Centralized vs Individual Models for Decision Making in Interconnected Infrastructure

Stephanie Allen [1] **John P. Dickerson** [2 1] **Steven A. Gabriel** [3 1 4]

Abstract

The 2013 *National Infrastructure Protection Plan* (DHS, 2013) outlines the need for interconnected infrastructure systems to coordinate more and recognize their interdependencies. We model the two extremes of this coordination spectrum using two different multi-agent models: (a) a model called the "centralized model" in which the agents act as one unit in making decisions and (b) a model called the "individual model" in which the agents act completely separately and have either a pessimistic or optimistic assumption regarding the damages of the other infrastructure systems controlled by the other agents. We then use the individual model to establish a point along the coordination spectrum by providing the individual agents with delayed information from the other player(s). To test this framework, we use a small but illustrative model from a Yu & Baroud paper in which there is a power and a water network, and we assume that there are operators for both networks that would like to maximize flow in the network (2020). Our results comparing partially repaired networks using the two models find that: (i) the centralized model acts as an upper bound on the individual model in terms of our flow metric and (ii) the delayed information individual model leads to less variability in results compared to the other individual model assumptions which points to the value of some coordination in decision making.

[1]Applied Mathematics, Statistics, & Scientific Computation Program, University of Maryland, College Park, Maryland, US [2]Department of Computer Science, University of Maryland, College Park, Maryland, US [3]Department of Mechanical Engineering, University of Maryland, College Park, Maryland, US [4]Department of Industrial Economics and Technology Management, Norwegian University of Science and Technology, Trondheim, Norway. Correspondence to: Stephanie Allen <sallen7@umd.edu>.

2nd AI4Science Workshop at the 39^{th} International Conference on Machine Learning (ICML), 2022.

1. Introduction

Modeling the decisions made by those responsible for the components of interconnected infrastructure systems in response to disasters helps us better understand the best ways to approach coordination between these entities. Indeed, the first three tenets in the 2013 *National Infrastructure Protection Plan* (DHS, 2013) state:

- "Risk should be identified and managed in a coordinated and comprehensive way across the critical infrastructure community to enable the effective allocation of security and resilience resources."
- "Understanding and addressing risks from cross-sector dependencies and interdependencies is essential to enhancing critical infrastructure security and resilience."
- "Gaining knowledge of infrastructure risk and interdependencies requires information sharing across the critical infrastructure community."

The first and third items point to the need for coordination between infrastructure components, and the second item elucidates the existence of interdependencies among infrastructure components. Our paper explores the points brought up in this report by modeling the two ends of the coordination spectrum, with a model in which infrastructure components are completely centralized in their decision making and a model in which the components make their own decisions with either very pessimistic or very optimistic assumptions regarding the other network. We also use the individual model to simulate the situation of delayed information sharing between agents, meaning the individual agents still make decisions separately but have knowledge of the previous state of the other players' network states. Overall, as our experiments demonstrate, the centralized model acts as best case scenario model against which to compare the individual models for performance.

2. Literature Review

We include literature in this review that pertains to Markov decision process (MDP) models or reinforcement learning techniques applied to interconnected infrastructure in

the event of some kind of disaster. First, there are single-player Markov decision process models for interconnected infrastructure (Espada Jr, 2014) that use linear programming (Huang et al., 2017) along with decomposition techniques (Huang et al., 2018) as well as approximate dynamic programming techniques (Nozhati, 2021; Nozhati et al., 2020) and Monte Carlo techniques (Khouj et al., 2014; 2018). These models then move into reinforcement learning (Lopez et al., 2018; Alutaibi, 2017; Sun & Zhang, 2020) and deep reinforcement learning (Ishigaki et al., 2020; Memarzadeh & Pozzi, 2019) techniques for single-player critical infrastructure applications. Our concentration is on cooperative, multi-agent MDP and multi-agent reinforcement learning methods for disaster affected interconnected infrastructure systems, which is an underdeveloped literature. Some exceptions are a multi-agent reinforcement learning paper (Megherbi et al., 2013) and three deep reinforcement learning papers (Rajulapati et al., 2020a;b; Srikanth et al., 2021) that fit this description. There are a few additional papers that focus upon *non-cooperative* infrastructure applications (Panfili et al., 2018; Ni & Paul, 2019).

In Megherbi et al., they focus on experiments with reinforcement learning in which the multiple agents involved either did or did not pass information to each other, in which they found that passing information leads to quicker learning (2013). Our paper is different from Megherbi et al. because: (i) Megherbi et al. does not truly deal with interconnected infrastructure networks in that they do not model connections between different infrastructure systems and (ii) we focus on centralized versus individualized control, along with a delayed information sharing model, thus presenting more cases for interaction among agents and providing a best case scenario against which to compare models (2013). Rajulapati et al.'s papers use the multi-agent deep deterministic policy gradient (MADDP) algorithm in which the agents involved are trained using "centralized training" and "decentralized execution" techniques, meaning that the agents are provided with global information during the training phase but are not provided with this information after the training phase (2020a; 2020b). Srikanth et al. use the MADDP algorithm in the context of Covid-19 interconnected infrastructure presumably with the same training scheme (2021). Our paper differs from these last three papers because they only consider the case of decentralized control and never consider the case in which complete centralization could make the situation better. We also consider network measures of reward, whereas the reward mapping is not explicitly specified in these three papers.

2.1. Our Contribution

Our contribution to the literature is to provide some additional models for multi-agent coordination in interconnected infrastructure. We also shine a light on the policy objective of having interconnected infrastructure systems coordinate more by showing the two ends of the spectrum of coordination along with a point along the continuum where there is delayed information sharing. Furthermore, we contribute to the literature by explicitly modeling interconnected infrastructure outcomes using a network measurement of reward.

3. Markov Decision Processes

Before moving into our multi-agent models, we provide some background regarding Markov decision processes. As defined by Sutton and Barto, a Markov decision process for an agent i over T total time steps can be defined by a few sets and functions as follows (2018):

- $\mathcal{S}$ as the set of states in which the agent can exist.
- $\mathcal{A}$ as the set of actions the agent can take. $\mathcal{A}(s)$ can be a function of the state the agent in which the agent is currently.
- $p(s', r|s, a) = Pr\{R_{t+1} = r, S_{t+1} = s'|S_t = s, A_t = a\}$, as the probability of transitioning to state s' and obtaining reward r at time $t+1$ given that the agent began at state s and took action a at time t. This definition expresses the Markov Property of the next state only depending upon the previous state and action.
- $r(s, a) = \mathbb{E}[R_{t+1}|S_t = s, A_t = a] = \sum_{r\in\mathcal{R}} r \sum_{s'\in\mathcal{S}} p(s', r|s, a)$, as the reward at state s and action a, requiring us to take the expectation over all rewards.
- $\pi(a|s)$ as the policy. It represents the probability of the agent choosing action a given that the agent is at state s.
- The state-value function is defined, with discount factor $\gamma \in [0, 1)$, as:

$$v_\pi(s) = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s \right] \tag{1}$$

- The state-action-value function is defined as:

$$q_\pi(s, a) = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | S_t = s, A_t = a \right] \tag{2}$$

The Bellman Equation is defined as the optimal state-action-value function for a pair (s, a) in the logical progression as

follows (Sutton & Barto, 2018):

$$\begin{aligned} q_*(s,a) &= \max_{\pi} q_\pi(s,a) \\ &= \mathbb{E}\left[R_{t+1} + \gamma v_*(S_{t+1}) | S_t = s, A_t = a\right] \\ &= \mathbb{E}\left[R_{t+1} + \gamma \max_{a'} q_{\pi *}(S_{t+1}, a') | S_t = s, A_t = a\right] \\ &= \sum_{s',r} p(s', r|s, a)\left[r + \gamma \max_{a'} q_*(s', a')\right] \end{aligned} \tag{3}$$

We obtain a recursive equation which defines the Bellman equation. The third step in (3) illustrates the greedy nature of the process of finding an optimal policy (Sutton & Barto, 2018) because it demonstrates that, as long as one knows the next steps ahead are optimal, one can choose the action that maximizes reward in this time period. Indeed, this is the underpinning of the dynamic programming algorithms which suppose that policies can be improved with this greedy procedure (Sutton & Barto, 2018).

For our multi-agent models, our transition function is deterministic so, although we presented the MDP model with a probabilistic transition function for the sake of generality, we will define our multi-agent models with a deterministic transition function that still preserves the Markov Property.

4. Multi-Agent Models

The definition of a multi-agent Markov decision process (MMDP) model comes jointly from Boutilier (1996) and Oliehoek & Amato (2016) with Boutilier describing it as thinking of the agents as having "one mind" and Oliehoek & Amato describing it "as a regular MDP with a 'puppeteer' agent that selects joint actions". Our centralized model flows directly from these two definitions, with the agents acting "as one mind" (Boutilier, 1996) and representing the idealized situation in which the agents would be able to take advantage of "global information" toward a shared goal (Oliehoek & Amato, 2016). The individual model, on the other hand, treats the agents entirely separately, with each agent making its own decisions and receiving its own rewards given either essentially no information about the other player or delayed information about the other player, which will be explained in the more detailed explanation of the model. Therefore, we can call the centralized model a multi-agent Markov decision process, but the individual model needs a new term, which we assign as "multiple MDP model" (MuMDP) because it recognizes that multiple single agent MDP models are being solved for multiple agents.

These models will be defined specifically for our application of an interconnected water and power system, and we consider the specific disaster of an earthquake for the transition functions.

Centralized Model: We define the centralized model as follows:

- State S_t: A vector that can be split into two parts: S_t^{water} and S_t^{power}. Each part has three components, with each part the length of the number of nodes of the infrastructure sub-network (water and power in our case). The three components are:
  - the initial damage state;
  - the number of days that have been spent repairing the network; and
  - the amount of damage the node currently has.

  Importantly, there are 5 initial damage states, taken from the FEMA HAZUS Earthquake Manual (FEMA, 2020), that correspond to undamaged, slight, moderate, extensive, and complete (with corresponding numbers 0, 1, 2, 3, and 4, respectively).

- Action $A_t = A_t^{\text{water}} \times A_t^{\text{power}}$: Represents the joint pair of actions that the water and power operators take to repair nodes in their respective networks. We assume each time period is a day, and thus the operators can send resources to repair one node per day. Depending upon the initial damage, this repair will lead to different amounts of repair of damaged components.

- Transition $S_{t+1} = T(S_t, A_t)$: Represents the transition function from S_t, A_t to S_{t+1}. This function adjusts S_t according to the A_t by adding 1 more day of work to any component that has been chosen by A_t. It also adjusts the damage levels using a combination of (a) the initial damage and (b) the number of days that work has been done on each of the nodes. The FEMA HAZUS Documentation (FEMA, 2020) has restoration functions for the water and power systems according to their initial damage categorization, so our code implements these curves. Thus, the correct restoration curve is chosen for the correct component (supply, demand, transition/transmission), and then the number of days of work is plugged into this restoration curve. These curves each have their own mean and standard deviation regarding the percentage of functionality the network component has attained for the number of days upon which it has been worked. We use the cumulative distribution functions.

- Reward $R(S_{t+1})$: Represents the reward obtained at time t. We calculate the reward based upon the maximum amount of flow that can be sent between:
  - The water supply and water demand nodes (water flow);
  - The power supply and power demand nodes (power flow); and

– The water demand and power supply nodes (water flow) and the power demand and water supply nodes (power flow).

Importantly, we should note that, based upon the implementation of our algorithms, we do not make the distinction between arcs devoted to water and arcs devoted to power. The arcs in the network are assumed to be able to handle both water and power. The capacity of the arcs is determined by the nodes incident to them. The determination occurs by taking the minimum of the repair levels of the two nodes i and j, which produces the capacity level for the arc (i, j). We use the Python package `networkx` to determine the maximum flow between the pairs of nodes listed above (Hagberg et al., 2008). This approach to measuring critical infrastructure repair was inspired by Dueñas-Osorio et al. (2007).

Individual Model: The individual model shares elements with the centralized model, but the key difference is that the water and power operators act independently with either a completely pessimistic of each other's situations, a completely optimistic view of each other's situations, or delayed information regarding each other's situations. This approach was inspired by Riel et al. (2017) and Talebiyan & Duenas-Osorio (2020).

- State S_t: There are two *separate* vectors S_t^{water} and S_t^{power}. They have the three components discussed in the previous formulation.
- Action A_t: There are separate action spaces for water and power instead of a joint action space. Therefore, we have A_t^{water} and A_t^{power} separately.
- Transition $S_{t+1}^{\text{water}} = T(S_t^{\text{water}}, A_t^{\text{water}})$ and $S_{t+1}^{\text{power}} = T(S_t^{\text{power}}, A_t^{\text{power}})$: The transition functions perform the same routines as the transition function from the previous set up, but we make it clear that there are separate transition functions for the two players.
- Reward $R^{\text{water}}(S_{t+1}^{\text{water}})$ and $R^{\text{power}}(S_{t+1}^{\text{power}})$: This is the most important part of this MuMDP formulation. Water and power have explicitly separate reward functions, in which the water reward function places all 0s, all 1s, or the previous state capacities for the power node capacities and the power reward function places all 0s, all 1s, or the previous state capacities for the water node capacities. The 0s correspond to each player thinking the other has been completely destroyed by the disaster (pessimistic assumption), the 1s correspond to each player thinking the other sustained no damage after the disaster (optimistic assumption), and the previous state capacities represent delayed information about the other player's state after the disaster. These modeling choices were certainly influenced by Talebiyan & Duenas-Osorio's Judgement Call mechanic which assumes different information assumptions between players (2020). The reward functions then calculate the maximum amount of flow according to the three tuples outlined in the reward function sub-section of the centralized model for each of the new augmented state vectors.

5. Algorithms to Solve Multi-Agent Models

As discussed in the beginning of Section 4, we have the centralized MMDP model and the individual MuMDP model. As discussed by both Boutilier (1996) and Oliehoek & Amato (2016), we can use single agent MDP solution techniques for MMDP models. We can also use single agent MDP solution techniques for our MuMDP model because this model is comprised of single agent MDPs. Consequently, we use policy iteration for the centralized MMDP model and Q learning for the individual MuMDP model (Sutton & Barto, 2018). As described by Sutton and Barto, policy iteration falls in the class of dynamic programming algorithms and, essentially, it works by iteratively updating the value function with the current policy and, then, using the updated value function to find a better policy (2018). In Q learning, the algorithm makes use of the Bellman Equation discussed in Section 3 by taking a linear combination of the previous value of the state-action pair and the bracketed part of the last line of equation (3) to produce the following update (Sutton & Barto, 2018):

$$\begin{aligned} Q(S_t, A_t) \leftarrow & (1-\alpha) Q(S_t, A_t) \\ & + \alpha \left(R_{t+1} + \gamma \max_a Q(S_{t+1}, a) \right) \end{aligned} \tag{4}$$

See Algorithms 1 and 2 for more details regarding policy iteration applied to the centralized model and Q learning applied to the individual model. We note for the Q learning algorithm (Algorithm 2) the following; when we run the Q learning algorithm, we run it for a specific S_0 so, for every iteration k, the S_0 is the same. This means that the Q learning algorithm does not have a set of initialized states from which it has to start searching; it repeatedly starts from one initial state (Sutton & Barto, 2018). Thus, for the experiments described in Section 6, Q learning has to be run for each trial. We also note for Algorithm 2 that the "current belief regarding state of other network" refers to either the pessimistic, optimistic, or delayed information assumptions discussed in Section 4. As a final note, readers may notice that we do not engage with deep reinforcement learning (DRL) algorithms for this work. This is due to time limitations; our previous experience with DRL informed us

Algorithm 1 Policy Iteration for Centralized MMDP (Sutton & Barto, 2018)

Initialize $V(s) \in \mathbb{R}$, $\pi(s) \in \mathcal{A}(s) = A^{\text{water}} \times A^{\text{power}}\ \forall s$, $\theta = 1e-6$, and stable $= 0$
while stable $= 0$ **do**
  $\Delta = 100$;
  while $\Delta > \theta$ **do**
    $\Delta = 0$
    for $s \in \mathcal{S}$ **do**
      $v = V(s)$;
      $s' = T(s, \pi(s))$;
      $V(s) = R(s') + \gamma V(s')$;
      $\Delta = \max(\Delta, |v - V(s)|)$
    end for
  end while
  stable $= 1$;
  for $s \in \mathcal{S}$ **do**
    old_joint_action $= \pi(s)$;
    $\pi(s) = \text{argmax}_a[R(T(s,a)) + \gamma V(T(s,a))]$;
    If old_joint_action $\neq \pi(s)$, then stable $= 0$;
  end for
end while

Algorithm 2 Q Learning Algorithm for Individualized MuMDP (Sutton & Barto, 2018)

Initialize $Q^{\text{w}}(s,a), \forall s \in \mathcal{S}, a \in \mathcal{A}^{\text{w}}(s)$, arbitrarily, and $Q^{\text{w}}(\text{terminal state}, \cdot) = 0$ and $Q^{\text{p}}(s,a), \forall s \in \mathcal{S}, a \in \mathcal{A}^{\text{p}}(s)$, arbitrarily, and $Q^{\text{p}}(\text{terminal state}, \cdot) = 0$
for $k = 1, K$ **do**
  Initialize S^{w} and S^{p} with S_0 value;
  for $t = 1, T$ **do**
    Choose A^{w} from $\mathcal{A}^{\text{w}}(S)$ using policy derived from Q^{w} (such as ϵ-greedy);
    Choose A^{p} from $\mathcal{A}^{\text{p}}(S)$ using policy derived from Q^{p} (such as ϵ-greedy);
    Take action A^{w}, observe $S^{\text{w}'}$, and obtain R^{w} by concatenating $S^{\text{w}'}$ with current belief regarding state of other network;
    Take action A^{p}, observe $S^{\text{p}'}$, and obtain R^{p} by concatenating $S^{\text{p}'}$ with current belief regarding state of other network;
    $B^{\text{w}} \leftarrow R^{\text{w}} + \gamma \max_a Q^{\text{w}}(S^{\text{w}'}, a)$
    $Q^{\text{w}}(S^{\text{w}}, A^{\text{w}}) \leftarrow (1-\alpha) Q^{\text{w}}(S^{\text{w}}, A^{\text{w}}) + \alpha B^{\text{w}}$;
    $B^{\text{p}} \leftarrow R^{\text{p}} + \gamma \max_a Q^{\text{p}}(S^{\text{p}'}, a)$
    $Q^{\text{p}}(S^{\text{p}}, A^{\text{p}}) \leftarrow (1-\alpha) Q^{\text{p}}(S^{\text{p}}, A^{\text{p}}) + \alpha B^{\text{p}}$;
    $S^{\text{w}} \leftarrow S^{\text{w}'}$;
    $S^{\text{p}} \leftarrow S^{\text{p}'}$;
  end for
end for

that DRL algorithms can take some time to tune. In potential future work on this subject, DRL algorithms would be an interesting avenue to explore.

6. Experimental Set-up

For the experimental set-up, we produce 10 random damage states for the water and power networks by drawing a random integer number between 0 to 4 for each node for each of the 10 trials. These random numbers represent the 5 damage categories for each sub-node of the water and power systems according to FEMA in the event of an earthquake (FEMA, 2020). We then use the policy iteration Algorithm 1 to solve the centralized MMDP model for three time periods into the future to produce a partially repaired network for each of the 10 damage states. We then use Q-learning (Algorithm 2) to solve the individual MuMDP model for three time periods into the future to produce a partially repaired network for each of the 10 damage states. We run the individual MuMDP model for each of the 10 damage states for a pessimistic (all 0s capacity for the opposing network), an optimistic (all 1s capacity for the opposing network), and a delayed information (the previous capacity state for the opposing network) perspectives on the opposing networks. We then compare the maximum flow using the same metric as for the reward for both partially repaired networks under the two models (and for the pessimistic, the optimistic, and the delayed approaches for the individual MuMDP model).

We run our experiment on a small but illustrative network from Yu and Baroud (2020). See Figure 1. It has 8 water nodes (in blue and on the left), 8 power nodes (in red and on the right), and 18 total links in the network (including interconnected links between the two networks). The water and power networks each have supply (S), demand (D), and transmission/transition (T) nodes (Yu & Baroud, 2020).

For Algorithm 1, the θ value is set at $1e-6$. For Algorithm 2, the parameters values are:

- $K = 5000$
- $\epsilon = \frac{0.9}{1+e^{10\frac{k-(0.4)(K)}{K}}}$, which comes from (Yu et al., 2021)
- $\alpha = 0.99$
- $\gamma = 0.9$

7. Results

The results in Tables 1, 2, and 3 show that the centralized model solution maximum flow acts as an upper bound on the individual model solution maximum flow. This makes intuitive sense because, in the centralized model, the agents are acting as one unit whereas, in the individual model, the agents are not, which does not allow them to take advantage

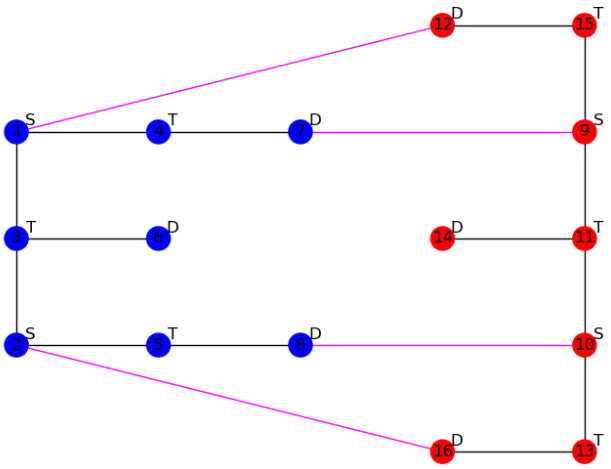

Figure 1. Toy Network (Yu & Baroud, 2020)

of coordination opportunities to maximize flow in the interconnected arcs/nodes. Figure 2, demonstrates that, among the individual models, the delayed information model leads to less variability in the percentage difference metric between the individual model and the centralized model when the ranges of the metric are compared. This indicates that there is value in delayed information sharing between agents, which requires very little coordination and perhaps suggests an important policy idea in improving disaster management practices between interconnected networks. We do note that the fourth quartiles of the pessimistic and optimistic individual models lie above the fourth quartile of the delayed information individual model but, overall, if we have to choose one model and aim to guard against comparatively large negative outliers, the experiments still point to the delayed information model because its first quartile lies well above the two other models' first quartiles.

As a note on timing for the experiments, these are the average timing results in seconds for the four experiments undertaken:

- Policy Iteration for the MMDP/Centralized Model: 3976 seconds
- Q Learning for the MuMDP/Individual Model with Pessimistic Assumption: 178 seconds
- Q Learning for the MuMDP/Individual Model with Optimistic Assumption: 562 seconds
- Q Learning for the MuMDP/Individual Model with Delayed Information Assumption: 545 seconds

Table 1. Flow Results of Centralized vs Individual Model Decisions after 3 Time Periods and Pessimistic Information Assumption for Individual Model

Initial Damage Flow	Centralized Solution Flow	Individual Solution Flow	Difference as Percentage of Centralized (%)
4.1	6.17	6.17	0.0
4.17	5.63	5.31	-5.77
3.11	4.23	4.0	-5.37
7.39	12.93	12.93	0.0
13.47	24.15	22.16	-8.23
7.59	12.0	11.67	-2.74
3.71	5.44	5.42	-0.5
1.56	2.63	1.98	-24.63
2.69	3.52	3.17	-9.96
3.61	5.05	5.05	0.0

Table 2. Flow Results of Centralized vs Individual Model Decisions after 3 Time Periods and Optimistic Information Assumption for Individual Model

Initial Damage Flow	Centralized Solution Flow	Individual Solution Flow	Difference as Percentage of Centralized (%)
4.1	6.17	6.17	0.0
4.17	5.63	5.57	-1.11
3.11	4.23	3.7	-12.67
7.39	12.93	12.02	-7.02
13.47	24.15	24.15	0.0
7.59	12.0	11.68	-2.66
3.71	5.44	5.19	-4.67
1.56	2.63	2.15	-17.98
2.69	3.52	3.38	-4.07
3.61	5.05	4.42	-12.31

8. Conclusions & Future Work

In conclusion, this work demonstrates the two ends of the spectrum regarding coordination in interconnected infrastructure in the aftermath of a disaster as well as a point on this spectrum in the form of delayed information sharing. We propose two models, the centralized and the individual model, that span this spectrum as well as a modified version of the individual model with delayed information sharing, and we demonstrate all of these models using a 10 trial experiment. From this experiment, we see that the centralized model acts as an upper bound on the individual model, and we showcase the value in at least limited coordination in the form of delayed information sharing.

In the future, we would test the two models on a larger network, which would require more sophisticated MDP and

Table 3. Flow Results of Centralized vs Individual Model Decisions after 3 Time Periods and Delayed Information Assumption for Individual Model

Initial Damage Flow	Centralized Solution Flow	Individual Solution Flow	Difference as Percentage of Centralized (%)
4.1	6.17	6.01	-2.55
4.17	5.63	5.38	-4.43
3.11	4.23	4.06	-4.09
7.39	12.93	12.02	-7.02
13.47	24.15	24.13	-0.07
7.59	12.0	11.35	-5.47
3.71	5.44	5.17	-5.04
1.56	2.63	2.53	-3.78
2.69	3.52	3.3	-6.31
3.61	5.05	4.98	-1.33

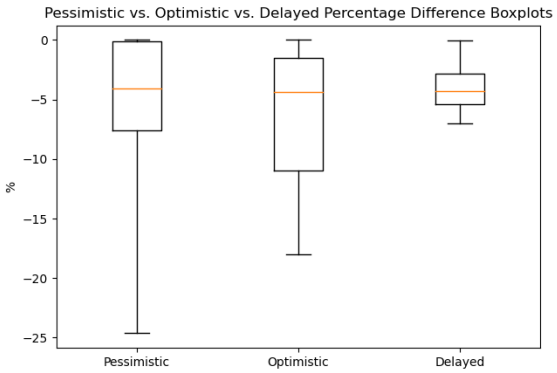

Figure 2. Boxplots Comparing Pessimistic vs Optimistic vs Delayed Percentage of Centralized (%)

reinforcement learning solution techniques. We would also test more realistic earthquake damage disaster situations, as opposed to the randomized damage scenarios we used for our experiments. Finally, we would explore deep reinforcement learning algorithms applied to this context.

9. Acknowledgements

We would like to thank Dr. Allison Reilly from University of Maryland, College Park for a very insightful conversation in Spring 2022 that helped inspire this work. She pointed us toward the NIPP 2013 (DHS, 2013) and informed us of the lack of communication between different infrastructure entities.

References

Alutaibi, K. *Decision support for emergency response in interdependent infrastructure systems*. PhD thesis, University of British Columbia, 2017.

Boutilier, C. Planning, learning and coordination in multiagent decision processes. In *TARK*, volume 96, pp. 195–210. Citeseer, 1996.

DHS. Nipp 2013: Partnering for critical infrastructure security and resilience, 2013. URL https://www.cisa.gov/sites/default/files/publications/national-infrastructure-protection-plan-2013-508.pdf.

Dueñas-Osorio, L., Craig, J. I., and Goodno, B. J. Seismic response of critical interdependent networks. *Earthquake engineering & structural dynamics*, 36(2):285–306, 2007.

Espada Jr, R. *Spatial analysis and modelling of flood risk and climate adaptation capacity for assessing urban community and critical infrastructure interdependency*. PhD thesis, University of Southern Queensland, 2014.

FEMA. Hazus earthquake model technical manual, 2020. URL https://www.fema.gov/sites/default/files/2020-10/fema_hazus_earthquake_technical_manual_4-2.pdf.

Hagberg, A. A., Schult, D. A., and Swart, P. J. Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T., and Millman, J. (eds.), *Proceedings of the 7th Python in Science Conference*, pp. 11 – 15, Pasadena, CA USA, 2008.

Huang, L., Chen, J., and Zhu, Q. A large-scale markov game approach to dynamic protection of interdependent infrastructure networks. In *International Conference on Decision and Game Theory for Security*, pp. 357–376. Springer, 2017.

Huang, L., Chen, J., and Zhu, Q. Distributed and optimal resilient planning of large-scale interdependent critical infrastructures. In *2018 Winter Simulation Conference (WSC)*, pp. 1096–1107. IEEE, 2018.

Ishigaki, G., Devic, S., Gour, R., and Jue, J. P. Deeppr: Progressive recovery for interdependent vnfs with deep reinforcement learning. *IEEE Journal on Selected Areas in Communications*, 38(10):2386–2399, 2020.

Khouj, M. T., Sarkaria, S., Lopez, C., and Marti, J. Reinforcement learning using monte carlo policy estimation for disaster mitigation. In *International Conference on Critical Infrastructure Protection*, pp. 155–172. Springer, 2014.

Khouj, M. T., Alsubaie, A., Alutaibi, K., Ahmed, H. M., Sarkaria, S., and Martí, J. R. Intelligent decision system for responsive crisis management. *International Journal of Critical Infrastructures*, 14(4):375–399, 2018.

Lopez, C., Marti, J. R., and Sarkaria, S. Distributed reinforcement learning in emergency response simulation. *IEEE Access*, 6:67261–67276, 2018.

Megherbi, D., Kim, M., and Madera, M. A study of collaborative distributed multi-goal & multi-agent-based systems for large critical key infrastructures and resources (ckir) dynamic monitoring and surveillance. In *2013 IEEE International Conference on Technologies for Homeland Security (HST)*, pp. 687–692. IEEE, 2013.

Memarzadeh, M. and Pozzi, M. Model-free reinforcement learning with model-based safe exploration: Optimizing adaptive recovery process of infrastructure systems. *Structural Safety*, 80:46–55, 2019.

Ni, Z. and Paul, S. A multistage game in smart grid security: A reinforcement learning solution. *IEEE transactions on neural networks and learning systems*, 30(9):2684–2695, 2019.

Nozhati, S. A resilience-based framework for decision making based on simulation-optimization approach. *Structural Safety*, 89:102032, 2021.

Nozhati, S., Sarkale, Y., Chong, E. K., and Ellingwood, B. R. Optimal stochastic dynamic scheduling for managing community recovery from natural hazards. *Reliability Engineering & System Safety*, 193:106627, 2020.

Oliehoek, F. A. and Amato, C. *A concise introduction to decentralized POMDPs*. Springer, 2016.

Panfili, M., Giuseppi, A., Fiaschetti, A., Al-Jibreen, H. B., Pietrabissa, A., and Priscoli, F. D. A game-theoretical approach to cyber-security of critical infrastructures based on multi-agent reinforcement learning. In *2018 26th Mediterranean Conference on Control and Automation (MED)*, pp. 460–465. IEEE, 2018.

Rajulapati, P. S., Nukavarapu, N., and Durbha, S. Deep learning-based critical infrastructure simulation model for disaster monitoring. In *2020 International Conference on Data Mining Workshops (ICDMW)*, pp. 828–835. IEEE, 2020a.

Rajulapati, P. S., Nukavarapu, N., and Durbha, S. Multi-agent deep reinforcement learning based interdependent critical infrastructure simulation model for situational awareness during a flood event. In *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*, pp. 6890–6893. IEEE, 2020b.

Srikanth, G., Nukavarapu, N., and Durbha, S. Deep reinforcement learning interdependent healthcare critical infrastructure simulation model for dynamically varying covid-19 scenario-a case study of a metro city. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pp. 8499–8502. IEEE, 2021.

Sun, J. and Zhang, Z. A post-disaster resource allocation framework for improving resilience of interdependent infrastructure networks. *Transportation Research Part D: Transport and Environment*, 85:102455, 2020.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

Talebiyan, H. and Duenas-Osorio, L. Decentralized decision making for the restoration of interdependent networks. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 6(2): 04020012, 2020.

van Riel, W., Langeveld, J., Herder, P., and Clemens, F. The influence of information quality on decision-making for networked infrastructure management. *Structure and Infrastructure Engineering*, 13(6):696–708, 2017.

Yu, J.-Z. and Baroud, H. Modeling uncertain and dynamic interdependencies of infrastructure systems using stochastic block models. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*, 6(2):020906, 2020.

Yu, L., Zhang, C., Jiang, J., Yang, H., and Shang, H. Reinforcement learning approach for resource allocation in humanitarian logistics. *Expert Systems with Applications*, 173:114663, 2021.