Dynamic multi-attribute inventory routing problem at Renault: dealing with the continental scale

Louis Bouvier[1], Axel Parmentier[1]

CERMICS, Ecole des Ponts, France

{louis.bouvier, axel.parmentier}@enpc.fr

Keywords : *Inventory routing problem, large neighborhood search, rolling horizon, random predictions.*

1 Introduction

The inventory routing problem (IRP) arises when a supplier manages the delivery of commodities to its customers on a multiple-day horizon in a centralized manner. It consists in deciding "who sends what to whom and when" and planning routes to deliver commodities from depots to customers with the objective of minimizing inventory and routing costs. This NP-hard problem has received significant attention in the operations research literature over the past 40 years.

The present paper is motivated by a partnership with Renault, a major European car manufacturer who must routinely solve IRP instances of unprecedented continental scale and complexity as part of their backward logistics. Indeed, they receive car parts from suppliers at their plants in packaging, and reuse the latter, which implies the need for backward packaging logistics. The goal of our partnership is to redesign their IRP algorithm. We embed it into a multi-stage optimization framework. Indeed, only the short-term decisions must be taken on each day in practice, based on the current information of the future.

This context leads to challenges that can be formalized in a stochastic optimization manner, as shown by [6]. However, even the deterministic version of the problem is very challenging due to its combinatorial structure. It prevents the use of stochastic heuristics such as progressive hedging [8], or Lagrangian relaxation [11], which require to solve many deterministic instances. We therefore design a rolling horizon policy [9] solving a unique deterministic instance per step, and leveraging statistical models of predictions. A prerequisite to be able to solve this dynamic IRP is to address the fixed-horizon deterministic IRP. Numerous variants of the inventory routing problem have been studied over the past decades. In our specific IRP, several vehicles can deliver various distinct commodities to clients, starting from several depots. It is thus a multi-depot, multi-vehicle and multicommodity IRP, close to the *multi-attribute* formulation of [5]. In addition, we have the continuous-time structure as detailed by [10]: routes can last several days, which is implied by the European scale. It induces tight links between routing and inventory dynamics, and leads to a more challenging problem. Indeed, the optimal order of a route not only depends on the distances, but also on the delays introduced in the inventory dynamics of the customers involved. Typical heuristics include route-based matheuristics [3], decomposition matheuristics [7], and metaheuristics [2]. Algorithms from the literature do not scale because of 1) the multi-attribute and continuous-time aspects, combined with 2) the scale of the European instances, with 30 commodities, 16 depots, 600 customers on average and a 21 days horizon.

Our contributions are the following. We design a large neighborhood search (LNS) to solve the fixed-horizon deterministic multi-attribute continuous-time IRP. To this end, we derive a new flow relaxation, we generalize dozens of neighborhoods from the routing literature to the continuous-time IRP, we define two new perturbations based on new MILP formulations, and a new large neighborhood deriving a localized multi-attribute generalization of a matheuristic [1]. We show we scale to the European instances through extensive numerical experiments. From this LNS, we define a policy to address the dynamic IRP, leveraging statistical models. This policy is currently being industrialized at Renault. We implement a simulator to evaluate our policy in a realistic framework, and to compare it with the current algorithm used in production at Renault. We highlight the potential gains: thousands of tons of CO_2 and millions of euros per year. Besides, we give access[1] to our open-source Julia package that implements the LNS, together with the library of realistic multi-attribute continuous-time fixed-horizon deterministic IRP instances.

2 Problem statement

General notations and data. Let M be the set of *commodities*, D the set of *depots* and C the set of *customers*, that respectively *release* and *demand* commodities $m \in M$. The *time horizon* is $T \in \mathbb{Z}^+$ days. At the beginning, each *vertex* v (depot or customer) has an *initial inventory* of commodity m denoted by I^0_{mv}. On each day $t \in [T]$, a customer c demands a quantity b^-_{mct} of commodity m. A depot d releases a quantity b^+_{mdt} of commodity m.

A *maximum inventory capacity* κ_{mvt} is set on the night of each day t per vertex v and commodity m. Below this capacity, no inventory cost is paid. Above, a cost is set to $c^{\texttt{exc}}_{mv}$ per unit, where "`exc`" stands for excess. A price $c^{\texttt{short}}_{mc}$ is paid per unit of unsatisfied demand for commodity m of customer c, where "`short`" stands for shortage. It corresponds to a soft constraint of non-negativity for the customers' inventories.

We approximate commodities and vehicles by one-dimensional objects. We associate a length ℓ_m to each commodity $m \in M$. We consider an infinite fleet of homogeneous vehicles of length L, to deliver the commodities from depots to customers. They are not assigned to a particular depot. A 1D bin packing problem must therefore be solved for vehicle loading.

The depots and customers are the vertices $\mathcal{V} = D \cup C$ of a directed graph $\mathcal{D} = (\mathcal{V}, \mathcal{A})$ that we name the *locations graph*. The directed aspect is used to model the fact that transport durations and distances depend on the trip direction. There is an arc $a = (u, v) \in \mathcal{A}$ for each vertex $u \in D \cup C$ and $v \in C$, $v \neq u$. We associate a *distance* Δ_a (in kilometers) and a *transport duration* τ_a (in hours) to each arc. The distances satisfy the triangular inequality. When planning a route, a cost is paid per vehicle $c^{\texttt{veh}}$, per stop (customer visited) $c^{\texttt{stop}}$, and per kilometer travelled $c^{\texttt{km}}$. The number of stops must not exceed S_{max}, which is a practical requirement of the car manufacturer. The *limit of driving hours per day* is $\tau_{\max}$. A route is then the combination of a feasible path in the locations graph, a date of departure, and the quantities of each commodity to be delivered to each customer along the path, respecting the vehicle capacity L.

Dynamic IRP. One difficulty comes from the fact that on day t, we do not know the true demand and release $\mathbf{b}_{t'}$ of the days $t' \geq t$. We only have access to noisy predictions over $H \in \mathbb{Z}^+$ days $\overline{\mathbf{b}^H_t} = (\overline{\mathbf{b}_{t't}})_{t' \in \{t,\ldots,\min(T,t+H)\}}$, made by Renault, and based on expert knowledge.

$$\underbrace{b^-_{mct'}}_{\text{True demand}} = \underbrace{\overline{b^-_{mct't}}}_{\text{Prediction}} + \underbrace{\xi^-_{mct't}}_{\text{Noise}}, \quad \underbrace{b^+_{mdt'}}_{\text{True release}} = \underbrace{\overline{b^+_{mdt't}}}_{\text{Prediction}} + \underbrace{\xi^+_{mdt't}}_{\text{Noise}}. \tag{1}$$

On day t, we reveal $\mathbf{b}_{t-1}$, the true demand and release of day $t-1$, as well as the expert predictions $\overline{\mathbf{b}^H_t}$ for the next H days. Other sources of randomness such as transport random delays or reporting errors are omitted. We model our problem as a multi-stage stochastic optimization problem. The state $\mathbf{X}_t$ on day t contains the routes that start before day t and do not reach their final stop before t, and the inventory at the depots and customers in the evening of $t-1$. The transition from the state $\mathbf{X}_t$ to the next $\mathbf{X}_{t+1}$ is expressed by a function f_t. A policy π decides the set of routes $\pi(\mathbf{X}_t, \overline{\mathbf{b}^H_t})$ that start on day t given the state $\mathbf{X}_t$ and predictions $\overline{\mathbf{b}^H_t}$.

[1] https://github.com/LouisBouvier/InventoryRoutingLNS.jl

The cost C_t paid on day t is the sum of the cost of the routes in $\pi(\mathbf{X}_t, \overline{\mathbf{b}_t^H})$, the excess inventory cost at the depots and customers for the night of day t, the shortage cost at the customers for the unsatisfied demand of day t, and the penalization induced by the infeasible decisions in $\pi(\mathbf{X}_t, \overline{\mathbf{b}_t^H})$ given $\mathbf{b}_t$. The multi-stage stochastic optimization problem we consider can then be written as:

$$
\begin{aligned}
\min_{\pi} \quad & \mathbb{E}_\pi\Big[\sum_{t=0}^{T} C_t\Big(\mathbf{X}_t, \pi(\mathbf{X}_t, \overline{\mathbf{b}_t^H}), \mathbf{b}_t\Big)\Big] && \text{(2a)}\\
\text{subject to} \quad & \mathbf{X}_0 = \mathbf{x}_0 && \text{(2b)}\\
& \mathbf{X}_{t+1} = f_t(\mathbf{X}_t, \pi(\mathbf{X}_t, \overline{\mathbf{b}_t^H}), \mathbf{b}_t), \quad \forall t \in \{0, ..., T-1\} && \text{(2c)}\\
& \pi(\mathbf{X}_t, \overline{\mathbf{b}_t^H}) \in \mathbb{U}_t(\mathbf{X}_t), \qquad \forall t \in \{0, ..., T\} && \text{(2d)}\\
& \sigma(\pi(\mathbf{X}_t, \overline{\mathbf{b}_t^H})) \subset \sigma(\mathbf{b}_0, ..., \mathbf{b}_{t-1}), \qquad \forall t \in \{0, ..., T\}. && \text{(2e)}
\end{aligned}
$$

In the formulation above, constraint (2d) defines the set of admissible policies given the state. They may not be feasible given the true demand and release $\mathbf{b}_t$. Constraint (2e) expresses the fact that the decision on day t is taken knowing the values of demand and release up to $t-1$, denoted by $(\mathbf{b}_0, ..., \mathbf{b}_{t-1})$.

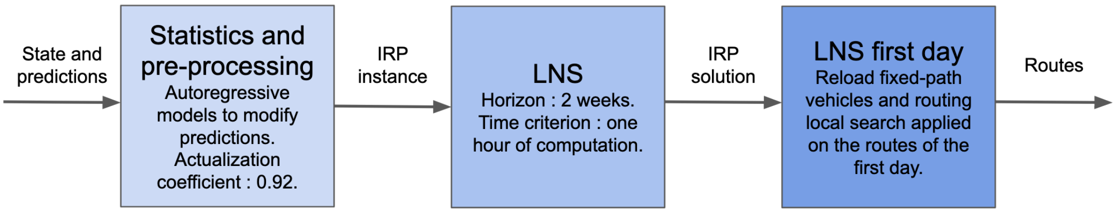
Simulator. We implement a simulator to be able to evaluate policies for the stochastic IRP. It includes the dynamic function f, the cost function C and uses historical data to compute the cost with respect to real scenarios over multiple weeks.

3 Policy

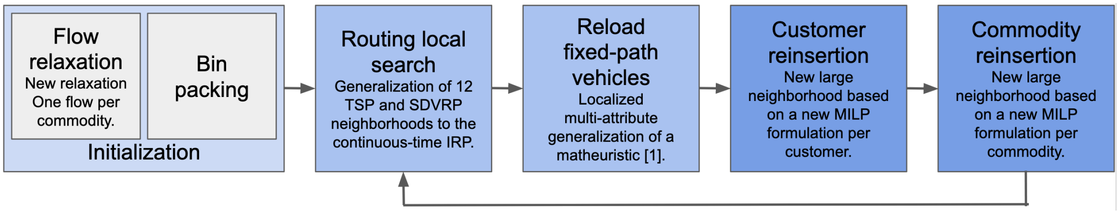
Policy. We derive a heuristic policy to solve the stochastic IRP, detailed in Figure 1 (a). The principle is the following: on each day t, it defines a deterministic IRP instance with fixed-horizon H based on the state $\mathbf{X}_t$ and predictions $\overline{\mathbf{b}_t^H}$. It uses an autoregressive model trained on historical data to set the values of demand and release given $\overline{\mathbf{b}_t^H}$. It also introduces an actualization coefficient $\gamma \in (0, 1)$. The cost of each day $t' \in \{t, ..., \min(T, t+H)\}$ is scaled by a factor $\gamma^{t'-t}$. We tune the parameter γ to take future into account without being too sensitive to the forecast errors. It then solves the IRP instance with our LNS described below, leading to an IRP solution $\mathbf{s}_t = (\mathbf{r}_t, ..., \mathbf{r}_{\min(T,t+H)})$. The set of routes $\mathbf{r}_t$ leaving on day t in $\mathbf{s}_t$ is modified using a simplified version of the LNS, applying the reload-fixed path vehicles and the routing local search subroutines. The reason for this last step is that small changes of $\mathbf{r}_t$ may improve the policy, but not enough to be applied in the time allocated to the LNS. The set of modified routes is then taken as policy: $\pi(\mathbf{X}_t, \overline{\mathbf{b}_t^H}) = \mathbf{r}_t$.

Although the decisions to be taken on day t only concern the routes leaving on day t, one key point of this approach is that it leverages the information it has at its disposal about the future H time steps. It does so by exactly modelling the inventory dynamics, and considering the potential routes in the near future as alternatives to the short-term decisions. It is one major distinction with regard to the algorithm currently in production at Renault.

Large neighborhood search. One crucial ingredient for the policy is the algorithm used to solve the fixed-horizon deterministic problem. As noted above, no approach from the literature scales as is to our instances. We design an LNS shown in Figure 1 (b) that involves the following five subroutines. The first one builds an initial solution. 1) The *initialization* subroutine solves a flow relaxation – thus a linear program – per commodity and deduces direct routes by approximately solving bin packing problems to respect vehicle capacity. The other four subroutines improve or perturb an existing solution, and can be applied any number of times in any order. 2) The *routing local search* subroutine takes a random subset of routes and



(a) Policy.



(b) Large neighborhood search.

FIG. 1: Details of the policy and main underlying contributions.

applies a local search with traveling salesman problem (TSP) and split delivery vehicle routing problem (SDVRP) neighborhoods adapted to the continuous-time IRP. 3) The *reload fixed-path vehicles* subroutine solves an MILP per depot to re-optimize the load of the routes starting from it, possibly cancelling some of them, as done by [1]. 4) The *customer reinsertion* subroutine removes a customer from every delivery of a solution and solves an MILP to reinsert it in the existing routes, also creating new direct routes. 5) The *commodity reinsertion* subroutine removes a commodity from every delivery of a solution and solves an MILP to reinsert it.

The main idea behind this LNS is to consider the structure of the IRP, "decompose" it along its major axes, and solve smaller natural problems to explore the solution space. The crux of the matter is to find a compromise between the size of the large neighborhoods and the time required to solve the MILPs behind them. Key contributions in each subroutine are given in Figure 1 (b). For more details, please consider [4].

4 Numerical experiments

Dynamic IRP. We are first interested in the quality of the policy described in Section 3 when applied to real-world scenarios, and compared with the current algorithm in production. We focus here on cost and CO_2 emissions. Additional indicators such as average vehicle loading and kilometers travelled are omitted to keep this paper short. With the Renault supply chain team[2], we select a specific period of three consecutive weeks of real activity. We thus have access to the data and predictions detailed in Section 2 for $T = 21$ days. Computing an expected cost requires statistical models both for the predictions and realizations of demand and release, which is hard to derive a priori. Therefore, using our simulator, we compare policies with respect to the true historical scenario. The average relative cost reductions are included in Table 1. They are brief for confidentiality reasons. We show that on average, over the three weeks of interest, the new policy saves 4.6% of the transport cost. Since this reduction is induced by the distance travelled, we have a proxy of the transport CO_2 emissions reduction. Besides, we show a significant shortage cost reduction of 57%. The quantity of commodities in shortage at the customers is divided by three thanks to our policy. This can be understood because the algorithm in production does not explicitly take future inventory dynamics into account. Therefore, it can send commodities to customers that do not need them at short term to fill vehicles, which induces less inventory at the depots for future demand. The estimation

[2]We especially thank Thaddeus Leonard for his help on industrial data and code.

of total cost saving is 29%, which motivates the current industrialization process, representing millions of euros and thousands of tons of CO_2 per year.

Transport cost reduction	Shortage cost reduction	Estimated total cost reduction
4.6%	57%	29%

TAB. 1: Estimated average cost reductions induced by our policy compared with the existing algorithm over three weeks of real data.

Fixed-horizon deterministic IRP. Ablation tests show that each of our subroutines visible in Figure 1 (b) enables to significantly improve the performance of the LNS. We present a subset of the results available in [4]. We build 71 preprocessed IRP instances, at the European scale and over roughly 20 days each, with 30 commodities, 16 depots, and 600 customers on average. The maximum number of stops is $S_{max} = 3$ to comply with the car manufacturer requirements. Our dataset and code are publicly available[3]. Algorithms are run with a 90 minutes time limit and same parameters over the 71 instances. *Initialization + local search* simply runs the initialization subroutine to build an initial solution, and then applies the routing local search to improve it. *Route-based matheuristic* applies iteratively the reload fixed-path vehicles subroutine after the initialization. The remaining algorithms considered are the LNS and its ablation versions.

On Figure 2, we plot the cumulative distributions over instances with respect to the gap of the solutions. Our lower bound is not tight, leading to high gaps by definition. It seems relevant to use them as a comparison metric, and not as an absolute indicator on a solution quality. First, the route-based matheuristic (blue curve) performs better than our initialization + local search algorithm (orange curve), but worse than the LNS, even when one of its components is removed. Besides, we show that whatever is the component removed from the LNS, the resulting algorithm with same time budget leads to worse solutions. Indeed, overall, the curve of the cumulative distribution of our whole LNS (brown) is above any other curve. This is particularly true when removing the reload fixed-path vehicles neighborhood (purple), but also for the customer (red) and commodity reinsertion (green) ablations that lead to similar performance. Both perturbation ablations result in 5% additional average gap compared to the whole LNS. We expect this trend to be accentuated when the time limit increases, since looping over neighborhoods and perturbations is key to escape from local minima.

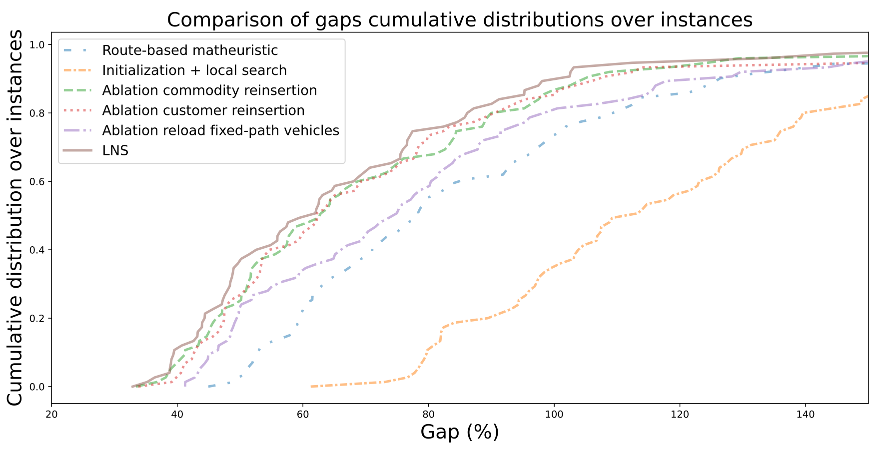


FIG. 2: Cumulative distributions of the gap among deterministic IRP instances solutions.

[3] https://github.com/LouisBouvier/InventoryRoutingLNS.jl

5 Conclusion and perspectives

In this short paper, we emphasize a practical use-case: the dynamic IRP behind the packaging return logistics at Renault. We derive a policy that leverages an LNS designed for the fixed-horizon deterministic IRP, which is itself a challenge. Through numerical experiments on real data, we compare our new policy with the existing algorithm at Renault. We show the potential gains in terms of CO_2 emissions, costs, and packaging in shortage. Those gains motivate the current work done to industrialize our algorithm. We plan to improve the latter with machine learning techniques in a future work.

Acknowledgments. We thank Vincent Leclère for his advice on stochastic optimization. We are also grateful to the Renault supply-chain and IT teams for the partnership we have set and the industrial motivation they provide us with, especially to Alain Nguyen, Thaddeus Leonard, Nicusor-Eugen Plescan, Christian Serrano, Ludovic Doudard and Aimé-Frédéric Rosenzweig.

References

[1] Claudia Archetti, Natashia Boland, and M. Grazia Speranza. A Matheuristic for the Multivehicle Inventory Routing Problem. *INFORMS Journal on Computing*, 29(3):377–387, April 2017.

[2] Thierry Benoist, Frédéric Gardi, Antoine Jeanjean, and Bertrand Estellon. Randomized Local Search for Real-Life Inventory Routing. *Transportation Science*, 45(3):381–398, March 2011.

[3] Luca Bertazzi, Leandro C. Coelho, Annarita De Maio, and Demetrio Laganà. A matheuristic algorithm for the multi-depot inventory routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 122:524–544, February 2019.

[4] Louis Bouvier, Guillaume Dalle, Axel Parmentier, and Thibaut Vidal. Solving a Continent-Scale Inventory Routing Problem at Renault, October 2022.

[5] Leandro Callegari Coelho, Annarita De Maio, and Demetrio Laganà. A variable MIP neighborhood descent for the multi-attribute inventory routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 144:102137, December 2020.

[6] Leandro Callegari Coelho, Gilbert Laporte, and Jean-François Cordeau. Dynamic and stochastic inventory-routing. Technical Report CIRRELT-2012-37, CIRRELT Montreal, 2012.

[7] Jean-François Cordeau, Demetrio Laganà, Roberto Musmanno, and Francesca Vocaturo. A decomposition-based heuristic for the multiple-product inventory-routing problem. *Computers & Operations Research*, 55:153–166, March 2015.

[8] Lars Magnus Hvattum and Arne Løkketangen. Using scenario trees and progressive hedging for stochastic inventory routing problems. *Journal of Heuristics*, 15(6):527–557, December 2009.

[9] Patrick Jaillet, Jonathan F. Bard, Liu Huang, and Moshe Dror. Delivery Cost Approximations for Inventory Routing Problems in a Rolling Horizon Framework. *Transportation Science*, 36(3):292–300, August 2002.

[10] Martin Savelsbergh and Jin-Hwa Song. Inventory routing with continuous moves. *Computers & Operations Research*, 34(6):1744–1763, June 2007.

[11] Yugang Yu, Chengbin Chu, Haoxun Chen, and Feng Chu. Large scale stochastic inventory routing problems with split delivery and service level constraints. *Annals of Operations Research*, 197(1):135–158, August 2012.