Inventory routing with non-stationary stochastic demands

Iliya Markov\textsuperscript{a}, Yousef Maknoon\textsuperscript{a}, Jean-François Cordeau\textsuperscript{b}
Sacha Varone\textsuperscript{c}, Michel Bierlaire\textsuperscript{a}

\textsuperscript{a}Transport and Mobility Laboratory
School of Architecture, Civil and Environmental Engineering
École Polytechnique Fédérale de Lausanne

\textsuperscript{b}HEC Montréal and CIRRELT

\textsuperscript{c}Haute École de Gestion de Genève
University of Applied Sciences Western Switzerland (HES-SO)

16\textsuperscript{th} Swiss Transport Research Conference (STRC)
Monte Verità / Ascona, May 18–20, 2016
Outline

1. Introduction
2. Related Literature
3. Sketch of Formulation
4. Methodology
5. Numerical Experiments
6. Conclusion
# Outline

1. Introduction
2. Related Literature
3. Sketch of Formulation
4. Methodology
5. Numerical Experiments
6. Conclusion
**Setup**

- Sensorized containers for recyclables periodically send waste level data to a central database.
Setup

- Sensorized containers for recyclables periodically send waste level data to a central database.
- Level data is used for container selection and tour planning.
Sensorized containers for recyclables periodically send waste level data to a central database.

Level data is used for container selection and tour planning.

Vehicles are dispatched to carry out the daily schedules produced by the routing algorithm.
Introduction

Setup

- Sensorized containers for recyclables periodically send waste level data to a central database.
- Level data is used for container selection and tour planning.
- Vehicles are dispatched to carry out the daily schedules produced by the routing algorithm.
- Efficient waste collection depends on the ability to:
  - forecast container levels,
  - select the containers to collect each day,
  - and route the vehicles in an (near-)optimal way.
The inventory routing problem (IRP) is a multi-day problem that determines simultaneously:

- the visit days,
- the delivery/collection quantities,
- the vehicle tours on each day.
Problem Definition

The inventory routing problem (IRP) is a multi-day problem that determines simultaneously:

- the visit days,
- the delivery/collection quantities,
- the vehicle tours on each day.

The routing component in our problem is schematically represented by Figure 1:

![Figure 1: Tour example](image-url)
Outline

1. Introduction
2. Related Literature
3. Sketch of Formulation
4. Methodology
5. Numerical Experiments
6. Conclusion
Related Stochastic IRP Literature

- Early research on optimal replenishment policies in a stochastic setting:

- Robust optimization:
  - Solyalı et al. (2012).

- Chance constraints:

- Scenario based:
  - Rollout/branch-and-cut: Bertazzi et al. (2013), Bertazzi et al. (2015),
Contributions

- Dynamic probabilistic information on overflows and route failures.
Contributions

- Dynamic probabilistic information on overflows and route failures.
- Demand forecasting model tested and validated on real data (Markov et al., 2015).
Contributions

- Dynamic probabilistic information on overflows and route failures.
- Demand forecasting model tested and validated on real data (Markov et al., 2015).
- A rich IRP with features traditionally absent or rarely considered in the IRP literature.
Related Literature

Contributions

- Dynamic probabilistic information on overflows and route failures.
- Demand forecasting model tested and validated on real data (Markov et al., 2015).
- A rich IRP with features traditionally absent or rarely considered in the IRP literature.
- ALNS algorithm performs excellently on IRP benchmarks from the literature.
- Benefit of considering uncertainty in the objective function evaluated on instances derived from real data.
Outline

1. Introduction
2. Related Literature
3. Sketch of Formulation
4. Methodology
5. Numerical Experiments
6. Conclusion
Basic Definitions and Ideas

- Demand is the amount deposited in a container on each day.
- It is random and non-stationary.
Basic Definitions and Ideas

- Demand is the amount deposited in a container on each day.
- It is random and non-stationary.
- The forecasting model gives:
  - The expected container demands on each day,
  - A consistent estimate of the forecasting error based on historical fit.

The distribution of the forecasting error can be approximated by the normal distribution, which is used to calculate probabilities of container overflows and route failures. They are dynamic and conditional, and depend on:
- The evolution of container state scenarios on each day (overflowing vs. not full),
- And the vehicle visits on each day.
Basic Definitions and Ideas

- Demand is the amount deposited in a container on each day.
- It is random and non-stationary.
- The forecasting model gives:
  - The expected container demands on each day,
  - A consistent estimate of the forecasting error based on historical fit.
- The distribution of the forecasting error can be approximated by the normal distribution, which is used to calculate probabilities of container overflows and route failures.
- They are dynamic and conditional, and depend on:
  - The evolution of container state scenarios on each day (overflowing vs. not full),
  - And the vehicle visits on each day.
Basic Definitions and Ideas

- **Container overflow:**
  - Container is full and all subsequent demand is placed beside it,
  - Overflow cost: paid on each day when there is an overflow,
  - Emergency visit cost: paid on each day when there is an overflow **and** no planned visit.

- **Route failure:**
  - Vehicle becomes full earlier than the next scheduled dump visit,
  - Entails the cost of visiting the closest dump.
Objective Function

Routing cost + Expected overflow and emergency visit cost + Expected route failure cost

Lower routing cost is counterbalanced by more overflows and route failures, and vice versa.

Our goal is to minimize the expected monetary value of all components.
Objective Function

- Lower routing cost is counterbalanced by more overflows and route failures, and vice versa.
- Our goal is to minimize the expected monetary value of all components.
Constraints

- Collection policy:
  - Order-up-to (OU),
  - No expected overflows over the planning horizon,
  - An overflow on day 0 is out of our control,
  - But the container must be collected on day 0 (single-day backorder limit).
Constraints

- **Collection policy:**
  - Order-up-to (OU),
  - No expected overflows over the planning horizon,
  - An overflow on day 0 is out of our control,
  - But the container must be collected on day 0 (single-day backorder limit).

- We also need to ensure/enforce the rich features of the routing component:
  - Point accessibilities,
  - Vehicle availabilities,
  - Vehicle capacities,
  - Time windows,
  - Maximum tour duration.
Outline

1. Introduction
2. Related Literature
3. Sketch of Formulation
4. Methodology
5. Numerical Experiments
6. Conclusion
Adaptive Large Neighborhood Search (ALNS)

- A meta-heuristic framework in which a number of fairly simple destroy and repair operators compete in modifying the current solution.
Adaptive Large Neighborhood Search (ALNS)

- A meta-heuristic framework in which a number of fairly simple destroy and repair operators compete in modifying the current solution.

- At each iteration, a destroy-repair operator couple is drawn based on past performance.

- The operator $i$ with weight $\omega_i$ is drawn from the destroy (repair) pool with a probability:

$$\mathbb{P}(i) = \frac{\omega_i}{\sum_{j \in \mathcal{O}} \omega_j} \quad (1)$$
Adaptive Large Neighborhood Search (ALNS)

- A meta-heuristic framework in which a number of fairly simple destroy and repair operators compete in modifying the current solution.

- At each iteration, a destroy-repair operator couple is drawn based on past performance.

- The operator $i$ with weight $\omega_i$ is drawn from the destroy (repair) pool with a probability:

$$P(i) = \frac{\omega_i}{\sum_{j \in O} \omega_j} \quad (1)$$

- The solution guiding mechanism relies on simulated annealing.
The Operators

Destroy operators:
- Remove $\rho$ containers randomly.
- Remove $\rho$ worst containers.
- Shaw removals (Shaw, 1997).
- Empty a random day.
- Empty a random vehicle.
- Remove a random dump.
- Remove the worst dump.
- Remove consecutive visits.

Repair operators:
- Insert $\rho$ containers randomly.
- Insert $\rho$ containers in the best way.
- Shaw insertions (Shaw, 1997).
- Swap $\rho$ random containers.
- Insert a dump randomly.
- Swap random dumps.
- Replace a random dump.
- Reorder dumps DP operator.
Outline

1. Introduction
2. Related Literature
3. Sketch of Formulation
4. Methodology
5. Numerical Experiments
6. Conclusion
### Table 1: Results on high cost instances

<table>
<thead>
<tr>
<th>$H$</th>
<th>$n$</th>
<th>ALNS Fast version</th>
<th>ALNS Slow version</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Runtime(s.)</td>
<td>Min Gap(%)</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>8</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>14</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>22</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>36</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>25</td>
<td>53</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>77</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>35</td>
<td>108</td>
<td>0.01</td>
</tr>
<tr>
<td>3</td>
<td>40</td>
<td>149</td>
<td>0.12</td>
</tr>
<tr>
<td>3</td>
<td>45</td>
<td>199</td>
<td>0.17</td>
</tr>
<tr>
<td>3</td>
<td>50</td>
<td>276</td>
<td>0.15</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>14</td>
<td>0.00</td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>28</td>
<td>0.00</td>
</tr>
<tr>
<td>6</td>
<td>15</td>
<td>53</td>
<td>0.00</td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td>81</td>
<td>0.04</td>
</tr>
<tr>
<td>6</td>
<td>25</td>
<td>128</td>
<td>0.19</td>
</tr>
<tr>
<td>6</td>
<td>30</td>
<td>189</td>
<td>0.08</td>
</tr>
</tbody>
</table>

Average | 90 | 0.05 | 0.22 | 361 | 0.02 | 0.11 |
Archetti et al. (2007) Instances

Table 2: Results on low cost instances

<table>
<thead>
<tr>
<th>$H$</th>
<th>$n$</th>
<th>ALNS Fast version</th>
<th>ALNS Slow version</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Runtime(s.)</td>
<td>Min Gap(%)</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>7</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>14</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>22</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>34</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>25</td>
<td>52</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>71</td>
<td>0.02</td>
</tr>
<tr>
<td>3</td>
<td>35</td>
<td>101</td>
<td>0.01</td>
</tr>
<tr>
<td>3</td>
<td>40</td>
<td>140</td>
<td>0.37</td>
</tr>
<tr>
<td>3</td>
<td>45</td>
<td>191</td>
<td>0.59</td>
</tr>
<tr>
<td>3</td>
<td>50</td>
<td>247</td>
<td>0.30</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>13</td>
<td>0.00</td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>28</td>
<td>0.00</td>
</tr>
<tr>
<td>6</td>
<td>15</td>
<td>49</td>
<td>0.00</td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td>77</td>
<td>0.08</td>
</tr>
<tr>
<td>6</td>
<td>25</td>
<td>121</td>
<td>0.25</td>
</tr>
<tr>
<td>6</td>
<td>30</td>
<td>182</td>
<td>0.67</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Average</td>
<td>84</td>
</tr>
</tbody>
</table>
Archetti et al. (2012) Instances

Table 3: Results on high cost 50-customer instances

<table>
<thead>
<tr>
<th>Instance</th>
<th>Runtime(s.)</th>
<th>Min Cost</th>
<th>Avg Cost</th>
<th>Min Gap(%)</th>
<th>Avg Gap(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>abs1n50</td>
<td>670</td>
<td>30,708.05</td>
<td>30,809.31</td>
<td>-1.41</td>
<td>-1.09</td>
</tr>
<tr>
<td>abs2n50</td>
<td>676</td>
<td>30,226.23</td>
<td>30,271.07</td>
<td>0.11</td>
<td>0.26</td>
</tr>
<tr>
<td>abs3n50</td>
<td>667</td>
<td>30,388.68</td>
<td>30,515.79</td>
<td>-0.11</td>
<td>0.31</td>
</tr>
<tr>
<td>abs4n50</td>
<td>671</td>
<td>32,103.17</td>
<td>32,213.62</td>
<td>0.64</td>
<td>0.99</td>
</tr>
<tr>
<td>abs5n50</td>
<td>666</td>
<td>29,646.74</td>
<td>29,797.79</td>
<td>0.43</td>
<td>0.95</td>
</tr>
<tr>
<td>abs6n50</td>
<td>652</td>
<td>32,336.81</td>
<td>32,420.63</td>
<td>-0.18</td>
<td>0.08</td>
</tr>
<tr>
<td>abs7n50</td>
<td>661</td>
<td>30,222.28</td>
<td>30,269.23</td>
<td>0.19</td>
<td>0.35</td>
</tr>
<tr>
<td>abs8n50</td>
<td>652</td>
<td>26,409.83</td>
<td>26,537.19</td>
<td>-0.03</td>
<td>0.46</td>
</tr>
<tr>
<td>abs9n50</td>
<td>656</td>
<td>30,543.31</td>
<td>30,630.53</td>
<td>-0.42</td>
<td>-0.13</td>
</tr>
<tr>
<td>abs10n50</td>
<td>635</td>
<td>31,937.51</td>
<td>32,065.85</td>
<td>-1.31</td>
<td>-0.92</td>
</tr>
<tr>
<td>Average</td>
<td>661</td>
<td>30,452.26</td>
<td>30,553.10</td>
<td>-0.21</td>
<td>0.13</td>
</tr>
</tbody>
</table>
### Table 4: Results on low cost 50-customer instances

<table>
<thead>
<tr>
<th>Instance</th>
<th>Runtime(s.)</th>
<th>Min Cost</th>
<th>Avg Cost</th>
<th>Min Gap(%)</th>
<th>Avg Gap(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>abs1n50</td>
<td>611</td>
<td>10,377.36</td>
<td>10,449.91</td>
<td>-0.31</td>
<td>0.39</td>
</tr>
<tr>
<td>abs2n50</td>
<td>643</td>
<td>10,927.83</td>
<td>11,014.20</td>
<td>0.43</td>
<td>1.22</td>
</tr>
<tr>
<td>abs3n50</td>
<td>622</td>
<td>10,702.05</td>
<td>10,924.09</td>
<td>-0.61</td>
<td>1.46</td>
</tr>
<tr>
<td>abs4n50</td>
<td>632</td>
<td>10,711.86</td>
<td>10,875.98</td>
<td>0.52</td>
<td>2.06</td>
</tr>
<tr>
<td>abs5n50</td>
<td>624</td>
<td>10,332.55</td>
<td>10,458.54</td>
<td>0.96</td>
<td>2.19</td>
</tr>
<tr>
<td>abs6n50</td>
<td>620</td>
<td>10,388.66</td>
<td>10,485.72</td>
<td>-1.38</td>
<td>-0.45</td>
</tr>
<tr>
<td>abs7n50</td>
<td>626</td>
<td>10,388.08</td>
<td>10,497.06</td>
<td>-0.70</td>
<td>0.35</td>
</tr>
<tr>
<td>abs8n50</td>
<td>623</td>
<td>10,683.31</td>
<td>10,771.40</td>
<td>2.61</td>
<td>3.46</td>
</tr>
<tr>
<td>abs9n50</td>
<td>610</td>
<td>10,416.97</td>
<td>10,472.96</td>
<td>1.08</td>
<td>1.62</td>
</tr>
<tr>
<td>abs10n50</td>
<td>598</td>
<td>10,047.06</td>
<td>10,153.50</td>
<td>-4.05</td>
<td>-3.03</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>621</strong></td>
<td><strong>10,497.57</strong></td>
<td><strong>10,610.33</strong></td>
<td><strong>-0.14</strong></td>
<td><strong>0.93</strong></td>
</tr>
</tbody>
</table>
Instances Based on Real Data

- 63 instances, each covering a week of white glass collections in Geneva, Switzerland in 2014, 2015, or 2016.

- Vehicle-related costs:
  - Per day: 100 CHF,
  - Per km: 2.95 CHF,
  - Per hour: 40 CHF.

- Container-related costs:
  - Overflow cost: 100 CHF,
  - Emergency collection cost: 100 CHF.
Numerical Experiments

Instances Based on Real Data

- 63 instances, each covering a week of white glass collections in Geneva, Switzerland in 2014, 2015, or 2016.

- **Vehicle-related costs:**
  - Per day: 100 CHF,
  - Per km: 2.95 CHF,
  - Per hour: 40 CHF.

- **Container-related costs:**
  - Overflow cost: 100 CHF,
  - Emergency collection cost: 100 CHF.

- **Two types of problem:**
  - Routing-only: Considers no overflow and route failure risk,
  - Complete: Considers full objective with the above costs.
### Numerical Experiments

#### Real Data: Cost Comparison

**Table 5:** Cost breakdown for real data instances

<table>
<thead>
<tr>
<th></th>
<th>Avg Cost (CHF)</th>
<th>Avg Routing Cost (CHF)</th>
<th>Avg Overflow Cost (CHF)</th>
<th>Avg Rte Failure Cost (CHF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing-only</td>
<td>430.61</td>
<td>430.61</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Complete</td>
<td>693.66</td>
<td>588.16</td>
<td>105.44</td>
<td>0.06</td>
</tr>
</tbody>
</table>

**Table 6:** Performance indicators for real data instances

<table>
<thead>
<tr>
<th></th>
<th>Avg Collected Volume (L)</th>
<th>Liters per Unit Cost</th>
<th>Liters per Unit Routing Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing-only</td>
<td>25,106.81</td>
<td>58.31</td>
<td>58.31</td>
</tr>
<tr>
<td>Complete</td>
<td>47,364.96</td>
<td>68.28</td>
<td>80.53</td>
</tr>
</tbody>
</table>
Figure 2: Cost percentiles of container overflows
Real Data: Taking Advantage of Probability Information

Figure 3: Container overflow percentiles for routing-only objective
Real Data: Taking Advantage of Probability Information

**Figure 4:** Container overflow percentiles for complete objective
Real Data: Taking Advantage of Probability Information

Figure 5: Route failure percentiles for routing-only objective
Real Data: Taking Advantage of Probability Information

Figure 6: Route failure percentiles for complete objective
Real Data: Explaining Overflows, Comparison

Table 7: Driving factors for the occurrence of container overflows

(a) Regressions on forecasting error

<table>
<thead>
<tr>
<th>Objective</th>
<th>75th percentile coefficient</th>
<th>75th percentile $R^2$</th>
<th>90th percentile coefficient</th>
<th>90th percentile $R^2$</th>
<th>95th percentile coefficient</th>
<th>95th percentile $R^2$</th>
<th>99th percentile coefficient</th>
<th>99th percentile $R^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing-only</td>
<td>0.16***</td>
<td>0.51</td>
<td>0.18***</td>
<td>0.52</td>
<td>0.19***</td>
<td>0.51</td>
<td>0.21***</td>
<td>0.51</td>
</tr>
<tr>
<td>Complete</td>
<td>0.02***</td>
<td>0.49</td>
<td>0.02***</td>
<td>0.53</td>
<td>0.03***</td>
<td>0.52</td>
<td>0.03***</td>
<td>0.57</td>
</tr>
</tbody>
</table>

(b) Regressions on number of containers in instance

<table>
<thead>
<tr>
<th>Objective</th>
<th>75th percentile coefficient</th>
<th>75th percentile $R^2$</th>
<th>90th percentile coefficient</th>
<th>90th percentile $R^2$</th>
<th>95th percentile coefficient</th>
<th>95th percentile $R^2$</th>
<th>99th percentile coefficient</th>
<th>99th percentile $R^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing-only</td>
<td>0.34***</td>
<td>0.25</td>
<td>0.37***</td>
<td>0.25</td>
<td>0.41***</td>
<td>0.26</td>
<td>0.47***</td>
<td>0.27</td>
</tr>
<tr>
<td>Complete</td>
<td>0.02**</td>
<td>0.08</td>
<td>0.02**</td>
<td>0.07</td>
<td>0.03**</td>
<td>0.09</td>
<td>0.03*</td>
<td>0.06</td>
</tr>
</tbody>
</table>

Significance codes: *** 99%, ** 95%, * 90%
Outline

1. Introduction
2. Related Literature
3. Sketch of Formulation
4. Methodology
5. Numerical Experiments
6. Conclusion
Conclusions

- A rich stochastic IRP with the relevant dynamic uncertainty components in the objective.
- An ALNS that produces excellent results on IRP benchmarks.
- Computational experiments on real-data instances demonstrate the practical relevance of our approach.
Conclusions

- A rich stochastic IRP with the relevant dynamic uncertainty components in the objective.
- An ALNS that produces excellent results on IRP benchmarks.
- Computational experiments on real-data instances demonstrate the practical relevance of our approach.
- Future research directions:
  - Decomposition methods,
  - Scenario generation,
  - Chance constraints,
  - Location-routing, open tours, online re-optimization, multiple flows...
Thank you.

Questions?


