The role of value of information in multi-agent deep reinforcement learning for optimal decision-making under uncertainty

Saifullah, Mohammad; Andriotis, Charalampos; Papakonstantinou, Konstantinos G.

The role of value of information in multi-agent deep reinforcement learning for optimal decision-making under uncertainty

Mohammad Saifullah
Graduate Student, Dept. of Civil Engineering, Penn State University, University Park, PA, USA

Charalampos P. Andriotis
Assistant Professor, Faculty of Architecture & the Built Environment, Delft University of Technology, Delft, The Netherlands

Konstantinos G. Papakonstantinou
Associate Professor, Dept. of Civil Engineering, Penn State University, University Park, PA, USA

ABSTRACT: To preserve structural safety of deteriorating engineering systems through optimal maintenance, it is imperative to efficiently integrate structural health information with decision-making optimization frameworks. Although there may be abundance of available data, these are often uncertain and incomplete. In addition, joint inspection and maintenance (I&M) optimization is inherently complex due to high-dimensional state and action spaces, stochastic objectives, long planning horizons, and various constraints, among others. As shown recently, these computational challenges can be effectively addressed through optimization principles of Partially Observable Markov Decision Processes (POMDPs) and constrained Deep Reinforcement Learning (DRL). The POMDP framework provides a way of updating the decision-maker's perception about the system state by naturally incorporating the Value of Information (VoI) in the optimality equations. As such, optimal observation-gathering actions are those which guide maintenance decisions towards reduced life-cycle costs and risks. The role of VoI in DRL-driven I&M has also been shown to be central to the formation of policy gradients, which are necessary to obtain the optimal I&M plan with deep learning actor-critic architectures. Leveraging this property, a recently devised DRL architecture is further examined in this work, consisting of fully decoupled 'maintainer' and 'inspector' actors, which allow for greater efficacy and interpretability in multi-agent DRL settings. Several numerical analyses are carried out to assess the performance of the relevant architectures on stochastic systems with a varying number of components, multiple maintenance-inspection actions per component, and system-level failure risks.

1. INTRODUCTION

Preserving the integrity and functionality of rapidly deteriorating infrastructure requires life-long inspection and maintenance (I&M) actions. Inspecting the condition of structural components during their operational life can effectively inform appropriate maintenance decisions, but both I&M actions have associated costs that must be weighed against risk implications and available resources. The decision maker’s goal is thus to minimize the total anticipated costs over the structural system lifetime while adhering to certain performance constraints. This defines an optimization problem with several challenges, including the curse of dimensionality of action and state spaces, the uncertainty in collected data, and the presence of multiple types of constraints.

Most existing I&M planning methods assume independence among components and focus on optimizing static or adaptive decision rules, built upon performance threshold principles (Straub, 2004; Saydam & Frangopol, 2014; Bocchini & Frangopol, 2011), and solved through direct policy search or gradient-based and genetic algorithms. These methods often provide suboptimal solutions, are hard to scale in high-dimensional spaces, and

may delimit the use of data in open-loop workflows.

Stochastic optimal control methods have also been deployed in I&M planning problems and have demonstrated significant closed-loop control capabilities for solving this optimization problem under uncertain real-time observation (Madanat, 1993 ; Papakonstantinou & Shinozuka, 2014b; Papakonstantinou, et al., 2018). For large-scale multi-components systems with high-dimensional state and action spaces, the I&M planning problem is effectively addressed using a combination of Partially Observable Markov Decision Processes (POMDPs) and multi-agent Deep Reinforcement Learning (DRL). The dynamic programming principles of POMDPs allow for adaptive reasoning under noisy data, as for example demonstrated in (Papakonstantinou & Shinozuka, 2014a; Memarzadeh & Pozzi, 2015). Within the POMDP framework, uncertain information can update the decision-maker's perception about the system state and the notion of Value of Information (VoI) is proven to be intrinsically present in the POMDP optimality equations (Andriotis, et al., 2021). As a result, the POMDP I&M policies provide the optimal observation-gathering actions, maximizing the data benefits in terms of reduced life-cycle costs.

The role of VoI within POMDP-DRL settings is further analyzed to play an important role in the formation of the gradients involved in training actor-critic deep network architectures to obtain the I&M policy, allowing us to decouple, in a mathematically consistent way, the searched policies to their maintenance and inspection constituents. Leveraging this property, the recently devised DRL architecture adopted here, exploits the natural sequential structure of inspection-maintenance actions, decomposing joint actors into independent 'maintainer' and 'inspector' actors (Andriotis & Papakonstantinou, 2022). Here, we investigate the computational and interpretability attributes of this approach in multi-agent DRL-driven I&M optimization, particularly when individual components have high action space dimensionality due to combinations of inspection-maintenance choices. The inspector-maintainer decomposition is applied to the family of deep decentralized multi-agent actor-critic DRL architectures developed in (Andriotis & Papakonstantinou, 2019; Andriotis & Papakonstantinou, 2021; Saifullah, et al., 2023) . Several numerical analyses are finally carried out and characteristics of the different architectures are reported for stochastic systems with varying number of components, multiple maintenance-inspection actions per component, and various system-level interactions and complexities.

2. BACKGROUND

A POMDP is comprised of several key components, including S (a set of states), $A = A_I \times A_M$ (a set of actions, where A_I and A_M are inspection and maintenance sets, respectively.), $\mathbf{P}$ (a model of transitions), $\mathbf{\Omega}$ (a set of possible observations), $\mathbf{O}$ (an observation model), $\mathbf{C}$ (a cost functions), and γ (a discount factor). In this framework, the decision-maker (agent) begins at a specific condition state, s_t, at a given time step, t. It takes an action, $a_t = (a_{I,t}, a_{M,t})$, which comprises an inspection ($a_{I,t}$) and maintenance ($a_{M,t}$), and incurs a cost, c_t, before transitioning to the next state, s_{t+1}, and receiving an observation, $o_{t+1} \in \mathbf{\Omega}$, based on an observation probability model, which depends on the state of the system and the action at the current step and is defined as a probability, $p(o_{t+1}|s_{t+1}, a_t)$. Due to the partial observability, the agent can only form a belief, $\mathbf{b}_t$ about the condition state, which is a probability distribution over the set of all possible discrete states, S. To calculate the belief $\mathbf{b}_{t+1}$, a Bayesian update is performed, i.e., $b(s_{t+1}) = p(s_{t+1}|o_{t+1}, a_t, \mathbf{b}_t)$, where probabilities $b(s_t)$, for all $s_t \in S$, form the belief vector $\mathbf{b}_t$ of length $|S|$, (Papakonstantinou & Shinozuka, 2014a). The goal of an agent is to minimize the expected future discounted cumulative cost, defined by the value function (Papakonstantinou & Shinozuka, 2014a). The optimal value function is written as:

$$V^{\pi^*}(\mathbf{b}_t) = \min_{a \in A} \sum_{s_t \in S} b(s_t)\, c(s_t, a_t) + \gamma \sum_{o_{t+1} \in \Omega} p(o_{t+1} \mid \mathbf{b}_t, a_t)\, V^{\pi^*}(\mathbf{b}_{t+1}) \tag{1}$$

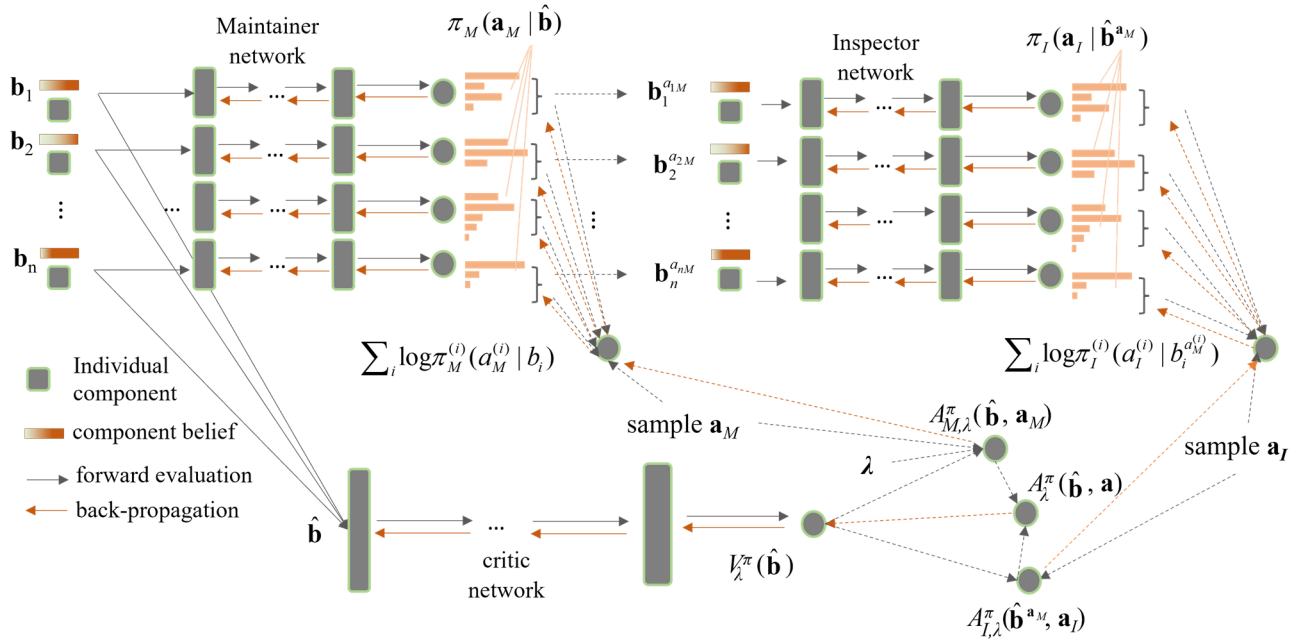

Figure 1: Maintainer-Inspector architecture.

where $c(s_t, a_t)$ may have multiple parts, such as maintenance cost $c_M(a_{M,t})$, inspection cost $c_I(a_{I,t})$, and risk related cost $c_R(s_t, a_{M,t})$, among others, and π^* is the optimal policy. The value function can be also expressed in terms of VoI, as shown in (Andriotis, et al., 2021):

$$V^{\pi^*}(\mathbf{b}_t) = \min_{a_{M,t}\in A_M}\left\{\sum_{s_t\in S} b(s_t)\left(c_M(a_{M,t}) + c_R(s_t, a_{M,t})\right) + \gamma V^*\left(\mathbf{b}_t^{a_M}\right) - \gamma \max_{a_{I,t}\in A_I} VoI_{net}\left(a_{I,t}\right)\right\} \quad (2)$$

where VoI_{net} denotes the net Value of Information (VoI) associated with inspection action a_I:

$$VoI_{net}\left(a_{I,t}\right) = V^{\pi^*}\left(\mathbf{b}_t^{a_M}\right) - \mathbb{E}_o\left[V^{\pi^*}\left(\mathbf{b}_{t+1}\right)\right] - c_I(a_{I,t}) \quad (3)$$

where, $\mathbb{E}_o[.]$ is an expectation over all possible observations. In essence, Eq. (2) explains how, when following a maintenance action ($a_{M,t}$) from optimal policy π^*, inspections ($a_{I,t}$) are selected based on the net value of information (VoI) (Andriotis, et al., 2021).

2.1. Multi-agent actor-critic DRL formulations

Multi-agent DRL is an effective approach to solving POMDP problems for large-scale systems in a decentralized way, where each agent in the system learns to act optimally based on local and global beliefs over the system state space. Actor-critic architectures are widely used in deep reinforcement learning, with actors and critics parametrizing the policy and value functions, respectively. Recently, the authors have developed various multi-agent actor-critic algorithms to solve POMDPs, including the Deep Centralized/Decentralized Multi-agent Actor-Critic (DCMAC/DDMAC) in (Andriotis & Papakonstantinou, 2019; Andriotis & Papakonstantinou, 2021), and DDMAC with centralized training with decentralized execution (CTDE) method in (Saifullah, et al., 2023). These actor-critic methods utilize offline training with experience replay and belong to the general actor-critic families that have shown capabilities of discovering powerful strategies in immense state spaces (Silver, et al., 2016; Mnih, et al., 2015) in various domains, such as cooperative navigation, resource allocation, and decentralized control.

POMDP problems under constraints can also be

solved with DRL approaches, as shown in (Andriotis & Papakonstantinou, 2021). The generalized value function including a risk can be given as:

$$V^{\pi^*}(\mathbf{b}_0) = \max_{\lambda \geq 0} \min_{\pi \in \Pi} \mathbb{E}\Bigg[\sum_{t=0}^{T} \gamma^t \left(c_t + \lambda c_F \Delta P_{F_t} \right) - \lambda \mathfrak{R}_{ult}^{\pi} \,\Big|\, a_t \sim \pi\left(o_{0:t}, a_{0:t-1}\right) \Bigg] \tag{4}$$

where λ is a Lagrange multiplier, c_F is the failure cost, P_{Ft} is the failure probability up to time t, and $\mathfrak{R}_{ult}$ is a prescribed life-cycle risk tolerance. The $c_{F.}$ can have two parts, an instantaneous failure cost and a perpetual cost due to the continual disruption if the component is not rebuilt. Several sources of risks can also be considered, including failures of individual components and system failure. Further details on implementing POMDP-DRL with risk and other constraints can be found in (Andriotis & Papakonstantinou, 2021; Saifullah, et al., 2023).

In the actor-critic algorithms used here, the value function is parameterized by the critic network, with parameters $\boldsymbol{\theta}_V$:

$$V_{\lambda}^{\pi}\left(\hat{\mathbf{b}}\right) \simeq V_{\lambda}^{\pi}\left(\hat{\mathbf{b}}; \boldsymbol{\theta}_V\right) \tag{5}$$

The policy network is parameterized with $\boldsymbol{\theta}_{\pi}^{(i)}$ for the i^{th} component, and each component is represented by a separate actor network:

$$\pi\left(\mathbf{a} \,|\, \hat{\mathbf{b}}\right) = \prod_{i=1}^{N_C} \pi_i\left(a^{(i)} \,|\, \mathbf{b}_i; \boldsymbol{\theta}_{\pi}^{(i)}\right) \tag{6}$$

In Eqs. (5) and (6) $\mathbf{b}_i$ is the belief vector for the i^{th} component, $\hat{\mathbf{b}}$ is the system's belief (i.e., the collection of all $\mathbf{b}_i$); $\mathbf{a}$ is a vector of actions $a^{(i)}$, and N_c is the total number of components. For concise notation we are also using $(\cdot)'$ for $(\cdot)$ at t+1 time step. The parameters of the critic network are updated based on the gradient obtained from mean squared error, and the policy network parameters are updated based on the policy gradient theorem (Sutton & Barto, 2018). Further details on parameter updates and gradient estimations can be found in (Andriotis & Papakonstantinou, 2021; Saifullah, et al., 2023).

3. MAINTAINER-INSPECTOR ARCHITECTURE

Leveraging the inherent sequential nature of maintenance and inspection actions, the recently devised DRL architecture in (Andriotis & Papakonstantinou, 2022) is further examined here. This architecture decomposes the actor network into fully decoupled, 'maintainer' and 'inspector' actors. The factored policy is then given as:

$$\pi\left(\mathbf{a} \,|\, \hat{\mathbf{b}}\right) = \underbrace{\prod_{i=1}^{Nc} \pi_M^{(i)}\left(a_M^{(i)} \,|\, \mathbf{b}_i\right)}_{\pi_M:\ \text{maintenance policy}} \underbrace{\prod_{i=1}^{Nc} \pi_I^{(i)}\left(a_I^{(i)} \,|\, \mathbf{b}_i^{a_{M,i}}\right)}_{\pi_I:\ \text{inspection policy}} \tag{7}$$

where $\pi_M^{(i)}$ and $\pi_I^{(i)}$ are maintenance and inspection policies for the i^{th} component, respectively. These policies can be parameterized with parameters $\boldsymbol{\theta}_M^{(i)}$ and $\boldsymbol{\theta}_I^{(i)}$, correspondingly, and can be updated by gradients, as follows:

$$\nabla_{\boldsymbol{\theta}_M^{(i)}} V_{\lambda}^{\pi} = \mathbb{E}_{\mathcal{M}}\Bigg[w_M A_{M,\lambda}^{\pi}\left(\hat{\mathbf{b}}, \mathbf{a}_M; \boldsymbol{\theta}_V\right) \cdot \cdot \sum_{i=1}^{N_{C,M}} \nabla_{\boldsymbol{\theta}_M^{(i)}} \log \pi_M^{(i)}\left(a_M^{(i)} \,|\, \mathbf{b}_i; \boldsymbol{\theta}_M^{(i)}\right) \Bigg] \tag{8}$$

$$\nabla_{\boldsymbol{\theta}_I^{(i)}} V_{\lambda}^{\pi} = \mathbb{E}_{\mathcal{M}}\Bigg[w_I A_{I,\lambda}^{\pi}\left(\hat{\mathbf{b}}^{\mathbf{a}_M}, \mathbf{a}_I; \boldsymbol{\theta}_V\right) \cdot \cdot \sum_{i=1}^{N_{C,I}} \nabla_{\boldsymbol{\theta}_I^{(i)}} \log \pi_I^{(i)}\left(a_I^{(i)} \,|\, \mathbf{b}_i^{a_M^{(i)}}; \boldsymbol{\theta}_I^{(i)}\right) \Bigg] \tag{9}$$

$$\nabla_{\lambda} V_{\lambda}^{\pi} \simeq c_F \sum_{t=0}^{T} \gamma^t \Delta P_{F_t} - \mathfrak{R}_{ult}^{\pi} \tag{10}$$

where, $\hat{\mathbf{b}}^{\mathbf{a}_M}$ is a system's belief after taking maintenance actions $\mathbf{a}_M$, $\mathbf{b}_i^{a_M^{(i)}}$ is a i^{th} component belief after taking maintenance $a_M^{(i)}$, w_M and w_I are importance sampling weights, $A_{M,\lambda}^{\pi}$, $A_{I,\lambda}^{\pi}$ are the maintenance and inspection advantage functions, respectively, and $\mathcal{M}$ is the experience replay containing information of past transitions and costs. The advantage functions take the form:

$$A^{\pi}_{M,\lambda}\left(\hat{\mathbf{b}},\mathbf{a}_M;\boldsymbol{\theta}_V\right) \simeq -\mathbb{E}_{\mathbf{s}}\left[c_M(\mathbf{s},\mathbf{a})\right] - \lambda c_F \Delta P_{F_i} - \gamma V^{\pi}_{\lambda}\left(\hat{\mathbf{b}}^{\mathbf{a}_M};\boldsymbol{\theta}_V\right) + V^{\pi}_{\lambda}\left(\hat{\mathbf{b}};\boldsymbol{\theta}_V\right) \quad (11)$$

$$A^{\pi}_{I,\lambda}\left(\hat{\mathbf{b}}^{\mathbf{a}_M},\mathbf{a}_I;\boldsymbol{\theta}_V\right) \simeq -\gamma c_I(\mathbf{a}_I) - \gamma V^{\pi}_{\lambda}\left(\hat{\mathbf{b}}';\boldsymbol{\theta}_V\right) + \gamma V^{\pi}_{\lambda}\left(\hat{\mathbf{b}}^{\mathbf{a}_M};\boldsymbol{\theta}_V\right) \quad (12)$$

From Eq. (12), it can be noticed that the inspection advantage function can directly provide the net value of Conditional VoI (CVoI) at every time step *t,* i.e., $CVoI_{net} = A^{\pi}_{I,\lambda}\left(\hat{\mathbf{b}}^{\mathbf{a}_M},\mathbf{a}_I\right)$. The CVoI, from standard terminology in reliability literature is defined as the difference between posterior and prior expected life-cycle benefits of using certain information (inspection/monitoring) at any time step *t,* conditioned over the collected observations.

As shown in Eq. (2), the VoI is inherently used in POMDPs to select optimal inspection actions. This can also be applied in DRL settings by parametrizing the maintenance policy and choosing the inspections that maximize the net VoI. However, due to computational challenges in large-scale systems with multiple components, it is generally difficult to estimate the net VoI. Instead, by parametrizing the inspection network and using the CVoI to train the network parameters, the optimal inspection behavior can be approximated through gradient descent.

Figure 1 shows the discussed maintainer-inspector architecture, where individual component beliefs are the input to the maintainer network. The obtained maintenance actions modify the system state and the agents' beliefs. These updated beliefs then become the inspector actors' input, which then suggest relevant observation actions for the final update step. The presented architecture is amenable to adding also other deterministic constraints, such as budget constraints, as considered in (Andriotis, et al., 2023). The network parameters get updated after each episode using the gradients mentioned in Eqs. (8)-(10).

4. ENVIRONMENT DETAILS

For our numerical experiments, a scalable multi-component reliability block system is considered,

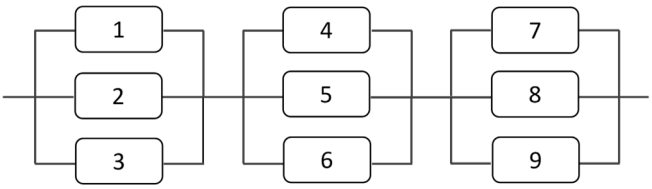

Figure 2: 9-component system reliability block diagram

with 9- and 35-components. The 9-component network is shown in Figure 2, where a 3x3x3 block architecture is considered. All components have identical deterioration, state, observation, and action characteristics. They are described by 4 states following non-stationary transitions, including a 5th failure state. The non-zero, non-stationary transition probabilities for a service horizon of 30 steps from (Andriotis & Papakonstantinou, 2021) are used. In addition, the considered failure probabilities are $P_f = 0.0019$ if the component is in state 1 (intact), $P_f = 0.0067$ if it is in state 2 (minor damage), $P_f = 0.0115$ if in state 3 (major damage), and $P_f = 0.0177$ if the component is in state 4 (severe damage). Similarly, five 7-component blocks are considered for the 35-component system, with the same characteristics for each component.

Several maintenance and inspection actions per component are considered to investigate the characteristics of the decoupled architecture. The complexity of the action space is varied in different cases, with 5, 50, and 100 available actions per component. This makes the joint space of system actions at every step ranging from 5^9 to 10^{70}, for the 9-component system with 5 actions per component and the 35-component system with 100 actions, respectively.

The 5-action case includes the combinations of 2 inspection actions (no-inspection and inspection), 2 maintenance actions (do-nothing and repair), and 1 rebuild action. The no-inspection action does not provide any additional information, while the inspection action follows an observation probability model given in (Andriotis, et al., 2023). The do-nothing action has no effect on the component and the repair action reverses the component's damage state by one condition without modifying its deterioration rate. The

rebuilding action restores the component to its intact state. Additionally, we have cases with 50 (7 inspection x 7 maintenance+1 rebuild) and 100 (9 inspection x 11 maintenance + 1 rebuild) action combinations used in this study. The maintenance actions for these cases have probabilistic outcomes with suitably chosen action transition probabilities. The details are provided in (Andriotis, et al., 2023).

For the 5 actions case, the cost of rebuilding C_{reb} is 1.0, and repair and inspection action costs are 7.5% and 1.5% of C_{reb}, respectively. For 50 and 100 action combinations, C_{reb} is 1.0, and the cost of maintenance actions is linearly interpolated from 0 to 42% of C_{reb}, 42% being the highest maintenance action cost. Similarly, inspection action cost can be linearly interpolated from 0 to 10% of C_{reb}, 10% being the highest inspection cost. The cost of failure c_F is taken as 2x and 7.5x of C_{reb} for perpetual and instantaneous costs, respectively, in the case of component failure, and 5x and 10x of the system rebuild cost for system-level failure.

5. RESULTS

The separate maintainer and inspector actors are parameterized here for each component with 2x200 hidden layers; whereas the centralized critic network parametrizes the value function with 2x500 hidden layers. The maintainer-inspector decoupling architecture is applied to DDMAC with decentralized information (CTDE specifications), having the same characteristics for the actors and critics' hidden layers. All involved networks have been trained with Keras with Tensorflow backend version 1.5.0.

5.1. Policy evaluation and comparison

We compare the policies of all architectures for all cases of 9 and 35-component systems with varying number of actions per component, as discussed in Section 4.2. All policies are trained for a maximum of 10^6 episodes or until convergence, starting with an intact state, taking $\gamma = 0.975$. Table 2 shows the relative performance of the two methods based on the total life-cycle cost after convergence. As observed, for smaller systems with low number of actions per component, DDMAC-CTDE performs almost the same and better than the Maintainer-Inspector (MI) DDMAC-CTDE. As the number of actions grows, the MI architecture improves the performance by 20% and 55%, for 9 and 35-component systems respectively (with 100 actions per component). DDMAC with CTDE performs consistently better in the smaller action space case, regardless of the number of components, and it even performed better in the 9-component system with 50 actions case. However, as the number of components and actions increases, we see that the MI architecture improves the DDMAC-CTDE for both 9 and 35-component systems. Training insta-

Table 1: Comparison of different DRL methods in terms of the mean total life-cycle cost expressed in C_{reb}

Comp.	Methods	5 actions	50 actions	100 actions
9	DRL1	11.62	10.86	**10.85**
	DRL2	**9.68**	**10.40**	11.13
35	DRL1	43.86	**46.48**	**49.89**
	DRL2	**38.73**	48.44	80.55

DRL 1, and 2 represent maintainer-inspector, and DDMAC-CTDE architectures, respectively. The mean life-cycle cost estimated via $2x10^4$ simulations using the best DRL network weights. The 95% confidence bounds for DRL1 (with increasing actions) are [1.29, 1.49, 1.54] and [2.88, 3.18, 3.15] for 9- and 35-component systems, respectively. Similarly, for DRL2 [1.21, 1.34, 6.40] and [2.35, 12.92, 21.15] are the related confidence bounds.

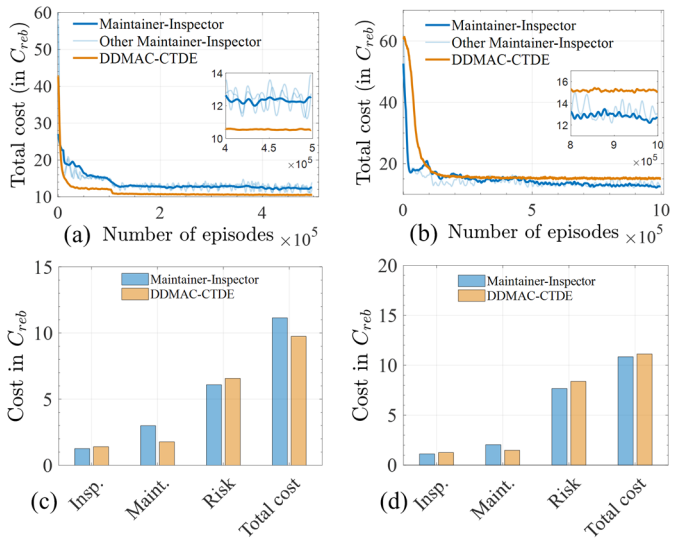

Figure 3: Total life cycle costs comparison of Maintainer-Inspector with other DRL methods for the 9 component system during training with (a) 5 actions and (b) 100 actions per component. Cost constituents during policy simulation (c) 5 actions and (d) 100 actions per component

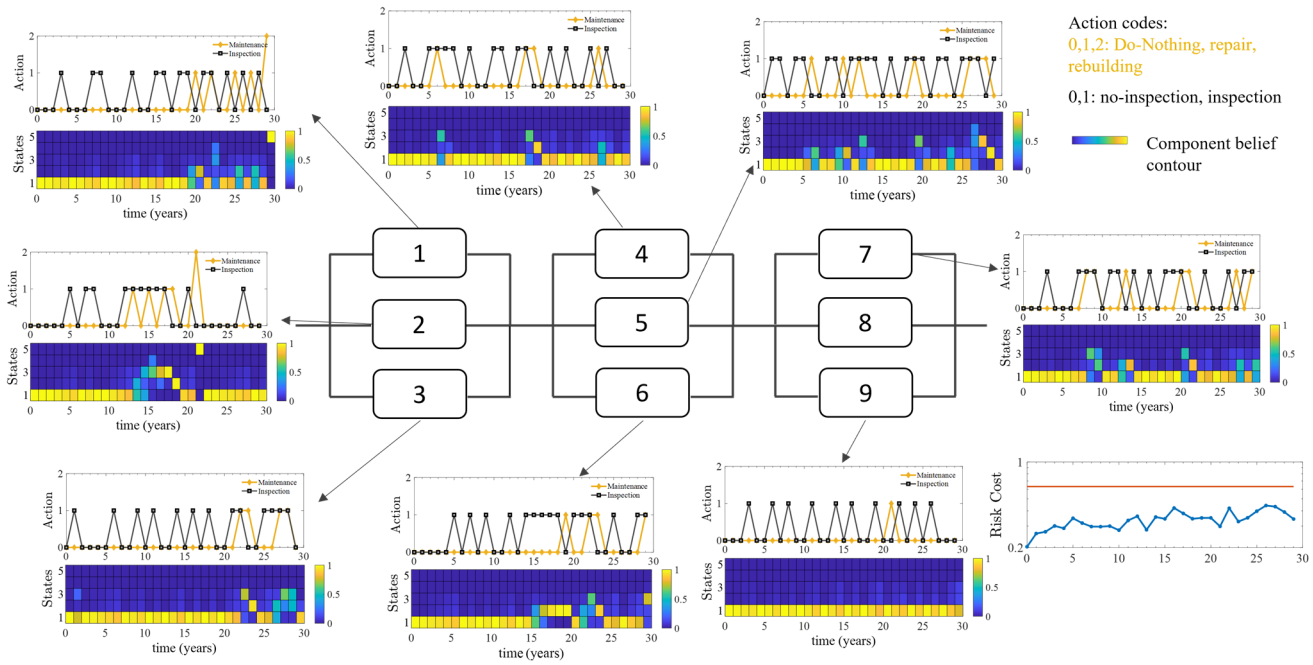

Figure 4: Life-cycle realization of the computed Maintainer-Inspector policy for the 9-component network with 5 actions per component.

nces for the 9-component network with 5 and 100 actions per components are shown in Figures 3(a) & (b), respectively. The episodes used in the respective cases are $5x10^5$ and 10^6.

Figures 3(c) & (d) show the cost constituents of the total cost (during simulation), that includes inspection, maintenance, and risk, for the 9-component network with 5 and 100 available actions per component, respectively. It can be observed that the MI DDMAC method is achieving the lower life-cycle risk than the DDMAC-CTDE. A policy realization is shown in Figure 4 to better understand and interpret the behavior of the converged policy for the MI architecture for the case with 5 actions. The figure illustrates actions generated by one realization of the policy and the evolution of component belief states is shown with contours. Figure 4 also displays the total risk (including system and components risk) over time, which is the only constraint considered here. The red line shows the constraint level, and the blue curve shows the system risk as it evolves in time. As expected, the risk is minimal at the beginning and increases with time, with downward jumps mainly due to the maintenance activities.

As we observe in Figure 4, the inspection and maintenance actions can be generally understood based on the belief states evolution. For example, the agents initially choose many do-nothing & no-inspection actions since the belief states start at the intact component conditions. As the conditions gradually worsen, relevant interventions are considered. The choice of rebuilding for components 1 and 2 after failure is consistent with the cost model. Similarly, for component 9, no repair action is selected for most of its life-cycle, as the component mostly remained in good condition.

6. CONCLUSIONS

This paper examines an actor-critic Deep Reinforcement Learning (DRL) approach for various inspection and maintenance settings for deteriorating multi-component systems. The presented approach utilizes decoupled maintenance and inspection actor networks, conditioned on post-inspection and post-maintenance beliefs, respectively. The inspection policy network is trained based on the net conditional Value of Information (VoI), to guide the inspection choices. The proposed approach is embedded in existing DRL techniques and

illustrated for 9 and 35-component systems, with varying number of actions per component. The new architectural configuration is found to improve baseline performance by significant margins as the number of system actions and components increases.

7. ACKNOWLEDGEMENTS

The authors acknowledge the support of the U.S. National Science Foundation under CAREER Grant No. 1751941 and LEAP-HI Grant No. 2053620, and the Center for Integrated Asset Management for Multimodal Transportation Infrastructure Systems, 2018 U.S. DOT Region 3 University Center. Dr. Andriotis would further like to acknowledge the support of the TU Delft AI Labs program.

8. REFERENCES

Andriotis, C. P. & Papakonstantinou, K. G., 2019. Managing engineering systems with large state and action spaces through deep reinforcement learning. *Reliability Engineering & System Safety,* Volume 191, p. 106483.

Andriotis, C. P. & Papakonstantinou, K. G., 2021. Deep reinforcement learning driven inspection and maintenance planning under incomplete information and constraints. *Reliability Engineering & System Safety,* 212, p. 107551.

Andriotis, C. P. & Papakonstantinou, K. G., 2022. Optimizing deep reinforcement learning policies for deteriorating systems considering ordered action structuring and value of information. International Conference on Structural Safety and Reliability (ICOSSAR).

Andriotis, C. P., Papakonstantinou, K. G. & Chatzi, E. N., 2021. Value of structural health information in partially observable stochastic environments. *Structural Safety,* 93, p. 102072.

Andriotis, C. P., Saifullah, M. & Papakonstantinou, K. G., 2023. Integrating value of information with deep reinforcement learning architectures for managing degrading systems. *under preparation.*

Bocchini, P. & Frangopol, D. M., 2011. A probabilistic computational framework for bridge network optimal maintenance scheduling. *Reliability Engineering & System Safety,* 96(2), pp. 332-49.

Madanat, S., 1993 . Optimal infrastructure management decisions under uncertainty. *Transportation Research Part C: Emerging Technologies,* 1(1), pp. 77-88.

Memarzadeh, M. & Pozzi, M., 2015. Integrated inspection scheduling and maintenance planning for infrastructure systems. *Computer-Aided Civil and Infrastructure Engineering,* 31(6), 403-415.

Mnih, V. et al., 2015. Human-level control through deep reinforcement learning. *Nature,* 518(7540), pp. 529--533.

Papakonstantinou, K. G., Andriotis, C. P. & Shinozuka, M., 2018. POMDP and MOMDP solutions for structural life-cycle cost minimization under partial and mixed observability. *Structure and Infrastructure Engineering,* 14(7), pp. 869-882.

Papakonstantinou, K. G. & Shinozuka, M., 2014a. Planning structural inspection and maintenance policies via dynamic programming and Markov processes. Part I: Theory. *Reliability Engineering & System Safety,* 130, pp. 202-213.

Papakonstantinou, K. G. & Shinozuka, M., 2014b. Optimum inspection and maintenance policies for corroded structures using partially observable Markov decision processes and stochastic, physically based models. *Probabilistic Engineering Mechanics,* 37, pp. 93-108.

Saifullah, M., Papakonstantinou, K. G., Andriotis, C. P. & Stoffels, S. M., 2023. Multi-agent deep reinforcement learning with centralized training and decentralized execution for transportation infrastructure management. *Transportation Research Part C: Emerging Technologies,* under review.

Saydam, D. & Frangopol, D. M., 2014. Risk-based maintenance optimization of deteriorating bridges. *Journal of Structural Engineering,* 141(4), p. 04014120.

Silver, D. et al., 2016. Mastering the game of Go with deep neural networks and tree search. *Nature,* 529(7587), pp. 484--489.

Straub, D., 2004. *Generic Approaches to Risk Based Inspection Planning for Steel Structures.* Zurich: PhD thesis, ETH Zurich.

Sutton, R. S. & Barto, A. G., 2018. *Reinforcement Learning: An Introduction.* 2nd ed. Cambridge, MA: MIT Press.