Meta-heuristic solution approach to the Isolated Community Evacuation Routing Problem (ICEP)

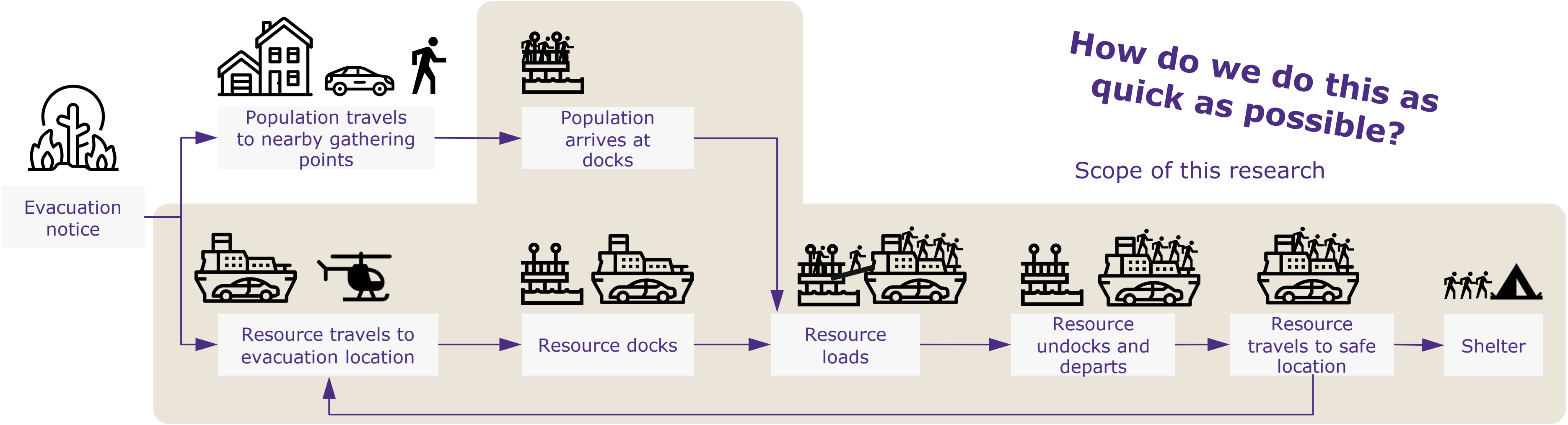
Klaas Fiete Krutein
Department of Industrial & Systems Engineering

UNIVERSITY of WASHINGTON

Motivation and Previous Work

Original Definition of the Isolated Community Evacuation Problem (ICEP) (Krutein & Goodchild, 2021)

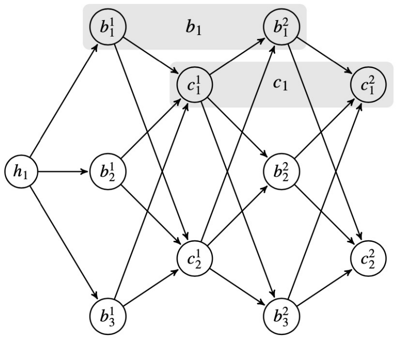
How to evacuate an isolated community without land-based evacuation routes as quickly as possible?

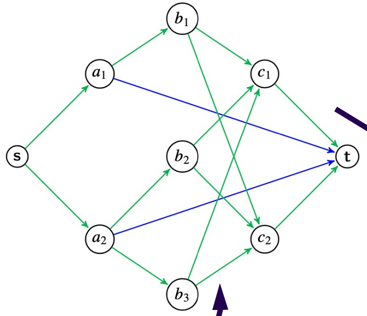


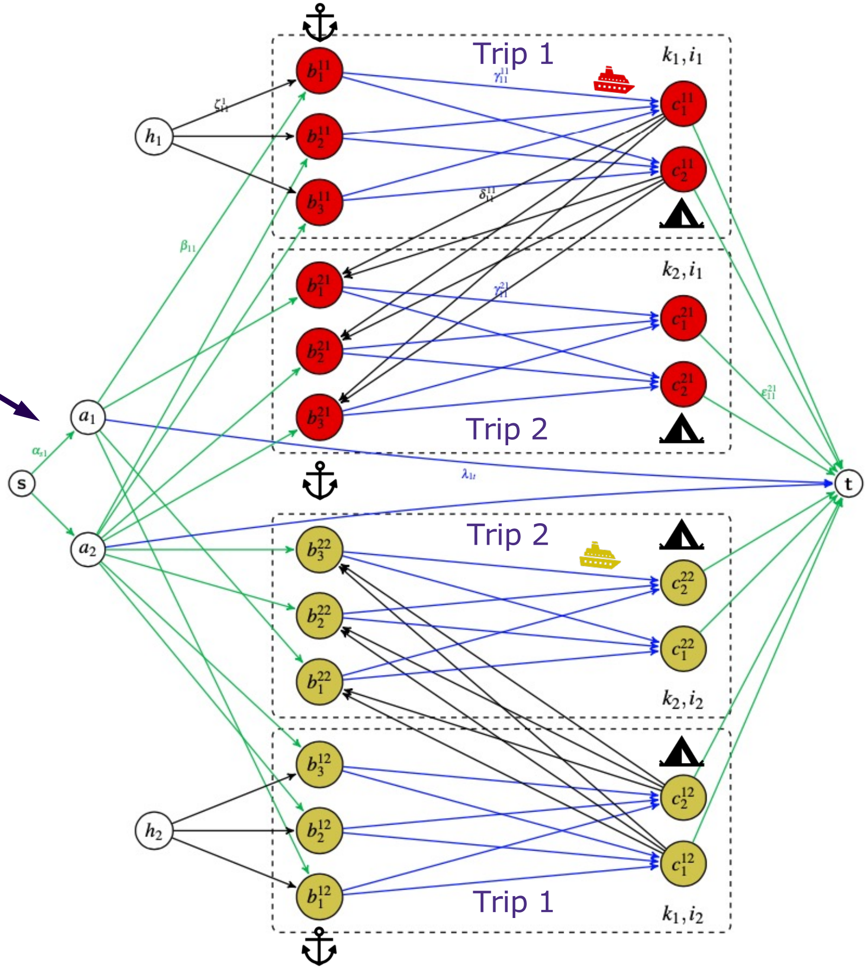
Icons provided by *Freepik* (house, dock, car), *Srip* (route), *Google* (pedestrian), *photo3idea_studio* (ferry), *monkik* (ship), *ultimatearm* (wildfire) from www.flaticon.com

ICEP Network

(Min total evacuation time)







Evacuation area

Pick-up location (dock)

Drop-off location

Evacuation resource 1

Evacuation resource 2

How to solve this problem?

Commercial solvers (e.g. CPLEX, Gurobi)

- Challenges:
  - Routing problems are NP-complete
  - Problem is very complex in structure and objective
  - Trip expansion generates many binary variables
- Consequences:
  - For many instances commercial solver takes very long

Previous Attempt: Constructive Greedy Heuristic

> 2 phase heuristic

– 1st phase goal: greedily generate a feasible route plan

– 2nd phase goal: improve the route plan through local search

Results from Heuristic Testing versus Gurobi 9.1

> Significantly faster than Gurobi for larger instances

> Not optimal in all cases (optimality gap)

> More complex decision rules cause run time increase

> Other ideas? → meta-heuristics

Literature Review

Meta-heuristics successfully applied to related routing problems

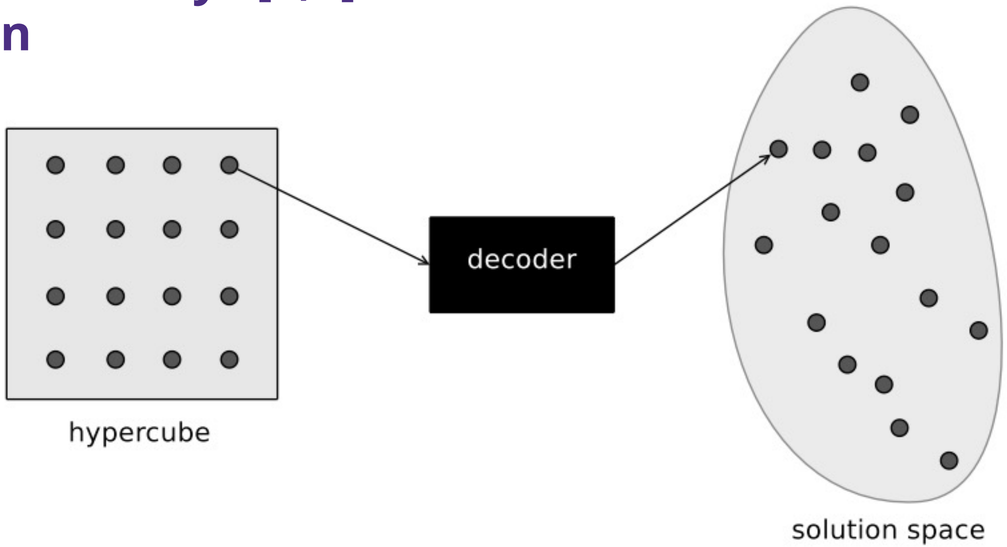
- **Simulated Annealing** (Kirckpatrick, 1983)
- **Tabu Search** (Glover, 1986, Goerigk, et al., 2014)
- **Greedy Randomized Adaptive Search** (Resende & Ribeiro, 2016)
- **(Biased) Random-Key Genetic Algorithm** (Bean 1994, Gonçalves & Resende, 2011)

Chosen Methodology: Biased Random Key Genetic Algorithm (BRKGA)

> Reasons:
  - Feasible region of ICEP very complex
  - BRKGA generates feasible solution in every iteration
  - Population based structure is promising to avoid local minima effectively
  - Proven track record for solving routing problems

Random-Key Genetic Algorithm (Bean, 1994)

- Simplification of solution representation
- Use random keys [0,1] instead of variable values to represent solution



Source: Gonçalvez and Resende, 2011

Developed Methodology

Developed Chromosome Decoder Logic

Step 1

1	2	…	s	s+1	s+2	…	t	t+1	…	n
0.2	0.9	…	0.8	0.4	0.6	…	0.3	0.2		0.3

1	2	…	s	s+1	s+2	…	t
0.2	0.9	…	0.8	0.4	0.6	…	0.3

Scenario level

Resource level

s+1	s+2	t-1	t
0.4	0.6	0.1	0.3

Mapping	Index	Dock
0	0	None
0.167	1	Evac Dock 1
0.333	2	Evac Dock 2
0.5	3	Safe Dock 1
0.667	4	Safe Dock 2
0.833	5	Safe Dock 3

Route plan Resource 1
Evac Dock 2
Safe Dock 1
None (Stay)
Evac Dock 1

Developed Chromosome Decoder Logic

Step 2

Route plan Resource 1
Evac Dock 2
Safe Dock 1
None (Stay)
Evac Dock 1

Route plan Resource 2
Evac Dock 1
Evac Dock 2
Safe Dock 1
None (Stay)

1. Order all arrivals

Ordered arrivals	Arrival time
R2: initial loc → Evac Dock 1	3:00 pm
R1: initial loc → Evac Dock 2	3:05 pm
R2: Evac Dock 1 → Evac Dock 2	3:20 pm
R1: Evac Dock 2 → Safe Dock 1	3:25 pm
R2: Evac Dock 2 → Safe Dock 1	3:40 pm

2. Allocate evacuees

Evacuees allocated
min(remaining evac. at ED1, remaining cap. R2)
min(remaining evac. at ED2, remaining cap. R1)
min(remaining evac. at ED2, remaining cap. R2)
Unload all evacuees on R1
Unload all evacuees on R2

3. Delete all trips after full allocation

4. Evaluate fitness of plan

Experiment Results

Experiment Results

Data label	No. resources	No. docks	Scenarios	Gurobi 9.1		BRKGA (concurrent)		BRKGA (Parallelized)	
				Solution time	Objective	Solution time	Objective	Solution time	Objective
Test 1	6	7	2	5.51s	101.03	109.77s (last imp.)	172.00	142.42s	124.00
Test 2	4	5	2	2.36s	56	188.13s (last imp.)	56.67	17.65s	56.67
Test 3	2	5	2	116.15s	229	375.28s (last imp., ran for 3600s)	324.00	928.2s	232.64
Test 4	5	8	3	3600s (aborted)	113.04	805.57s (last imp., ran for 3600s)	291.39	671.39s	259.73
Test 5	20	6	4	3600s (aborted)	78.04	1217.39s (last imp.)	218	908.63s	108.03

Findings

- Many solutions generated by decoder are sub-optimal
- Solution discrimination is still difficult
- Evolution in BRKGA is too slow to compete with Gurobi, even in parallelized case
- Decoder gets stuck at local optima in the large feasible region

Conclusions

> Escaping local minima is an ongoing challenge

Next Steps

> Experiment with algorithm restarts, adaptive randomization rates and path relinking
> Adding bias to decoder

Questions and Answers