4OR-Q J Oper Res (2014) 12:35–48
DOI 10.1007/s10288-013-0230-7

RESEARCH PAPER

Internet shopping with price sensitive discounts

Jacek Blazewicz · Pascal Bouvry ·
Mikhail Y. Kovalyov · Jedrzej Musial

Received: 27 December 2011 / Revised: 17 January 2013 / Published online: 17 March 2013

Abstract A customer would like to buy a given set of products in a given set of Internet shops. For each Internet shop, standard prices for the products are known as well as a concave increasing discounting function of total standard and delivery price. The problem is to buy all the required products at the minimum total discounted price. Computational complexity of various special cases is established. Properties of optimal solutions are proved and polynomial time and exponential time solution algorithms based on these properties are designed. Two heuristic algorithms are suggested and computationally tested.

Keywords Internet shopping · Computational complexity · Optimization · Algorithms

Mathematics Subject Classification (2000) 90C59 · 90C27 · 68T20 · 68Q25

Electronic supplementary material The online version of this article (doi:10.1007/s10288-013-0230-7) contains supplementary material, which is available to authorized users.

J. Blazewicz · J. Musial (✉)
Institute of Computing Science, Poznan University of Technology,
ul. Piotrowo 2, 60-965 Poznan, Poland
e-mail: Jedrzej.Musial@cs.put.poznan.pl

P. Bouvry · J. Musial
Computer Science and Communications Research Unit, University of Luxembourg,
6 Rue Coudenhove-Kalergi, 1359 Luxembourg, Luxembourg

M. Y. Kovalyov
United Institute of Informatics Problems, National Academy of Sciences of Belarus,
Surganova 6, 220012 Minsk, Belarus

1 Introduction

Report by Foundation (2008) states that 32 % of EU customers make purchases in Internet stores and that the prospects for the e-commerce sector look very healthy over the next few years. Forecast predict 70 % growth in terms of number of people shopping online world-wide.

Price sensitive discounts are often used in Internet shops to attract customers. A typical example is the following advertisement in an Internet shop selling computer parts: ”If the value of your purchase is at least 100 euros, then your discount is 2; 200–5; 300–10, 500–15; 1,000–20 %”. Many Internet shops offer free delivery if the price of a purchase exceeds a certain threshold.

In this paper, we study an optimization aspect of Internet shopping (Blazewicz et al. 2010) with price sensitive discounts from customer perspective. Specifically, we consider a problem in which a customer would like to buy products of a given set $N = \{1, \dots, n\}$ in a given set of Internet shops $M = \{1, \dots, m\}$ at the minimum total final price. There are the following given parameters and decision variables:

d_i—delivery price of all products from shop i to the customer;
p_{ij}—standard price of product j in shop i, $p_{ij} = p_j$ if standard prices of product j are the same in all shops;
N_i—subset of products of the set N in shop i (eligible products for shop i), $N_i \subseteq N$;
M_j—subset of shops in which product j can be bought (eligible shops for product j), $M_j \subseteq M$;
S_i—subset of products selected by the customer in shop i (basket of shop i, decision variable), $N = \cup_{i=1}^{m} S_i$ and $S_i \cap S_j = \phi$, $i \neq j$, for a feasible solution;
$T_i(S_i) = d_i + \sum_{j \in S_i} p_{ij}$ - total delivery and standard price in shop i for a given set of products $S_i \subseteq N_i$; if there is no ambiguity, notation S_i in $T_i(S_i)$ can be omitted;
$f_i(T)$—discounting function for final price, a concave increasing differentiable or concave piecewise linear function of total delivery and standard price T in shop i at all points $T > 0$, $f_i(0) = 0$.

We denote the above problem as IS, where the abbreviation stands for Internet Shopping. Its mathematical program can be written as follows:

$$\begin{aligned}
&\min \sum_{i=1}^{m} f_i\left(d_i y_i + \sum_{j \in N_i} p_{ij} x_{ij}\right),\\
&\text{s.t.} \sum_{i \in M_j} x_{ij} = 1,\ j = 1, \dots, n,\\
&0 \le x_{ij} \le y_i,\ i = 1, \dots, m,\ j = 1, \dots, n,\\
&x_{ij} \in \{0, 1\},\ y_i \in \{0, 1\},\ i = 1, \dots, m,\ j = 1, \dots, n.
\end{aligned}$$

The rest of the paper is organized as follows. We classify subproblems of problem IS according to their most important characteristics. Computational complexity of every subproblem is established either by proving its NP-hardness or by providing a polynomial time solution algorithm. On top of our classification, there are two classes: problems, in which not every shop may have all the required products, and those, in which every shop has all the required products. These two classes are studied Sects. 2 and 3, respectively. Clearly, the first class represents our original problem IS. Problems of both classes are NP-hard in the strong sense. Subclasses are distinguished on the basis of whether or not the same product has the same standard price in each shop, and whether or not shops actually offer discounts. Section 5 describes two heuristics for the general problem IS and the results of computer experiments with them. Conclusions and suggestions for future research are given in Sect. 6.

2 Each shop may not have all products

In this section, we study the general case of problem IS, in which $N_i \neq N$ can be satisfied for $i \in \{1, \ldots, m\}$. We first prove that a special case of problem IS is NP-hard in the strong sense.

Theorem 1 *Problem* IS *is NP-hard in the strong sense if* $N_i \neq N$ *for* $i \in \{1, \ldots, m\}$, $p_{ij} = 1$, $j = 1, \ldots, n$, $d_i = 1$, $f_i(T) = T$, $i = 1, \ldots, m$, *each product appears in at most three shops and each shop has exactly three products.*

Proof A decision version of the above mentioned special case, which asks for the existence of a solution with value not exceeding $4n/3$, can be re-formulated as follows: Given a family $C = \{C_1, \ldots, C_m\}$ of 3-element subsets (shops) of the set $N = \{1, \ldots, n\}$ (products to be bought), where $n = 3k$, does C contain an exact cover of N, i.e., a subfamily $Y \subseteq C$ (selection of shops) such that each $j \in N$ belongs to exactly one 3-element set in Y?

This re-formulation coincides with the formulation of the problem Exact Cover by 3-sets by Garey and Johnson (1979), which is NP-complete in the strong sense even if each element $j \in N$ appears in at most three subsets C_i. □

Note that if each product appears in at most two shops, then the special case in Theorem 1 is polynomially solvable, see Garey and Johnson (1979).

Problem IS can be solved by the following enumeration approach. Construct all n-dimensional sequences $(i_1, \ldots, i_n)$ of shop indices, where $i_j \in M_j$ is an eligible shop in which product j can be bought. There exists a one-to-one correspondence between these sequences and feasible solutions of problem IS. The number of these sequences does not exceed m^n. In $O(n)$ time calculate solution value (final total price) for every sequence and choose the best as an optimal solution.

The time complexity of this approach is $O(nm^n)$. It is polynomial if the number of products n is a given constant, and it can be acceptable for real-life computations if there are about ten products and about ten shops.

2.1 Same standard prices

Let us consider a subproblem of problem IS, in which $p_{ij} = p_j$, $j = 1, \ldots, n$, $i = 1, \ldots, m$. We denote it as IS- Same- Prices. This problem has the following useful property.

Theorem 2 *There exists an optimal solution of problem* IS- SAME- PRICES, *in which all the products are bought in shops $M_{i_1}, M_{i_2}, \ldots, M_{i_h}$ such that in shop M_{i_1} all its eligible products of the set N_{i_1} are bought, in shop M_{i_2} all its eligible products except for those selected in the shop M_{i_1} are bought, and so on.*

Proof Consider an optimal solution. Assume without loss of generality that the products are bought in the shops $1, \ldots, k$, $k \geq 2$, and that corresponding total delivery and standard prices are $T_1, \ldots, T_k$. Let shop i^0 have minimum derivative value $f'_i(T_i)$ among $i = 1, \ldots, k$ (in the case of concave piecewise linear functions f_i, minimum slope of the line segment $[T_i, T_i + \varepsilon]$ for an arbitrarily small $\varepsilon > 0$). If products of the set N_{i^0} are selected not only in the shop i^0 but also in other shops, then move them to the basket of shop i^0. Due to the concavity of functions f_i, relation $f_{i^0}(T_{i^0}+\delta) - f_{i^0}(T_{i^0}) \leq f_i(T_i) - f_i(T_i - \delta)$ is satisfied for any $\delta > 0$ and $i = 1, \ldots, k$. Therefore, the product move does not increase total final price. We deduce that solution with $S_{i^0} = N_{i^0}$ is optimal. Repetition of this argument completes the proof. A small difference of the next iterations of this argument is that products of shops selected in the previous iterations are not moved. Corresponding derivatives or slopes of functions f_i for these shops are not greater than those for not selected shops because of the concavity assumption. □

Theorem 2 is a basis of the following solution approach for problem IS- SAME- PRICES. Construct all shop sequences $(i_1, \ldots, i_m)$ and corresponding solutions $(S_{i_1}, \ldots, S_{i_k})$, $k \leq m$, such that $S_{i_1} = N_{i_1}$, $S_{i_r} = N_{i_r} \backslash (\cup_{l=1}^{r-1} S_{i_l})$, $r = 1, \ldots, k$, and $N = \cup_{l=1}^{k} S_{i_l}$. The number of these sequences does not exceed $m!$. In $O(n)$ time calculate solution value (final total price) for every sequence and choose solution corresponding to the best sequence as an optimal solution.

The time complexity of this approach is $O(nm!)$, which is better than the previous $O(nm^n)$ time solution approach. This approach is linear in n, and it can be acceptable for real-life computations if there are thousands of products and tens of shops. It can be used as a heuristic for a more general case of problem IS, in which prices of the same product are slightly different in different shops. It can also be adapted to solve the following realistic special case of problem IS- SAME- PRICES in a polynomial time.

Problem IS- SAME- PRICES- SUBSETS: $p_{ij} = p_j$, $j = 1, \ldots, n$, $i = 1, \ldots, m$, and N is partitioned into subsets $N^{(0)}$, $N^{(1)}$ and $N^{(2)}$ such that all products of the set $N^{(0)}$ are present in each shop, each product of the set $N^{(1)}$ is present in one shop, and $|N^{(2)}| \leq c$, where c is a given constant.

Theorem 3 *There exists an optimal solution of problem* IS- SAME- PRICES- SUBSETS, *in which all products of the set $N^{(0)}$ are bought in one shop.*

Proof The proof is the same as that of Theorem 2. □

Theorem 3 and an observation that the shop to buy a product of the set $N^{(1)}$ is determined uniquely justify the following solution algorithm for problem IS- SAME- PRICES- SUBSETS. Consider candidate shops $l = 1, \ldots, m$ to buy all products of the set $N^{(0)}$ in one shop. For each l, perform the following computations. Assign products of the set $N^{(0)}$ to shop l. Assign each product $j \in N^{(1)}$ to its unique eligible shop.

Let $N^{(2)} = \{1, \ldots, c\}$. Enumerate all feasible assignments of products $1, \ldots, c$ to the shops, which is equivalent to enumerating shop sequences $(i_1, i_2, \ldots, i_c)$ such that product j is assigned to shop $i_j \in M_j$, $|M_j| \leq m$, $j = 1, \ldots, c$. The number of these sequences does not exceed m^c. For each l and the above mentioned shop sequence $(i_1, i_2, \ldots, i_c)$, calculate the value of the corresponding solution. Select a solution with the minimum value as an optimal solution. The run time of this algorithm is $O(nm^{c+1})$.

2.2 No discounts

In this subsection we consider a subproblem of problem IS, in which $f_i(T) = T$, $i = 1, \ldots, m$. We denote it as IS- No- Discounts. This problem is equivalent to the well known Facility Location Problem (FLP). Main characteristics of the FLP are a space, a metric, given customer locations and given or not positions for facility locations. A traditional FLP is to open a number of facilities in arbitrary positions of the space (continuous problem) or in a subset of the given positions (discrete problem) and to assign customers to the opened facilities such that the sum of opening costs and costs related to the distances between customer locations and their corresponding facility locations is minimized. The equivalence of the traditional discrete FLP and problem IS- No- Discounts is easy to see if customers are treated as products, positions for facility locations as shops, opening costs as delivery prices, and cost related to the distance between position i and customer j as the price p_{ij} of product j in shop i. Note, however, that the general problem IS with price sensitive discounts cannot be treated as a traditional discrete FLP because there is no evident motivation for a discount on the cumulative cost in the sense of distances.

Discussions of FLPs can be found in Eiselt and Sandblom (2004), Iyigun and Ben-Israel (2010), Krarup et al. (2002), Melo et al. (2009), Revelle et al. (2008), Stratila (2009). The traditional discrete FLP is NP-hard in the strong sense, so is the problem IS- No- Discounts. Solution approaches for discrete FLPs include Integer Programming formulations, Lagrangian relaxations, Linear Programming relaxations, Genetic Algorithms, graph-theoretical approaches, and Tabu Search, whose descriptions can by found in Revelle et al. (2008) and Stratila (2009). All these approaches can be used to solve problem IS- No- Discounts.

Stratila (2009) studied a *concave cost facility location problem*, whose mathematical program can be written as

$$\begin{aligned}
&\min \sum_{i=1}^{m} \left(\phi_i \left(\sum_{j=1}^{n} d_j x_{ij} \right) + \sum_{j=1}^{n} p_{ij} x_{ij} \right), \\
&\text{s.t.} \sum_{i=1}^{m} x_{ij} = 1, \ j = 1, \ldots, n, \\
&x_{ij} \in \{0, 1\}, \ i = 1, \ldots, m, \ j = 1, \ldots, n,
\end{aligned}$$

where ϕ_i are concave nondecreasing functions. It can be noticed that this problem and problem IS are not sub-cases of one another, while the traditional discrete FLP is a special case of any of these problems.

3 Each shop has all products

In this section, we study a subproblem of problem IS, in which $N_i = N$, $i = 1, \dots, m$. Note that this assumption does not mean that every shop offers the same assortment. It means that every shop has all products of the set N required by the customer, which can happen in many real-life situations. This subproblem is NP-hard in the strong sense as demonstrated in the following theorem.

Theorem 4 *Problem* IS *is NP-hard in the strong sense if* $N_i = N$, $p_{ij} \in \{1, 2\}$, $d_i = 1$, $f_i(T) = T$, *each product* j *has at most three values* $p_{ij} = 1$ *and each shop* i *has exactly three products with value* $p_{ij} = 1, i = 1, \dots, m, j = 1, \dots, n$.

Proof Consider the following problem: Is there a solution for the special case in the theorem statement with value at most $4n/3$? The decision version of problem IS in the proof of Theorem 1 evidently reduces to it by choosing $p_{ij} = 1$ if product j is present in shop i and $p_{ij} = 2$, otherwise. □

3.1 Same standard prices

Let us consider a subproblem of problem IS, in which $N_i = N$ and $p_{ij} = p_j$, $j = 1, \dots, n$, $i = 1, \dots, m$. We denote it as IS- Same- Products- And- Prices. In this subproblem, total standard price of all products does not depend on the solution. Denote it by $P = \sum_{j=1}^{n} p_j$.

Problem IS- Same- Products- And- Prices possesses the following property.

Theorem 5 *There exists an optimal solution of problem* IS- Same- Products- And- Prices, *in which all the products are bought in one shop.*

Proof The proof is the same as that of Theorem 2. The only difference is that, since $N_i = N$ for $i = 1, \dots, m$, for an optimal solution with $k \geq 2$ shops, all the products can be moved to the shop i^0 with minimum derivative value $f_i'(T_i)$ among $i = 1, \dots, k$ (in the case of concave piecewise linear functions f_i, minimum slope of the line segment $[T_i, T_i + \varepsilon]$ for an arbitrarily small $\varepsilon > 0$). □

Problem IS- Same- Products- And- Prices can be solved as follows. Determine shop $i^0 = \arg\min\{f_i(d_i + P) \mid i = 1, \dots, m\}$. Construct optimal solution, in which all products are selected in shop i^0. The time complexity of this approach is $O(n+m)$.

3.2 Proportional standard prices, piecewise linear discounts

In this subsection we assume that $N_i = N$, discounting functions are concave piecewise linear: $f_i(T) = a_i^{(r)} + b_i^{(r)} T$ for $T \in (T_{r-1}, T_r]$, $r \in \{1, \dots, k\}$, $0 = T_0 < T_1 < T_2 < \cdots < T_k$, and standard prices in different shops are proportional: $p_{ij} = h_i p_j$. Here $a_i^{(r)} \geq 0$, $b_i^{(r)} > 0$ and $h_i > 0$, $r = 1, \dots, k$, $i = 1, \dots, m$. This structure of standard prices often happens in reality, when shops with well established reputation apply extra charges to the product price comparing to that in shops with ordinary reputation. We denote corresponding subproblem as IS- SAME- PRODUCTS- LINEAR- CHARGES. For this subproblem, final total price of the products selected in shop i can be calculated as follows:

$$f_i\left(T_i(S_i)\right) = a_i^{(r)} + b_i^{(r)} \left(d_i + h_i \sum_{j \in S_i} p_j \right) = a_i^{(r)} + b_i^{(r)} h_i \left(d_i / h_i + \sum_{j \in S_i} p_j \right).$$

This formula shows that problem IS- SAME- PRODUCTS- LINEAR- CHARGES reduces to the problem IS- SAME- PRODUCTS- AND- PRICES with convex piecewise linear discounting functions $g_i(T) = a_i^{(r)} + b_i^{(r)} h_i T$ for $T \in (\frac{T_{r-1}}{h_i}, \frac{T_r}{h_i}]$, $r \in \{1, \dots, k\}$, and delivery prices d_i / h_i. Therefore, it can be solved in $O(n+m)$ time by the approach in Sect. 3.1.

Since Location Problems are well known for dozens of years we analyze literature and provide a descripiton on some papers that could be relevant to the IS topic. First of all, it is worth noticing that a publication by Soland (1974) describes Facility Location Problem with concave costs of the transport of goods. However, the two problems differ significantly. Main formulation of the Soland's problem is as follows: There is a company which wants to sell a number of units r_j of a given (one) product. Assumption is that the period is one year. Clients are the cities j, $j = 1, \dots, n$. All production in the ammount of r units is distributed from the plants i, $i = 1, \dots, m$. x_{ij} is the annual amount to be shipped from plant i to the city j, and $y_i = \sum_{j=1}^{n} x_{ij}$ is the annual production in plant i. Let a_i, $a_i = 1, \dots, m$ be the maximum capacity of plant i and $t_{ij}(x_{ij})$ be the total discounted cost over the T-year period of shipping $x_i j$. Mathematical programming formulation can be written as

$$\begin{aligned}
&\min \sum_{i=1}^{m} f_i(y_i) + \sum_{i=1}^{m} \sum_{j=1}^{n} t_{ij}(x_{ij}), \\
&\text{s.t.} \sum_{i=1}^{m} x_{ij} = r_j, \\
&\sum_{j=1}^{n} x_{ij} = y_i \le a_i, \\
&x_{ij} \ge 0,\, y_i \ge 0,\; i = 1, \dots, m,\; j = 1, \dots, n.
\end{aligned}$$

Nonetheless, even if this problem contains discounting function, it can be noticed that this problem and problem IS are not sub-cases of one another.

Another interesting, somehow similar to IS, is Facility location problem with staircase costs. As an example we can define a staircase cost function as a finite piecewise linear increasing function with a finite set of discontinuities, each corresponding to a certain size of a facility. A formulation of the problem (see Holmberg (1994) for the details) is a linear MIP-problem formulation with $mn + mq$ continuous variables and mq integer variables. In the paper by Holmberg (1994) authors investigated solution methods based on convex piecewise linearization and Benders decomposition.

Using convex piecewise linearization, only the integer variables that are needed to improve the approximation are introduced, and computational tests show that in average only a few are needed. Authors show that Benders decomposition can be used to solve the resulting problem. Moreover, Holmberg and Ling (1997) developed and compared heuristic solution methods for the capacitated facility location problem with staircase shaped production cost functions, a linear mixed integer programming problem with a large proportion of integer variables. In the same paper authors (Holmberg and Ling) proposed a Lagrangean heuristic, including Lagrangean relaxation and subgradient optimization as a base for an efficient primal heuristic. They showed also how to use convex piecewise linearization of the staircase shaped cost functions to get good initial upper and lower bounds as well as initial dual solutions. Based on the solution of the Lagrangean relaxation, a transportation problem which yields primal feasible solutions, was constructed.

However, it can be seen that staircase costs are somewhat different than pricewise sensitive discounts described within the problem IS, so in our case different approaches are needed.

4 Branch and Bound exact algorithm

To calculate the optimal solution of the instances evaluated, the Branch and Bound Algorithm Land and Doig (1960) was implemented, as shown in Algorithms 1 and 2.

Algorithm 1 starts by calculating an upper bound (UB) employing the solution given by the G1 greedy algorithm (Sect. 5), and proceeds to branch the first level of the search tree in the stack, through Algorithm 2.

Subsequently, the algorithm pops the top element of the stack and evaluates the objective value it would have if it were part of the current solution. If the partial solution exceeds the limit given by the upper bound, UB, the current branch is fathomed, otherwise, if it was not a leaf of the tree, the algorithm piles up the following elements within the stack, but if it were, it would mean that the current solution is better than the best global solution found so far, so it would update the upper bound with the new value found.

This process continues as long as there are elements in the stack, which means that the whole search tree has been explored, the upper bound founded is now considered as the optimal solution of the instance being evaluated.

5 Heuristics

In our first heuristic algorithm for problem IS, denoted as G1, products are considered in a certain order. The algorithm is run for various product orders and the best solution found is presented to the customer. Let the products be ordered $1, \dots, n$. Values of the total delivery and standard price for all shops are initially set as $T_i = d_i, i = 1, \dots, m$. In iteration j of algorithm G1, product j is selected in its eligible shop $i \in M_j$ with minimum value $f_i(T_i + p_{ij})$, and the corresponding T_i-value is re-set: $T_i := T_i + p_{ij}$.

Algorithm 1 *BranchandBound*

```
Require: $Instance\ instance$
Require: $Solution\ solution$

$UB = cellPILS(instance, solution)$
$branch(instance, stack, 0)$
$lastProduct = -1$
repeat
    $element = stack.pop()$
    if $lastProduct \geq element.product$ then
        for $i = element.product$ to $lastProduct$ do
            $unassignStore(instance, solution, i)$
        end for
    end if
    $assignStore(instance, solution, element.product, element.store)$
    if $element.product < instance.products$ AND $solution.total < UB.total$ then
        if $element.product == instance.products - 1$ then
            $UB = solution$
        else
            $branch(instance, stack, element.product + 1)$
        end if
    end if
    $lastProduct = element.product$
until $stack \neq \emptyset$ return $UB$
```

Algorithm 2 *branch*

```
Require: $Instance\ instance$
Require: $Stack\ stack$
Require: $int\ product$

for $store = instance.stores - 1$ to 0 do
    $stack.push(store, product)$
end forreturn $stack$
```

Table 1 Price structure for poor performance of algorithm G1

	Prod 1	Prod 2	...	Prod n	Delivery
Shop 1	$W - 1$	$W - 1$	...	$W - 1$	0
Shop 2	0	0	...	0	W

We observed that algorithm G1 demonstrates very good performance on the experimental data. However, it can provide a solution, whose value is n times worse than the optimum. Consider product and delivery prices in Table 1.

For any product sequence, algorithm G1 selects all the products in shop 1, which costs $nW - n$, while an optimal solution is to select all the products in shop 2, which costs W.

In order to hedge against the instances similar to that in Table 1, we developed another heuristic algorithm, denoted as G2. In the first step of this algorithm, each set N_i of eligible products for shop i is evaluated by the value

$$V_i = \left(d_i + \sum_{j \in N_i} p_{ij} \right) / |N_i|, \quad i = 1, \ldots, n, \tag{1}$$

and the cheapest $\alpha\%$ of products of the set N_i with the minimal value V_i are selected in this shop. In every following step, sets N_i are updated by removing selected products, and values V_i are calculated according to (1) if no product is selected in shop i, and $V_i := \sum_{j \in N_i} p_{ij}/|N_i|$ if at least one product is selected in shop i. In the latter case, set N_i contains only unselected products.

We performed computer experiments, in which the best solutions obtained by algorithms G1 and G2 were compared against optimal solutions and those provided by algorithms of *Price Comparison Sites* (Chu et al. 2005; Hof 2003) for the examples of problem IS, which are prepared on the basis of data from the online book industry reported by Clay et al. (2001). In these examples, $m \in \{10, 20, 30\}$, $n \in \{2, 3, 4, \ldots, 15\}$, and discounting functions

$$f_i(P) = \begin{cases} P, & \text{if } 0 < P \leq 50, \\ 50 + 0.95(P - 50), & \text{if } 50 < P \leq 100, \\ 50 + 0.95 * 50 + 0.9(P - 100), & \text{if } 100 < P \leq 150, \\ 50 + 0.95 * 50 + 0.9 * 100 + 0.85(P - 150), & \text{if } P > 150. \end{cases}$$

where P is the total standard price of books selected in bookstore i. It is assumed that each bookstore has all the required books. For each pair (n, m), 10 instances were generated. In each instance, the following values were randomly generated for all i and j in the corresponding ranges. Delivery price: $d_i \in \{5, 10, 15, 20, 25, 30\}$, *publisher's recommended price* of book $j : r_j \in \{5, 10, 15, 20, 25\}$, and price of book j in bookstore $i : p_{ij} \in [a_{ij}, b_{ij}]$, where $a_{ij} \geq 0.69 r_j, b_{ij} \leq 1.47 r_j$, and the structure of intervals $[a_{ij}, b_{ij}]$ follow information in Table V in Clay et al. (2001). For each instance, algorithm G1 was run two times—for a sequence of books in the non-decreasing order of the recommended price and for the reverse sequence, and algorithm G2 was run four times—for $\alpha = 25, 50, 75, 100$. Table 2 contains the average solution values for the 50 instances, which are obtained by algorithms G1 and G2, an optimal algorithm B&B and algorithms of Price Comparison Sites.

One can see that the solution found by algorithm G1 was slightly worse than optimal solution for a lower number of products. The difference between G1 and B&B exact algorithm rises with the increasing number of products (see Table 3—from 101.51 % for 2 products among 10 shops, up to 110.55 % for 15 products from 30 stores). All values [for every pair (n, m)] are represented as average values from 50 independent tests.

Algorithm G2 computes a little bit better (than G1 results for a low number od products (up to 5–8). For a bigger number of products it computes worse results than algorithm G1. Results of computations using Price Comparison Sites algorithm PCS are incomparably worse than G1 and G2 algorithms. An upgraded version of PCS algorithm which takes into account delivery prices is much better than raw PCS algorithm; however, it is significantly inferior to the two proposed heuristic algorithms.

Table 2 Average objective values in the experiment

Products	Shops	G1	G2	PCS	PCS+	B&B
2	10	30.99	30.77	53.69	34.22	30.53
3	10	48.34	48.17	84.17	53.80	47.60
4	10	57.97	57.93	103.03	64.11	56.54
5	10	68.07	68.10	119.85	74.63	66.22
6	10	83.54	82.87	146.46	90.22	80.41
7	10	99.30	98.01	169.08	107.02	94.57
8	10	104.91	105.11	176.85	110.66	98.47
9	10	118.80	119.56	194.41	125.44	111.82
10	10	127.46	129.42	205.60	136.68	120.41
11	10	140.22	139.03	235.38	148.45	128.02
12	10	158.00	156.07	251.98	163.43	142.26
13	10	167.01	168.40	255.19	174.71	152.10
14	10	178.27	178.18	283.84	186.49	160.96
15	10	184.68	188.83	292.74	195.59	168.60
2	20	31.45	31.04	57.29	33.35	30.86
3	20	42.71	42.16	80.25	47.03	41.49
4	20	57.25	55.63	104.11	64.40	55.10
5	20	70.86	69.96	129.98	78.41	67.97
6	20	83.16	81.86	153.77	92.04	79.38
7	20	95.93	95.16	183.04	102.26	90.64
8	20	106.69	107.47	206.19	112.64	100.82
9	20	112.98	112.64	213.00	121.93	105.62
10	20	128.02	129.38	250.51	134.43	119.72
11	20	137.04	137.41	255.22	143.81	124.39
12	20	147.93	151.11	279.03	158.33	136.35
13	20	169.80	172.89	316.73	180.45	156.18
14	20	175.73	176.19	310.30	184.69	158.80
15	20	184.48	185.42	342.62	192.21	163.76
2	30	35.05	34.23	59.91	37.17	34.08
3	30	43.32	42.46	79.06	48.11	42.10
4	30	55.04	53.93	109.89	60.11	52.59
5	30	65.82	65.88	131.69	71.95	63.76
6	30	80.60	77.92	148.62	85.20	76.10
7	30	93.59	91.79	184.30	101.36	87.81
8	30	103.23	102.37	211.37	111.97	96.79
9	30	116.54	118.14	240.84	127.22	109.16
10	30	128.86	130.79	241.40	139.31	120.84
11	30	139.30	141.14	271.44	150.68	129.00

Table 2 continued

Products	Shops	G1	G2	PCS	PCS+	B&B
12	30	149.33	150.99	301.28	156.97	136.83
13	30	169.24	170.06	329.62	180.69	152.67
14	30	176.25	177.83	349.03	183.10	158.26
15	30	179.76	183.80	357.24	193.08	162.60

Table 3 Average objective ratios in the experiment (as a percentage distance from the optimum value)

Products	Shops	$\frac{G1}{B\&B}$ (%)	$\frac{G2}{B\&B}$ (%)	$\frac{PCS}{B\&B}$ (%)	$\frac{PCS+}{B\&B}$ (%)
2	10	101.51	100.80	175.89	112.10
3	10	101.56	101.20	176.83	113.02
4	10	102.52	102.45	182.22	113.38
5	10	102.79	102.83	181.00	112.70
6	10	103.89	103.06	182.14	112.19
7	10	105.01	103.64	178.80	113.17
8	10	106.53	106.73	179.59	112.37
9	10	106.25	106.93	173.87	112.19
10	10	105.86	107.49	170.76	113.51
11	10	109.53	108.61	183.87	115.97
12	10	111.07	109.71	177.13	114.88
13	10	109.80	110.71	167.78	114.87
14	10	110.75	110.70	176.34	115.86
15	10	109.54	112.00	173.63	116.01
2	20	101.91	100.57	185.63	108.07
3	20	102.95	101.63	193.42	113.35
4	20	103.90	100.97	188.94	116.88
5	20	104.26	102.92	191.23	115.36
6	20	104.77	103.13	193.72	115.95
7	20	105.84	104.99	201.95	112.82
8	20	105.82	106.60	204.52	111.72
9	20	106.97	106.64	201.67	115.44
10	20	106.93	108.07	209.24	112.28
11	20	110.17	110.46	205.18	115.61
12	20	108.50	110.83	204.65	116.13
13	20	108.72	110.70	202.79	115.54
14	20	110.66	110.95	195.40	116.30
15	20	112.65	113.22	209.22	117.37
2	30	102.86	100.45	175.81	109.08
3	30	102.89	100.85	187.81	114.29
4	30	104.66	102.55	208.95	114.29

Table 3 continued

Products	Shops	$\frac{G1}{B\&B}$ (%)	$\frac{G2}{B\&B}$ (%)	$\frac{PCS}{B\&B}$ (%)	$\frac{PCS+}{B\&B}$ (%)
5	30	103.23	103.33	206.54	112.85
6	30	105.91	102.39	195.28	111.95
7	30	106.58	104.54	209.89	115.43
8	30	106.65	105.76	218.37	115.68
9	30	106.76	108.23	220.63	116.54
10	30	106.63	108.23	199.77	115.28
11	30	107.98	109.41	210.42	116.80
12	30	109.14	110.35	220.19	114.72
13	30	110.86	111.40	215.91	118.36
14	30	111.37	112.37	220.55	115.70
15	30	110.55	113.03	219.70	118.75

Algorithms G1 and G2 cannot compute optimum solutions (however it happends in some cases) nonetheless, computation times are much faster than the ones from B&B algorithm (for a bigger number of products/shops it appears to be up to 1,000 times slower). Algorithm G1 is much faster than G2 (it depends on instance size; from 2 to 9 times).

6 Conclusion and suggestions for future research

In this note, we introduced a new Internet shopping problem with price sensitive discounts, denoted as IS. We proved that problem IS is NP-hard in the strong sense if not every shop has all the required products, or if every shop has all the required products and product standard prices are distinct. The general case of problem IS is solvable in $O(nm^n)$ time, the case in which $p_{ij} = p_j$ for all shops is solvable in $O(nm!)$ time, and the case in which there are no discounts is equivalent to the well studied discrete facility location problem. Polynomial time exact algorithms are presented for problems IS- SAME- PRICES- SUBSETS, IS- SAME- PRODUCTS- AND- PRICES and IS- SAME- PRODUCTS- LINEAR- CHARGES. Two heuristics are suggested for the general problem. Computer experiments demonstrated their good performance on a set of examples generated by using data from the online book industry.

Modeling, computational complexity and algorithmic results presented in this paper is an initial step in studying and solving the Internet shopping problem. In the future, we will concentrate on the development of efficient exact and approximate solution procedures. Specifically, it is interesting to develop Integer Programming formulations and compare their suitability for various scenarios of input parameters and various existing standard software. Heuristics can be a good option for solving large-scale Internet shopping problems. Their development and efficiency verification through computer experiments as well as a worst-case analysis is another direction for future research.

Acknowledgments The work was partially supported by the grant from the National Council of Science of Poland no. N N519 643340. Also, Jedrzej Musial acknowledges the support of the Fonds National de la Recherche Luxembourg grant no. FNR/11/AM2c/21.

References

Blazewicz J, Kovalyov MY, Musial J, Urbanski AP, Wojciechowski A (2010) Internet shopping optimization problem. Appl Math Comput Sci 20(2):385–390

Chu W, Choi B, Song MR (2005) The role of on-line retailer brand and infomediary reputation in increasing consumer purchase intention. Int J Electron Commer 9:115–127

Clay K, Krishnan R, Wolff E (2001) Prices and price dispersion on the web: evidence from the online book industry. NBER Working Papers 8271, National Bureau of Economic Research, Inc

Eiselt H, Sandblom CL (2004) Decision analysis, location models, and scheduling problems. Springer, Berlin

Foundation TF (2008) E-commerce across Europe—progress and prospects

Garey MR, Johnson DS (1979) Computers and intractability: a guide to the theory of NP-completeness. Freeman, New York

Hof R (2003) More ways to price-shop online. BusinessWeek 3851:14

Holmberg K (1994) Solving the staircase cost facility location problem with decomposition and piecewise linearization. Eur J Oper Res 75(1):41–61. http://EconPapers.repec.org/RePEc:eee:ejores:v:75:y:1994:i:1:

Holmberg K, Ling J (1997) A lagrangean heuristic for the facility location problem with staircase costs. Eur J Oper Res 97(1):63–74. http://ideas.repec.org/a/eee/ejores/v97y1997i1p63-74.html

Iyigun C, Ben-Israel A (2010) A generalized weiszfeld method for the multi-facility location problem. Oper Res Lett 38(3):207–214

Krarup J, Pisinger D, Plastria F (2002) Discrete location problems with push-pull objectives. Discret Appl Math 123(1–3):363–378

Land A, Doig A (1960) An automatic method for solving discrete programming problems. Econometrica 28:497–520

Melo MT, Nickel S, da Gama FS (2009) Facility location and supply chain management—a review. Eur J Oper Res 196(2):401–412

Revelle CS, Eiselt HA, Daskin MS (2008) A bibliography for some fundamental problem categories in discrete location science. Eur J Oper Res 184(3):817–848

Soland RM (1974) Optimal facility location with concave costs. Oper Res 22(2):373–382

Stratila D (2009) Combinatorial optimization problems with concave costs. Massachusetts Institute of Technology, Sloan School of Management, Operations Research Center