$(1-u_{ijr})x_{ijr} = 0 \Rightarrow \begin{cases} r = zy \\ y = x_{ijr}, y \in [0, M] \\ z = 1-u_{ijr}, z = \{0,1\} \\ r = 0 \\ M = f_j^{max} \end{cases}$$

$$(1-u_{ijr})x_{ijr} = 0 \Rightarrow \begin{cases} 1-u_{ijr} = 0 \Rightarrow x_{ijr} \leq M \\ 1-u_{ijr} = 1 \Rightarrow \ x_{ijr} = 0 \end{cases} \Rightarrow \begin{cases} u_{ijr} = 1 \Rightarrow \ x_{ijr} \leq M \\ u_{ijr} = 0 \Rightarrow \ x_{ijr} = 0 \end{cases} \Rightarrow \ x_{ijr} \leq u_{ijr}M$$

Constraints (17) could be rewritten as follows:

$$\begin{cases} x_{ijr} \leq u_{ijr} f_j^{max} \\ x_{ijr} \geq u_{ijr} \\ \sum_{r=0}^{1} u_{ijr} \leq 1 \end{cases} \text{for all } i = 1,\dots,S,\ j = 1,\dots,P,\ r \in \{0,1\} \quad (25)$$

In SSAP-2, constraints (20) are not linear, therefore they were linearized. Constraints (20) could be rewritten as follows:

$$u_{ijr}a_{jr} \leq i \Rightarrow \begin{cases} u_{ijr}a_{jr} = d_{ijr} \\ d_{ijr} \leq i \end{cases} \Rightarrow \begin{cases} r = zy \\ y = a_{jr}, y \in [1, M] \\ z = u_{ijr}, z = \{0,1\} \Rightarrow \\ r = d_{ijr} \\ M = S \end{cases}$$

$$\Rightarrow \begin{cases} a_{jr} - (1-u_{ijr})S \leq d_{ijr} \leq a_{jr} + (1-u_{ijr})S \\ 0 \leq d_{ijr} \leq S \\ d_{ijr} \leq i \end{cases} \Rightarrow \begin{cases} a_{jr} - (1-u_{ijr})S \leq d_{ijr} \leq a_{jr} + (1-u_{ijr})S \\ 0 \leq d_{ijr} \leq i \end{cases}$$

lower and upper shelf number where the product is placed

$$\begin{cases} b_{jr} \geq u_{ijr} i \\ \sum_{r=0}^{1} b_{jr} \leq S \\ \sum_{r=0}^{1} a_{jr} \geq 1 \\ a_{jr} \leq b_{jr} \\ a_{jr} - (1-u_{ijr})S \leq d_{ijr} \leq a_{jr} + (1-u_{ijr})S \\ 0 \leq d_{ijr} \leq i \end{cases} \text{for all } i = 1,\dots,S,\ j = 1,\dots,P,\ r \in \{0,1\} \quad (26)$$

4. Computational experiment

4.1. Experimental data

The computational experiments show the possibility of finding the optimal solution due to applying linearization techniques to both problems (SSAP-1 and SSAP-2) on different problem sizes, especially on very large instances. The data were simulated based on real-life data. In accordance with Yang (2001), Lim et al. (2004), and Bai and Kendall (2005), the authors generated 23 product sets randomly, with a normal distribution with different parameters.

Here is a description of how the product parameters could be adopted. First, a real planogram with products of the definite category was selected. In a real store, on a planogram all product width, depth, height, price, etc. are shown, based on this, the mean or expectation of the distribution and standard deviation for the category could be estimated. Next, the number of products in the tested set is set, and then the achieved values can be applied in the formula of the normal distribution, for example, Microsoft Excel includes a random value generator of a normal distribution with mean and standard deviation input parameters. These values should be randomly assigned to the products on the tested set, and then another number of products set, repeating the parameter generation process for them, or repeating the parameter generation process for the same number of products which could represent another test product set.

The generated 23 sets of products contain 10, 15, …, 300 products. Each set of products contains five products more than the previous one. Thus five planogram widths were modelled of 250 cm, 375 cm, 500 cm, 625 cm, and 750 cm. The same widths of all the shelves on the planogram were chosen because this is the most frequent case at a store. Each planogram contains three, four or five shelves. Product sets and shelves were assigned to three horizontal sales potentials subcategories.

The optimal solution was found using the commercial solver IBM ILOG CPLEX Optimization Studio Version: 12.7.1.0.

4.2. SSAP-1

In SSAP-1, 0 and 0 show the results where solutions exist. They do not show instances with unfeasible data where all the entries are at implied bounds, e.g. too many products were tried to be placed on too short shelves. 0 shows the very large numbers of variables and constraints that were modelled in SSAP-1. Nevertheless, the optimal solution was found for all of the presented product sets.

Tabela 2. Number of variables and constraints in shelf space allocation problem SSAP-1

Shelves	Products	Constraints	Decision variables		Nonzero constraint coefficients
			Integer variables	Binary variables	
3	10	595	240	10	1,270
	15	907	360	15	1,921
	20	1,207	480	20	2,536
	25	1,505	600	25	3,173
	30	1,803	720	30	3,786
	35	2,103	840	35	4,425
	40	2,405	960	40	5,042
	45	2,707	1,080	45	5,683
	50	3,005	1,200	50	62,296
	55	3,309	1,320	55	6,933
	60	3,601	1,440	60	7,546
	70	4,189	1,680	70	9,280
	80	4,807	1,920	80	10,648
	90	5,399	2,160	90	11,990

4	10	790	320	10	1,700
	15	1,180	480	15	2,574
	20	1,580	640	20	3,372
	25	1,972	800	25	4,221
	30	2,364	960	30	5,038
	35	2,750	1,120	35	5,881
	40	3,148	1,280	40	6,704
	45	3,534	1,440	45	7,547
	50	3,938	1,600	50	8,376
	55	4,352	1,760	55	9,234
	60	4,728	1,920	60	10,048
	70	5,524	2,250	70	12,382
	90	7,038	2,880	90	15,916
	125	9,888	4,000	125	22,221
	150	11,860	4,800	150	26,606
	175	13,790	5,600	175	31,125
5	10	975	400	10	2,120
	15	1,457	600	15	3,177
	25	2,449	1,000	25	5,279
	30	2,931	1,200	30	6,296
	35	3,425	1,400	35	7,365
	40	3,899	1,600	40	8,374
	45	4,395	1,800	45	9,445
	50	4,893	2,000	50	10,478
	55	5,389	2,200	55	11,534
	60	5,879	2,400	60	12,574
	90	8,811	3,600	90	19,976
	125	12,271	5,000	125	27,781
	150	14,703	6,000	150	33,248
	175	17,135	7,000	175	38,935
	200	19,593	8,000	20	44,448
	275	26,947	11,000	275	61,002

Source: authors' work.

Table 3 shows the time which is needed to find an optimal solution in SSAP-1. It was observed that 93 out of the 134 instances were solved in less than a second. Obviously, for very large instances, the time increased up to approximately 1 minute. The longest time of 75 seconds was spent finding a solution for 50 products on four shelves of 250 cm. The average solution time was 3.73 seconds.

Table 3. Solution time for shelf space allocation problem SSAP-1

No. of shelves eq. 3			No. of shelves eq. 4			No. of shelves eq. 5		
No. of products	Shelf width [cm]	Solution time [s]	No. of products	Shelf width [cm]	Solution time [s]	No. of products	Shelf width [cm]	Solution time [s]
10	250	0.297	10	250	0.015	10	250	0.062
	375	0.344		375	0.031		375	0.390
	500	0.031		500	0.203		500	0.187
	625	0.422		625	1.718		625	0.187
	750	0.188		750	0.016		750	0.125
15	250	0.078	15	250	0.578	15	250	1.078
	375	0.094		500	0.671	25	250	2.015
	500	1.984		625	1.609		375	5.108
	625	0.250		750	0.359		750	5.546
	750	0.109	20	250	0.062	30	500	0.234

20	250	0.047		375	0.531		625	2.556
	375	0.297		625	0.125		750	0.281
	500	0.391		750	0.313	35	250	0.219
	625	0.578	25	250	0.062		375	0.906
	750	0.265		375	0.485	40	750	43.787
25	250	0.047		500	0.438	45	250	0.796
	375	0.031	30	250	9.326		375	0.672
	500	0.047		375	21.792		500	3.921
	625	0.578		500	0.453		625	9.357
	750	0.453		625	0.328	50	250	23.776
30	250	0.047		750	0.297		500	0.391
	375	0.218	35	250	2.265	55	500	0.359
	625	2.749		375	0.484		625	4.562
	750	0.062		625	4.015		750	2.655
35	250	0.031		750	1.188	60	250	0.656
	375	0.281	40	250	0.328		375	0.641
	500	9.545		500	4.281		625	0.531
	625	0.594	45	250	0.563		750	0.390
	750	0.312		625	6.842	90	375	0.140
40	250	0.109		750	0.281	125	500	7.045
	500	0.953	50	250	75.139	150	625	5.311
	625	0.500		375	0.516		750	4.780
	750	0.203		500	0.047	175	625	2.125
45	250	0.188		750	1.250		750	18.636
	375	0.156	55	625	1.828	200	625	10.123
	500	0.157	60	250	40.935		750	1.781
	625	0.391		375	0.406	275	750	20.277
	750	0.250	70	375	0.437			
50	375	0.109		500	0.453			
	500	0.016	90	500	1.125			
	625	0.375		750	10.279			
	750	0.047	125	625	61.251			
55	750	0.281	150	750	0.937			
60	250	0.016	175	750	2.218			
	375	0.047						
	500	0.281						
	625	9.451						
	750	0.140						
70	500	0.078						
	625	0.500						
	750	26.415						
80	750	0.062						
90	750	0.593						

Source: authors' work.

Fig. 4, Fig. 5 and Fig. 6 show the time needed for CPLEX mixed-integer programming optimisation in SSAP-1 for three, four, and five shelves, respectively. The fastest solution was found for three shelves because the number of constraints and decision variables was the smallest. Only one instance was solved in more than 20 seconds. For four shelves, four instances were solved in more than 20 seconds. For five shelves, three instances were solved in over 20 seconds, but also very few instances were solved in the time shorter than 1 second as was the case with the three shelves.

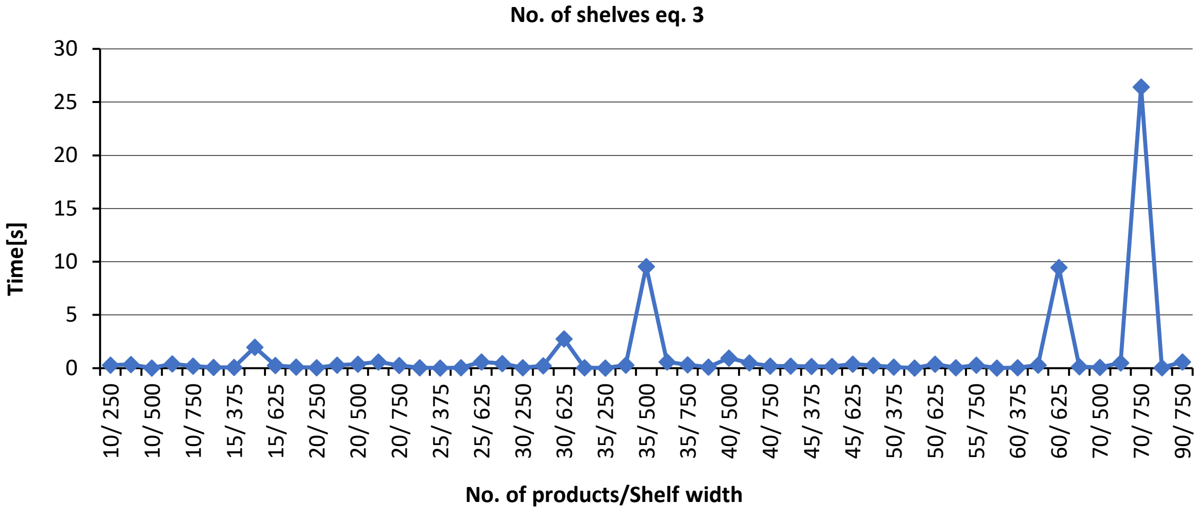

Fig. 4. CPLEX MIP optimisation for three shelves for SSAP-1

Source: authors' work.

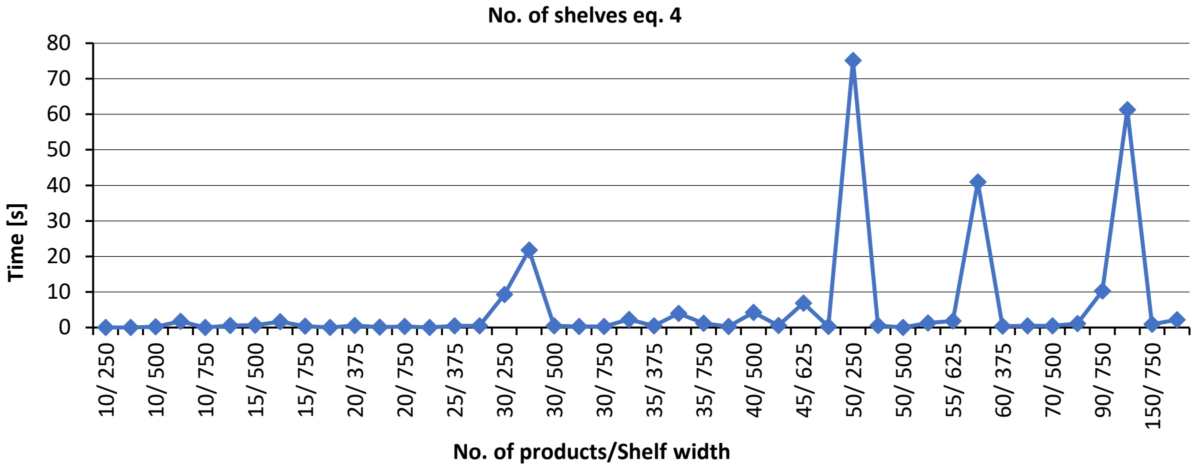

Fig. 5. CPLEX MIP optimisation for four shelves for SSAP-1

Source: authors' work.

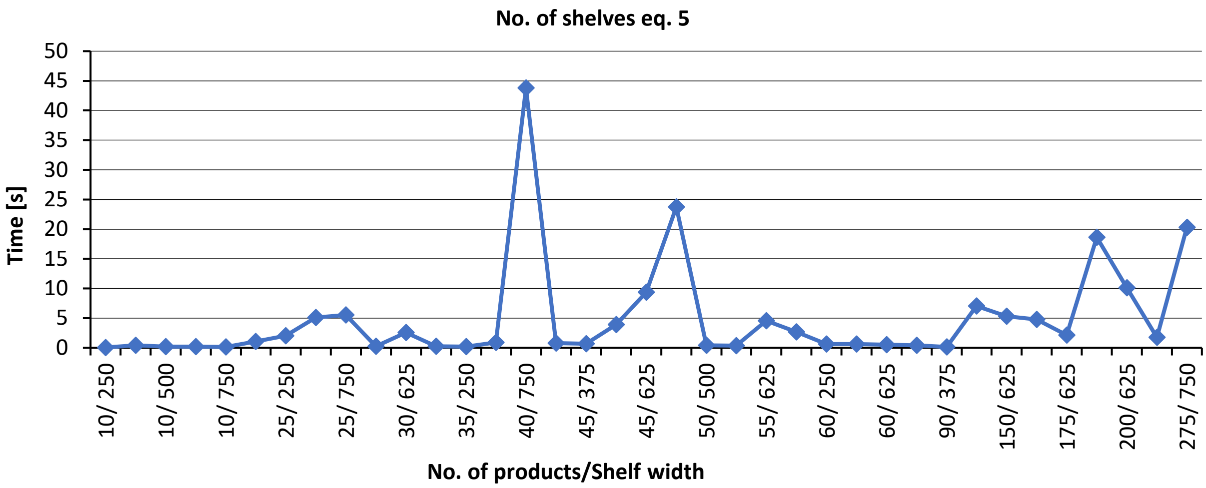

Fig. 6. CPLEX MIP optimisation for five shelves for SSAP-1

Source: authors' work.

4.3. SSAP-2

In SSAP-2, Table 4 and Table 5 show the results where solutions exist. They do not show instances with unfeasible data where all the entries are at implied bounds, e.g. trying to place too many products on too short shelves. Table 4 shows the very large numbers of variables and constraints that were modelled in SSAP-2. Nevertheless, the optimal solution was found for all of the presented product sets.

Table 4. Number of variables and constraints in shelf space allocation problem SSAP-2

Shelves	Products	Constraints	Decision variables		Nonzero constraint coefficients
			Integer variables	Binary variables	
3	10	1,107	340	70	2,398
	15	1,669	510	105	3,589
	20	2,255	680	140	4,888
	25	2,197	850	175	6,041
	30	3,387	1,020	210	7,362
	35	3,967	1,190	245	8,661
	40	4,471	1,360	280	9,626
	45	5,029	1,530	315	10,831
	50	5,589	1,700	350	12,032
	60	6,733	2,040	420	14,554
	70	7,737	2,380	490	17,032
	90	9,935	3,060	630	218,554
4	10	1,456	440	90	3,164
	15	2,171	660	135	4,711
	20	2,944	880	180	6,428
	25	3,653	1,100	225	7,945
	30	4,426	1,320	270	9,686
	35	5,177	1,540	315	11,389
	40	5,836	1,760	360	12,656
	45	6,555	1,980	405	14,231
	50	7,300	2,200	450	15,824
	55	8,079	2,420	495	17,542
	60	8,804	2,640	540	19,152
	70	10,138	3,080	630	22,438
	90	12,936	3,960	810	28,708
	125	18,061	5,500	1,125	39,913
	150	21,658	6,600	1,350	47,798
	175	25,229	7,700	1,575	55,881
5	10	1,795	540	110	3,920
	15	2,677	810	165	5,837
	20	3,639	1,080	220	7,974
	25	4,519	1,350	275	9,859
	30	5,471	1,620	330	12,016
	35	6,415	1,890	385	14,145
	40	7,209	2,160	440	15,694
	45	8,151	2,430	495	17,665
	50	9,033	2,700	550	19,638
	55	9,979	2,970	605	21,754
	60	10,899	3,240	660	23,774
	70	12,521	3,780	770	27,826
	90	16,071	4,860	990	35,696
	125	22,331	6,750	1,375	49,521
	150	26,763	8,100	1,650	59,288
	175	31,215	9,450	1,925	69,355
	200	35,683	10,800	2,200	79,208

Source: authors' work.

Table 5. Solution time for shelf space allocation problem SSAP-2

No. of shelves eq. 3			No. of shelves eq. 4			No. of shelves eq. 5		
Products	Shelf width	Time [s]	Products	Shelf width	Time [s]	Products	Shelf width	Time [s]
10	250	0.282	10	250	0.063	10	250	0.203
	375	0.391		375	0.078		375	0.562
	500	0.656		500	2.609		500	0.625
	625	0.297		625	1.328		625	0.453
	750	0.094		750	1.468		750	0.468
15	250	0.032	15	250	1.546	15	250	0.203
	375	0.016		375	2.453		375	2.937
	500	0.047		500	2.562	20	250	1.171
	625	0.047		625	1.390		375	6.623
	750	0.093		750	0.859		625	1.828
20	375	0.032	20	250	0.125	25	250	0.453
	500	0.047		375	0.281		375	1.515
	625	0.094		500	0.235		625	2.234
	750	0.079		625	0.188		750	0.750
25	250	0.047		750	0.562	30	250	7.904
	375	0.047	25	250	1.187		375	1.546
	500	0.078		375	0.281	35	250	1.624
	625	0.093		500	2.515		625	31.868
	750	0.094		625	0.594	40	750	8.857
30	375	0.062		750	9.903	45	375	2.077
	500	0.031	30	250	0.156		500	51.676
	625	0.031		375	0.172		625	19.496
	750	0.032		500	1.703	50	250	0.969
35	375	0.047		625	0.953		375	0.906
	500	0.094		750	0.234		500	2.187
	625	0.110	35	250	0.109		625	0.594
	750	0.094		375	0.828	55	500	1.156
40	375	0.344		500	0.859		625	39.209
	500	0.156		625	0.281		750	12.997
	625	0.499		750	0.921	60	250	1.297
	750	0.656	40	250	0.296		500	1.734
45	375	0.109		500	0.063		625	3.171
	500	0.156		625	1.109		750	1.562
	625	0.094		750	1.546	70	500	28.165
	750	0.109	45	250	0.985		750	18.730
50	500	0.063		625	1.406	90	375	1.687
	625	0.093		750	0.860	125	500	2.328
	750	0.078	50	250	0.140		625	7.060
60	375	0.093		375	0.328	150	625	2.937
	500	0.281		500	0.234		750	22.291
	625	0.235		625	0.906	175	625	59.376
	750	0.703		750	1.249		750	36.382
70	500	0.188	55	625	0.813	200	625	14.419
	625	0.390		750	1.203		750	10.091
	750	8.514	60	250	6.045			
90	750	0.952		375	18.465			
				500	6.749			
				625	5.014			
			70	500	1.812			
				625	23.853			
			90	500	1.828			
				750	4.062			
			125	625	2.765			
				750	2.827			
			150	750	1.312			
			175	750	6.936			

Source: authors' work.

Table 5 shows the time needed to find the optimal solution in SSAP-2. It could be observed that 84 out of the146 instances were solved in less than a second. Obviously, for very large instances, the time increased to approximately 1 minute. The longest time of 59 seconds was spent to find a solution for 175 products on eight shelves of 625 cm. The average solution time is 3.838 seconds.

Fig. 7, Fig. 8 and Fig. 9 show the time needed for CPLEX MIP optimisation in SSAP-2 for three, four and five shelves, respectively. The fastest solution was found for three shelves because the number of constraints and decision variables was the smallest. Only one instance was solved in 8-9 seconds, whereas most were solved in less than a second. For four shelves, also 1 instance was solved in more than 20 seconds. For five shelves, seven were solved in over 20 seconds, but also very few instances were solved in the time shorter than one second as in the case with the three shelves. Hence, five shelves case took more time compared to 3-shelf and 4-shelf cases.

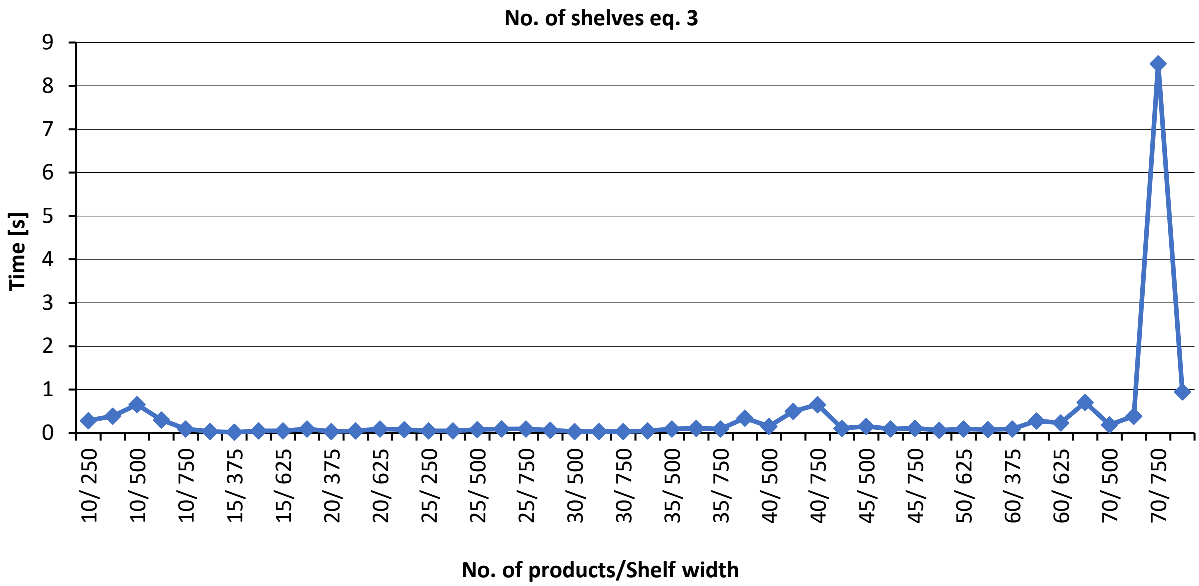

Fig. 7. CPLEX MIP optimisation for three shelves for SSAP-2

Source: authors' work.

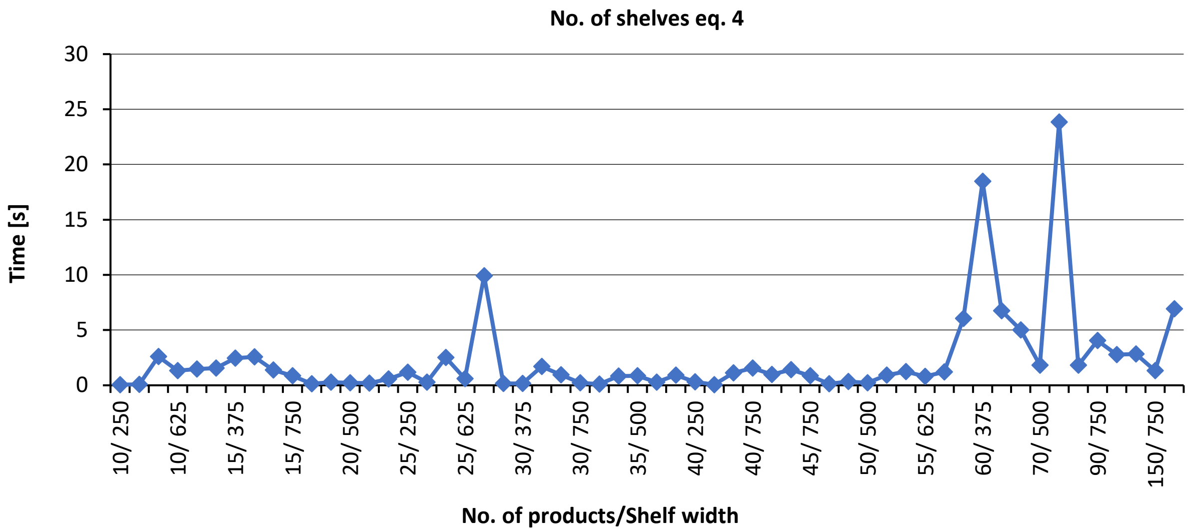

Fig. 8. CPLEX MIP optimisation for four shelves for SSAP-2

Source: authors' work.

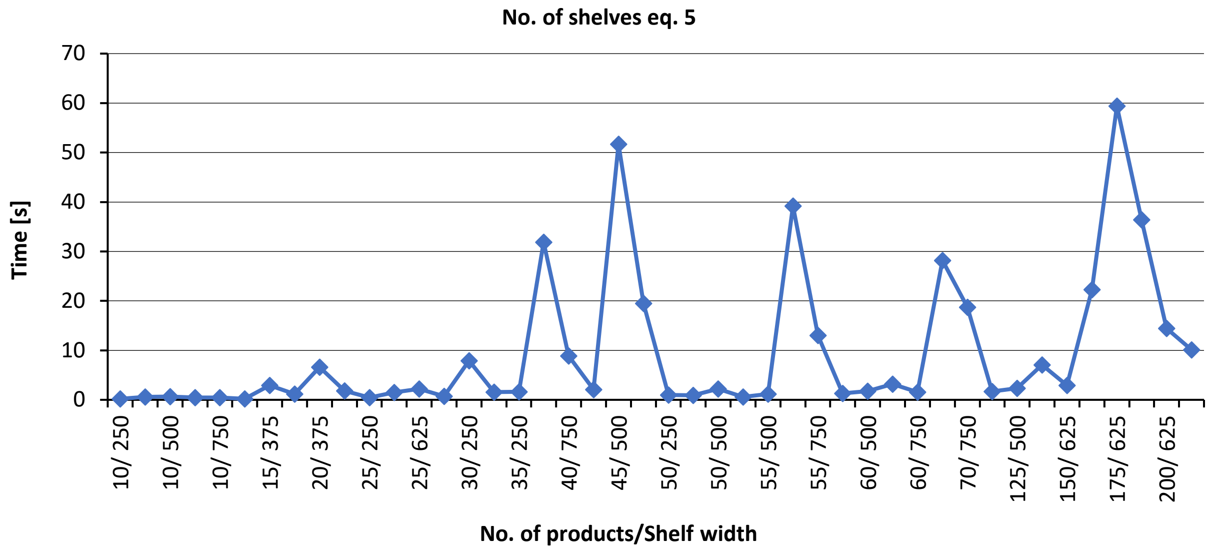

Fig. 9. CPLEX MIP optimisation for five shelves for SSAP-2

Source: authors' work.

The model timing results can be useful in the retail store case designs to improve shelf space planning and management operations. The generalisation of the computational experiment given here may be possible with more research conducted in additional market sectors.

5. Conclusion

SSAP is of crucial importance for retailers. By determining the number of items to be placed on the shelves and offered to customers, retailers make fundamental decisions that influence brand visibility as well as customer satisfaction which results in the obtained store profit.

Several important issues in the shelf space were identified. Generally, SSAP is modelled as non-linear, it is known to be NP-hard because of a large number of constraints. In this article, the authors presented two SSAPs, in each one, non-linear constraints existed, which made it impossible to find an optimal solution, especially on large problem instances. SSAP-1 includes shelf, product, and product group constraints. SSAP-2 includes additional multi-shelf constraints; both models maximise the total profit while allocating products on a planogram.

The authors adjusted the linearization technique and rewrote come constraints in a linear form which allowed to find the optimal solution with the help of the commercial CPLEX solver. Most of the instances were solved in less than a second, which makes this method to be very powerful for real retail problem cases, and outperforms alternative approaches. This method allows retailers to find the optimal number of product items that should be placed on shelves without human intervention. The proposed method could also be used by category managers in analysing the economic impacts of shelf space planning in retail stores, as the process of shelf space allocation is mostly dictated by operational constraints. Considering the merchandising rules, capping and nesting allows to adjust the arrangement of products to the customer's needs in a better way. The research findings could be utilised to construct a shelf space allocation module in retail information systems.

The SSAP literature is rich in various models. For example, other SSAPs include dividing the shelf into segments of various attractiveness, hierarchical product categorisation, product grouping possibilities, store traffic management, assortment optimisation etc. Each of these directions provides interesting insights for SSAP modelling. Future research could be focused on analysing other non-linear SSAP models in order to propose the methods of decision variables and constraints modelling so that they can be rewritten in a linear form. Thus, there are numerous of intriguing, practically relevant problems and research directions that can be pursued in the future.

The project is financed by the Ministry of Science and Higher Education in Poland under the programme "Regional Initiative of Excellence" 2019-2022 project number 015/RID/2018/19 total funding amount 10,721,040.00 PLN.

References

Anic, I. D., Radas, S., & Lim, L. K. S. (2010). Relative effects of store traffic and customer traffic flow on shopper spending. *International Review of Retail, Distribution and Consumer Research* 20(2), 237–250. https://doi.org/10.1080/09593961003701841

Bai, R., & Kendall, G. (2005). An investigation of automated planograms using a simulated annealing based hyper-heuristic. *Operations Research/ Computer Science Interfaces Series*, 32, 87–108. https://doi.org/10.1007/0-387-25383-1_4

Bianchi-Aguiar, T. (2015). *The Retail Shelf Space Allocation Problem: New Optimization Methods Applied to a Supermarket Chain*. PhD thesis, Faculdade de Engenharia da Universidade do Porto, Portugal.

Bianchi-Aguiar, T., Silva, E., Guimarães, L., Carravilla, M. A., & Oliveira, J. F. (2018). Allocating products on shelves under merchandising rules: Multi-level product families with display directions. *Omega (United Kingdom)*, 76, 47–62. https://doi.org/10.1016/j.omega.2017.04.002

Chandon, P., Hutchinson, J. W., Bradlow, E. T., & Young, S. H. (2009). Does in-store marketing work? Effects of the number and position of shelf facings on brand attention and evaluation at the point of purchase. *Journal of Marketing*, 73(6), 1–17. https://doi.org/10.1509/jmkg.73.6.1

Chen, M. C., & Lin, C. P. (2007). A data mining approach to product assortment and shelf space allocation. *Expert Systems with Applications*, 32(4), 976–986. https://doi.org/10.1016/j.eswa.2006.02.001

Corstjens, M., & Doyle, P. (1981). A model for optimizing retail space allocations. *Management Science*, 27(7), 822–833. https://doi.org/10.1287/mnsc.27.7.822

Curhan, R. C. (1972). The relationship between shelf space and unit sales in supermarkets. *Journal of Marketing Research*, 9(4), 406–412. https://doi.org/10.1177/002224377200900408

Czerniachowska, K. (2021). A genetic algorithm for the retail shelf space allocation problem with virtual segments. *OPSEARCH*. https://doi.org/10.1007/s12597-021-00551-3

Czerniachowska, K., & Hernes, M. (2020). A genetic algorithm for the shelf-space allocation problem with vertical position effects. *Mathematics*, 8(11), 1881. https://doi.org/10.3390/math8111881

Czerniachowska, K., & Hernes, M. (2021). Simulated annealing hyper-heuristic for a shelf space allocation on symmetrical planograms problem. *Symmetry*, 13(7), 1182. https://doi.org/10.3390/sym13071182

Dantzig, G. B. (1960). On the significance of solving linear programming problems with some integer variables. *Econometrica*, 28(1), 30–44. https://doi.org/10.2307/1905292

Desrochers, D. M., & Nelson, P. (2006). Adding consumer behavior insights to category management: Improving item placement decisions. *Journal of Retailing*, 82(4), 357–365. https://doi.org/10.1016/j.jretai.2006.08.009

Elbers, T. (2016). The effects of in-store layout and shelf designs on consumer behaviour. http://edepot.wur.nl/369091

Gabrielli, V., & Cavazza, N. (2014). The influence of in-store product holders on orientation towards the product and on purchase intention. *International Review of Retail, Distribution and Consumer Research*, 24(3), 311–327. https://doi.org/10.1080/09593969.2013.862507

Lim, A., Rodrigues, B., & Zhang, X. (2004). Metaheuristics with local search techniques for retail shelf-space optimization. *Management Science*, 50(1), 117–131. https://doi.org/10.1287/mnsc.1030.0165

Park, H. H., Jeon, J. O., & Sullivan, P. (2015). How does visual merchandising in fashion retail stores affect consumers' brand attitude and purchase intention? *International Review of Retail, Distribution and Consumer Research*, 25(1), 87–104. https://doi.org/10.1080/09593969.2014.918048.

Schaal, K., & Hübner, A. (2018). When does cross-space elasticity matter in shelf-space planning? A decision analytics approach. *Omega (United Kingdom)*, 80, 135–152. https://doi.org/10.1016/j.omega.2017.08.015.

Yang, M. H. (2001). Efficient algorithm to allocate shelf space. *European Journal of Operational Research*, 131(1), 107–118. https://doi.org/10.1016/S0377-2217(99)00448-8

Received: April 2022, revised: September 2022