Using Reinforcement Learning to provide decision support in multi-domain mass evacuation operations

Mark Rempel and Nicholi Shiell

Centre for Operational Research and Analysis

17 October 2022

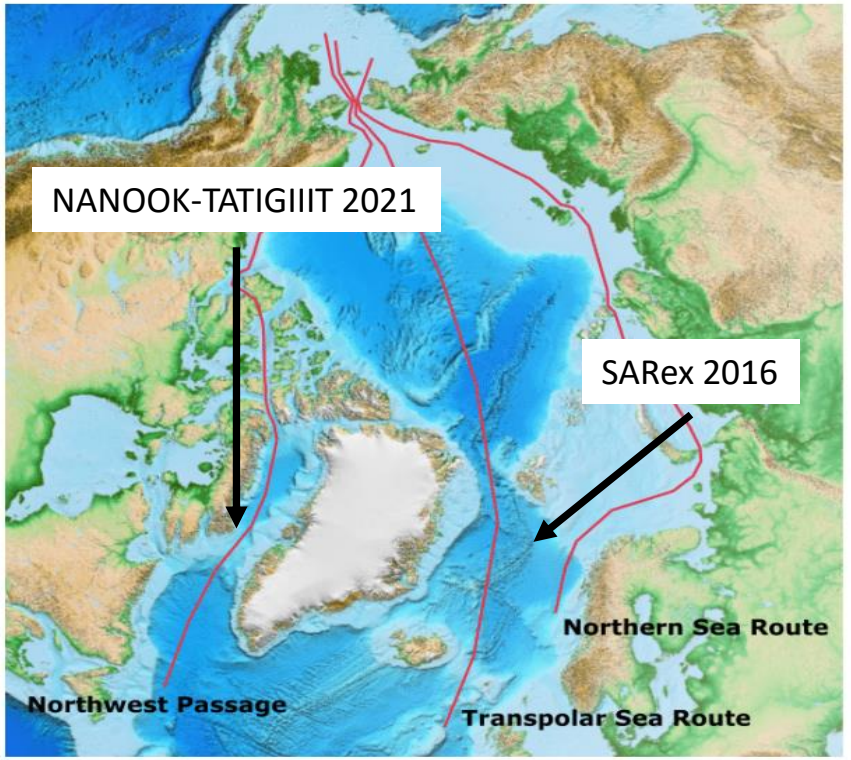
Outline

- Introduction
- Problem definition
- Methodology
  - Markov decision process formulation
  - Approximate dynamic programming formulation
- Computational results
- Conclusion

Introduction

Introduction

- Arctic sea ice has been decreasing for decades, the result …navigation of the Arctic sea routes will become commonplace
- Polar code sets requirements for ships operating in the regions, but governments play a key role in Search and Rescue (SAR)
- Many factors affect the outcome of a MAJMAR: response time, environment, infrastructure, decision policies, etc.
- **Decision policy**: “a rule (or function) that determines a decision given the information available” (Powell, 2011, p. 221)
- Study a Multi-Domain Operation (MDO) in response to
  - a Arctic MAJMAR that occurs in a remote location, with limited resources available to respond, and evac of ~2000; and
  - seek a policy to decide who to evacuate in each domain in order to maximize the number of survivors



Exercises focusing on responding to a Major Maritime Disaster (MAJMAR) in the Arctic

Introduction—what is unique about this study?

- Seeking a policy for allocation of resources to maximize the number of survivors:
  - Beyond defence: blunt trauma, mass casualty events, healthcare facility evacuation, etc.
    - Focus on a relatively small number of people, 10 to 100 (Jacobson et al., 2012; Shin and Lee, 2020)
    - Large number of people, ~100,000, but in a large urban centre (Caglayan, 2021)
  - Within defence:
    - Air domain operation in response to a MAJMAR (~2000 people) (Rempel et al., 2021)
- Large-scale sequential decision problem, Reinforcement Learning (RL) / Approximate Dynamic Programming (ADP) methods may be useful to find near-optimal policies
  - Applying RL / ADP in the context of MDOs is sparse in the open literature

To the best of our knowledge, using RL / ADP to seek near-optimal decision policies in MDOs is either not wide spread or non-existent in the open literature.

Introduction—two main contributions

#1: Formulate multi-domain mass evacuation operation as a Markov Decision Process (MDP) using Powell's Notation for a sequential decision problem

#2: Learn a near-optimal evacuation decision policy that coordinates efforts across multiple domains with the objective to maximize the number of survivors

#2a: Design an ADP-based algorithm to learn a policy using a representative scenario

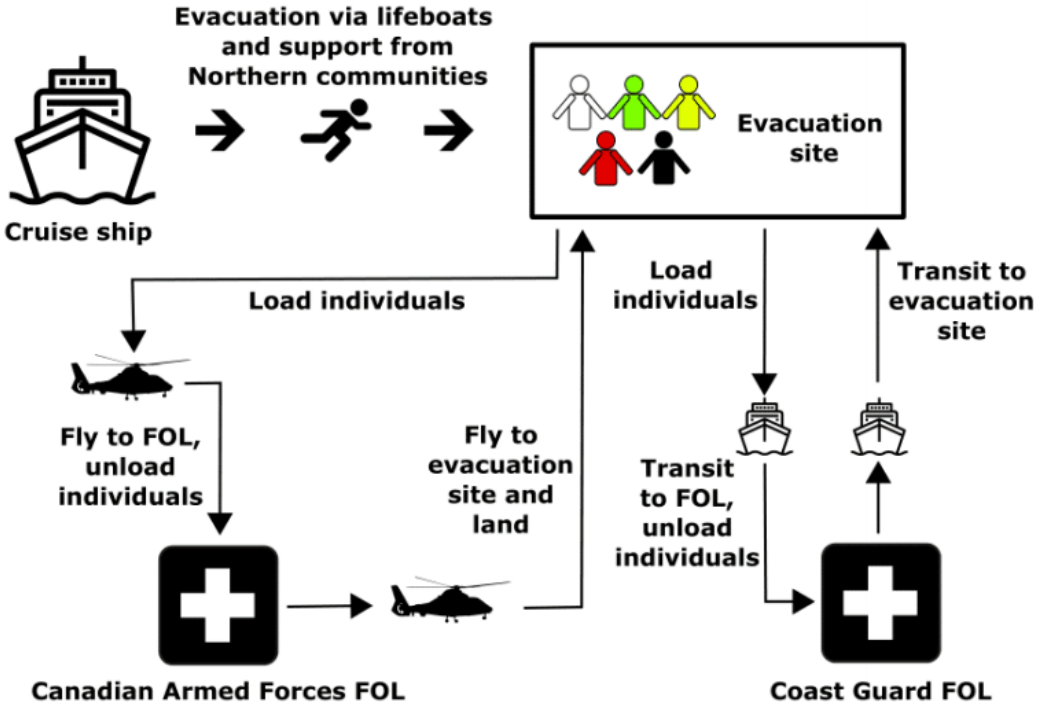
#2b: Compare the ADP-generated policy with benchmark policies

#2c: Test the ADP-generated policy in a range of scenarios and compare to outcomes achieved using non-coordinated policies

Problem definition

Problem definition—representative scenario

- MAJMAR scenario in the Arctic, focusing on transport of 2000 individuals from an evacuation site to a Forward Operating Location (FOL)—based on Hunter et al., 2021; Rempel et al., 2021



Initial distribution of evacuees medical status. Transition between categories follows an exponential distribution.

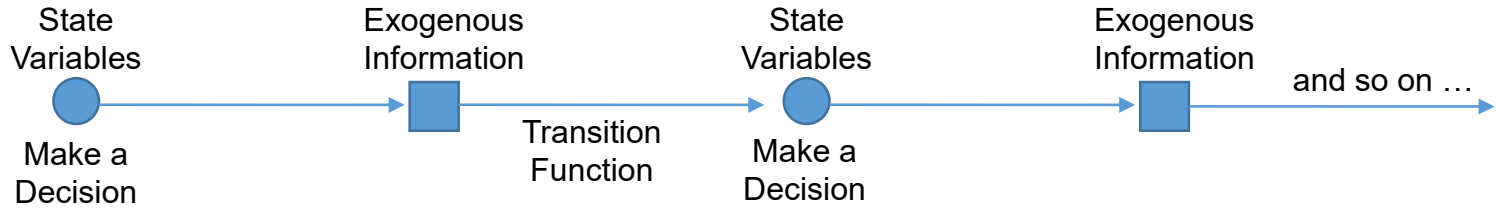
Category	Treatment	Initial count	Stretcher?	Mean time [h]
White	None	1900	No	120
Green	Routine	40	No	48
Yellow	Early	30	Yes	8
Red	Immediate	30	Yes	1.5
Black	Deceased	0	No	-

Methodology

Markov decision process formulation

Contribution #1

- Markov Decision Process—model of sequential decision making under uncertainty
- Powell's sequential decision modelling notation (Powell, 2011, ch. 5)
  - **State variables**—describe the state of the system at a given decision epoch—time, event (helicopter or ship ready to load), individuals in each triage category at the evacuation site
  - **Decision variables**—variables the decision maker controls—how many to load from each triage category constrained by available space
  - **Exogenous information**—information that arrives after a decision is made—change in number in each triage category, type of next transport to arrive, time of next arrival
  - **Transition function**—describes how the system evolves from one state to the next
  - **Objective function**—describes the decision makers goal—maximize the expected number of individuals that are loaded onto the helicopter(s) or ship(s)



Optimal policy is given by Bellman's equation (Bellman, 1957)

Approximate dynamic programming formulation

- Curses of dimensionality: size of state space, size of decision space, size of exogenous info
  - Billions of possible state space transitions due to changes in medical condition
- ADP overcomes these curses, but seeks a near-optimal policy
  - Post-decision state variable—four dimensional vector, number of individuals per triage
  - Approximating the value function—lookup table, state aggregation based on hierarchical aggregation and discretization of the state space into intervals

$$V_t(S_t) = \max_{x_t \in \mathcal{X}_t} (C(S_t, X^\pi(S_t)) + \gamma \mathbb{E}(V_{t+1}(S_{t+1}|S_t))$$

Bellman's equation

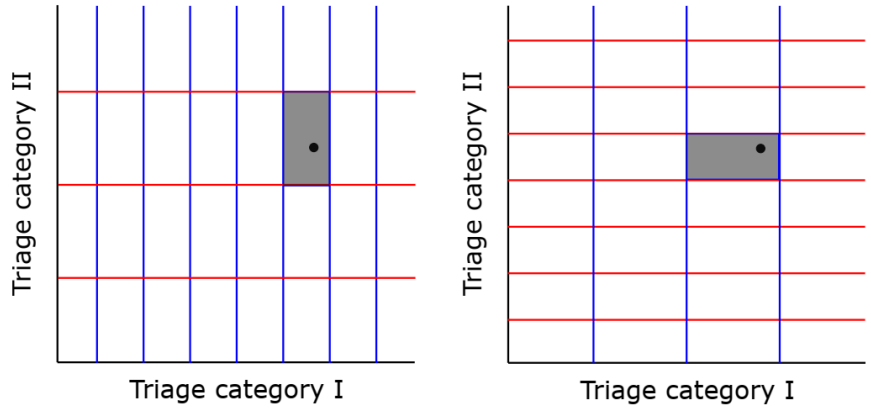
$$V^x_{t-1}(S^x_{t-1}) = \mathbb{E}(\max_{x_t \in \mathcal{X}_t} (C(S_t, X^\pi(S_t)) + \gamma V^x_t(S^x_t)|S^x_{t-1}))$$

with approx.

Approx. of the future value of a decision

Post-decision state variable

Two-dimensional example of state aggregation approach employed



Computational results

Computational results—ADP-based algorithm

Contribution #2a

- Designed an ADP-based algorithm to learn a near-optimal policy in the representative scenario
  - One helo (capacity 10, arrives hour 48, 3 hour revisit), one ship (capacity 50, arrives hour 4, 16 hour revisit)—individuals consume the same capacity on each
  - Policy performance—expected total number of individuals loaded onto helo and ship
- State aggregation scheme—four encodings
- Approximate Value Iteration (AVI) algorithm
  - Number of iterations: 10^7
  - Learning rate: Generalized harmonic stepsize
  - Exploration rate: 0.25
- Output is a learned / coordinated policy for the helo and ship evacuation routes

Scenario parameters

Type	h^H/h^S	i	r	Δ^H/Δ^S
Helo	10	48	3	[1,1,3,3]
Ship	50	4	16	[1,1,3,3]

State aggregation scheme—number of bins per dimension

Encoding 1	Encoding 2	Encoding 3	Encoding 4
$\{50, 50, 50, 100\}$	$\{50, 50, 100, 50\}$	$\{50, 100, 50, 50\}$	$\{100, 50, 50, 50\}$

Computational results—policy evaluation

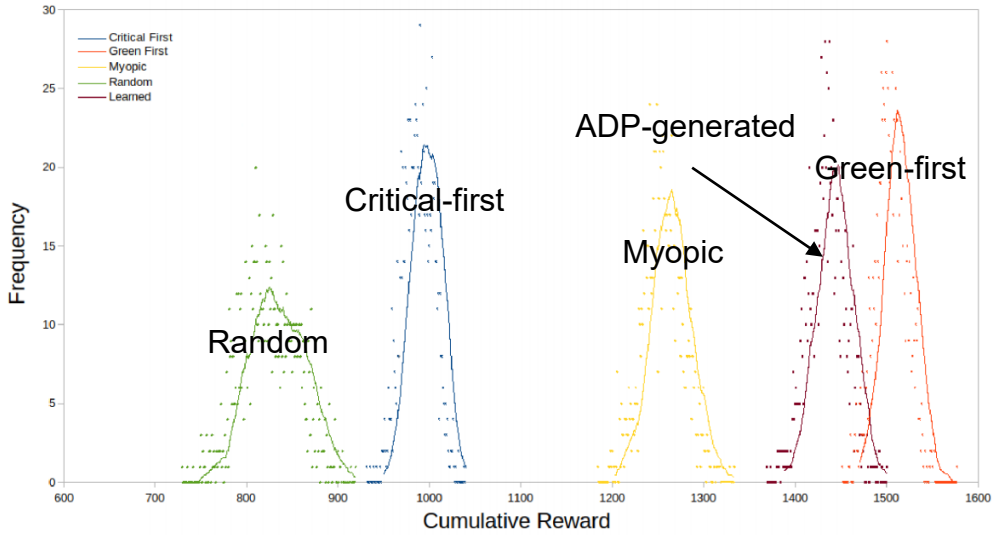
Contribution #2b

- ADP-generated policy compared to four benchmark policies
  - **Green-first**: Green, White, Red, Yellow (PFA)
  - **Myopic**: Maximize the number of individuals per epoch (CFA)
  - **Critical-first**: Red, Yellow, Green, White (CFA)
  - **Random**: Random selection (PFA)

Key takeaway: Results demonstrate that in the context of the scenario, evacuating healthy individuals first maximizes the number of survivors—"[starting] from the most urgent jobs and [moving] onto those that are less urgent as resources become available" (Jacobson et al., 2012, p. 813) is not the best policy.

Benchmarks vs ADP-generated

Policy	Mean	Variance	Comparison	Rank
Green-First	1504	17.0	-5 ± 3	1
ADP-generated	1434	21.2	-	2
Myopic	1255	23.0	12 ± 6	3
Critical-First	987	16.5	31 ± 17	4
Random	823	31.3	42 ± 16	5

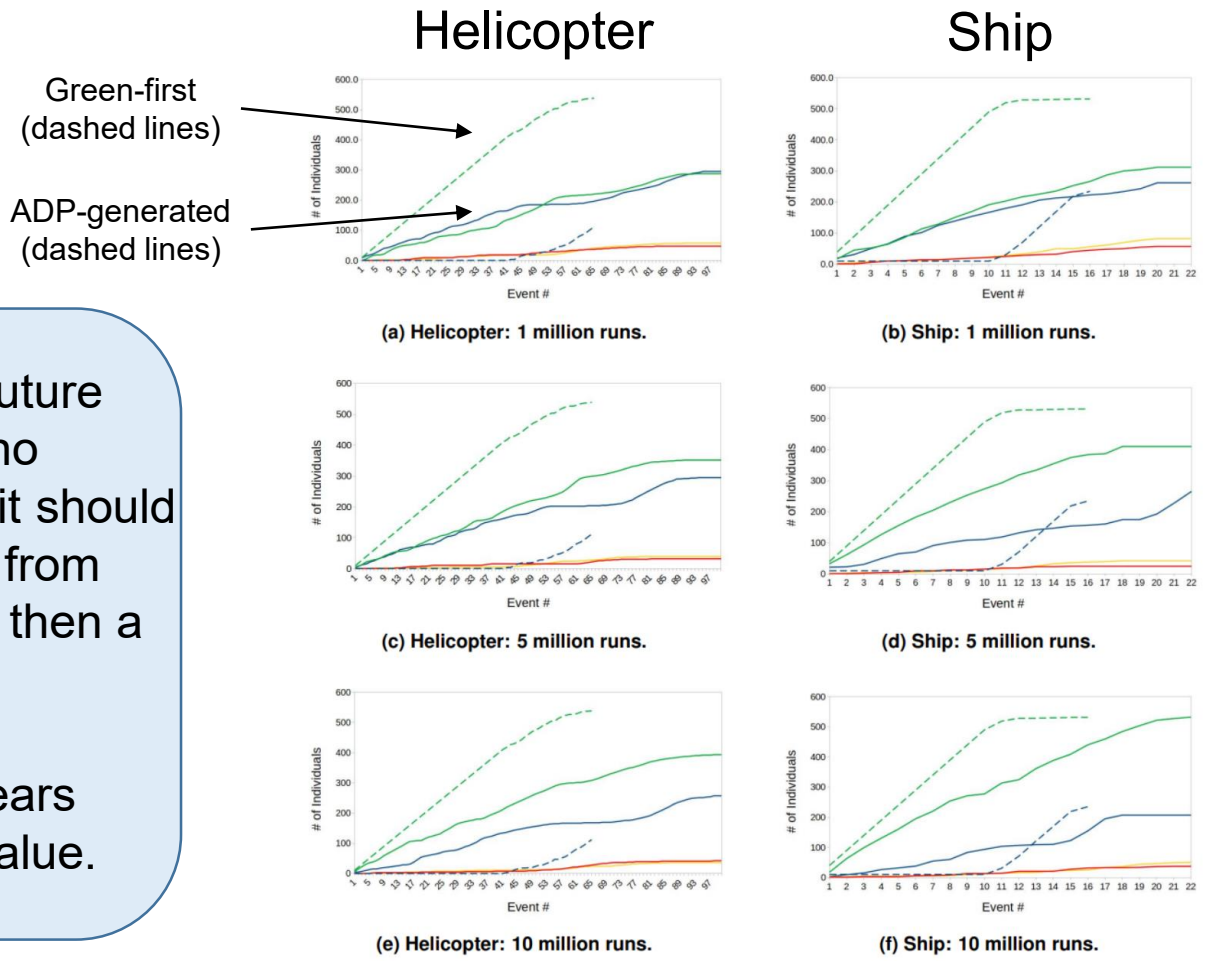


Computational results—policy evaluation (continued)

- ADP-based algorithm learns a policy (labelled ADP-generated) close to the Green-first policy
- This gives us confidence that the ADP-generated is near-optimal

Key takeaway: Results indicate that incorporating the future value of decision within the evacuation policy provides no value to the decision maker. This agrees with Powell—"it should not be surprising to find out that if it is possible to move from any state to any other state (instantly and with no cost), then a myopic policy will be optimal" (Powell, 2011, p. 594).

A myopic may be a PFA or CFA, and in this case it appears to be a PFA (Green-first) generates the best objective value.



Computational results—test scenarios

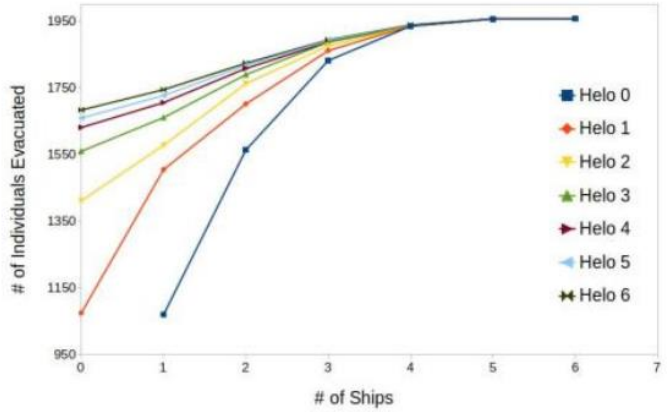
Contribution #2c

- Test scenarios vary the number of helicopters (zero to six) and ships (zero to six)—48 scenarios in total, MDP executed 35 times for each
- Green-first outperforms the ADP-generated policy in all scenarios
- When four or more ships are available, helicopters provide little benefit

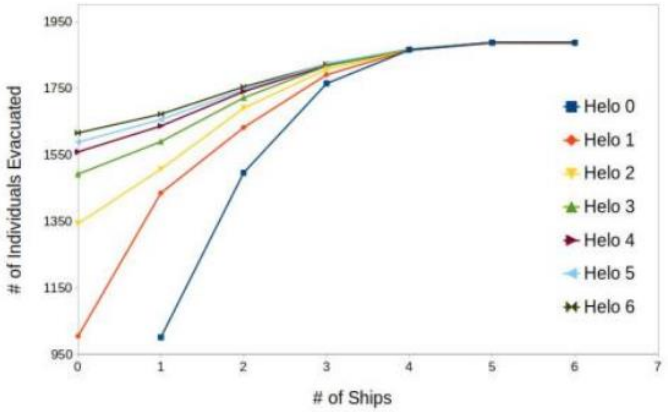
	Initial arrival time (h)	
	Helicopter	Ship
1	48	4
2	49	5
3	50	6
4	51	7
5	52	8
6	53	9

Key takeaway: Coordination of loading policies may not be required—ADP-generated policy performed similar to the Green-first policy.

.



(a) Green-first policy.



(b) ADP-generated policy.

Conclusion

Conclusion

- Examined a MAJMAR scenario in a remote Arctic location, 2000 individuals must be evacuated, and their medical condition stochastically deteriorates over time
- MDO, individuals are evacuated by ship or air, with the aim to maximize the number of survivors—modelled the problem as an MDP using Powell's notation, and solved it via ADP using a lookup table representation based on state aggregation
- Compared the ADP-generated policy to four benchmark policies (random, myopic, critical-first, Green-first)—ADP-generated policy performed similar to the Green-first policy
- Incorporating a decision's future value may provide no benefit in the scenario studied, and thus coordination between the domains' policies may not be required—focus on the here and now
- Although the learned policy did not outperform the benchmarks, AI frameworks may be used to determine and test policies and provide effective decision support in MDOs

Questions?