DOI: 10.1016/j.orl.2021.09.007

Operations Research Letters 49 (2021) 874–876

Contents lists available at ScienceDirect

Operations Research Letters

www.elsevier.com/locate/orl

Improved approximation for the capacitated inventory access point problem ☆

Miguel Ángel Marfurt Alarcón *, Lehilton Lelis Chaves Pedrosa

Institute of Computing, Av. Albert Einstein, 1251 – Cidade Universitária, Campinas, 13083-852, São Paulo, Brazil

ARTICLE INFO

Article history:
Received 21 May 2021
Received in revised form 7 September 2021
Accepted 9 September 2021
Available online 30 October 2021

Keywords:
Approximation algorithms
Inventory routing
Inventory access point
Facility location

ABSTRACT

The Inventory Access Point (IAP) is the single-item lot-sizing problem where a single customer faces demands in a discrete planning horizon, and the goal is to find a replenishment policy that minimizes the total inventory and ordering costs. While the uncapacitated version is polynomial, only a 3-approximation is known for the capacitated case. We improve this factor to 2.619 and, as a byproduct, we also improve the best factor for SIRPFL, which is a variant with multiple depots and customers.

1. Introduction

Inventory and routing problems are observed in many sectors of the industry, as inventory management and delivery planning normally contribute to a large share of a company's operating costs. Since both types of decision are highly correlated, unified models are used to minimize all costs at once [12]. The Inventory Routing Problems (IRPs) form a class of problems that integrate inventory and routing variables [2]. In such a model, given a set of depots and a fleet of vehicles, the goal is to find a best delivery policy to serve a set of customers. Each customer faces demands for items over a discrete planning horizon and has associated storage costs.

Even when considering decisions of inventory or routing individually, several studied problems are NP-hard, such as Lot-Sizing Problem [13], the Joint Replenishment Problem [7], the Facility Location Problem [10], the Vehicle Routing Problem [6], among others. Thus, many works on unified problems have considered heuristics or approximation algorithms. For general versions of IRP, when the routing cost is a function of the set the participating customers, there are sub-logarithm approximation algorithms [3,11], and it is open whether such versions admit constant-factor approximations. Recent studies have focused on designing constant-factor approximation for special cases of IRP [4,5,8,9].

One of these simplifications is the Star Inventory Routing Problem with Facility Location (SIRPFL), introduced by Jiao and Ravi [9]. This problem allows for multiple depots, but assumes that trips are direct connections from the depot to the associated customers. The special case with only a single depot and a single customer is known as the Inventory Access Point (IAP) and is of particular importance, as an algorithm for IAP can be used as a sub-routine for SIRPFL [4].

1.1. Problem definitions

In SIRPFL, we are given a metric space V with distance function w and a number T of periods indexed from 1 to T. There is a subset $\mathcal{F} \subseteq V$ representing depots (a.k.a. facilities) and a subset $\mathcal{N} \subseteq V$ representing customers. For each customer i and period t, there is a demand for $d_{it} \geq 0$ items that must be shipped from an open depot j and delivered at i in a period $s \leq t$. There are three costs involved: (i) the cost to open a depot j is $f_j \geq 0$, (ii) the cost to ship items from an open depot j to a customer i is $w(j,i) \geq 0$, and (iii) the cost to hold a unit of demand from period s to period t in the inventory of customer i is $h^i_{st} \geq 0$. The goal is to open a subset of depot as well as determining which customers are served by each open depot in each period, while minimizing the total opening, shipment and holding costs.

We assume that holding costs are monotonic, that is, for each customer i and any three periods $r \leq s \leq t$, we have $h^i_{rt} \geq h^i_{st}$. This implies that zero-inventory policies are optimal for the uncapacitated case, i.e., a trip for a customer is only placed when there are no items in its inventory [1]. In the *Uncapacitated SIRPFL*, we as-

☆ This project was supported by São Paulo Research Foundation (FAPESP) grants #2015/11937-9, and National Council for Scientific and Technological Development (CNPq) grants #425340/2016-3, #422829/2018-8, and #312186/2020-7.

* Corresponding author.
E-mail address: miguel.alarcon@students.ic.unicamp.br (M.Á. Marfurt Alarcón).

https://doi.org/10.1016/j.orl.2021.09.007
0167-6377/

Table 1
Improved approximation ratios for IAP and SIRPFL.

Problem	Previous work	This paper
Capacitated Splittable IAP	3 [9]	2.619
Capacitated Unsplittable IAP	6 [9]	4.562
Capacitated Splittable SIRPFL	3.236 [4]	2.905
Capacitated Unsplittable SIRPFL	6.029 [4]	4.649

sume that a trip can include any number of items, while in the *Capacitated SIRPFL*, each trip must include at most U items. The capacitated variant has two forms: in the *Capacitated Splittable SIRPFL*, the d_{it} items demanded at customer i in period t can be split into multiple trips, whereas in the *Capacitated Unsplittable SIRPFL*, all d_{it} items must be delivered by a single trip.

IAP is the special case of SIRPFL with a single customer and a single depot. Thus, we only need to decide in what periods we place trips, and how many items we drop off in these periods. For IAP, we can assume without loss of generality that all demands (for all periods) are strictly positive. As in the case of SIRPFL, there are three variants of IAP: *Uncapacitated IAP*, *Capacitated Splittable IAP*, and *Capacitated Unsplittable IAP*.

1.2. Related works

The uncapacitated version of IAP reduces to the classical single-item lot-sizing problem, that can be solved by a dynamic programming algorithm in polynomial time [13]. Jiao and Ravi [9] showed that the capacitated unsplittable version is NP-hard, and it is currently open whether the splittable case is also NP-hard. They gave a 3-approximation for Capacitated Splittable IAP, which implies a 6-approximation for Capacitated Unsplittable IAP, which has currently the best known factor. Later, they used the algorithms for IAP to give LP-rounding approximation algorithms the Uncapacitated, Capacitated Splittable and Capacitated Unsplittable versions of SIRPFL, that achieve approximation factors of 12, 24 and 48, respectively.

Subsequently, Byrka and Lewandowski [4], also using algorithms for IAP as sub-routines, reduce SIRPFL to a concave cost variant of Uncapacitated Facility Location (UFL), leading to a 1.488-approximation for Uncapacitated SIRPFL, a 3.236-approximation for Capacitated Splittable SIRPFL, and a 6.029-approximation for Capacitated Unsplittable SIRPFL.

1.3. Our results

In this work, we improve the approximation factor for Capacitated Splittable IAP from 3 to 2.619. As a consequence, we also improve the approximation factor for Capacitated Unsplittable IAP, Capacitated Splittable SIRPFL, and Capacitated Unsplittable SIRPFL. The key idea for the improvement is modifying the 3-approximation by Jiao and Ravi, leading to a tighter analysis of the approximation factor. A summary of our results is presented in Table 1.

The rest of the text is organized as follows. First, in Section 2, we review the 3-approximation by Jiao and Ravi for Capacitated Splittable IAP. Then, in Section 3, we present our improvement of the algorithm for Capacitated Unsplittable IAP and discuss the implications of these changes for Capacitated Unsplittable IAP and the capacitated versions of SIRPFL.

2. The 3-approximation by Jiao and Ravi

Let W be the distance between the unique depot and the unique customer, d_t be the customer demand in period t, h_{st} be the cost to hold one item at the costumer from period s to t, and $H_{st} = h_{st}d_t$. The Capacitated Splittable IAP can be formulated as the following mixed integer linear program, where variable x_{st} represents the fraction of demand d_t that is delivered by a trip in period s, and y_s indicates the number of trips on period s.

$$\min \sum_{s=1}^{T} W y_s + \sum_{t=1}^{T}\sum_{s=1}^{t} H_{st}x_{st}$$

$$\text{s.t.} \quad \sum_{s=1}^{t} x_{st} \geq 1, \qquad t \leq T, \tag{1}$$

$$y_s \geq \sum_{t=s}^{T} \frac{x_{st}d_t}{U}, \qquad s \leq T, \tag{2}$$

$$y_s \geq x_{st}, \qquad t \leq T, s \leq t, \tag{3}$$

$$x_{st} \geq 0, \qquad t \leq T, s \leq t, \tag{4}$$

$$y_s \in \mathbb{Z}^+, \qquad s \leq T. \tag{5}$$

Constraints (1) ensure that all demands must be satisfied before its expiration period. Constraints (2) provide a lower bound on the number of trips that can be placed considering the vehicle's capacity. Constraints (3) are valid inequalities for the model since it imposes the condition that if the customer is not visited once in the period s, then no demand can be satisfied in period s.

Let (x, y) be an optimal solution of the linear relaxation of the formulation, and, for each $t \leq T$, define s_t as the maximum period such that $\sum_{s=s_t}^{t} x_{st} \geq \frac{1}{2}$. The LP-rounding procedure proposed by Jiao and Ravi is given in Algorithm 1.

Algorithm 1 Visit rule for Capacitated Splittable IAP.

```
1: Initialize A ← ∅, S ← ∅
2: while there is any unsatisfied demand do
3:    Denote by t the unsatisfied demand period with the latest s_t
4:    A ← A ∪ {t}
5:    S ← S ∪ {s_t}
6:    Satisfy t by dropping off d_t items on period s_t
7:    for unsatisfied demand period t̂ ≥ s_t do
8:       satisfy t̂ by dropping off d_t̂ items on period s_t
9:    end for
10: end while
11: return the visit set S
```

Note that each demand $t \in A$ is satisfied in s_t (these demands are called anchors) and all demands $\hat{t}$ that have been satisfied in s_t have $s_{\hat{t}} \leq s_t$, since otherwise $\hat{t}$ would have been processed before t. Also, note that all intervals of the form $[s_t, t]$ for an anchor $t \in A$ are disjoint, by the construction of the algorithm.

Jiao and Ravi [9] showed that the solution produced by the algorithm has holding cost at most 2 times the holding cost in the objective function of the linear relaxation, and the routing cost at most 3 times the routing cost in the objective function, thus implying that their algorithm is a 3-approximation. To derive an improved approximation, we would like to balance the factors that multiply holding and routing costs. Intuitively, one might want to decrease the routing cost, while possibly increasing the holding cost.

3. An improved 2.619-approximation

In the improved approximation, we use the same LP-rounding procedure given in Algorithm 1, but define s_t as the maximum period such that $\sum_{s=s_t}^{t} x_{st} \geq \alpha$ for some parameter $0 < \alpha < 1$. Thus, the original algorithm corresponds to the choice of $\alpha = 1/2$.

In the following, we analyze the modified algorithm. Let (x, y) be an optimal solution of the relaxation. First, we bound the holding cost.

Lemma 1. *The holding cost of the solution is at most* $\frac{1}{1-\alpha}\sum_{t=1}^{T}\sum_{s=1}^{t} H_{st}x_{st}$.

Proof. Consider the demand of some period t. We first analyze the case that $t \in A$. It follows that t is served on period s_t, incurring a holding cost of $H_{s_t t}$. Since $\sum_{s=1}^{t} x_{st} = 1$, we have $\sum_{s=1}^{s_t} x_{st} \geq 1-\alpha$, as otherwise s_t would not be maximal. Since H_{st} is monotonic, this implies

$$H_{s_t t}(1-\alpha) \leq H_{s_t t}\sum_{s=1}^{s_t} x_{st} \leq \sum_{s=1}^{s_t} H_{st}x_{st} \leq \sum_{s=1}^{t} H_{st}x_{st}. \tag{6}$$

Now, we assume that $t \notin A$. Let s' be the last period in S such that $s' \leq t$. Then, the holding cost of t is $H_{s't}$. Since t was not chosen to be an anchor, we know that $s_t \leq s'$. Again we have $\sum_{s=1}^{s_t} x_{st} \geq 1-\alpha$, then $\sum_{s=1}^{s'} x_{st} \geq 1-\alpha$. Thus, as before, we have

$$H_{s't}(1-\alpha) \leq H_{s't}\sum_{s=1}^{s'} x_{st} \leq \sum_{s=1}^{s'} H_{st}x_{st} \leq \sum_{s=1}^{t} H_{st}x_{st}. \tag{7}$$

By adding up (6) or (7) for every period t, the lemma follows. □

The routing cost is bounded by the following lemma.

Lemma 2. *The routing cost of the solution is at most* $(1+\frac{1}{\alpha})\sum_{s=1}^{T} W y_s$.

Proof. Let D^s be the set of periods satisfied in the period s by the Algorithm 1. The number of trips in a period $s \in S$ can be computed as

$$n(s) = \left\lceil \sum_{t\in D^s} \frac{d_t}{U} \right\rceil \leq \sum_{t\in D^s} \frac{d_t}{U} + 1. \tag{8}$$

Thus, the total number of trips in the solution is

$$n(S) = \sum_{s\in S} n(s) \leq \sum_{s\in S}(\sum_{t\in D^s} \frac{d_t}{U} + 1) = \sum_{s\in S}\sum_{t\in D^s} \frac{d_t}{U} + |S|$$
$$= \sum_{t=1}^{T} \frac{d_t}{U} + |S|. \tag{9}$$

From constraints (1) and (2), we can bound the first term in the right side of (9) as

$$\sum_{s=1}^{T} y_s \geq \sum_{s=1}^{T}\sum_{t=s}^{T} \frac{d_t}{U}x_{st} = \sum_{t=1}^{T}\sum_{s=1}^{t} \frac{d_t}{U}x_{st} = \sum_{t=1}^{T} \frac{d_t}{U}. \tag{10}$$

For the second term in the right side of (9), we use $|S| = |A|$. By construction, the intervals $[s_t, t]$ are disjoint for $t \in A$ and $\sum_{s=s_t}^{t} x_{st} \geq \alpha$, thus

$$\frac{1}{\alpha}\sum_{s=1}^{T} y_s \geq \frac{1}{\alpha}\sum_{t\in A}\sum_{s=s_t}^{t} y_s \geq \frac{1}{\alpha}\sum_{t\in A}\sum_{s=s_t}^{t} x_{st} \geq \frac{1}{\alpha}\sum_{t\in A}\alpha = |A| = |S|, \tag{11}$$

where the second inequality follows from constraint (3).

Combining (10) and (11), we get $n(S) \leq (1+\frac{1}{\alpha})\sum_{s=1}^{T} y_s$. Since each trip incurs cost W, the lemma follows. □

Now, consider a (λ_h, λ_r)-approximation for the Capacitated Splittable IAP being a bi-factor approximation algorithm that provides a solution with a cost limited to λ_h times the optimal holding cost plus λ_r times the optimal routing cost. According to Lemmas 1 and 2, the following is immediate.

Lemma 3. *Algorithm 1 is a* $(\frac{1}{1-\alpha}, 1+\frac{1}{\alpha})$*-approximation algorithm for Capacitated Splittable IAP.*

To optimize the approximation factor, we equate the factors of holding and routing costs, by fixing $\alpha = \frac{\sqrt{5}-1}{2} \approx 0.618$. By substituting α in Lemma 3, we obtain the following result.

Theorem 1. *There is a* 2.619*-approximation algorithm for Capacitated Splittable IAP.*

Consequences for related problems Jiao and Ravi [9] showed that it is possible to convert any Capacitated Splittable IAP feasible solution into a Capacitated Unsplittable IAP feasible solution that has the same holding cost and that makes at most twice as many visits as the splittable solution. Combined with Lemma 3, this implies that there is a $(\frac{1}{1-\alpha}, 2+\frac{2}{\alpha})$-approximation algorithm for Capacitated Unsplittable IAP. Optimizing the approximation factors again, by fixing $\alpha = \frac{\sqrt{17}-1}{4} \approx 0.781$, we obtain a 4.562-approximation for Capacitated Unsplittable IAP.

Byrka and Lewandowski [4] considered a variant of Uncapacitated Facility Location called the Per-Client Non-decreasing Concave Connection Cost Facility Location (NCC-FL). They reduced each version of SIRPFL to NCC-FL, by solving instances of the corresponding versions of IAP as a sub-routine to compute the connection costs between each customer and depot. They showed that a ρ-approximation for IAP implies a $\max\{\lambda_f, \rho(1+2e^{-\lambda_f})\}$-approximation for SIRPFL, where $\lambda_f \geq 1$ is a parameter of the algorithm for NCC-FL. By substituting ρ and choosing λ_f appropriately, we improve the approximation factors of Capacitated Splittable SIRPFL and Capacitated Unsplittable SIRPFL to 2.905 and 4.649, respectively.

References

[1] E. Arkin, D. Joneja, R. Roundy, Computational complexity of uncapacitated multi-echelon production planning problems, ORL (1989).
[2] W.J. Bell, L.M. Dalberto, M.L. Fisher, A.J. Greenfield, R. Jaikumar, P. Kedia, R.G. Mack, P.J. Prutzman, Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer, Interfaces (1983).
[3] T. Bosman, N. Olver, Improved approximation algorithms for inventory problems, CoRR (2019).
[4] J. Byrka, M. Lewandowski, Concave connection cost facility location and the star inventory routing problem, arXiv (2019).
[5] M. Cheung, A. Elmachtoub, R. Levi, D. Shmoys, The submodular joint replenishment problem, Math. Program. (2015).
[6] G.B. Dantzig, J.H. Ramser, The truck dispatching problem, Manag. Sci. (1959).
[7] A. Federgruen, Y.-S. Zheng, The joint replenishment problem with general joint cost structures, Oper. Res. (1992).
[8] T. Fukunaga, A. Nikzad, R. Ravi, Deliver or hold: approximation algorithms for the periodic inventory routing problem, in: Leibniz International Proceedings in Informatics, LIPIcs, 2014.
[9] Y. Jiao, R. Ravi, Inventory routing problem with facility location, arXiv (2019).
[10] J. Krarup, P.M. Pruzan, The simple plant location problem: survey and synthesis, Eur. J. Oper. Res. (1983).
[11] V. Nagarajan, C. Shi, Approximation algorithms for inventory problems with submodular or routing costs, Math. Program. (2015).
[12] A. Sarmiento, R. Nagi, A review of integrated analysis of production-distribution systems, IIE Trans. (1999).
[13] H.M. Wagner, T.M. Whitin, Dynamic version of the economic lot size model, Manag. Sci. (1958).