An iterated local search metaheuristic for the single-product inventory-routing problem

Aldair Alvarez
Departamento de Engenharia de Produção, Universidade Federal de São Carlos
Rodovia Washington Luís - Km 235, CEP 13565-905, São Carlos-SP, Brasil
aldair@dep.ufscar.br

Pedro Munari
Departamento de Engenharia de Produção, Universidade Federal de São Carlos
Rodovia Washington Luís - Km 235, CEP 13565-905, São Carlos-SP, Brasil
munari@dep.ufscar.br

Reinaldo Morabito
Departamento de Engenharia de Produção, Universidade Federal de São Carlos
Rodovia Washington Luís - Km 235, CEP 13565-905, São Carlos-SP, Brasil
morabito@ufscar.br

RESUMO

O problema de roteamento de estoques vem recebendo maior atenção nos últimos anos, principalmente porque integra duas atividades essenciais que compõem a gestão da cadeia de suprimentos, a saber: controle de estoques e roteamento de veículos. Para resolver este problema, um algoritmo metaheurístico baseado em busca local iterada é proposto. Além disso, duas funções objetivo diferentes são abordadas. A primeira minimiza o custo total, o qual é composto de custos de estocagem e de viagem, enquanto a segunda minimiza uma razão logística. Experimentos computacionais mostram que a metaheurística proposta pode fornecer soluções razoavelmente boas em tempos computacionais relativamente curtos para ambas as funções objetivo quando aplicada a instâncias da literatura. Para algumas dessas instâncias, foram encontradas soluções com valores melhores que aqueles conhecidos até o momento na literatura.

PALAVRAS CHAVE. Problema de roteamento de estoques; Busca local iterada; Razão logística.

L&T – Logística e Transportes; MH – Metaheurísticas

ABSTRACT

The inventory routing problem has received increased attention in recent years mainly because it integrates two essential activities that compose the supply chain management, namely: inventory control and vehicle routing. To solve this problem, we propose a metaheuristic algorithm based on iterated local search. Also, we tackle two different objective functions. The first minimizes total cost, composed of inventory and traveling costs and the second minimizes a logistic ratio. Computational experiments show that the proposed metaheuristic can provide reasonably good solutions in relatively short running times for both objective functions when applied to problem instances from the literature. The solutions obtained for some instances improve upon previous best-known solutions.

KEYWORDS. Inventory routing problem; Iterated local search; Logistic ratio.

L&T – Logistics and transportation; MH – Metaheuristics

1. Introduction

Nowadays, companies have realized that optimizing the operations of the supply chain can yield considerable gains. Therefore, many strategies have appeared aiming to integrate the activities that compose the supply chain management. The Inventory Routing Problem (IRP) models situations in which suppliers are allowed to manage the inventory of their customers aiming to reduce logistics costs and improve the overall performance of the supply chain [Archetti and Speranza, 2016]. In return, suppliers must ensure that no stockout will occur at the customers. Hence, the IRP consists of determining the best visit schedule, delivery quantities and vehicle routing plan to meet the demands of the customers over the planning horizon.

In this paper, we address the basic variant of the IRP, i.e., the single-item, multi-vehicle and multi-period IRP. We propose a metaheuristic algorithm based on Iterated Local Search (ILS) to solve this variant considering two different types of objective function. Computational experiments show that the metaheuristic can find good feasible solutions in reasonably short running times.

The IRP has been studied for more than 30 years [Coelho et al., 2014] since the paper of Bell et al. [1983], who solved an integrated inventory and vehicle routing problem for the bulk gas industry. Thenceforth, many IRP variants have been proposed and solved in the literature, such as, long-term IRPs [Dror et al., 1985; Dror and Ball, 1987], IRPs minimizing transportation costs only [Campbell and Savelsbergh, 2004], with pickup and delivery and routes spanning multiple time periods [Savelsbergh and Song, 2007, 2008], with backlogging [Abdelmaguid et al., 2009], with perishability constraints [Le et al., 2013], multi-product IRP [Cordeau et al., 2014], single product IRP (basic variant) [Archetti et al., 2007; Solyalı and Süral, 2011; Shiguemoto and Armentano, 2010; Coelho et al., 2012; Archetti et al., 2012; Coelho and Laporte, 2013, 2014; Coelho et al., 2014; Adulyasak et al., 2014b; Archetti et al., 2014; Santos et al., 2016; Desaulniers et al., 2016; Alvarez et al., 2017], among other variants [Andersson et al., 2010; Coelho et al., 2014]. Observe that the basic variant of the IRP addressed in this paper has received an increasing attention in the last years.

The remainder of this paper is organized as follows. In Section 2, we describe the characteristics of the addressed IRP. In Section 3, we present the solution method proposed to solve the problem. The results of the computational experiments with the algorithm are described in Section 4 and the conclusion of this work is presented in Section 5.

2. Problem description

In the IRP the supplier is responsible for delivering the single product to the customers and controlling their inventory levels, as well as its own inventory level. To define the problem, consider the following sets and parameters:

$0:$ the supplier;
$\mathcal{C} =$ $\{1, \ldots, N\}$: set of customers;
$\mathcal{N} =$ $\{0\} \cup \mathcal{C}$: set of facilities;
$\mathcal{T} =$ $\{1, \ldots, T\}$: planning horizon;
$\mathcal{K} =$ $\{1, \ldots, K\}$: set of vehicles;
$d_{it}:$ consumption (demand) of customer $i \in \mathcal{C}$ in period $t \in \mathcal{T}$;
$r_t:$ production quantity in the supplier at period $t \in \mathcal{T}$;
$I_{i0}:$ initial inventory available at the start of the planning horizon in facility $i \in \mathcal{N}$;
$h_{it}:$ unit holding cost at facility $i \in \mathcal{N}$ at period $t \in \mathcal{T}$;
$c_{ij}:$ transportation cost when a vehicle travels from i to j, $\forall\, i, j \in \mathcal{N}$;
$C_i:$ holding capacity of customer $i \in \mathcal{C}$;
$Q:$ capacity of the vehicles (homogeneous fleet);

The consumption (demand) of the product (d_{it}) is the minimum amount of product that the supplier must guarantee to be available at the customer i in time period t. A quantity larger than d_{it} may be delivered before or in period t and the surplus of product can be stocked to meet future demands. Each vehicle can perform a single route each time period and no delivery splitting is allowed inside the period. All vehicle routes must originate from and end at the supplier.

The IRP consists of determining when to visit each customer, how much to deliver in each visit and how to combine those deliveries into feasible routes in each time period. We address two objective functions. The first minimizes the sum of the transportation costs and inventory holding costs at the supplier and at the customers, i.e.,

$$\min \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{k\in\mathcal{K}}\sum_{t\in\mathcal{T}} c_{ij}x_{ij}^{kt} + \sum_{i\in\mathcal{N}}\sum_{t\in\mathcal{T}} h_{it}I_{it},$$

where x_{ij}^{kt} is binary value that is equal to 1 if and only if vehicle k goes from i to j in time period t; and I_{it} is the inventory level of i at time period t. The second objective function consists of minimizing the logistic ratio, given by the total transportation costs divided by the total quantity delivered through the planning horizon:

$$\min \frac{\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}}\sum_{k\in\mathcal{K}}\sum_{t\in\mathcal{T}} c_{ij}x_{ij}^{kt}}{\sum_{i\in\mathcal{C}}\sum_{t\in\mathcal{T}} q_{it}},$$

where q_{it} is the total quantity delivered to customer i at time period t. Holding costs are not included in this objective. For mathematical formulations of the IRP see Coelho and Laporte [2014].

3. Solution method

In this section, we introduce the metaheuristic algorithm proposed to solve the IRP. First, we detail a construction heuristic used as source of good feasible solutions and then we present the ILS algorithm and their components.

3.1. Construction heuristic

A construction heuristic was proposed to provide initial solutions for the metaheuristic. In each iteration, the heuristic separates the decisions of the problem into two phases. In the first phase, the heuristic defines which customers to visit and how much to deliver to them. Then, in the second phase feasible delivery routes are defined to visit the customers selected in the first phase.

The heuristic is based on the inventory levels I_{it} of each customer $i \in \mathcal{C}$ at each time period $t \in \mathcal{T}$. Inventory levels are first computed at the beginning of the heuristic, for all $i \in \mathcal{C}$ and $t \in \mathcal{T}$. Then, they are updated at the end of each iteration based on the deliveries of the routes obtained in the iteration.

There is one iteration of the heuristic for each time period t, starting from $t = 1$. In the first phase of iteration t, to determine the customers that will be visited and the quantities that will be delivered to them, the heuristic separates the customers into two sets. The first set is composed by those customers for which stockout will occur if they are not visited in the current period. The remaining customers are selected to enter in the second set based on the degree of urgency of the customer (if a stockout may occur in the following time periods) and the potential profitability of the visit. Then, in the second phase of the iteration a standard nearest-neighbor insertion heuristic is used to determine feasible routes for the customers of these sets. The resulting routes are used to update the inventory levels at the end of the current time period. If the updated level I_{it} of at least one customer in the first set remains negative, then the heuristic terminates with no solution. Otherwise, a new iteration is started for the next time period $(t+1)$, until the end of the time horizon.

3.2. Iterated local search

ILS is a metaheuristic that applies a local search algorithm to solutions resulting from the perturbation of the previously visited search point, leading to a randomized walk on the space of local optimal solutions [Lourenço et al., 2003; Stützle, 2006; Alvarez and Munari, 2016]. A basic

ILS algorithm is composed by a construction heuristic, a local search procedure and a perturbation mechanism.

We propose an ILS algorithm that uses the construction heuristic of Section 3.1 as source of initial solutions, a multi-start randomized variable neighborhood descent (RVND) heuristic to perform the local search and a multi-operator algorithm as perturbation mechanism. Finally, as acceptance criterion the heuristic keeps the reached solution only if its objective value is better than the current best solution. A pseudo-code of the proposed ILS is shown in Algorithm 1.

Algorithm 1: Iterated local search

input : Instance, parameters;
output: Best solution S^*;
begin
  $S_0 \leftarrow$ Construct initial solution;
  if $S_0 \neq \emptyset$ **then**
    $S^* \leftarrow multi\text{-}RVND(S_0)$;
    while *stop criterion is not met* **do**
      $S^{'} \leftarrow perturb(S^*)$;
      $S^{'} \leftarrow multi\text{-}RVND(S^{'})$;
      if $f(S^{'}) < f(S^*)$ **then** $S^* \leftarrow S^{'}$;
    end
  end
end

For the local search procedure, a variable neighborhood descent heuristic [Mladenovic and Hansen, 1997] with random neighborhood ordering is used. This heuristic explores different neighborhoods of the current incumbent solution by applying a set of local search operators. In our implementation, all operators exhaustively explore the search space using the best improvement strategy and only allowing feasible solutions through the search process.

Seven local search operators are used by the heuristic. The first three are routing operators and the others are inventory-routing operators, as follows: *Or-opt(k)*, $k \in \{1, 2, 3\}$; *Shift(k)*, $k \in \{1, 2, 3\}$; *Swap(k_1, k_2)*, $k_1, k_2 \in \{1, 2\}, k_1 \geq k_2$; *Increase/reduce deliveries*; *Merge visits*; *Transfer visits* and *Insert visits*.

Since RVND is a non-deterministic algorithm (local optimal solutions can be different for distinct local search operators and random ordering of the operators), we used a multi-start RVND heuristic to boost the intensification of the local search and reduce the variability of the results. The number of times the algorithm repeats the process is controlled by the parameter *maxIterRVND*. We set to 5 its value which, according to prior tests, provides an adequate balance between solution quality and computational effort.

Perturbation must be applied to escape from local optimal solutions. The performance of the ILS is strongly dependent on strength of the perturbation applied because it defines much of the behavior of the metaheuristic. Thus, given the multiple decisions made simultaneously on the IRP, a perturbation mechanism composed by four (distinct) operators is used in the ILS. The operators are: *Random shift of visits*; *Random delivery reductions*; *Random insertion of visits*; and *Random split of deliveries*. Each call to the perturbation algorithm activates a single operator chosen at random until one of them changes at least one element of the solution. A maximum of $max\_perturb$ elements of the solution can be perturbed. The value of this parameter was calibrated using the ParamILS algorithm of Hutter et al. [2009], which determined that the most suitable value for this parameter was 7. This configuration was adopted in all the subsequent experiments.

Finally, it is worth mentioning that Santos et al. [2016] also proposed an ILS-based hybrid method for the IRP, with similar local search and perturbation algorithms. Different to their approach, our ILS metaheuristic applies the local search heuristic in a multi-start approach to explore more intensively and effectively the search space and no mathematical programming component is used in our method.

4. Computational experiments

This section describes the computational experiments carried out to evaluate the performance of the proposed ILS to solve the IRP. The algorithm was coded using C++ language and run on an Intel Core i7-2600 3.4 GHz processor and 16 GB of RAM.

The metaheuristic was tested on the benchmark instances proposed by Archetti et al. [2007] for the single-vehicle IRP. This set is composed of 160 instances involving from 5 to 50 customers and grouped into four sets: H3, L3, H6 and L6. H (L) means that the inventory holding costs in the instances are high (low) in comparison with the travel costs, while the number 3/6 indicates the number of time periods in the instances of the set. The travel costs correspond to Euclidean distances rounded to the nearest integer. We use two to five vehicles in addition to the single-vehicle case, resulting in 800 instances. The original vehicle capacity is divided by the number of available vehicles and then rounded to the nearest integer when the objective function is the total cost and, following Archetti et al. [2016], rounded to the nearest lower integer when the logistic ratio is used as an objective function.

In the first computational experiment, we analyze the performance of the proposed metaheuristic algorithm according to different running time limits when minimizing the total cost. As the metaheuristic has random components, each instance was run five times for a time limit of 5 and 30 seconds. It is worth mentioning that longer running times were tested, but no significant gains were achieved compared to the additional computational effort. Table 1 summarizes the best out of five results within 5 and 30 seconds. The first three columns in the table give the name of the instance set, the number of available vehicles and the total number of instances in the group, respectively. We grouped the instances of each set according to the number of vehicles, so that each row in the table represents the average over all instances in the corresponding group (same set and same number of vehicles). For both time limits, the table shows the value of the objective function of the solutions (total cost); the time required to find the best solution (time to best); the optimality gap (opt gap), measured as the relative difference to the lower bounds (LB) reported by the BPC algorithm of Desaulniers et al. [2016] and the B&C of Coelho and Laporte [2014]; and the relative difference between the results obtained by the method and the best upper bounds (UB) reported in the literature (best UB gap). We took the best UBs provided by the Adaptive Large Neighborhood Search (ALNS)-based hybrid method of Adulyasak et al. [2014b], the B&C of Adulyasak et al. [2014a] and the previously mentioned exact methods. The average of each column is shown in the last row of the tables (avg).

The results of the experiment show that for both time limits the metaheuristic can find good feasible solutions, with and approximated average optimality gap of 2%. Notice that the time to find the best solution is very short when compared to the time limit and, as expected, it tends to increase with the number of vehicles and periods. The average optimality gaps of the solutions also tend to increase with the number of vehicles and periods. However, as many LBs may not correspond to optimal solutions this conclusion must be taken cautiously. Negative average gaps to the best UBs indicate that, on average, the solutions found are better than the best solutions reported in the literature. Specifically, 66 new best known solutions were found. Finally, it can be observed that the optimality gaps are quite low considering the relatively short running times.

Regarding the characteristics of the instances, notice that when the number of vehicles increases for the same instance set, the average total cost also increases as a result of larger total travel costs. The average time to find the best solution of the algorithms also increases with the number of vehicles as a consequence of the enlargement of the search space.

To further analyze the performance of the metaheuristic regarding different characteristics of the instances, Table 2 shows the detailed results (considering the average of the five executions for

Set	nV	# of instances	5 seconds total cost	time to best	opt gap[a]	best UB gap[b]	30 seconds total cost	time to best	opt gap[a]	best UB gap[b]
H3	1	50	9279.03	0.40	0.43%	0.43%	9280.45	2.07	0.37%	0.37%
	2	50	9625.80	0.84	0.54%	0.54%	9612.83	5.35	0.33%	0.33%
	3	50	10081.15	1.01	0.66%	0.59%	10060.74	5.17	0.47%	0.40%
	4	50	10569.55	1.44	0.90%	0.54%	10538.93	7.15	0.67%	0.32%
	5	50	11012.82	1.76	1.10%	0.03%	10993.96	7.27	0.97%	-0.10%
L3	1	50	3026.58	0.67	1.62%	1.62%	3019.71	3.11	1.41%	1.41%
	2	50	3420.15	1.06	3.20%	3.18%	3393.57	6.57	2.32%	2.31%
	3	50	3865.27	1.25	3.14%	2.80%	3850.93	5.62	2.82%	2.48%
	4	50	4315.54	1.54	2.51%	1.35%	4302.44	6.43	2.24%	1.10%
	5	50	4767.22	1.51	3.33%	0.63%	4734.09	8.52	2.69%	0.03%
H6	1	30	13001.29	2.15	0.71%	0.71%	12999.55	11.21	0.72%	0.72%
	2	30	14071.98	2.90	1.72%	1.65%	14005.35	15.23	1.24%	1.17%
	3	30	15299.94	3.07	2.34%	1.67%	15256.78	17.02	2.18%	1.51%
	4	30	16617.15	3.35	2.99%	-0.07%	16539.82	14.06	2.50%	-0.55%
	5	29	18326.07	3.17	3.71%	-0.53%	18219.63	18.22	3.19%	-1.02%
L6	1	30	5902.19	2.33	1.91%	1.91%	5898.11	6.71	1.72%	1.72%
	2	30	6915.65	3.01	3.12%	2.88%	6856.92	15.24	2.32%	2.07%
	3	30	8189.08	2.84	4.65%	3.14%	8112.11	15.14	3.79%	2.30%
	4	30	9476.69	2.98	5.09%	0.53%	9400.22	17.99	4.32%	-0.19%
	5	29	11003.49	2.89	6.23%	-3.18%	10936.91	13.68	5.66%	-3.70%
avg			9438.33	2.01	2.50%	1.02%	9400.65	10.09	2.10%	0.63%

[a] Best LB from: Desaulniers et al. [2016] (*BPC*) and Coelho and Laporte [2014] (*B&C*).
[b] Best UB from: Desaulniers et al. [2016] (*BPC*), Coelho and Laporte [2014] (*B&C*) and Adulyasak et al. [2014a] (*B&C* & ALNS).

Table 1: Best results of the metaheuristic.

the 30 seconds time limit) for the instance sets H3 and L6 with one and five vehicles, respectively. These instances represent extreme features of all tested instances, namely the easiest (L3, with only one vehicle) and most difficult (H6, with 5 vehicles). It is worth highlighting that all solutions for the instances with one single vehicle were solved to optimality by exact methods [Desaulniers et al., 2016; Coelho and Laporte, 2014]. The results are grouped according to the number of customers (nC). Notice that the average time required to find the obtained solutions (time to best) and the optimality gaps (opt gap) increase with the number of customers in the instance. Finally, observe the negative average gap to the best UBs for the larger instances (H6 with 25 and 30 customers).

Set	nV	nC	# of instances	total cost	time to best	opt gap	best UB gap
H3	1	5-10	10	3280.14	0.01	0.38%	0.38%
		15-20	10	6339.82	0.94	0.13%	0.13%
		25-30	10	10005.15	3.40	0.53%	0.53%
		35-40	10	12091.18	2.75	0.95%	0.95%
		45-50	10	14802.14	6.55	1.14%	1.14%
L6	5	5-10	9	8944.13	7.52	5.12%	4.10%
		15-20	10	11225.25	16.94	7.63%	2.86%
		25-30	10	12930.16	19.75	8.76%	-13.10%

Table 2: Detailed results for the some set of instances.

We compared the results of our metaheuristic against the ALNS-based hybrid method of Adulyasak et al. [2014b], as this is the only heuristic method whose results are publicly available. Table 3 reports the average relative differences between the results found by the ILS (best out of five runs within 30 seconds) and the results of Adulyasak et al. [2014b]. The results are grouped according to the number of customers (nC). It can be observed that the metaheuristic outperforms the ALNS-based hybrid method on average, as only negative relative differences are presented for the instance sets. Unfilled cells (–) are due to incomplete report of the results of the ALNS-based hybrid method.

Set	nC	2 veh	3 veh	4 veh
H3	5-50	-1.88%	-3.29%	-3.00%
L3	5-50	-1.75%	-3.99%	-6.83%
H6	5-25	-1.57%	-0.66%	–
L6	5-25	-3.03%	-1.79%	–
avg		-2.06%	-2.43%	-4.91%

Table 3: Relative differences to the ALNS of Adulyasak et al. [2014b].

Regarding the performance of the metaheuristic when addressing the IRP with logistic ratio as objective function, Table 4 shows the relative gaps of the best (out of five) results found by ILS within 30 seconds with respect to the optimal logistic ratios reported by Archetti et al. [2016]. In that paper, the authors solved the instance set H3 (three time periods and high inventory holding cost) and extended the instances of this set to four and five time periods, resulting in sets H4 and H5, respectively. The instance are grouped based on the number of vehicles (nV) and the number of customers (nC), given in the first two columns. For each instance set (H3, H4 and H5), column lr* gives the average of the optimal logistic ratios over all instances in the group and column ILS gives the average relative difference of the logistic ratios obtained by ILS in relation to lr*. The results show that the metaheuristic is able to find optimal or near optimal solutions for most of the analyzed instances, demonstrating the ability of the developed algorithm to also address the logistic ratio as the objective function in the IRP. For the instances solved in the sets H3, H4 and H5, ILS found solutions with average logistic ratio of 2.77, 3.39 and 3.41, respectively. Unfilled cells (–) are due to that Archetti et al. [2016] could not find the optimal solutions for those instances.

nV	nC	H3		H4		H5	
		lr*	ILS	lr*	ILS	lr*	ILS
1	5	2.54	0.79%	2.66	1.13%	2.59	10.42%
	10	1.42	0.00%	1.43	0.70%	1.42	0.70%
	15	1.15	0.87%	–	–	–	–
2	5	3.18	1.57%	3.28	0.91%	3.19	2.19%
	10	1.86	2.15%	1.87	0.53%	1.86	1.61%
	15	1.41	2.13%	–	–	–	–
3	5	4.19	0.00%	4.15	2.41%	4.26	2.11%
	10	2.35	0.85%	2.27	2.20%	2.34	3.85%
	15	1.68	0.60%	–	–	–	–
4	5	5.06	1.38%	5.34	1.69%	5.31	3.01%
	10	2.80	0.00%	2.78	2.16%	2.81	2.49%
	15	1.97	2.54%	–	–	–	–
5	5	6.02	1.00%	6.38	1.41%	6.10	2.13%
	10	3.26	1.23%	3.27	1.22%	3.29	1.52%
	15	2.29	1.31%	–	–	–	–
avg		2.75	1.09%	3.34	1.44%	3.32	3.01%

Table 4: Comparison to the optimal logistic ratios.

5. Conclusions

In this paper, we presented a metaheuristic algorithm based on Iterated Local Search (ILS) to solve the inventory routing problem. A construction heuristic and a multi-start randomized variable neighborhood descent heuristic are used in the algorithm. Two different objective functions were addressed, namely, total cost minimization and logistic ratio minimization. All the experiments were based on over 800 instances from the literature. The results showed that the proposed algorithm can effectively handle both objective functions and provide good feasible solutions in short running times. New best known solutions were found for 66 instances when using the first objective function.

An interesting perspective for future research is to extend the metaheuristic approach to deal with richer IRP variants, such as IRP with time windows and/or with heterogeneous fleet. Other interesting lines of research are to combine this metaheuristic with mathematical programming components yielding effective hybrid methods.

Acknowledgments

This research was partially supported by CAPES, CNPq and FAPESP (grant numbers 2016/23366-9 and 2017/06664-9).

References

Abdelmaguid, T. F., Dessouky, M. M., and Ordóñez, F. (2009). Heuristic approaches for the inventory-routing problem with backlogging. *Computers & Industrial Engineering*, 56(4):1519–1534.

Adulyasak, Y., Cordeau, J.-F., and Jans, R. (2014a). Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems. *INFORMS Journal on Computing*, 26(1):103–120.

Adulyasak, Y., Cordeau, J.-F., and Jans, R. (2014b). Optimization-based adaptive large neighborhood search for the production routing problem. *Transportation Science*, 48(1):20–45.

Alvarez, A. and Munari, P. (2016). Metaheuristic approaches for the vehicle routing problem with time windows and multiple deliverymen. *Journal of Management & Production (Gestão e Produção)*, 23(2):279–293.

Alvarez, A., Munari, P., and Morabito, R. (2017). Iterated local search and simulated annealing algorithms for the inventory routing problem. Technical report, Industrial Engineering Department, Federal University of São Carlos.

Andersson, H., Hoff, A., Christiansen, M., Hasle, G., and Løkketangen, A. (2010). Industrial aspects and literature survey: Combined inventory management and routing. *Computers & Operations Research*, 37(9):1515–1536.

Archetti, C., Bertazzi, L., Hertz, A., and Speranza, M. G. (2012). A hybrid heuristic for an inventory routing problem. *INFORMS Journal on Computing*, 24(1):101–116.

Archetti, C., Bertazzi, L., Laporte, G., and Speranza, M. G. (2007). A branch-and-cut algorithm for a vendor-managed inventory-routing problem. *Transportation Science*, 41(3):382–391.

Archetti, C., Bianchessi, N., Irnich, S., and Speranza, M. G. (2014). Formulations for an inventory routing problem. *International Transactions in Operational Research*, 21(3):353–374.

Archetti, C., Desaulniers, G., and Speranza, M. G. (2016). Minimizing the logistic ratio in the inventory routing problem. *EURO Journal on Transportation and Logistics*, pages 1–18.

Archetti, C. and Speranza, M. G. (2016). The inventory routing problem: The value of integration. *International Transactions in Operational Research*, 23(3):393–407.

Bell, W. J., Dalberto, L. M., Fisher, M. L., Greenfield, A. J., Jaikumar, R., Kedia, P., Mack, R. G., and Prutzman, P. J. (1983). Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer. *Interfaces*, 13(6):4–23.

Campbell, A. M. and Savelsbergh, M. (2004). A decomposition approach for the inventory-routing problem. *Transportation Science*, 38(4):488–502.

Coelho, L. C., Cordeau, J.-F., and Laporte, G. (2012). Consistency in multi-vehicle inventory-routing. *Transportation Research Part C: Emerging Technologies*, 24:270–287.

Coelho, L. C., Cordeau, J.-F., and Laporte, G. (2014). Thirty years of inventory routing. *Transportation Science*, 48(1):1–19.

Coelho, L. C. and Laporte, G. (2013). A branch-and-cut algorithm for the multi-product multi-vehicle inventory-routing problem. *International Journal of Production Research*, 51(23):7156–7169.

Coelho, L. C. and Laporte, G. (2014). Improved solutions for inventory-routing problems through valid inequalities and input ordering. *Intern. Journal of Production Economics*, 155:391–397.

Cordeau, J.-F., Laganà, D., Musmanno, R., and Vocaturo, F. (2014). A decomposition-based heuristic for the multiple-product inventory-routing problem. *Computers & Operations Research*, 55(2):153–166.

Desaulniers, G., Rakke, J. G., and Coelho, L. C. (2016). A branch-price-and-cut algorithm for the inventory-routing problem. *Transportation Science*, 50(3):1060–1076.

Dror, M. and Ball, M. (1987). Inventory/routing: Reduction from an annual to a short-period problem. *Naval Research Logistics (NRL)*, 34:891–905.

Dror, M., Ball, M., and Golden, B. (1985). A computational comparison of algorithms for the inventory routing problem. *Annals of Operations Research*, 4(1):1–23.

Hutter, F., Hoos, H. H., Leyton-Brown, K., and Stützle, T. (2009). ParamILS: an automatic algorithm configuration framework. *Journal of Artificial Intelligence Research*, 36(1):267–306.

Le, T., Diabat, A., Richard, J.-P., and Yih, Y. (2013). A column generation-based heuristic algorithm for an inventory routing problem with perishable goods. *Optimization Letters*, 7(7):1481–1502.

Lourenço, H., Martin, O., and Stützle, T. (2003). Iterated local search. In Glover, F. and Kochenberger, G. A., editors, *Handbook of Metaheuristics*, pages 321–353. Springer.

Mladenovic, N. and Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24:1097–1100.

Santos, E., Ochi, L., Simonetti, L., and González, P. (2016). A hybrid heuristic based on iterated local search for multivehicle inventory routing problem. *Electronic Notes in Discrete Mathematics*, 52:197–204.

Savelsbergh, M. and Song, J.-H. (2007). Inventory routing with continuous moves. *Computers & Operations Research*, 34(6):1744–1763.

Savelsbergh, M. and Song, J.-H. (2008). An optimization algorithm for the inventory routing problem with continuous moves. *Computers & Operations Research*, 35(7):2266–2282.

Shiguemoto, A. L. and Armentano, V. A. (2010). A tabu search procedure for coordinating production, inventory and distribution routing problems. *International Transactions in Operational Research*, 17(2):179–195.

Solyalı, O. and Süral, H. (2011). A branch-and-cut algorithm using a strong formulation and an a priori tour-based heuristic for an inventory-routing problem. *Transportation Science*, 45(3):335–345.

Stützle, T. (2006). Iterated local search for the quadratic assignment problem. *European Journal of Operational Research*, 174(3):1519–1539.