Solution Representations and Local Search for the bi-objective Inventory Routing Problem

Thibaut Barthelemy\textsuperscript{1}  Martin J. Geiger\textsuperscript{2}  Marc Sevaux\textsuperscript{3}

\texttt{barthelt@hsu-hh.de}
\texttt{m.j.geiger@hsu-hh.de}
\texttt{marc.sevaux@univ-ubs.fr}

\textsuperscript{1}University of Nantes
\textsuperscript{2}Helmut Schmidt University
\textsuperscript{3}University of South-Brittany

\textit{ORO Master}
\textit{Logistic Management Dept.}
\textit{Lab-STICC, CNRS}

\textit{Nantes, France}
\textit{Hamburg, Germany}
\textit{Lorient, France}

EU/ME 2012 – Copenhagen
May 10–11, 2012
The Inventory Routing Problem (IRP)

- One depot
- A set of customers

Barthelemy, Geiger, Sevaux
The Inventory Routing Problem (IRP)

- One depot
- A set of customers
- A time horizon
The Inventory Routing Problem (IRP)

- One depot
- A set of customers
- A time horizon
- A known demand for each customer and each date
The Inventory Routing Problem (IRP)

- One depot
- A set of customers
- A time horizon
- A known demand for each customer and each date
- Minimize **Inventory** cost
The Inventory Routing Problem (IRP)

- One depot
- A set of customers
- A time horizon
- A known demand for each customer and each date
- Minimize *Inventory* cost
- Minimize *Routing* cost, time, distance, $CO_2$ emissions, ...
The Inventory Routing Problem (IRP)

- One depot
- A set of customers
- A time horizon
- A known demand for each customer and each date
- Minimize **Inventory** cost
- Minimize **Routing** cost, time, distance, CO₂ emissions, ...

IRP is a **real bi-objective** optimization problem
Important decisions

For solving the IRP, we must make the following decisions:

### Decisions

1. When deliver customers?
2. How much deliver?
3. With which routes?

All these decisions are linked together:

- increase delivery quantities $\rightarrow$ change routes or periods
- change periods $\rightarrow$ adapt delivery quantities
- ...
Important references

**First papers**

[Bell et al., 1983]
Improving the distribution of industrial gases with and on-line computerized routing and scheduling optimizer

[Federgruen and Zipkin, 1984]
A combined vehicle routing and inventory allocation problem

**Recent surveys**

[Bertazzi et al., 2008]
Inventory routing

[Cordeau et al., 2011]
Short-haul routing
Some guidelines/targets of our work

- Main goal: helping companies where IRP is important
- Propose a simple output
- Use simple rules (that can be understood)
- Use simple implementation that can be reproduced
- Give different alternatives

We want to help the decision maker
A major assumption in our work

To follow a common strategy in companies dealing with IRP, we have separated the decisions

1. Determine quantities for each period
2. Compute best routing for each period
Simplifications

On visiting dates

- Visits are periodic.
- Each customer $i$ has a delivery period $p_i$.
- The periodic sequence starts at $s_i$.

One delivery before the starting date is made if required to cover first demands of some customers.

Purpose

Reduction of the number of searched variables.
Simplifications

**On delivered amounts**
Any customer is replenished just enough to satisfy its consumptions until next visit.

**Purpose**
Served quantities are not part of the searched variable set.

**Standard replenishment rules**
- **DD** Day-to-day delivery policy
  - If not enough in stock, deliver the missing demand
- **OU** Order-up-to level policy
  - When you ship, ship the maximum (customer capacity)
- **ML** Maximum level policy (misleading)
  - Any quantity less than the maximum level
Evaluation of solutions

Each solution is measured with two criteria

**Inventory cost**
Sum of all inventory levels at customers’ at the end of each date. This can be computed in $\mathcal{O}(nH)$

**Routing cost**
Sum of all distances run by the trucks at every date: VRP solutions.
As it is a $\mathcal{NP}$-hard problem, we use:
- Clarke & Wright heuristic in $\mathcal{O}(n'^2 \log n)$,
- Two-optimization in $\mathcal{O}(n'^2)$ on average.
We have implemented several initial solutions
We have implemented several initial solutions

- **Identical period**: $p_i$ is the same for each customer
Initial solutions

We have implemented several initial solutions

- **Identical period**: $p_i$ is the same for each customer
- ** Totally random**: $p_i$ is chosen at random for each customer
We have implemented several initial solutions

- **Identical period:** $p_i$ is the same for each customer
- ** Totally random:** $p_i$ is chosen at random for each customer
- **Controlled random:** $p_i$ is chosen at random between two bounds
Initial solutions

Initial solution set of a real instance
<table>
<thead>
<tr>
<th></th>
<th>( s_i - 1 )</th>
<th>( s_i + 1 )</th>
<th>( p_i - 1 )</th>
<th>( p_i + 1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>((7, 4, 1, 6, 5))</td>
<td>((9, 4, 1, 6, 5)) &amp; ((8, 5, 1, 6, 5)) &amp; ((8, 4, 2, 6, 5))</td>
<td>((8, 4, 1, 6, 5)) &amp; ((8, 4, 1, 6, 5)) &amp; ((8, 4, 1, 6, 5))</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>((3, 2, 2, 2, 3))</td>
<td>((3, 2, 2, 2, 3)) &amp; ((3, 2, 2, 2, 3)) &amp; ((3, 2, 2, 2, 3))</td>
<td>((3, 2, 1, 2, 3)) &amp; ((3, 2, 1, 2, 3)) &amp; ((3, 2, 1, 2, 3))</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>((8, 4, 1, 6, 5))</td>
<td>((8, 4, 1, 6, 5)) &amp; ((8, 4, 1, 6, 5)) &amp; ((8, 4, 1, 6, 5))</td>
<td>((8, 4, 1, 6, 5)) &amp; ((8, 4, 1, 6, 5)) &amp; ((8, 4, 1, 6, 5))</td>
<td></td>
</tr>
</tbody>
</table>

\( s = (8, 4, 1, 6, 5) \)
\( p = (3, 2, 2, 2, 3) \)
General algorithm

Algorithm 1: Improvement

Initialization: create an initial population
   → Identical period + controlled random

Cleanup: remove dominated solutions

repeat
   Select $R$ solutions
   Generate $4n$ neighbors for each solution
      → add $\pm 1$ to each starting date and each period
   Rebuild archive with non-dominated solutions only
until no more improvements

- At each iteration, $4nR$ neighbors are evaluated.
- So $it$ iterations imply $ev = it4nR$ evaluations.
- Data of 1 customer is modified in each neighbor.
- So $\frac{ev}{4n^2R}$ modifications per customer are done on average.
Selection strategies

Output for an instance with 50 customers (2485 alternatives)
Selection strategies
Selection strategy 1 - by reference points

Thousands of non-dominated solutions. How to handle them?

First solution: the **reference set** strategy
Selection strategy 1 - by reference points

- Thousands of non dominated solutions. How to handle them?

First solution: the **reference set** strategy
(here with \( R=7 \) reference points)
Selection strategy 1 - by reference points

- Thousands of non dominated solutions. How to handle them?

First solution: the **reference set** strategy (here with \( R=7 \) reference points)
Selection strategy 2 - by crowding distances

Random selection

- Inspired from NSGA-II
- Likelihood depends on crowding distances
A new set of instances

Derived from VRP instances of Nicos Christofides et al.\(^1\)

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Possible values</th>
</tr>
</thead>
<tbody>
<tr>
<td># customers, (n)</td>
<td>{50, 75, 100, 150, 199}</td>
</tr>
<tr>
<td>Horizon, (H)</td>
<td>{30}</td>
</tr>
<tr>
<td>Truck capacity, (\hat{Q})</td>
<td>{480, 500, 600}</td>
</tr>
<tr>
<td>Average demand</td>
<td>{Constant, Increasing, Sinus}</td>
</tr>
</tbody>
</table>

This forms a new set of 45 instances available at [http://logistik.hsu-hh.de/IRP](http://logistik.hsu-hh.de/IRP)

\(^1\) *The vehicle routing problem* in Combinatorial Optimization, John Wiley and Sons, 1979
Period only / Starting date + Period

Pareto front obtained for GS-05-30a
Period only / Starting date + Period

Hyper-volume evolution obtained for GS-05-30a
Changing the number of reference points

Even if we change $R$, the method is still robust

![Graph showing inventory-routing alternatives for different reference point values.](graph.png)
Solution selection rule

Pareto front obtained for GS-01-30b
Solution selection rule

Hyper-volume evolution obtained for GS-02-30a
Removal of identical solutions

Pareto front obtained for GS-01-30c
Removal of identical solutions

Hyper-volume evolution obtained for GS-01-30c
Progression of our algorithm

n=50, sinusoidal consumptions
Progression of our algorithm

\[ n=100, \text{ linear increasing consumptions} \]
Progression of our algorithm

n=200, constant consumptions
Progression of our algorithm

Comparison of progression rates for different instances
Archetti’s instances

Provided by Claudia Archetti et al.\textsuperscript{2}

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Possible values</th>
</tr>
</thead>
<tbody>
<tr>
<td># customers, $n$</td>
<td>${5, 10, \ldots, 45, 50}$</td>
</tr>
<tr>
<td>Horizon, H</td>
<td>${3, 6}$</td>
</tr>
<tr>
<td>Demand</td>
<td>${\text{Constant}}$</td>
</tr>
</tbody>
</table>

At provider: Inventory cost & Production rate.

\textsuperscript{2}A branch-and-cut algorithm for a vendor-managed inventory-routing problem in Transportation Science vol 41, 2007
Progression of our algorithm
Separation of costs

Pareto front we obtain vs. position of the optimum of Archetti
## Performances

<table>
<thead>
<tr>
<th>size</th>
<th>type</th>
<th>ML/OU (%)</th>
<th>GSB/OU (%)</th>
<th>OU (s)</th>
<th>GSB (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>low-C H3</td>
<td>-13.23</td>
<td>-4.47</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>low-C H3</td>
<td>-15.94</td>
<td>-8.39</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>low-C H3</td>
<td>-13.68</td>
<td>-3.48</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>high-C H6</td>
<td>-12.53</td>
<td>-10.81</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>high-C H6</td>
<td>-10.66</td>
<td>-6.15</td>
<td>530</td>
<td>2</td>
</tr>
<tr>
<td>30</td>
<td>high-C H6</td>
<td>-9.03</td>
<td>-7.18</td>
<td>4100</td>
<td>20</td>
</tr>
</tbody>
</table>

**ML/OU** gain arising from the ML replenishment rule with regard to the OU rule, for optimal results.

**GSB/OU** gain arising from our heuristic with regard to exact results for the OU replenishment rule.
Conclusion

What we learnt

- Movable starting dates improve only solutions with high inventory costs.
- Random selection provides more homogeneous fronts but worse solutions than reference point selection.
- Solutions with identical objective values must be removed, without view of decision variables.
- Relevant stopping criteria: ratio modifications / customer.
- For large instances, we are not so far from optimum and we are considerably faster.

In progress

- Customer clustering (acts like a pre-solving)
- Ideal representation for crossover: solving by NSGA-II
Our web pages

Project page
http://logistik.hsu-hh.de/IRP
OR-group of Lab-STICC
http://www.univ-ubs.fr/or/

Contact
m.j.geiger@hsu-hh.de
marc.sevaux@univ-ubs.fr