Optim Lett (2016) 10:1577–1592
DOI 10.1007/s11590-015-0950-z

ORIGINAL PAPER

A capacitated facility location and inventory management problem with single sourcing

Ali Diabat[1]

Received: 14 May 2014 / Accepted: 15 September 2015 / Published online: 15 December 2015

Abstract The current paper addresses the integrated location and inventory problem with capacity constraints. Adopting more realistic assumptions comes at the cost of increased complexity and inability to solve the model with existing methods, mainly due to the non-linear terms that arise. We attempt to render the extended formulation solvable by linearizing its non-linear terms. Certain terms are replaced by exact reformulations while for the rest a piecewise linearization is implemented. The contribution of this work is not only the development of a formulation that is more practical, but also the reformulation that enables its solving with commercial software. We test our proposed approach on a benchmark dataset from the literature, including both small and large instances of the problem. Results clearly demonstrate the superiority of this approach in terms of both solution quality and computational time.

Keywords Supply chain · Location-inventory · Integer programming · Piecewise linearization

1 Introduction and background

The decisions involved in supply chain management can be distinguished, based on the time horizon of impact, into strategic level decisions, such as facility location, tactical level decisions, such as inventory policy and operational level decisions, such as transportation and routing [11]. The impact of strategic level decisions spans over a greater period than tactical decisions, while the latter have a longer impact than

✉ Ali Diabat
adiabat@masdar.ac.ae

[1] Department of Engineering Systems and Management, Masdar Institute of Science and Technology, Abu Dhabi, United Arab Emirates

operational decisions [21]. The integration of decisions across different levels has been increasingly studied in recent research streams, due to the significant benefits it can deliver for supply chains. Despite the fact that management at the strategic and tactical level concerns long-term and short-term decisions respectively, they are nonetheless interrelated and thus more information is exploited when integrating them.

Several authors have addressed integration in supply chains; Daskin et al. [1] integrate location and inventory decisions at distribution centers (DCs) and develop a mixed integer non-linear program (MINLP) which they solve with the help of Lagrangian relaxation (LR). The same model was studied by Shen et al. [19], who tackle it using the Branch-and-Price method. Risk pooling was accounted for, in the sense that a retailer can act as a distribution center. More thoroughly, risk pooling was discussed by Vidyarthi et al. [23]. The model addressed by Daskin et al. [1] and Shen et al. [19] was expanded by Ozsen et al. [15] to include capacity constraints for the single-sourcing case, in which there is a single plant that supplies all the retailers. Multiple sourcing for the capacitated problem was studied in later work of Diabat et al. [5], for which the authors develop an efficient Genetic Algorithm (GA) and report satisfactory results.

While the aforementioned works consider inventory only at the distribution centers (DCs), it is more practical in many cases to also consider inventory at the level of the retailers, although this leads to a more complicated formulation. This was done by Diabat et al. [9], where the authors simultaneously decide the location as well as inventory policy at both DCs and retailers. Once again, an LR based heuristic is developed to obtain results. In later work Diabat et al. [3], develop a set of propositions in order to improve the solutions obtained through the sub-problems, which were obtained through heuristics in their earlier work, while Diabat et al. [7] develop a GA to solve a capacitated location inventory problem. The integrated location inventory problem was also addressed by Diabat et al. [3], but this time for a closed-loop supply chain, meaning that the authors model both the forward and the reverse logistics network, the latter involving the collection of used products and their delivery to remanufacturing centers. Their model is implemented as a MINLP and solved with the help of an LR heuristic. Diabat [2] uses a hybrid genetics/simulated annealing algorithm to solve a vendor managed inventory (VMI) problem.

Shu [20] and Teo and Shu [22] consider location, distribution and inventory replenishment in a formulation solved with the help of a greedy heuristic and column generation, respectively. The heuristic is better able to tackle large sized problems than column generation. Another notable heuristic is developed by Jayaraman and Ross [12], who integrate strategic with operational level decisions and further consider cross-docking in the supply chain. Diabat and Richard [8] develop a nested LR approach for an integrated location and inventory model, formulated as a Mixed Integer Linear Program (MILP). Diabat and Theodorou [10] use the idea of reformulation and piecewise linearization in solving the uncapacitated joint location inventory problem that was developed by Diabat et al. [9]. Aside from the integration of strategic and tactical level decisions, or strategic and operational level decisions, the integration of tactical and operational level decisions has also been addressed and yields benefits for the supply chain. Le et al. [14] study the integrated inventory and routing problem for perishable products and implement a column-generation based solution approach. A tabu search heuristic was also developed by Diabat et al. [4] for the integrated inventory and routing problem with perishable products.

As mentioned, previous work of Diabat et al. [9] focuses on developing an LR approach for the multi-echelon joint location-inventory model that simultaneously decides the location of DCs as well as inventory policies at both DCs and retailers. The continuous relaxation of the formulation is non-convex and thus the problem cannot be solved to optimality using commercial software, even for small instances. This was the motivation for developing an LR approach; however, despite its contribution, the developed method posed certain drawbacks. These included the fact that the subproblems were not solved to optimality. The authors observed that the value of the lower bound was sometimes greater than the upper bound, producing a negative gap. Furthermore, another disadvantage of the method is that any minor change in the objective function or in the constraints could result in the method not functioning and requiring further calibration. Therefore, the formulation does not provide flexibility in modifying or extending it to account for realistic considerations, such as capacity constraints. Later work of Diabat et al. [6] develops an improved LR based heuristic to solve the problem. Yet, the problem remains of developing an extended formulation that can be solved without the use of heuristics, but rather with the help of commercial software.

In light of the aforementioned points, the aim of the current work is to extend the formulation studied in Diabat et al. [6,9] and Diabat and Theodorou [10] to account for capacity constraints and to develop an appropriate method to address the extended formulation. The LR approach developed in Diabat et al. [6,9] is not capable of solving the extended problem with capacity constraints. Thus, the main contribution of this work is the transformation of the problem into one that is not only solvable using commercial software but also capable of finding a high quality feasible solution through commercial software, by the linearization of the non-linear terms. While LR thus far has contributed in producing a set of lower bounds for the problem at hand, the best of which is then converted into a feasible upper bound with the help of heuristics, there is no guarantee that even the optimal Lagrangian bound will be within a certain gap of the optimal solution. This is overcome in our new approach due to the fact that the generated solution is feasible, so there is no need for more heuristics to modify the lower bound and make it feasible.

The current paper is structured as follows: Sect. 2 presents the problem statement as well as the formulation of the problem, while Sect. 3 is dedicated to the computational analysis conducted and the discussion of the results. Finally, Sect. 4 summarizes the conclusions of the current work, while providing certain directions for future research.

2 Problem statement and formulation

The current section will begin by describing the multi-echelon joint inventory location problem, without capacity constraints and with the adoption of the power-of-two inventory policy. Subsequently, the capacity constraints will be modeled and described, before being added to create the extended formulation. Finally, the current section will conclude with the linearization of the non-linear terms encountered in the extended formulation.

2.1 Multi-echelon joint inventory-location model (MJIL)

The objective of the multi-echelon joint inventory-location problem is to simultaneously determine the number and location of distribution centers (DCs), the retailers assigned to each DC and the amount and delivery time of products to each facility (retailer or DC). The retailers, with deterministic demand and with the ability to hold inventory, are assigned to DCs from which they will be supplied with the product. On their part, DCs order a product at regular intervals from the manufacturer and they also hold inventory resulting from product that has been ordered without having been shipped to the retailer. The aim of the model is to minimize the incurred costs, which include the fixed costs of placing any order, the inventory and shipping costs, as well as the fixed costs of setting up a DC. For a complete description of the model parameters the reader is referred to Diabat et al. [9]. At this point, only certain parameters as well as the sets and decision variables of the problem are presented. These hold for all formulations provided beyond this point, while subsequent models will introduce certain new variables.

Sets

I: set of retailers, indexed by $i = 1, \ldots, |I|$
J: set of possible DC locations, indexed by $j = 1, \ldots, |J|$

Parameters

f_j :	fixed cost of establishing a DC at location j
$\hat{k}_j, k_i, b_{ij}, c_{ij}, e_{ij}$:	parameters calculated for convenience, see [9]
M:	sufficiently large number
ε :	sufficiently small number

Decision variables

T_{ij} : cycle-time (time between orders placed) of retailer i when served by DC j
$\hat{T}_j$: cycle-time of DC j

Binary decision variables

$$X_j = \begin{cases} 1 \text{ if a DC is opened at location } j \\ 0 \text{ otherwise} \end{cases}$$

$$Y_{ij} = \begin{cases} 1 \text{ if a retailer } i \text{ is served by DC at location } j \\ 0 \text{ otherwise} \end{cases}$$

The formulation of the original problem is presented as follows:

$$\min_{X,Y,\hat{T},T} \sum_{j\in J}\left(f_j + \frac{\hat{k}_j}{\hat{T}_j}\right)X_j + \sum_{j\in J}\sum_{i\in I}\left(\frac{k_i}{T_{ij}} + b_{ij} + c_{ij}T_{ij} + e_{ij}max\left\{\hat{T}_j, T_{ij}\right\}\right)Y_{ij} \tag{1}$$

Subject to:

$$\sum_{j\in J} Y_{ij} = 1 \qquad \forall i \in I \tag{2}$$

$$Y_{ij} \leq X_j \qquad \forall i \in I, j \in J \tag{3}$$

$$Y_{ij}, X_j \in \{0, 1\} \qquad \forall i \in I, j \in J \tag{4}$$

$$\frac{\hat{T}_j}{T_{ij}} = 2^{N_{ij}} \qquad \forall i \in I, j \in J \tag{5}$$

$$N_{ij} \in \mathbb{Z} \qquad \forall i \in I, j \in J \tag{6}$$

$$\hat{T}_j, T_{ij} \geq \varepsilon \qquad \forall i \in I, j \in J \tag{7}$$

The objective function aims to minimize the sum of inventory, ordering, location and shipping costs. Constraints (2) ensure that each retailer is assigned to exactly one DC, while constraints (3) prohibit retailers to be assigned to DCs that are not opened. Constraints (4) ensure that the location and assignment variables are binary, while constraints (5)–(7) represent the power-of-two inventory policy adopted. Regarding the inventory, a power-of-two policy is adopted, according to which the ratio of the time elapsed between orders for the DC over the time between orders at the retailer is a power of two. It has been shown by Roundy [18] that this policy is 98 % effective at finding an optimal policy. Thus, adopting this simplifying policy can greatly reduce the complexity, without greatly sacrificing the accuracy of the solution.

2.2 MJIL extended with capacity constraints

A reformulation of the problem is presented at this point, with the addition of a set of capacity constraints and the elimination of the power-of-two inventory policy constraints. The resulting non-linear formulation will then be linearized through exact and piecewise linearization of the various non-linear terms. The extended model will be more practical, but at the same time more difficult to solve mainly due to the fact that it is not possible to know in advance which retailers will be assigned to each DC.

As mentioned, one of the contributions of the current work is the addition of capacity constraints to the formulation presented in Diabat et al. [6,9] and Diabat and Theodorou [10]. When adding capacity constraints, two types of problems can be distinguished: the problem with single-sourcing, in which the entire demand of a retailer is satisfied by one DC, and that with multiple-sourcing, according to which each retailer can be served by more than one DC. In this paper we consider single sourcing. It is important to consider whether the cycle time of the DC is greater than the cycle times of all retailers assigned to it, or in other words, it is important to focus on the case where $\hat{T}_j > T_{ij}$ for any retailer i that is assigned to DC j. The reason behind this consideration is that if the opposite holds, $T_{ij} \geq \hat{T}_j$, it is implied that the DC will place an order every time retailer i places an order. Therefore, the DC would not carry any inventory to be shipped to retailer i. We introduce the following binary variable:

$$Z_{ij} = \begin{cases} 1 & \text{if} \quad T_{ij} \leq \hat{T}_j \\ 0 & \text{otherwise} \end{cases}$$

The problem that arises now is that the capacity constraint for each DC will depend on which retailers are assigned to it. However, as we do not know the assignment in advance, the capacity constraint modeled in Eq. (14) requires multiplying the demand of all retailers by Y_{ij}. Further, let us define q_j, which denotes the maximum capacity of DC j. Then, the capacity constraints can be written as follows:

$$\hat{T}_j \sum_{i \in I} d_i Y_{ij} Z_{ij} \leq q_j \qquad \forall j \epsilon J \tag{8}$$

$$M Z_{ij} \geq \hat{T}_j - T_{ij} \qquad \forall i \in I;\ j \in J \tag{9}$$

$$M(Z_{ij} - 1) \leq \hat{T}_j - T_{ij} \qquad \forall i \in I;\ j \in J \tag{10}$$

Constraints (8) state that DC j is responsible for carrying inventory that is enough for all retailers assigned to it, with cycle times that are not greater than the DC cycle time. Constraints (9) state that if $\hat{T}_j$ is larger than T_{ij} then Z_{ij} must be equal to 1, whereas if $\hat{T}_j$ is not larger than T_{ij}, then Z_{ij} can be either 0 or 1. In a similar manner, constraints (10) ensure that if $\hat{T}_j$ is less than T_{ij}, then Z_{ij} must be equal to 0. On the other hand, if $\hat{T}_j$ is not less than T_{ij}, then Z_{ij} can be either 0 or 1. At this point, it is worth noting the case where $\hat{T}_j = T_{ij}$. In this case, constraints (9)–(10) are redundant. However, recall that for the case that $Y_{ij} = 1$ and $\hat{T}_j = T_{ij}$, DC j still does not need to carry any inventory to be shipped to retailer i. Therefore, constraints (8) will still hold for this case. Another important note is that since the problem described by Eqs. (1)–(10) is solved without considering the power-of-two constraints, and then the optimal values of $\hat{T}_j$ and T_{ij} are rounded to the nearest power-of-two values, the relation between $\hat{T}_j$ and T_{ij} may change before and after the rounding. However, for any possible change in the relationship, the capacity restriction will still be respected. For example, if $\hat{T}_j < T_{ij}$ changes to $\hat{T}_j = T_{ij}$, then in both cases DC j will not need to carry any inventory to be shipped to retailer i. If $\hat{T}_j > T_{ij}$ changes to $\hat{T}_j = T_{ij}$, then the left hand side of constraints (8) would be larger than its actual value and therefore this constraint would still be satisfied. After dropping the power-of-two constraints, adding the capacity constraints and dropping the index j from T_{ij}, the model can be stated as follows:

$$\min \sum_{j \in J} \left(f_j + \frac{\hat{k}_j}{\hat{T}_j} \right) X_j + \sum_{j \in J} \sum_{i \in I} \left(\frac{k_i}{T_i} + b_{ij} + c_{ij} T_i + e_{ij} max \left\{ \hat{T}_j, T_i \right\} \right) Y_{ij} \tag{11}$$

s.t.

$$\sum_{j \in J} Y_{ij} = 1 \qquad \forall i \in I \tag{12}$$

$$Y_{ij} \leq X_j \qquad \forall i \in I, j \in J \tag{13}$$

$$\hat{T}_j \sum_{i \in I} d_i Y_{ij} Z_{ij} \leq q_j \qquad \forall j \epsilon J \tag{14}$$

$$M Z_{ij} \geq \hat{T}_j - T_i \qquad \forall i \in I, j \in J \tag{15}$$

$$M(Z_{ij} - 1) \leq \hat{T}_j - T_i \qquad \forall i \in I, j \in J \tag{16}$$

$$Y_{ij}, X_j, Z_{ij} \in \{0, 1\} \qquad \forall i \in I, j \in J \tag{17}$$

$$\hat{T}_j, T_i \geq \varepsilon \qquad \forall i \in I, \forall j \in J \tag{18}$$

It is possible to observe by comparing the formulation presented by (11)–(18) with the formulation by (1)–(7) that the additional constraints are (14)–(16) (capacity constraints), while constraints (5) and (6) (power-of-two inventory policy) have been eliminated. Here, constraints (14) ensure that the total demand of retailers assigned to a DC will not exceed its capacity. Constraints (15) and (16) determine the binary variable Z_{ij} based on the cycle times of retailers and DCs. This non-linearity is one of the reasons for which this new problem cannot be solved using the Lagrangian relaxation heuristic developed in Diabat et al. [6,9]. The objective of this paper is to reformulate some of the non-linear terms and use piecewise linearization to approximate the remaining non-linear terms to convert the MINLP (11)–(18) to an MILP that is solvable using commercial solvers, such as CPLEX. We also test the quality of the new approach by comparing the objective function values of the obtained solutions to the objective function values of the solutions generated by solving the problem using a different heuristic that is explained in the following section of the computational analysis.

2.3 Piecewise linear based approximation

In this section we propose an MILP approximation for problem (11)–(18). The approximation is based on finding an exact linear formulation for some of the non-linear terms in the problem and linearizing the remaining non-linear terms using piecewise linear approximation.

2.3.1 An equivalent linear reformulation

We define the variables T_{ij}^{max}, W_{ij}^{max}, W_{ij} and $\hat{W}_{ij}$ as follows:

$$max\left\{\hat{T}_j, T_i\right\} = T_{ij}^{max}$$
$$T_{ij}^{max} Y_{ij} = W_{ij}^{max}$$
$$T_i Y_{ij} = W_{ij}$$
$$Z_{ij} \hat{T}_j Y_{ij} = \hat{W}_{ij}$$

Then the following problem is equivalent to problem (11)–(18)

$$\min \sum_{j\in J}\left(f_j + \frac{\hat{k}_j}{\hat{T}_j}\right) X_j + \sum_{j\in J}\sum_{i\in I}\left(\frac{k_i}{T_i} Y_{ij} + b_{ij} Y_{ij} + c_{ij} W_{ij} + e_{ij} W_{ij}^{max}\right) \quad (19)$$

s.t.

$$\sum_{j\in J} Y_{ij} = 1 \qquad \forall i \in I \quad (20)$$

$$Y_{ij} \le X_j \qquad \forall i \in I, j \in J \quad (21)$$

$$\sum_{i\in I} d_i \hat{W}_{ij} \le q_j \qquad \forall j \in J \quad (22)$$

$$M Z_{ij} \ge \hat{T}_j - T_i \qquad \forall i \in I, j \in J \quad (23)$$

$$M(Z_{ij}-1) \leq \hat{T}_j - T_i \qquad \forall i \in I, j \in J \tag{24}$$

$$T_{ij}^{max} \geq \hat{T}_j \qquad \forall i \in I, j \in J \tag{25}$$

$$T_{ij}^{max} \geq T_i \qquad \forall i \in I, j \in J \tag{26}$$

$$T_{ij}^{max} \leq W_{ij}^{max} + M(1-Y_{ij}) \qquad \forall i \in I, j \in J \tag{27}$$

$$T_i \leq W_{ij} + M(1-Y_{ij}) \qquad \forall i \in I, j \in J \tag{28}$$

$$\hat{T}_j \leq \hat{W}_{ij} + M(2-Y_{ij}-Z_{ij}) \qquad \forall i \in I, j \in J \tag{29}$$

$$Y_{ij}, X_j \in \{0,1\} \qquad \forall i \in I, j \in J \tag{30}$$

$$T_{ij}^{max}, W_{ij}^{max}, W_{ij}, \hat{W}_{ij} \geq 0 \qquad \forall i \in I, j \in J \tag{31}$$

$$T_i, \hat{T}_j \geq \varepsilon \qquad \forall i \in I, j \in J \tag{32}$$

Note how this formulation contains additional constraints, namely (26)–(29), as part of the linearization, while the objective function and constraints (22) have been adapted based on the newly introduced variables. Regarding the new constraints, (25) and (26) ensure that the new variable T_{ij}^{max} will be equal to the greatest among T_i and $\hat{T}_j$, thus replacing $max\left\{\hat{T}_j, T_i\right\}$ from the objective function. Constraints (27)–(29) help define the new variables $W_{ij}^{max} W_{ij}$ and $\hat{W}_{ij}$ which replace non-linear terms in the objective function and in constraints (22).

2.3.2 Piecewise linearization

The only non-linear terms remaining in problem (19)–(31) are $\frac{\hat{k}_j}{\hat{T}_j}X_j$ and $\frac{k_i}{T_i}Y_{ij}$. To linearize these two terms, we use piecewise linear functions to approximate $\frac{1}{\hat{T}_j}$ and $\frac{1}{T_i}$. Since $\frac{1}{\hat{T}_j}$ and $\frac{1}{T_i}$ are convex functions, they can be linearized by allowing the binary variables that are used in the linearization to be continuous. This is very important as this linearization does not result in a higher number of binary variables.

To linearize $\frac{1}{\hat{T}_j}$, assume that we divide the range of $\hat{T}_j$ into $n = 1, \ldots, N$ linear pieces using equidistant placement of the breakpoints. This will lead to $N+1$ break points $\hat{t}_j^1, \hat{t}_j^2, \ldots, \hat{t}_j^{(N+1)}$. Then, $\frac{1}{\hat{T}_j} \cong \frac{1}{\hat{t}_j^n}\hat{U}_j^n + \frac{1}{\hat{t}_j^{n+1}}(1-\hat{U}_j^n) = \frac{1}{\hat{t}_j^{n+1}} + (\frac{1}{\hat{t}_j^n} - \frac{1}{\hat{t}_j^{n+1}})\hat{U}_j^n$, where $\hat{U}_j^n \in [0,1]$. Let $\hat{V}_j^n = \hat{U}_j^n X_j$, then, $\frac{1}{\hat{T}_j}X_j \cong \frac{1}{\hat{t}_j^{n+1}}X_j + (\frac{1}{\hat{t}_j^n} - \frac{1}{\hat{t}_j^{n+1}})\hat{V}_j^n$. Similarly, $\frac{1}{T_i} \cong \frac{1}{t_i^{n+1}} + (\frac{1}{t_i^n} - \frac{1}{t_i^{n+1}})U_i^n$. Now, by letting $V_{ij}^n = U_i^n Y_{ij}$ we get $\frac{1}{T_i}Y_{ij} \cong \frac{1}{t_i^{n+1}}Y_{ij} + (\frac{1}{t_i^n} - \frac{1}{t_i^{n+1}})V_{ij}^n$

Therefore, problem (19)–(31) can be approximated as follows

$$\min \sum_{j\in J} f_j X_j + \sum_{j\in J}\sum_{n\in N} \frac{\hat{k}_j}{\hat{t}_j^{n+1}} X_j + \sum_{j\in J}\sum_{n\in N} \hat{k}_j \left(\frac{1}{\hat{t}_j^n} - \frac{1}{\hat{t}_j^{n+1}}\right) \hat{V}_j^n + \sum_{j\in J}\sum_{i\in I}\sum_{n\in N} \frac{k_i}{t_i^{n+1}} Y_{ij}$$
$$+ \sum_{j\in J}\sum_{i\in I}\sum_{n\in N} k_i \left(\frac{1}{t_i^n} - \frac{1}{t_i^{n+1}}\right) V_{ij}^n + \sum_{j\in J}\sum_{i\in I} \left(b_{ij} Y_{ij} + c_{ij} W_{ij} + e_{ij} W_{ij}^{max}\right) \tag{33}$$

s.t.

$$\sum_{j\in J} Y_{ij} = 1 \qquad \forall i \in I \tag{34}$$

$$Y_{ij} \le X_j \qquad \forall i \in I, j \in J \tag{35}$$

$$\sum_{i\in I} d_i \hat{W}_{ij} \le q_j \qquad \forall j \epsilon J \tag{36}$$

$$M Z_{ij} \ge \hat{T}_j - T_i \qquad \forall i \in I, j \in J \tag{37}$$

$$M(Z_{ij} - 1) \le \hat{T}_j - T_i \qquad \forall i \in I, j \in J \tag{38}$$

$$T_{ij}^{max} \ge \hat{T}_j \qquad \forall i \in I, j \in J \tag{39}$$

$$T_{ij}^{max} \ge T_i \qquad \forall i \in I, j \in J \tag{40}$$

$$T_{ij}^{max} \le W_{ij}^{max} + M(1 - Y_{ij}) \qquad \forall i \in I, j \in J \tag{41}$$

$$T_i \le W_{ij} + M(1 - Y_{ij}) \qquad \forall i \in I, j \in J \tag{42}$$

$$\hat{T}_j \le \hat{W}_{ij} + M(2 - Y_{ij} - Z_{ij}) \qquad \forall i \in I, j \in J \tag{43}$$

$$\sum_{n\in N} \hat{U}_j^n = 1 \qquad \forall j \in J \tag{44}$$

$$\sum_{n\in N} U_{ij}^n = 1 \qquad \forall i \in I, j \in J \tag{45}$$

$$X_j - 1 \le \hat{V}_j^n - \hat{U}_j^n \qquad \forall n \in N \quad \forall j \in J \tag{46}$$

$$Y_{ij} - 1 \le \hat{V}_{ij}^n - \hat{U}_{ij}^n \qquad \forall n \in N, j \in J, i \in I \tag{47}$$

$$\hat{V}_j^n, \hat{U}_j^n, \hat{V}_{ij}^n, \hat{U}_{ij}^n \in [0, 1] \qquad \forall i \in I, j \in J \tag{48}$$

$$X_j, Y_{ij}, Z_{ij} \in \{0, 1\} \qquad \forall i \in I, j \in J \tag{49}$$

$$\hat{T}_j, T_i, T_{ij}^{max}, W_{ij}^{max}, W_{ij}, \hat{W}_{ij} \ge 0 \qquad \forall i \in I, j \in J \tag{50}$$

The additional constraints in this formulation are constraints (44)–(47), which represent the piecewise linearization for the terms found in the previous objective function, $\frac{\hat{k}_j}{\hat{T}_j} X_j$ and $\frac{k_i}{T_i} Y_{ij}$. Aside from these, the formulation is identical to that described by (19)–(32).

3 Computational analysis

In this section we present the results obtained from the computational analysis. The data sets used for the current analysis were found in Diabat et al. [6,9] and they include both small and large size instances of the problem. For small sizes of the problem, i.e. $(|I| \times |J|) = (6 \times 3, 8 \times 4, 10 \times 5)$, we are able to obtain an optimal solution with the help of the commercial software GAMS (General Algebraic Modeling System), using the BARON solver. Despite the high computational time required for this solver, and its inability to solve certain instances to optimality, nevertheless we compare its results with those obtained using the piecewise linearization, which constitute an upper bound to the objective value, in order to assess the quality of the bound and evaluate the proposed approach.

We conduct our experiments on a total of 48 instances, 12 different instances for each of these sizes, namely (6 × 3), (8 × 4), (10 × 5) and (15 × 10). The results are summarized in Tables 1, 2, 3, 4 and they include the number of retailers and DCs, as well as the value obtained from each method and the time required to reach a solution, reported in both seconds and hours. Finally, the time and value ratio are given in order to compare the two solution approaches. From all tables we can observe the very high computational time for BARON, while Table 4 shows that the solver reaches its limit as GAMS runs out of memory and optimality is not achieved.

Results indicate that the linearization achieves very good results, with solutions being on overall average for all instances within 1 % of the solution obtained from the commercial solver, or 1.02 % if we exclude the instances of Table 4, for which BARON solver achieved worse results than the linearization. In addition, the gap remains fairly stable even as the instance size increases: from Table 1, the average solution gap is 1.0226, while from Tables 2 and 3 it is 1.0268 and 1.0247, respectively.

Table 1 Comparative results for instance (6 × 3)

Inst.	BarGAMS	Time (s)	Time (h)	Linear	Time (s)	Time (h)	Time ratio	Value ratio
1	3133	44,076	12.2	3181	467	0.1	0.0106	1.0152
2	2551	44,132	12.3	2643	427	0.1	0.0097	1.0363
3	1456	43,700	12.1	1482	396	0.1	0.0091	1.0183
4	3272	43,427	12.1	3318	472	0.1	0.0109	1.0142
5	2634	43,519	12.1	2639	453	0.1	0.0104	1.0020
6	2634	43,318	12.0	2682	405	0.1	0.0094	1.0183
7	3369	43,959	12.2	3403	402	0.1	0.0091	1.0101
8	2107	43,873	12.2	2216	414	0.1	0.0094	1.0515
9	1511	43,992	12.2	1587	406	0.1	0.0092	1.0504
10	2599	135,109	37.5	2688	1388	0.4	0.0103	1.0341
11	3479	43,868	12.2	3504	404	0.1	0.0092	1.0070
12	3466	43,455	12.1	3515	396	0.1	0.0091	1.0142
							Avg.	**1.0226**

Table 2 Comparative results for instance (8 × 4)

Inst.	BarGAMS	Time (s)	Time (h)	Linear	Time (s)	Time (h)	Time ratio	Value ratio
1	6253	158,056	43.9	6554	1703	0.5	0.0108	1.0482
2	3680	163,598	45.4	3720	1568	0.4	0.0096	1.0111
3	6844	145,075	40.3	6998	1372	0.4	0.0095	1.0225
4	5095	159,677	44.4	5324	1544	0.4	0.0097	1.0449
5	3396	123,804	34.4	3505	1133	0.3	0.0092	1.0320
6	4220	168,564	46.8	4280	1823	0.5	0.0108	1.0142
7	4889	122,298	34.0	5025	1302	0.4	0.0107	1.0277
8	5223	143,303	39.8	5413	1576	0.4	0.0110	1.0363
9	3285	43,585	12.1	3312	414	0.1	0.0095	1.0081
10	7436	131,469	36.5	7580	1330	0.4	0.0101	1.0194
11	6021	167,490	46.5	6252	1570	0.4	0.0094	1.0384
12	6973	190,372	52.9	7101	1968	0.5	0.0103	1.0183
							Avg.	**1.0268**

Table 3 Comparative results for instance (10 × 5)

Inst.	BarGAMS	Time (s)	Time (h)	Linear	Time (s)	Time (h)	Time ratio	Value ratio
1	10,726	123,529	34.3	10,747	1238	0.3	0.0100	1.0020
2	10,726	142,762	39.7	11,219	1366	0.4	0.0096	1.0460
3	8107	172,790	48.0	8272	1574	0.4	0.0091	1.0204
4	8855	118,768	33.0	8900	1270	0.4	0.0107	1.0050
5	5280	206,297	57.3	5558	2098	0.6	0.0102	1.0526
6	7691	214,656	59.6	7745	2354	0.7	0.0110	1.0070
7	8689	195,886	54.4	8715	2094	0.6	0.0107	1.0030
8	8481	127,479	35.4	8689	1154	0.3	0.0091	1.0246
9	4906	188,804	52.4	5016	1894	0.5	0.0100	1.0225
10	10,643	205,876	57.2	11,191	2215	0.6	0.0108	1.0515
11	12,472	184,592	51.3	12,938	1776	0.5	0.0096	1.0373
12	7733	209,277	58.1	7923	2202	0.6	0.0105	1.0246
							Avg.	**1.0247**

These results were obtained within a fraction of the time, on average 0.01 %. The small gap demonstrates the high solution quality obtained through the linearization, while its significantly reduced time further enhances its attractiveness.

For large instances a different heuristic is developed, which relies on solving the two problems sequentially. Essentially, the new heuristic approach is based on solving the facility location problem and the inventory management problem separately. The first step is to solve the following uncapacitated facility location problem (UFLP).

$$\min \sum_{j \in J} f_j X_j + \sum_{j \in J} \sum_{i \in I} b_{ij} Y_{ij}$$

Table 4 Comparative results for instance (15 × 10)

Inst.	BarGAMS	Time (s)	Time (h)	Linear	Time (s)	Time (h)	Time ratio	Value ratio
1	23,073	288,000	80.0	18,320	2606	0.7	0.0091	0.7940
2	17,945	288,000	80.0	14,123	3119	0.9	0.0108	0.7870
3	21,221	288,000	80.0	16,425	2933	0.8	0.0102	0.7740
4	29,054	288,000	80.0	24,551	3149	0.9	0.0109	0.8450
5	33,469	288,000	80.0	28,114	3070	0.9	0.0107	0.8400
6	27,203	288,000	80.0	22,061	2926	0.8	0.0102	0.8110
7	32,330	288,000	80.0	24,280	2681	0.7	0.0093	0.7510
8	35,748	288,000	80.0	28,527	2794	0.8	0.0097	0.7980
9	22,218	288,000	80.0	18,041	3073	0.9	0.0107	0.8120
10	42,300	288,000	80.0	35,489	2956	0.8	0.0103	0.8390
11	32,472	288,000	80.0	24,971	2769	0.8	0.0096	0.7690
12	24,497	288,000	80.0	18,691	2785	0.8	0.0097	0.7630

s.t.

$$\begin{aligned} &\sum_{j\in J} Y_{ij} = 1 && \forall i \in I \\ &Y_{ij} \leq X_j \quad \forall i \in I; && \forall j \in J \\ &X_j, Y_{ij} \in \{0, 1\} && \forall i \in I; \quad \forall j \in J \end{aligned}$$

From solving this problem, we obtain the optimal values corresponding to the set of open DCs and the set of assignments, X_j^* and Y_{ij}^*, respectively. Then, we define $J_o$to be the set of open DCs, i.e. $J_o = \left\{ j \backslash X_j^* = 1 \right\}$. Also, let I_j be the set of retailers assigned to DC $j \in J_o$, i.e. $I_j = \left\{ i \backslash Y_{ij}^* = 1 \right\}, \forall j \in J_o$. Having defined these subsets, we can now decompose the problem into multiple one-DC multi-retailer problems. For every $j \in J_o$, we have a one-DC multi-retailer inventory problem, as follows, which we solve using the algorithm by Roundy [18].

$$\min \frac{\hat{k}_j}{\hat{T}_j} + \sum_{i\in I_j} \left(\frac{k_i}{T_i} + c_{ij} T_i + e_{ij} max \left\{ \hat{T}_j, T_i \right\} \right)$$

This decomposition allows for easier solving of the problem, since we know the assignment of retailers to DCs. Now, we solve for $\hat{T}_j^*$ and T_i^*. Then, for every $j \in J_o$ and for every $i \in I_j$ if $\hat{T}_j^* > T_i^*$ we set $Z_{ij}^* = 1$, otherwise we set $Z_{ij}^* = 0$. At this point we need to check whether the capacity constraints (14) are satisfied. Given that we have solved the inventory problem without considering the capacity constraints, they will most likely be violated. Let J_V be the set of open DCs for which the capacity constraint is violated, i.e. $J_V = \{ j \in J_o \backslash \hat{T}_j^* \sum_{i\in I_j} d_{ij} Y_{ij}^* Z_{ij}^* > q_j \}$. For these DCs, the demand of retailers assigned with cycle time less than that of the DC exceeds the capacity. Thus, it is important to adjust the cycle time of some of the retailers, in order to make it equal to the cycle time of the DC so that the latter does not carry inventory for

Table 5 Comparative results for instance (20 × 15)

Inst.	Linear	Time (s)	Time (h)	SeqHeu	Time (s)	Time (h)	Time ratio	Value ratio
1	32,627	3263	0.9	58,995	3393	0.9	0.9615	0.5530
2	48,474	4847	1.3	90,643	3733	1.0	1.2987	0.5348
3	68,371	6837	1.9	126,004	5128	1.4	1.3333	0.5426
4	47,231	4723	1.3	103,262	4345	1.2	1.0870	0.4574
5	90,734	9073	2.5	182,419	8257	2.3	1.0989	0.4974
6	53,446	5345	1.5	87,429	4650	1.3	1.1494	0.6113
7	57,796	5780	1.6	99,499	5375	1.5	1.0753	0.5809
8	60,282	6028	1.7	114,586	4762	1.3	1.2658	0.5261
9	59,350	5935	1.6	118,700	5994	1.7	0.9901	0.5000
10	50,028	5003	1.4	110,004	4152	1.2	1.2048	0.4548
11	54,689	5469	1.5	91,413	4539	1.3	1.2048	0.5983
12	55,266	5527	1.5	114,309	4532	1.3	1.2195	0.4835

Table 6 Comparative results for instance (25 × 15)

Inst.	Linear	Time (s)	Time (h)	SeqHeu	Time (s)	Time (h)	Time ratio	Value ratio
1	81,430	8143	2.3	184,704	7084	2.0	1.1494	0.4409
2	67,603	6760	1.9	103,657	6287	1.7	1.0753	0.6522
3	80,387	8039	2.2	168,388	7637	2.1	1.0526	0.4774
4	52,622	5262	1.5	99,861	4210	1.2	1.2500	0.5270
5	107,934	10,793	3.0	196,710	8743	2.4	1.2346	0.5487
6	71,060	7106	2.0	138,506	6537	1.8	1.0870	0.5130
7	65,682	6568	1.8	110,917	5452	1.5	1.2048	0.5922
8	79,473	7947	2.2	128,544	6596	1.8	1.2048	0.6183
9	61,841	6184	1.7	137,292	5628	1.6	1.0989	0.4504
10	67,141	6714	1.9	122,754	5438	1.5	1.2346	0.5470
11	111,775	11,177	3.1	205,009	9501	2.6	1.1765	0.5452
12	57,175	5717	1.6	115,759	5432	1.5	1.0526	0.4939

that retailer. Having detected the retailers for which the cycle time should be adjusted, we solve the knapsack problem for every DC in the violated set, or for every $j \in J_V$:

$$\max \sum_{i \in I_j} Z_{ij}^V$$

s.t.

$$\begin{aligned} &\sum_{i \in I_j} d_i Z_{ij}^V \leq \frac{q_j}{\hat{T}_j^*} && \forall j \\ &Z_{ij}^V \in \{0, 1\} && \forall i, j \end{aligned}$$

Table 7 Comparative results for instance (30 × 15)

Inst.	Linear	Time (s)	Time (h)	SeqHeu	Time (s)	Time (h)	Time ratio	Value ratio
1	128,345	12,835	3.6	199,996	10,524	2.9	1.2195	0.6417
2	109,162	10,916	3.0	199,580	9170	2.5	1.1905	0.5470
3	91,977	9198	2.6	174,256	7266	2.0	1.2658	0.5278
4	62,574	6257	1.7	111,913	4881	1.4	1.2821	0.5591
5	133,826	13,383	3.7	304,752	13,918	3.9	0.9615	0.4391
6	125,148	12,515	3.5	219,725	9511	2.6	1.3158	0.5696
7	126,518	12,652	3.5	269,936	9868	2.7	1.2821	0.4687
8	99,483	9948	2.8	166,288	8755	2.4	1.1364	0.5983
9	117,840	11,784	3.3	182,145	11,077	3.1	1.0638	0.6470
10	109,086	10,909	3.0	172,795	9054	2.5	1.2048	0.6313
11	126,061	12,606	3.5	226,163	12,480	3.5	1.0101	0.5574
12	89,881	8988	2.5	179,449	7011	1.9	1.2821	0.5009

Table 8 Comparative results for instance (50 × 20)

Inst.	Linear	Time (s)	Time (h)	SeqHeu	Time (s)	Time (h)	Time ratio	Value ratio
1	290,292	29,029	8.1	510,453	23,223	6.5	1.2500	0.5687
2	129,950	12,995	3.6	273,204	11,825	3.3	1.0989	0.4757
3	209,597	20,960	5.8	325,725	17,816	4.9	1.1765	0.6435
4	213,789	21,379	5.9	437,469	20,524	5.7	1.0417	0.4887
5	302,868	30,287	8.4	640,254	25,744	7.2	1.1765	0.4730
6	270,380	27,038	7.5	539,822	26,768	7.4	1.0101	0.5009
7	204,357	20,436	5.7	436,013	16,757	4.7	1.2195	0.4687
8	198,069	19,807	5.5	392,048	19,015	5.3	1.0417	0.5052
9	197,021	19,702	5.5	351,824	17,141	4.8	1.1494	0.5600
10	284,004	28,400	7.9	600,377	29,536	8.2	0.9615	0.4730
11	304,964	30,496	**8.5**	648,260	24,397	**6.8**	1.2500	0.4704
12	227,413	22,741	6.3	352,935	20,012	5.6	1.1364	0.6443

Now, for any $j \in J_V$ and $i \in I_j$ with $Z_{ij}^V = 0$, we change Z_{ij}^* from 1 to 0, and T_i^* from its old value to $\hat{T}_j^*$. The objective is to maintain the values of Z_{ij}^* wherever possible; however, some will have to become equal to 0, only enough to make the solution feasible. The aim of this problem is to decide which will become 0 and which will remain 1. Thus, this problem will dictate which retailers are best removed from the capacity constraint, which means we would have to adjust their cycle times. Next, we calculate the objective function value of the sequential heuristics approach as follows:

$$SeqHeu = \sum_{j \in J_o} \left(f_j + \frac{\hat{k}_j}{\hat{T}_j^*} \right) + \sum_{j \in J_o} \sum_{i \in I_j} \left(\frac{k_i}{T_i^*} + b_{ij} + c_{ij} T_i^* + e_{ij} max \left\{ \hat{T}_j^*, T_i^* \right\} \right)$$

For the significantly larger instances, where (retailers × DCs) are (20 × 15), (25 × 15), (30 × 15) and (50 × 20) we compare the piecewise linear heuristics and the sequential heuristic in terms of the objective function value obtained, as well as the solution time. For each of these sizes, we run 12 different instances, leading to a total of 48 instances, and the results are shown in Tables 5, 6, 7, 8.

From these tables it is possible to observe that the longest time used by the piecewise linearization based heuristic was 8.5 h for solving the instance of size (50 × 20), while for the sequential based heuristic this time was 6.8 h, reported in Table 8. While the linearization requires a slightly higher computational time, on total average 3.3 h compared to 2.9 h for the sequential heuristic, its solutions are better by an average of 50 %, weighing the trade-off between time and quality to its advantage.

4 Conclusion

The current work addresses the capacitated facility location inventory problem. The addition of capacity constraints leads to a model of higher practical value. However, it poses an additional challenge in terms of solving the problem, due to the non-linear terms that occur. In light of this, we aim to reformulate some of the terms to make them linear and implement piecewise linearization for the rest, in order to render the problem solvable with the help of commercial software. For large problems, we develop a sequential heuristic approach. We solve both small and large instances of the problem to evaluate the linearization and the sequential heuristic. Our results demonstrate the overall superiority of the linearization in terms of time and solution quality.

The promising results obtained by using the linearization demonstrate the need for further investigating such methods in future research. For example, future work could focus on implementing for the same problem a strategy for optimally placing breakpoints for the linearization and selecting the placement which minimizes either the maximal deviation of the piecewise linear function from the original one or the area between the functions. This can be found in the works of [13,16,17]. The current work could also be extended by adding multiple products or various transportation options to the supply chain.

References

1. Daskin, M., Coullard, C., Shen, Z.-J.: An inventory-location model: formulation, solution algorithm and computational results. Ann. Oper. Res. **110**(1–4), 83–106 (2002)
2. Diabat, A.: Hybrid algorithm for a vendor managed inventory system in a two-echelon supply chain. Eur. J. Oper. Res. **238**(1),114–121 (2014)
3. Diabat, A., Abdallah, T., Henschel, A.: A closed-loop location-inventory problem with spare parts consideration. Comput. Oper. Res. **54**, 245–256 (2015)
4. Diabat, A., Abdallah, T., Le, T.: A hybrid tabu search based heuristic for the periodic distribution inventory problem with perishable goods. Ann. Oper. Res. (2014). doi:10.1007/s10479-014-1640-4
5. Diabat, A., Aouam, T., Ozsen, L.: An evolutionary programming approach for solving the capacitated facility location problem with risk pooling. Int. J. Appl. Dec. Sci. **2**(4), 389–405 (2009)
6. Diabat, A., Battaïa, O., Nazzal, D.: An improved lagrangian relaxation-based heuristic for a joint location-inventory problem. Comput. Oper. Res. **61**, 170–178 (2015)

7. Diabat, A., Deskoores, R.M.: A hybrid genetic algorithm based heuristic for an integrated supply chain problem. J. Manufact. Syst. (2015). doi:10.1016/j.jmsy.2015.04.011
8. Diabat, A., Richard, J.-P.: An integrated supply chain problem: a nested Lagrangian relaxation approach. Ann. Oper. Res. **229**(1), 303–323 (2015)
9. Diabat, A., Richard, J.-P., Codrington, C.: A Lagrangian relaxation approach to simultaneous strategic and tactical planning in supply chain design. Ann. Oper. Res. **203**(1), 55–80 (2013)
10. Diabat, A., Theodorou, E.: A location-inventory supply chain problem: reformulation and piecewise linearization. Comput. Ind. Eng. **90**, 381–389 (2015)
11. Dolgui, A., Proth, J.-M.: Supply Chain Engineering: Useful Methods and Techniques. Springer, New York (2010)
12. Jayaraman, V., Ross, A.: A simulated annealing methodology to distribution network design and management. Eur. J. Oper. Res. **144**(3), 629–645 (2003)
13. Kallrath, J., Rebennack, S.: Computing area-tight piecewise linear overestimators, underestimators and tubes for univariate functions. In: Optimization in Science and Engineering. Springer, New York, pp. 273–292 (2014)
14. Le, T., Diabat, A., Richard, J.-P., Yih, Y.: A column generation-based heuristic algorithm for an inventory routing problem with perishable goods. Optim. Lett. **7**(7), 1481–1502 (2013)
15. Ozsen, L., Coullard, C.R., Daskin, M.S.: Capacitated DC location model with risk pooling. Naval Res. Log. (NRL) **55**(4), 295–312 (2008)
16. Rebennack, S., Kallrath, J.: Continuous piecewise linear δ-approximations for MINLP problems. I. Minimal breakpoint systems for univariate functions . Tech. Rep., Colorado School of Mines, Division of Economics and Business. Working paper series 2012-12 (2012)
17. Rebennack, S., Kallrath, J.: Continuous Piecewise Linear δ-Approximations for MINLP Problems. II, Bivariate and Multivariate Functions. Tech. Rep., Colorado School of Mines, Division of Economics and Business. Working paper series 2012-13 (2012)
18. Roundy, R.: 98%-Effective Integer-Ratio Lot-Sizing for One-DC Multi-Retailer Systems. Manag. Sci. **31**(11), 1416–1430 (1985)
19. Shen, Z.-J.M., Coullard, C., Daskin, M.S.: A joint location-inventory model. Transp. Sci. **37**(1), 40–55 (2003)
20. Shu, J.: An efficient Greedy Heuristic for DC-retailer network design optimization. Transp. Sci. **44**(2), 183–192 (2010)
21. Simchi-Levi, D., Kaminsky, P., Simchi-Levi, E.: Designing and managing the supply chain: concepts, strategies, and case studies. Mc-Graw-Hill, New York (2003)
22. Teo, C.-P., Shu, J.: DC-retailer network design problem. Oper. Res. **52**(3), 396–408 (2004)
23. Vidyarthi, N., Çelebi, E., Elhedhli, S., Jewkes, E.: Integrated production-inventory-distribution system design with risk pooling: model formulation and heuristic solution. Transp. Sci. **41**(3), 392–408 (2007)