Augmented Resource Allocation Framework for Disaster Scenarios

Final Presentation - Luke Guerdan

Panacea’s Cloud
Overview

1. Problem Motivation
2. Background
3. Previous Works
4. Solution
5. Methodology
6. Experimental Results
7. Conclusion
Disaster Resource Allocation: Motivation
How can we leverage?

- Centralized information storage
- Dynamic routing algorithms
- Human knowledge of the situation
Overview

1. Problem Motivation
2. Background
3. Previous Works
4. Solution
5. Methodology
6. Experimental Results
7. Conclusion
Responder: Individual who alleviates disaster conditions - firefighter, police, search and rescue

Patient: Anyone needing help from a responder - injured, trapped in a building, missing pet

Incident Commander: Makes decisions about how to respond
Overview

1. Problem Motivation
2. Background
3. Related Work
4. Solution
5. Experimental Results
6. Conclusion
Related Work: Analytic Hierarchy Process

- Each factor in response ranked by importance
- Importance determined by expert in disaster management

Problems:
- Requires manual configuration each emergency
- Priority changes
- Different for each emergency
Related Work: K shortest paths

- Uses Dijkstra’s shortest path algorithm
- Multiple ‘short’ paths generated

- Short paths determined by
  - Infrastructure damage
  - Patient priority
  - Accessibility
Definitions:
- $G(V, E)$: weighted directed graph, with set of vertices $V$ and set of directed edges $E$.
- $w(u, v)$: cost of directed edge from node $u$ to node $v$ (costs are non-negative).

Links that do not satisfy constraints on the shortest path are removed from the graph.
- $s$: the source node

$YenKSP(C, H, 3)$
Overview

1. Problem Motivation
2. Background
3. Related Work
4. Solution
5. Experimental Results
6. Conclusion
Augmented Resource Allocation
Augmented Annealing Algorithm

input: patient priority queue \( P \), road network matrix \( R.N. \)

output: shortest path matrix

\[ R.N. \leftarrow \text{initial } R.N. \text{ costs;} \]

while \( P \) is not \( \emptyset \) do

\[ \text{Patient} \leftarrow \text{dequeue highest priority } P; \]

\[ \text{recommended} \leftarrow \text{KDi}jkstra(R.N,\text{Patient}); \]

\[ \text{present possible paths to Incident Commander;} \]

if other than shortest selected then

\[ \text{decrease weights} \leftarrow \text{longer path set minus shorter path;} \]

\[ \text{increase weights} \leftarrow \text{shorter path set minus longer path;} \]

else

\[ \text{keep } R.N. \text{ constant;} \]

end

end
Overview

1. Problem Motivation
2. Background
3. Related Works
4. Solution
5. Case Study
6. Conclusion
Panacea’s Cloud
Overview

1. Problem Motivation
2. Background
3. Related Work
4. Solution
5. Case Study
6. Conclusion
Conclusion

- Disaster’s becoming more common
- Highly reliant on technology and internet access
- Need to adapt new innovations to disaster situations

- We’ve now established platform for future work
- Make Panacea’s Cloud more robust
- Leverage progress for future intelligent decisions
Thank You!

Luke Guerdan

lmg4n8@mail.missouri.edu