International Journal of Retail & Distribution Management

Heuristics for retail shelf space allocation problem with linear profit function

Hasmukh K. Gajjar Gajendra K. Adil

IJRDM
39,2

Received 3 July 2009
Revised 9 April 2010
Accepted 15 July 2010

Heuristics for retail shelf space allocation problem with linear profit function

Hasmukh K. Gajjar
Indian Institute of Management Indore, Indore, India, and
Gajendra K. Adil
Indian Institute of Technology-Bombay, Mumbai, India

Abstract

Purpose – Shelf space is often retailer's critical resource. Growing number of products has posed a challenge to the retailers for efficient allocation of available shelf space to them. The paper aims to consider a retail shelf space allocation problem with linear profit function and aims to develop efficient heuristics to solve this problem.

Design/methodology/approach – The paper develops three heuristics to solve a shelf space allocation problem. It compares three heuristics with existing heuristic using empirical study.

Findings – In an empirical study of 320 randomly generated instances of problems with size (products, shelves) varying from (25, 5) to (200, 50), it was found that all three new heuristics are competitive with existing heuristic. The best amongst three heuristics found solution with average objective value of 99.59 percent of upper bound in a reasonable central processing unit time.

Research limitations/implications – The linearity assumption of the profit function is based on earlier findings that marginal returns to space first increase and then decrease in an S-shaped curve. Hence, linearity assumption for profit function is justified by the fact that retails would want to operate on linear (or approximately linear) and more strongly increasing part of the curve.

Practical implications – The proposed heuristics are applied to a case of existing retail store which gave more profit than the current allocation scheme.

Originality/value – The paper proposes new initial constructor and neighbourhood move strategy to develop efficient heuristic. Heuristics proposed in this paper are competitive with existing heuristics.

Keywords Retailing, Shelf space, Profit

Paper type Research paper

1. Introduction

Shelf space is often retailer's critical resource. Efficient shelf space allocation is one of key determinants to gain an edge in the highly competitive retail industry (Lim *et al.*, 2004). Growing number of products has posed a challenge to the retailers in allocating available shelf space to them efficiently. Several models are developed for allocation of shelf space to a large number of products to optimize retailer's objective under certain operating conditions within a store. Yang and Chen (1999) emphasize the importance of store space allocation decisions and their powerful incentives to the researchers to make such decisions correctly. Decisions involved in shelf space allocation can be viewed at two levels:

(1) How much shelf space to allocate to a particular product category?

(2) How much shelf space to allocate to each different product within a product category? (Reyes and Frazier, 2007).

International Journal of Retail & Distribution Management
Vol. 39 No. 2, 2011
pp. 144-155

0959-0552
DOI 10.1108/09590551111109094

Literature on shelf space consists of:

- experimental studies on shelf space;
- industrial practices on shelf space allocation; and
- optimization models on shelf space allocation problem (SSAP).

A number of experiments (Curhan, 1972) found positive relationship between shelf space allocated to a product and its sales which is described as space elasticity of product. While some other experimental studies (Drèze *et al.*, 1994) found that the level in the shelf at which the product is displayed significantly impact its sales. Retailers employ planogram tool and other shelf space management systems (e.g. Apollo, Spacemen, and PROGALI) that use relatively simple heuristics for shelf space allocation (Zufryden, 1986; Yang, 2001; Bai, 2005) and they are clearly sub optimal (Curhan, 1972; Yang, 2001).

A number of optimization models on shelf space allocation have also been developed by the researchers. Corstjens and Doyle (1981) developed a geometric programming model considering space elasticity and cross elasticity of products and solved it using the branch and bound algorithm as proposed by Gochet and Smeers (1979). They extended this model to a dynamic model (Corstjens and Doyle, 1983) that incorporates the goodwill or carryover effects which retailers accrue from a given merchandise area over a period. Thus, it can encourage the retailers to allocate more space to new products and divest earlier from declining ones. Zufryden (1986) proposed a dynamic programming model in which demand function considered space elasticity, cost of sales and demand-related marketing variables while neglecting cross elasticity between products and fixing non-space marketing variables for tractability and efficiency of the solution procedure.

Few integrated models jointly considered product assortment, shelf space allocation and inventory replenishment decisions to maximize retailer's profit under the given operating constraints (Urban, 1998, 2002; Hwang *et al.*, 2005; Hariga *et al.*, 2007; Bai and Kendall, 2008). More recently, several other approaches like goal programming (Reyes and Frazier, 2007), game theory (Amrouche and Zaccour, 2007) and data mining (Chen and Lin, 2007) are used for product assortment and allocation problems in retailing.

The several optimization models on SSAP found in the literature are complex and have some practical limitations. There is lack of exact method and efficient heuristics in literature which can be implemented to allocate shelf space to a large number of products in a retail store. However, several efforts were recently made by researchers (Yang and Chen, 1999, Yang, 2001; Bai, 2005; Gajjar and Adil, 2010; Lim *et al.*, 2004) towards development of tractable models and efficient heuristics to solve them.

The existing models often neglected the effect of product's location which was found to be significant by Drèze *et al.* (1994). Therefore, Yang and Chen (1999) considered location effects in non-linear model of Corstjens and Doyle (1981) and proposed a comprehensive model. Bai (2005) simplified the comprehensive model of Yang and Chen (1999) by dropping product availability constraints and ignoring effects of cross elasticity, non-space variables and location effects in demand function. Several meta-heuristics and hyper-heuristic approaches were applied to solve his model. Gajjar and Adil (2010) proposed an alternate linear model for the non-linear model of Bai (2005) using piecewise linearization. They obtained the tight upper bounds using linear programming relaxation in their linear model. Gajjar and Adil (2010) developed a local search heuristic which obtained better results compared to greedy heuristics, simulated annealing heuristics and heuristics proposed by Bai (2005).

Yang and Chen (1999) simplified their comprehensive model and developed a practicable linear shelf space allocation (LSSA) model. They assumed a linear objective function and dropped product availability constraints in proposed LSSA model for which a heuristic solution approach (similar to the algorithm used for solving a knapsack problem) was subsequently proposed by Yang (2001). For this problem, Lim *et al.* (2004) proposed efficient hybrid solution strategy, five-phase Squeaky-Wheel Optimization with Local (SWOL) search, by modifying the heuristic proposed by Yang (2001) and tested it to obtain near optimal solution for small problems. The initial constructor and neighbourhood strategy used by them can be improved further and hybrid solution strategy can be simplified.

In this paper, we consider a retail shelf space allocation problem with linear profit function and develop fast and efficient heuristics using new initial constructor and neighbourhood search strategy to solve this problem. LSSA model was developed by Yang and Chen (1999) for this problem and its mathematical formulation is presented in Section 2. In Section 3, we present existing and three new heuristics consisting of two phases:

(1) construction of initial solution; and

(2) improvement using neighbourhood search.

Experimental results and their analysis are shown in Section 4. In Section 5, we use four heuristics to allocate shelf space to various products in a category for an existing retail store. The paper is concluded with major findings in Section 6.

2. Mathematical model

Yang and Chen (1999) considered location effects found by Drèze *et al.* (1994). They developed a practicable LSSA model by simplifying their comprehensive model assuming a linear objective function and dropping product availability constraint. Lim *et al.* (2004) justified the linearity assumption of profit function with the fact that retailers like to operate on linear portion of the S-shaped curve of marginal returns to space as explained in Bultez and Naert (1988) and Drèze *et al.* (1994). The LSSA model is described next.

In a retail store, there are m shelves (indexed from $k = 1$ to m) available where the retailer wish to display n products (indexed from $i = 1$ to n). Product i has face length of a_i and display requirements of minimum facings (L_i) and maximum facings (U_i). Let p_{ik} be per facing profit of product i when displayed on shelf k. The available space of shelf k is measured by its length T_k. Under the given operating conditions, decisions for the retailer are the number of facings x_{ik} to be displayed on shelf k and the total number of facings to be displayed in a store x_i for each product i. Using the above notations, mathematical formulation of the LSSA model is given as follows.

LSSA model

$$\text{Maximize total profit, } TP = \sum_{i=1}^{n} \sum_{k=1}^{m} p_{ik} x_{ik} \qquad (1)$$

subject to:

$$\sum_{i=1}^{n} a_i x_{ik} \leq T_k \;\; \forall \;\; k = 1, \ldots, m \tag{2}$$

$$L_i \leq \sum_{k=1}^{m} x_{ik} \leq U_i \;\; \forall\, i = 1, \ldots, n \tag{3}$$

$$x_{ik} \geq 0 \, \text{and integer} \quad \forall\, i = 1, \ldots, n; k = 1, \ldots, m \tag{4}$$

The objective function (1) of the model maximizes retailer's total profit. Constraints (2) enforce that total allocated space to products does not exceed the available space on a given shelf. Constraints (3) ensure that total allocated facings of products in a store are between the specified lower limit (minimum number of facings) and upper limit (maximum number of facings). Constraints (4) show that number of facings of products is always a nonnegative integer.

LSSA problem is NP-hard (Yang and Chen, 1999) and is similar to multi-knapsack problem. Therefore, there is merit in developing efficient heuristics that obtain near optimal solution for large practical problems. We show in Section 4.2 that the linear programming relaxation of LSSA model generates a good upper bound; hence, it is used in this paper to assess the quality of heuristic solution. Next section discusses existing and proposed heuristics to solve LSSA model.

3. Heuristics approaches to LSSA model

Yang (2001) developed heuristics to solve his LSSA model described in Section 2. The original heuristic consisted of preparatory, allocation and termination phases, and it was similar to the greedy algorithm used for solving knapsack problem. He modified original heuristic adding an adjustment phase with three adjustment methods. Lim *et al.* (2004) proposed a new heuristic (termed as new neighbourhood moves) where initial solution was obtained using Yang's (2001) original heuristic in first phase and it was further improved by three new neighbourhood moves in second phase. In this paper, we propose three new heuristics consist of two phases similar to Lim *et al.* (2004). Section 3.1 discuses existing and new methods of constructing initial feasible solution used in first phase. While Section 3.2 discusses existing and new neighbourhood moves used to improve upon the initial feasible solution in second phase.

3.1 Phase I – construction of initial feasible solution

We first discuss the existing method to construct initial feasible solution and then propose a new method which constructs good initial solution to facilitate neighbourhood moves to obtain near optimal solution.

3.1.1 Existing method to construct initial solution

Initial constructor IC1. Lim *et al.* (2004) used original heuristic of Yang (2001) as initial constructor in their heuristic to obtain initial feasible solution. It is similar to the algorithm used for solving knapsack problems and consists of preparatory, allocation and termination phases. The profit of each product *i* per displayed length (p_{ik}/a_i) on a shelf *k* is considered as a weight and ranking order of weight is used as priority index in the process of space allocation.

Preparatory phase checks the feasibility of a given problem and then creates a set of priority indexes. The allocation phase assigns available space to products one by one in the order of priority. Minimum facing requirements of each product is first satisfied and then additional facings of each product (as given in a set of priority index) are added till maximum facings of that product is reached or capacity of the shelf is exhausted. This phase respects all the three constraints (2)-(4) of LSSA model. Termination phase calculates the objective value of the final solution.

3.1.2 New method to construct initial solution

Initial constructor IC2. Using a commercial solver, we solve linear programming problem obtained by relaxing the integrality constraints (4) of LSSA model. As discussed in Section 2, it also gives an upper bound to LSSA model. We attempt to construct initial solution by rounding down fractional solution obtained by LP relaxation of LLSA model to integer values. It may be possible that such rounding may make the solution infeasible due to violation of minimum facings for some products. If that happens, we attempt to make the solution feasible using two steps. In first step, we add facings of products to the empty space of shelf in attempt to satisfy minimum facings condition using the priority index defined by Yang (2001). If feasible initial solution is still not obtained, then we go for the second step. In the second step, we remove "x" number of facings of product i which has more facings than the required minimum and add "y" number of facings of product j in the same shelf k which does not have the required minimum facings when incremental objective value ($y\ {}^{*}p_{jk} - x\ {}^{*}p_{ik}$) is highest for combination (i, j and k).

3.2 Phase II – improvement using neighbourhood moves

In this section, we discuss the existing neighbourhood moves proposed by Yang (2001) and Lim *et al.* (2004) in their heuristics to improve the initial solution.

3.2.1 Existing neighbourhood moves. Yang (2001) proposed adjustment phase with three adjustment methods as neighbourhood moves to improve the initial solution in his original heuristic before termination phase. First method attempts to improve a solution by swapping one facing for a pair of products allocated on the same shelf. Second method interchanges one facing for a pair of products allocated on two shelves. Third method attempts to add more facings to remaining shelf space left after implementing second method. Yang (2001) inserted adjustment phase with a combination of above three adjustment methods to improve the initial solution in his original heuristic. However, Lim *et al.* (2004) found shortcomings of above adjustment methods which do not allow swap or interchange of multiple facings for a pair of products. Hence, they modified three adjustment methods considering many-to-many neighbourhood moves which are described next.

Move M1 – multishift move. It attempts to improve solution by swapping multiple facings for a pair of products allocated on the same shelf. It repeatedly removes a number of facings of one product and adds equivalent number of facings of another product into the same shelf till such remove–add generates positive incremental profit respecting all the constraints (2)-(4).

Move M2 – multiexchange move. It repeatedly interchanges multiple facings for a pair of products allocated on two shelves till such interchanges generate positive incremental profit respecting all the constraints (2)-(4).

Move M3 – multiadd and exchange move. This move is an extension of the second move. After interchanging a number of facings between two products on two shelves, there may be enough shelf space left over to be reallocated to other products.

Thus, this move adds more facings of most profitable product on a given shelf respecting all the constraints (2)-(4).

Lim *et al.* (2004) used above neighbourhood moves in various combinations and they found that sequence(M2, M3 and M1) performed well which is referred as Neighbourhood Search-1 (NS1) in this paper.

3.2.2 New neighbourhood move

Move M4 – shift and add. We propose a better neighbourhood move compared to move 1. As shown in Figure 1, this move shifts "*x*" number of facings of a product *i* from shelf *k* to shelf *l* in order to make enough space in shelf *k* and then adds "*y*" number of facings of other product *j* to empty space of shelf k if incremental profit ($x{*}p_{il} - x{*}p_{ik} + y{*}p_{jk}$) is positive. Thus, this move repeatedly does shift and add whenever such move is profitable respecting all the constraints (2)-(4) of LSSA model.

We use the two existing moves (M2 and M3) and a new move (M4) in the sequence (M4, M2 and M3) which is referred as Neighbourhood Search-2 (NS2).

3.3 Heuristics

In this section, we first discuss the existing heuristic proposed by Lim *et al.* (2004) and then build three heuristics using various combinations of two methods of constructing initial solution and two neighbourhood searches as explained previously. Design of existing and proposed heuristics is shown in Table I.

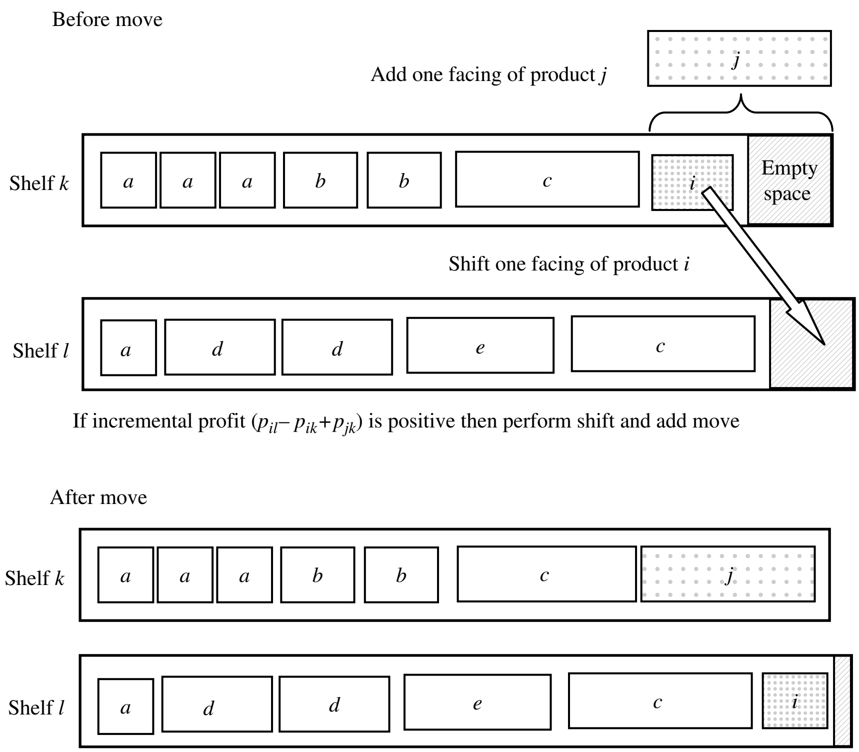

Figure 1.
An illustration of move M4 (shift and add)

3.3.1 Existing heuristic

Heuristic *H1*. Lim *et al.* (2004) modified Yang's (2001) heuristic and developed a new heuristic called "new neighbourhood moves". They used original heuristic of Yang (2001) as an initial constructor (IC1) to build initial feasible solution in first phase of their heuristic. Initial feasible solution was further improved by existing neighbourhood search (NS1) given by Lim *et al.* (2004) in second phase.

3.3.2 New heuristics

Heuristic *H2*. This heuristic, in the first phase, uses original heuristic of Yang (2001) as an initial constructor (IC1) to build initial feasible solution which is further improved in the second phase by applying new neighbourhood search (NS2).

Heuristic *H3*. This heuristic, in the first phase, uses new initial constructor (IC2) to build initial feasible solution which is further improved in the second phase by applying existing neighbourhood search (NS1).

Heuristic *H4*. This heuristic, in the first phase, uses new initial constructor (IC2) to build initial feasible solution which is further improved in the second phase by applying new neighbourhood search (NS2).

4. Experimental results and analysis

4.1 Experimental setup

A number of simulated problems are randomly generated in absence of benchmark data. We followed the data generation scheme similar to one given in Lim *et al.* (2004). As shown in Table II, four problem sizes (number of products n, number of shelves m) are considered with largest problem size (200, 50). There is no evidence in the literature that the problem of this size has been solved before. Two levels of face length a_i, minimum number of facings L_i and maximum number of facings U_i, are used for each problem size (n, m). Therefore, we have $2 \times 2 \times 2 = 8$ problems and we use ten random instances which gives 80 problem instances for each of the four problem sizes (25, 5),

Table I. Design of existing and new heuristics

Heuristics	Phase-I (initial constructors)	Phase-II (neighborhood search)
Existing		
H1 (Lim *et al.*, 2004)	IC1	NS1 (M2, M3 and M1)
New		
H2	IC1	NS2 (M4, M2 and M3)
H3	IC2	NS1 (M2, M3 and M1)
H4	IC2	NS2 (M4, M2 and M3)

Table II. Parameter values and ranges in experimental setup

Parameter	Random value	Range
(n, m)	–	(25, 5), (100, 30), (150, 40), (200, 50) [four combinations]
p_{ik}	U [0, 10]	–
a_i	U [1, A]	$A = 50, 100$ [two settings]
L_i	U [0, L]	$L = 5, 10$ [two settings]
Δ_i	U [0, Δ]	$\Delta = 25, 50$ [two settings]
U_i	$L_i + \Delta_I$	–
T_k	U [$T_L/4$, T_U]	$T_L = \Sigma_i L_i a_i / m$, $T_U = \Sigma_i (L_i + \Delta_i) a_i / m$

(100, 30), (150, 40) and (200, 50). All tests were performed on single Intel Xeon 2.8 GHz processor with 1 GB RAM and average performance was computed for 80 problem instances for each problem size (n, m).

4.2 Upper bound

A profit upper bound value (P_{ub}) is obtained by solving the linear programming relaxation of LSSA model dropping integrality constraints (4). To evaluate the quality of upper bound, optimal profit value (P_o) was also obtained for smaller problem size (25, 5) which could be solved in reasonable central processing unit (CPU) time using commercial IP solver (ILOG CPLEX 9.1®). The average value of upper bound was found to be close to optimal value (within 0.18 percent) as shown in Table III.

4.3 Comparison of initial constructors (IC1) and (IC2)

We compare the solution obtained by new method of constructing initial feasible solution (IC2) with that obtained using existing method (IC1). It can be seen from Table III that proposed method of constructing initial solution (IC2) always gives better initial feasible solution than that obtained by existing method (IC1). However, IC2 takes slightly more CPU time than IC1 to construct initial feasible solution.

4.4 Comparison of existing and new heuristics

We have used three performance measures: average profit value P_h, average CPU time and average P_h/P_{ub} in percent to compare existing and new heuristics.

It can be observed from Table III that all three new heuristics (*H2*, *H3* and *H4*) give competitive objective values and also take less CPU time than existing heuristic (*H1*) given by Lim *et al.* (2004).

H1 versus H2. *H1* and *H2* use existing initial constructor (IC1) but different neighbourhood searches NS1 and NS2, respectively. *H1* and *H2* give similar objective values but *H2* takes less CPU time than *H1* which can be attributed to new neighbourhood search (NS2).

H1 and H2 versus H3 and H4. *H3* and *H4* give better objective values and take less CPU time as compared to *H1* and *H2*. This indicates the effectiveness of new method (IC2) of constructing initial solution.

H3 versus H4. *H4* gives better objective value than that obtained using *H3*. Both heuristics use new method (IC2) of constructing initial solution but *H4* used neighbourhood search (NS2) with new neighbourhood move (M4) while *H3* used neighbourhood search (NS1) with existing neighbourhood moves. Thus, neighbourhood search (NS2) performs better than neighbourhood search (NS1).

H1 versus H4. New heuristic (*H4*) gives much better objective value and requires substantially less CPU time as compared to existing heuristic (*H1*) given by Lim *et al.* (2004). This shows that new method (IC2) of constructing initial solution and neighbourhood search (NS2) with a new neighbourhood move (M4) give superior performance compared to existing method (IC1) and neighbourhood search (NS1) with existing neighbourhood moves (M1, M2 and M3).

H4 versus meta-heuristics. Lim *et al.* (2004) developed and implemented two meta-heuristics (Tabu search and five-phase SWOL) along with heuristic *H1* (new neighbourhood moves). It was found that five-phase SWOL outperformed *H1* giving better objective value (with average P_h/P_{ub} of 99.52 per cent) but it took

Table III.
Performance of initial constructors and heuristics

				Initial constructors		Heuristics			
				Existing	New	Existing		New	
(n,m)	Performance measures	Optimal[a]	Upper bound	IC1	IC2	*H1*	*H2*	*H3*	*H4*
(25,5)	Average profit value	3,059.81	3,065.29[b]	2,973.30	3,034.18	3,012.50	3,009.45	3,046.71	3,051.28
	Average time (secs)	11.7994	0.0006	0.0041	0.0051	0.0093	0.0086	0.0075	0.0079
	Average P_h/P_{ub} (%)	–	–	97.00	98.99	98.28	98.18	99.39	99.54
	Average P_h/P_o (%)	–	–	97.17	99.16	98.45	98.35	99.57	99.72
(100,30)	Average profit value	–	14133.05	13814.77	13916.66	13901.66	13906.60	13990.18	14066.83
	Average time (secs)	–	0.01	0.15	0.18	6.85	5.53	1.69	1.76
	Average P_h/P_{ub} (%)	–	–	97.75	98.47	98.36	98.40	98.99	99.53
(150,40)	Average profit value	–	2,2025.43	2,1635.81	2,1732.03	2,1732.94	2,1745.53	2,1825.40	2,1940.88
	Average time (secs)	–	0.02	0.51	0.56	33.50	26.12	7.60	7.24
	Average P_h/P_{ub} (%)	–	–	98.23	98.67	98.67	98.73	99.09	99.62
(200,50)	Average profit value	–	2,9543.71	2,9075.32	2,9171.53	2,9182.84	2,9202.64	2,9288.33	2,9443.36
	Average time (secs)	–	0.05	1.32	1.41	119.82	86.88	23.97	22.46
	Average P_h/P_{ub} (%)	–	–	98.41	98.74	98.78	98.85	99.14	99.66

Notes: [a]Optimal solution was obtained for small problem instances due to time constraints; [b]average percentage gap of upper bound (P_{ub}) from optimal (P_o) is 0.18 for problem size (25, 5) substantially higher CPU time compared to *H1*. In our experiment, we found that our new heuristic *H4* gives average P_h/P_{ub} of 99.59 per cent which is in the same order as obtained by five-phase SWOL. But *H4* takes substantially less CPU time compared to *H1* implying that it is computationally much more efficient than five-phase SWOL.

5. Case study of ABC's retail

ABC's retail is the large supermarket chain in India with 400 stores across 65 cities covering a retail trading area of two million square feet and an astonishing 3.5 million customers a month. XYZ large store of ABC's retail has been chosen for the analysis because XYZ is the largest format of ABC's retail catering every needs of the customers and is located at well-known hyper city in a metro city in India. XYZ stocks 35,000 articles (SKUs) which are grouped into seven segments.

We implement LSSA model and four heuristics, *H1-H4*, for shelf space allocation to "bath soaps" category. There are 117 products in this category that are currently located on four identical racks, each having six shelves with length of 120 cm. Hence, 24 shelves each having a capacity of 120 cm is available to display the products under "bath soaps" category. The current allocation of shelf space to 117 products is available for the month of April from ABC's retail. The face length of each product is provided by the store. Sales data of each product is collected from the store for last six months. The lower and upper limit of number of facings for each product was calculated using minimum sales and maximum sales during last six months. The profit per facing of a product for different shelves was calculated using the average sales, current allocation, unit profit margin and location effect judged by the store personnel.

We obtained allocation using the proposed four heuristics and results are shown in Table IV. It can be seen that the solution obtained by heuristics (*H1*, *H2*, *H3* and *H4*) resulted in higher profits (71.7, 71.6, 71.2 and 70.8 per cent, respectively) than the monthly profit using the current allocation scheme for "bath soaps" category. *H4* has resulted in slightly inferior objective function value than *H1* in this case instance; however, computation time of *H4* is still much lower than *H1*.

6. Conclusion

In this paper, we proposed three heuristics to solve LSSA model given by Yang and Chen (1999). These heuristics consist of construction of initial solution and improvement using neighbourhood searches. Using fractional solution of linear programming relaxation of LSSA model, we developed a new method of constructing

				Initial constructors		Heuristics			
				Existing	New	Existing	New		
(n,m)	Performance measures	Current	Upper bound	IC1	IC2	*H1*	*H2*	*H3*	*H4*
	Average profit value	31,170	53,778	4,8,524	52,159	53,510	53,502	53,372	53,251
(117,24)	Average time (secs)	–	0.01	0.15	0.15	18.80	18.82	5.53	3.81
	Average P_h/P_{ub} (%)	–	–	90.23	96.99	99.50	99.49	99.24	99.02
	Percent improvement in profit from current	–	–	55.7	67.3	71.7	71.6	71.2	70.8

Table IV. Performance of initial constructors and heuristics for the case study initial solution (IC2) which performs better than existing method (IC1) adopted from Yang (2001). New neighbourhood search NS2 using a proposed neighbourhood move (M4) also gives better improvement in initial solution. Further, it was found from empirical results that three proposed heuristics (*H2*, *H3* and *H4*) outperform existing heuristic (*H1*) developed by Lim *et al.* (2004). The new heuristic (*H4*) produced best results amongst the four heuristics with average P_h/P_{ub} of 99.59 percent in reasonable on CPU time. This indicates that the new methods of constructing initial solution (IC2) and neighbourhood search (NS2) using proposed move (M4) perform well both on objective value and CPU time. The application of three heuristics to a case of retail store found that use of these heuristics has potential to increase the earnings of the retailers.

References

Amrouche, N. and Zaccour, G. (2007), "Shelf-space allocation of national and private brands", *European Journal of Operational Research*, Vol. 180 No. 2, pp. 648-63.

Bai, R. (2005), "An investigation of novel approaches for optimizing retail shelf-space allocation", unpublished PhD thesis, University of Nottingham, Nottingham.

Bai, R. and Kendall, G. (2008), "A model for fresh produce shelf-space allocation and inventory management with freshness-condition-dependent demand", *INFORMS Journal on Computing*, Vol. 20 No. 1, pp. 78-85.

Bultez, A. and Naert, P. (1988), "SH.A.R.P.: shelf allocation for retailers' profit", *Marketing Science*, Vol. 7 No. 3, pp. 211-31.

Chen, M.-C. and Lin, C.-P. (2007), "A data mining approach to product assortment and shelf space allocation", *Expert Systems with Applications*, Vol. 32 No. 4, pp. 976-86.

Corstjens, M. and Doyle, P. (1981), "A model for optimizing retail space allocations", *Management Science*, Vol. 27 No. 7, pp. 822-33.

Corstjens, M. and Doyle, P. (1983), "A dynamic model for strategically allocating retail space", *The Journal of the Operational Research Society*, Vol. 34 No. 10, pp. 943-51.

Curhan, R.C. (1972), "The relationship between shelf space and unit sales in supermarkets", *Journal of Marketing Research*, Vol. 9 No. 4, pp. 406-12.

Drèze, X., Hoch, S.J. and Purk, M.E. (1994), "Shelf management and space elasticity", *Journal of Retailing*, Vol. 70 No. 4, pp. 301-26.

Gajjar, H.K. and Adil, G.K. (2010), "A piecewise linearization for retail shelf-space allocation problem and a local search heuristic", *Annals of Operations Research*, Vol. 179 No. 1, pp. 149-67.

Gochet, W. and Smeers, Y. (1979), "Reversed geometric programming: a branch-and-bound method involving linear sub problems", *Operations Research*, Vol. 27 No. 5, pp. 982-96.

Hariga, M.A., Al-Ahmari, A. and Mohamed, A.A. (2007), "A joint optimisation model for inventory replenishment, product assortment, shelf space and display area allocation decisions", *European Journal of Operational Research*, Vol. 181 No. 1, pp. 239-51.

Hwang, H., Choi, B. and Lee, M. (2005), "A model for shelf space allocation and inventory control considering location and inventory level effects on demand", *International Journal of Production Economics*, Vol. 97 No. 2, pp. 185-95.

Lim, A., Rodrigues, B. and Zhang, X. (2004), "Metaheuristics with local search techniques for retail shelf-space optimization", *Management Science*, Vol. 50 No. 1, pp. 117-31.

Reyes, P.M. and Frazier, G.V. (2007), "Goal programming model for grocery shelf space allocation", *European Journal of Operational Research*, Vol. 181 No. 2, pp. 634-44.

Urban, T.L. (1998), "An inventory-theoretic approach to product assortment and shelf-space allocation", *Journal of Retailing*, Vol. 74 No. 1, pp. 15-35.

Urban, T.L. (2002), "The interdependence of inventory management and retail shelf management", *International Journal of Physical Distribution & Logistics Management*, Vol. 32 No. 1, pp. 41-58.

Yang, M. (2001), "An efficient algorithm to allocate shelf space", *European Journal of Operational Research*, Vol. 131 No. 1, pp. 107-18.

Yang, M. and Chen, W. (1999), "A study on shelf space allocation and management", *International Journal of Production Economics*, Vol. 60-61, pp. 309-17.

Zufryden, F.S. (1986), "A dynamic programming approach for product selection and supermarket shelf-space allocation", *The Journal of the Operational Research Society*, Vol. 37 No. 4, pp. 413-22.

About the authors

Hasmukh K. Gajjar is a Visiting Assistant Professor at the Indian Institute of Management Indore, India. He holds a PhD from the Shailesh J. Mehta School of Management, Indian Institute of Technology Bombay, India. He is a Graduate in Mechanical Engineering and MBA in Operations Management. He has six years of work experience in the field of Operations Management including two years of industry experience. His research work on retail shelf space allocation has been already accepted in the leading journals like *Annals of Operations Research* and *Asia-pacific Journal of Operational Research*. His major areas of interest are retail operations, modelling and optimization.

Gajendra K. Adil is a Professor at the Shailesh J. Mehta School of Management, Indian Institute of Technology, Bombay, India. He holds PhD (1994) in Industrial Engineering from the University of Manitoba, MSc (1990) in Industrial Engineering from the University of Regina, M.Tech (1987) in Mechanical Engineering from IIT-Kanpur and B.E (1985) in Mechanical Engineering from Ravishankar University, Raipur. He has worked as an Assistant Professor at the Bilkent University in Turkey and the City University of Hong Kong. He has also worked as a technology transfer consultant at i2 Technologies (Dallas, USA) and as an industrial engineer at Bristol Aerospace Ltd (Winnipeg, Canada). His publications have appeared in leading journals such as *Operations Research*, *IIE Transactions*, *Annals of Operations Research*, *European Journal of Operational Research*, *International Transaction in Operational Research*, *International Journal of Production Research* and *Omega and Journal of Manufacturing Systems*. His fields of interest include operations management and decision sciences. Gajendra K. Adil is the corresponding author and can be contacted at: adil@iitb.ac.in

This article has been cited by:

1. Tae Wan Kim, Ram Rajagopal, Martin Fischer, Calvin Kam. 2013. A knowledge-based framework for automated space-use analysis. *Automation in Construction* **32**, 165-176. [CrossRef]
2. Alexander H. HübnerBusiness Department, Operations Management, Catholic University Eichstätt-Ingolstadt, Ingolstadt, Germany Heinrich KuhnBusiness Department, Supply Chain Management and Operations, Catholic University Eichstätt-Ingolstadt, Ingolstadt, Germany Michael G. SternbeckBusiness Department, Supply Chain Management and Operations, Catholic University Eichstätt-Ingolstadt, Ingolstadt, Germany. 2013. Demand and supply chain planning in grocery retail: an operations planning framework. *International Journal of Retail & Distribution Management* **41**:7, 512-530. [Abstract] [Full Text] [PDF]
3. Alexander H. Hübner, Heinrich Kuhn. 2012. Retail category management: State-of-the-art review of quantitative research and software applications in assortment and shelf space management. *Omega* **40**:2, 199-209. [CrossRef]
4. Tuncay Ozcan, Şakir EsnafSwarm Intelligence Approaches to Shelf Space Allocation Problem with Linear Profit Function 22-41. [CrossRef]