A matheuristic for pre-positioning of disaster relief supplies

Renata Turkeš, Daniel Palhazi Cuervo, Kenneth Sörensen
March 17, 2016

This research was supported by the Interuniversity Attraction Poles (IAP) Programme initiated by the Belgian Science Policy Office (COMEX project).
$20.1 BILLION TO PROVIDE AID TO 87.6 MILLION PEOPLE IN 2016
Motivation

IRAQ

8.2M PEOPLE IN NEED OF ASSISTANCE
60% OF FRONTLINE HUMANITARIAN OPERATIONS AT RISK OF SHUTTING DOWN DUE TO UNDERFUNDING
Motivation

SYRIA

- 4M FORCED ACROSS BORDERS
- 4.8M PEOPLE LIVING IN HARD-TO-REACH OR BESIEGED LOCATIONS

Credit: UNHCR/B. Redford
Motivation

UKRAINE

5M in need of humanitarian assistance
2.2M uprooted by conflict
In the SAHEL alone, 9 million women, children and men will go hungry this year if aid doesn't reach them.
The objective of disaster response in the humanitarian relief chain is to rapidly provide relief (emergency food, water, medicine, shelter, and supplies) to areas affected by large-scale emergencies, so as to minimize human suffering and death. [1]

⇒ Optimization problem!
Problem description
The related academic literature in humanitarian logistics mostly falls into one of the three streams:
The related academic literature in humanitarian logistics mostly falls into one of the three streams:
The related academic literature in humanitarian logistics mostly falls into one of the three streams:

- Facility location
- Inventory management
- Routing
Pre-positioning of emergency supplies for disaster response

Develop emergency response pre-positioning strategy that determines

- number, location and size of storage facilities
- quantities of various types of emergency supplies stocked in each facility
- (distribution of the supplies to demand locations after an event,)

under uncertainty about if, or where, a natural disaster will occur.

This problem is NP-hard, as a generalization of facility location problem, that is NP-hard.
Problem description: Example

Scenario 1, $p_1 = 0.9$

Scenario 2, $p_2 = 0.1$
The objective is to minimize
▶ expected unmet demand
▶ expected response times
▶ expected spoilage cost

in lexicographic order, such that the cost of opening the facilities, acquisition cost and transportation cost respect their budget limitations (and the other capacity constraints are satisfied).
The objective is to minimize

- expected unmet demand
- expected response times
- expected spoilage cost

in lexicographic order, such that the cost of opening the facilities, acquisition cost, and transportation cost respect their budget limitations (and the other capacity constraints are satisfied).
The objective is to minimize
▶ expected unmet demand
▶ expected response times
▶ expected spoilage cost

in lexicographic order, such that the cost of opening the facilities, acquisition cost and transportation cost respect their budget limitations (and the other capacity constraints are satisfied).
The objective is to minimize

- expected unmet demand
- expected response times
- expected spoilage cost

in lexicographic order, such that the cost of opening the facilities, acquisition cost, and transportation cost respect their budget limitations (and the other capacity constraints are satisfied).
The objective is to minimize

- expected unmet demand
- expected response times
- expected spoilage cost

in lexicographic order, such that the cost of opening the facilities, acquisition cost and transportation cost respect their budget limitations (and the other capacity constraints are satisfied).
The objective is to minimize

- expected unmet demand
- expected response times
- expected spoilage cost

in lexicographic order, such that the cost of opening the facilities, acquisition cost and transportation cost respect their budget limitations (and the other capacity constraints are satisfied).
The objective is to minimize

- expected unmet demand
- expected response times
- expected spoilage cost

in lexicographic order, such that the cost of opening the facilities, acquisition cost and transportation cost respect their budget limitations (and the other capacity constraints are satisfied).
Matheuristic design: Example

Scenario 1, $p_1 = 0.9$

Scenario 2, $p_2 = 0.1$
Matheuristic design: Example

Scenario 1, $p_1 = 0.9$

Scenario 2, $p_2 = 0.1$
Matheuristic design: Example

Scenario 1, $p_1 = 0.9$

Scenario 2, $p_2 = 0.1$
Matheuristic design: Example

Scenario 1, $p_1 = 0.9$

Scenario 2, $p_2 = 0.1$
Matheuristic design: Example

Scenario 1, $p_1 = 0.9$

Scenario 2, $p_2 = 0.1$
Matheuristic design: Example

Scenario 1, \( p_1 = 0.9 \)
- [10, 2]
- [50, 5]
- [100, 10]

Scenario 2, \( p_2 = 0.1 \)
- [100, 100]
- [80, 70]
Matheuristic design: Example

Scenario 1, $p_1 = 0.9$

Scenario 2, $p_2 = 0.1$
Matheuristic design: Steps

1. Find a (reasonable) location configuration
2. Find a (reasonable) inventory configuration
3. Find an optimal assignment of cities to facilities
Matheuristic design: Steps

(1) Find a (reasonable) location configuration
Matheuristic design: Steps

(1) Find a (reasonable) location configuration
(2) Find a (reasonable) inventory configuration
Matheuristic design: Steps

(1) Find a (reasonable) location configuration
(2) Find a (reasonable) inventory configuration
(3) Find an optimal assignment of cities to facilities
Matheuristic design: Steps

(1) Find a (reasonable) location configuration
(2) Find a (reasonable) inventory configuration
(3) Find an optimal assignment of cities to facilities
Algorithm 1: Matheuristic

CurrentSolution = EmptySolution(Instance);
for $t = 1$ to NumberOfInitialSolutions do
    TempSolution = GreedyRandConstHeuristic(CurrentSolution);
    if TempSolution is better than CurrentSolution then
        CurrentSolution = TempSolution;
    end
end
CurrentSolution = IteratedLocalSearch(CurrentSolution);
return CurrentSolution
Algorithm 2: Iterated local search

BestSolution = CurrentSolution;
UnassignAllVertices(CurrentSolution);
for \( i = 1 \) to NumberOfLocationPerturbations do
    for \( j = 1 \) to NumberOfInventoryPerturbations do
        OptimalAssignment(CurrentSolution);
        for \( k = 1 \) to InventoryLocalSearchDepth do
            CurrentSolution = BestInventoryNeighbor(CurrentSolution, nbh1, nbh2);
            if CurrentSolution is better than BestSolution then
                BestSolution = CurrentSolution;
            end
        end
    end
    RandomInventoryPerturbation(CurrentSolution, pert);
end
RandomLocationPerturbation(CurrentSolution, pert);
return BestSolution
Matheuristic analysis: Parameter tuning

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>nbh1 (insert vertex)</td>
<td>on, off</td>
</tr>
<tr>
<td>nbh2 (remove vertex and insert vertex)</td>
<td>on, off</td>
</tr>
<tr>
<td>pert (perturbation factor)</td>
<td>0, 20, 40, 60, 80, 100</td>
</tr>
</tbody>
</table>
**Matheuristic analysis: Parameter tuning**

**Analysis of Variance table**

<table>
<thead>
<tr>
<th></th>
<th>Df</th>
<th>Sum Sq</th>
<th>Mean Sq</th>
<th>F value</th>
</tr>
</thead>
<tbody>
<tr>
<td>nbh1</td>
<td>1</td>
<td>68.320</td>
<td>68.320</td>
<td>37.9483</td>
</tr>
<tr>
<td>nbh2</td>
<td>1</td>
<td>171.859</td>
<td>171.859</td>
<td>95.4584</td>
</tr>
<tr>
<td>pert</td>
<td>1</td>
<td>0.546</td>
<td>0.546</td>
<td>0.3035</td>
</tr>
<tr>
<td>nbh1:nbh2</td>
<td>1</td>
<td>50.378</td>
<td>50.378</td>
<td>27.9826</td>
</tr>
<tr>
<td>nbh1:pert</td>
<td>1</td>
<td>0.028</td>
<td>0.028</td>
<td>0.0154</td>
</tr>
<tr>
<td>nbh2:pert</td>
<td>1</td>
<td>0.181</td>
<td>0.181</td>
<td>0.1008</td>
</tr>
<tr>
<td>nbh1:nbh2:pert</td>
<td>1</td>
<td>1.449</td>
<td>1.449</td>
<td>0.8050</td>
</tr>
</tbody>
</table>
Matheuristic analysis: Visualization

nbh1 mean plot

nbh2 mean plot

pert mean plot
Matheuristic analysis: Visualization
Matheuristic analysis: Visualization

(pert, nbh1) interaction plot

(pert, nbh2) interaction plot
Matheuristic analysis: Visualization

- Pert box plot, nbh1=0, nbh2=0
- Pert box plot, nbh1=1, nbh2=0
- Pert box plot, nbh1=0, nbh2=1
- Pert box plot, nbh1=1, nbh2=1
#### Matheuristic analysis: Validation set

<table>
<thead>
<tr>
<th>r-instance-20-2-2-30</th>
<th>Matheuristic</th>
<th>CPLEX</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>obj</td>
<td>time</td>
</tr>
<tr>
<td>r-instance-20-2-2-30</td>
<td>28.1479</td>
<td>5.72</td>
</tr>
<tr>
<td>r-instance-30-2-2-30</td>
<td>24.7584</td>
<td>4.83</td>
</tr>
<tr>
<td>r-instance-50-2-2-11</td>
<td>11.8394</td>
<td>7.35</td>
</tr>
<tr>
<td>r-instance-50-2-2-11</td>
<td>22.7065</td>
<td>20.78</td>
</tr>
<tr>
<td>r-instance-100-2-2-5</td>
<td>35.7950</td>
<td>10.4</td>
</tr>
</tbody>
</table>

⇒ average optimality gap = 0.29792%
Concluding remarks & Future research

Design and analysis of metaheuristics:

Problem structure important for the metaheuristic design

The effect of the algorithmic components should be studied, taking into account:

- Quality of the initial solution
- Instance complexity
- Stopping criteria
...

Future research:

- Investigate other parameters
- Investigate computation time: Is it worth it?
- Investigate robustness of the matheuristic over different runs
...

Concluding remarks & Future research

Design and analysis of metaheuristics:
  - Problem structure important for the metaheuristic design

Future research:
  - Investigate other parameters
  - Investigate computation time: Is it worth it?
  - Investigate robustness of the matheuristic over different runs
  - ...
Concluding remarks & Future research

Design and analysis of metaheuristics:

- Problem structure important for the metaheuristic design
- The effect of the algorithmic components should be studied, taking into account
Concluding remarks & Future research

Design and analysis of metaheuristics:

▶ Problem structure important for the metaheuristic design
▶ The effect of the algorithmic components should be studied, taking into account
  ▶ quality of the initial solution
▶ instance complexity
▶ stopping criteria

Future research:
▶ Investigate other parameters
▶ Investigate computation time: Is it worth it?
▶ Investigate robustness of the matheuristic over different runs
Concluding remarks & Future research

Design and analysis of metaheuristics:

- Problem structure important for the metaheuristic design
- The effect of the algorithmic components should be studied, taking into account
  - quality of the initial solution
  - instance complexity

Future research:
- Investigate other parameters
- Investigate computation time: Is it worth it?
- Investigate robustness of the matheuristic over different runs
Concluding remarks & Future research

Design and analysis of metaheuristics:

- Problem structure important for the metaheuristic design
- The effect of the algorithmic components should be studied, taking into account:
  - quality of the initial solution
  - instance complexity
  - stopping criteria

Future research:
- Investigate other parameters
- Investigate computation time: Is it worth it?
- Investigate robustness of the matheuristic over different runs
- ...
Concluding remarks & Future research

Design and analysis of metaheuristics:

- Problem structure important for the metaheuristic design
- The effect of the algorithmic components should be studied, taking into account
  - quality of the initial solution
  - instance complexity
  - stopping criteria
  - ...

Future research:

- Investigate other parameters
- Investigate computation time: Is it worth it?
- Investigate robustness of the matheuristic over different runs
  - ...

...
Concluding remarks & Future research

Design and analysis of metaheuristics:

▶ Problem structure important for the metaheuristic design
▶ The effect of the algorithmic components should be studied, taking into account
  ▶ quality of the initial solution
  ▶ instance complexity
  ▶ stopping criteria
  ▶ ...

Future research:

▶ Investigate other parameters
▶ Investigate computation time: Is it worth it?
▶ Investigate robustness of the matheuristic over different runs
▶ ...

21/25
Concluding remarks & Future research

Design and analysis of metaheuristics:

- Problem structure important for the metaheuristic design
- The effect of the algorithmic components should be studied, taking into account
  - quality of the initial solution
  - instance complexity
  - stopping criteria
  - ...

Future research:

- Investigate other parameters
Concluding remarks & Future research

Design and analysis of metaheuristics:

- Problem structure important for the metaheuristic design
- The effect of the algorithmic components should be studied, taking into account
  - quality of the initial solution
  - instance complexity
  - stopping criteria
  - ...

Future research:

- Investigate other parameters
- Investigate computation time: Is it worth it?
Concluding remarks & Future research

Design and analysis of metaheuristics:

- Problem structure important for the metaheuristic design
- The effect of the algorithmic components should be studied, taking into account:
  - quality of the initial solution
  - instance complexity
  - stopping criteria
  - ...

Future research:

- Investigate other parameters
- Investigate computation time: Is it worth it?
- Investigate robustness of the matheuristic over different runs
Concluding remarks & Future research

Design and analysis of metaheuristics:

- Problem structure important for the metaheuristic design
- The effect of the algorithmic components should be studied, taking into account
  - quality of the initial solution
  - instance complexity
  - stopping criteria
  - ...

Future research:

- Investigate other parameters
- Investigate computation time: Is it worth it?
- Investigate robustness of the matheuristic over different runs
- ...

...


S. Duran, M. A. Gutierrez, P. Keskinocak, *Pre-positioning of emergency items for CARE international*, Interfaces 41.3 (2011) 223-237


L. V. Snyder, *Facility location under uncertainty: A review*, IIE Transactions 38.7 (2006) 537-554
References

J. Uichanco, *A robust model for pre-positioning emergency relief items before a typhoon with an uncertain trajectory*, submitted to M& SOM


A matheuristic for pre-positioning of disaster relief supplies

Renata Turkeš, Daniel Palhazi Cuervo, Kenneth Sörensen
March 17, 2016

This research was supported by the Interuniversity Attraction Poles (IAP) Programme initiated by the Belgian Science Policy Office (COMEX project).