

Quantifying the Cost of Delaying Maintenance Actions on Transportation Infrastructure

Zachary Hamida Postdoc
James-A. Goulet Professeur

 Polytechnique Montréal, Canada
Département des génies civil, géologique et des mines

December 15, 2022

Funding:
Transportation Ministry of Quebec (MTQ)

🗺️ Outline

Recap

Challenges & Objectives

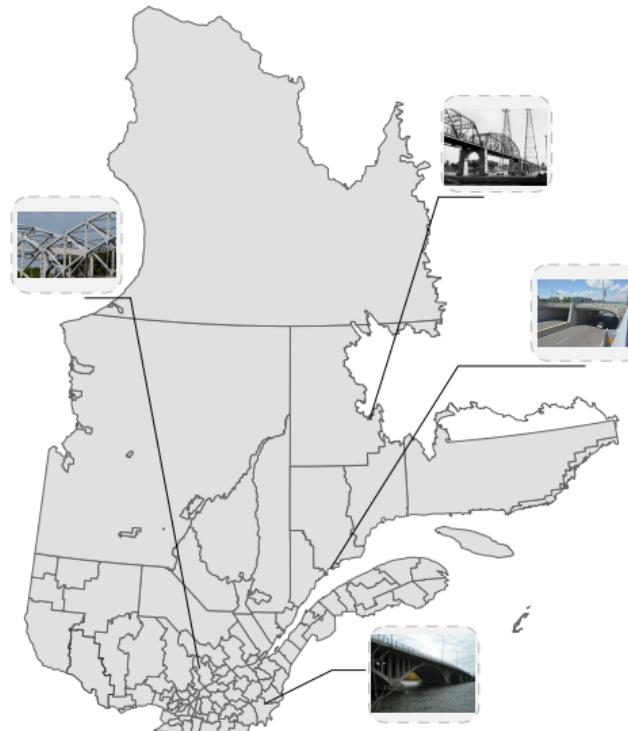
Decision-Making Framework

Expected Costs of Delays

Validation: Real Data

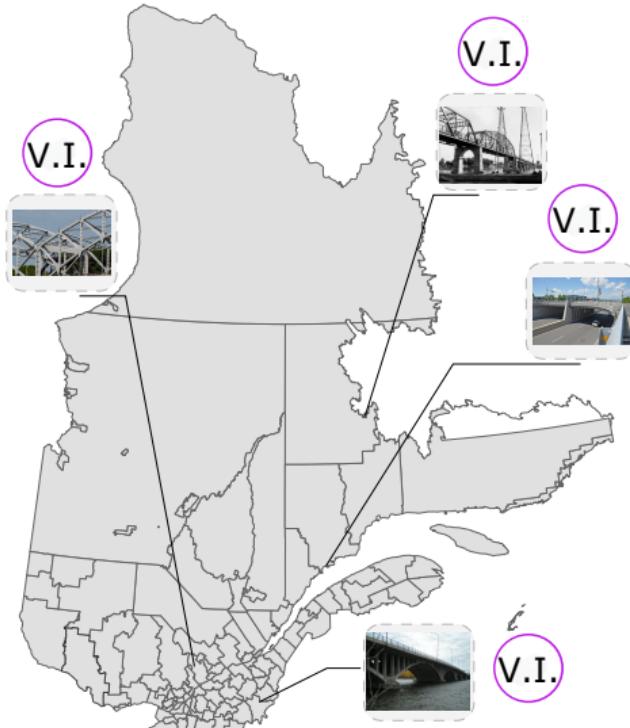
Quick Recap

Quick Recap



Quick Recap

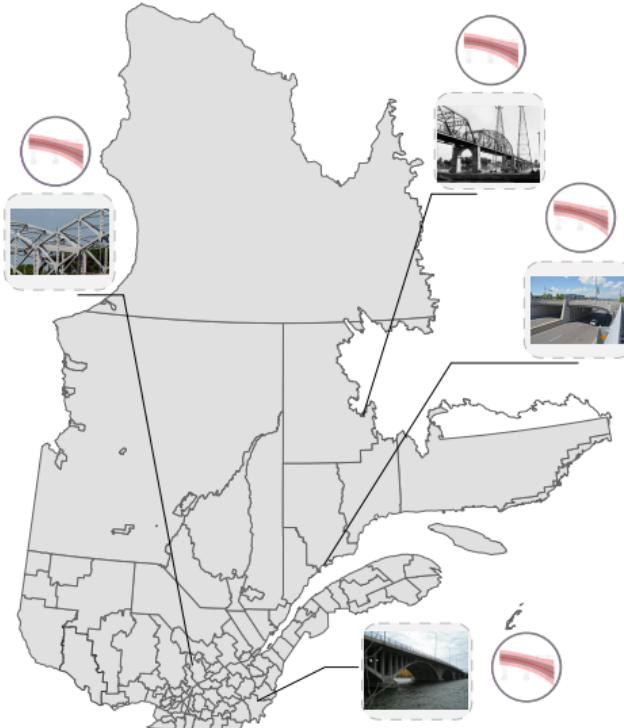
- ▷ Element-level monitoring



Quick Recap

Quick Recap

- ▷ Element-level monitoring
- ▷ Deterioration model:
 - ▷ Element → Bridge.



Quick Recap

Quick Recap

- ▷ Element-level monitoring
- ▷ Deterioration model:
 - ▷ Element → Bridge.
- ▷ Interventions data:



Quick Recap

- ▷ Element-level monitoring
- ▷ Deterioration model:
 - ▷ Element → Bridge.
- ▷ Interventions data:
 - ▷ Effect of interventions.



Quick Recap

- ▷ Element-level monitoring
- ▷ Deterioration model:
 - ▷ Element → Bridge.
- ▷ Interventions data:
 - ▷ Effect of interventions.
- * **Planning maintenance:**



Quick Recap

- ▷ Element-level monitoring
- ▷ Deterioration model:
 - ▷ Element → Bridge.
- ▷ Interventions data:
 - ▷ Effect of interventions.
- * **Planning maintenance:**
 - * **Maintenance timing.**



Quick Recap

- ▷ Element-level monitoring
- ▷ Deterioration model:
 - ▷ Element → Bridge.
- ▷ Interventions data:
 - ▷ Effect of interventions.
- * **Planning maintenance:**
 - * **Maintenance timing.**
 - * **Maintenance actions.**



Recap

○

Challenges & Objectives

●○○

Decision-Making Framework

○○○○

Expected Costs of Delays

○○

Validation: Real Data

○○○○○○○○○

Conclusions

○○

Challenges

Network-Scale Planning Challenges

Network-Scale Planning Challenges

Scale :

Network-Scale Planning Challenges

Scale : 
Elements $\times 10^5$

Network-Scale Planning Challenges

Scale :  \times 

Elements $\times 10^5$ Actions

Network-Scale Planning Challenges

Scale :  \times 

Elements $\times 10^5$ Actions

Uncertainty :

Network-Scale Planning Challenges

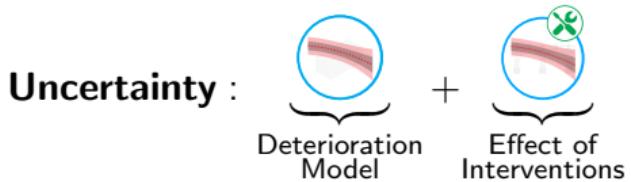
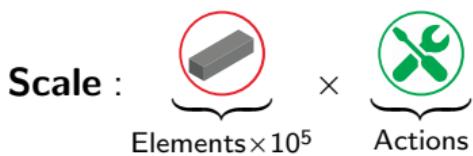
Scale :  \times 

Elements $\times 10^5$ Actions

Uncertainty : 

Deterioration Model

Network-Scale Planning Challenges



Alternative Prospective

Alternative Prospective

Maintenance Prioritization:

Alternative Prospective

Maintenance Prioritization:

- ▷ Structural safety, operational functionality.

Alternative Prospective

Maintenance Prioritization:

- ▷ Structural safety, operational functionality.
- ▷ Network connectivity :

Alternative Prospective

Maintenance Prioritization:

- ▷ Structural safety, operational functionality.
- ▷ Network connectivity :
 - ▷ Minimize travel time and user travel costs.

Alternative Prospective

Maintenance Prioritization:

- ▷ Structural safety, operational functionality.
- ▷ Network connectivity :
 - ▷ Minimize travel time and user travel costs.
 - ▷ Flow capacity.

Alternative Prospective

Maintenance Prioritization:

- ▷ Structural safety, operational functionality.
- ▷ Network connectivity :
 - ▷ Minimize travel time and user travel costs.
 - ▷ Flow capacity.

Limitations:

Alternative Prospective

Maintenance Prioritization:

- ▷ Structural safety, operational functionality.
- ▷ Network connectivity :
 - ▷ Minimize travel time and user travel costs.
 - ▷ Flow capacity.

Limitations:

- ▷ The prioritization analyses are relative.

Alternative Prospective

Maintenance Prioritization:

- ▷ Structural safety, operational functionality.
- ▷ Network connectivity :
 - ▷ Minimize travel time and user travel costs.
 - ▷ Flow capacity.

Limitations:

- ▷ The prioritization analyses are relative.
- ▷ No account for the incurred costs due to delayed maintenance.

Research Objective

Research Objective

- ▷ Quantify the cost of delaying maintenance for :

Research Objective

- ▷ Quantify the cost of delaying maintenance for :
 - ▷ A structural element.

Research Objective

- ▷ Quantify the cost of delaying maintenance for :
 - ▷ A structural element.
 - ▷ An entire bridge.

Research Objective

- ▷ Quantify the cost of delaying maintenance for :
 - ▷ A structural element.
 - ▷ An entire bridge.
- ▷ Validate the performance of the proposed framework using real data.

Element-Level Maintenance

Element-Level Maintenance

- ▷ Advantages:

Element-Level Maintenance

▷ Advantages:

- ▷ Small state-space $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.

Element-Level Maintenance

▷ Advantages:

- ▷ Small state-space $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.
- ▷ Small action-space $\mathcal{A}^e = \{a_0, a_1, a_2, a_3, a_4\}$.

Element-Level Maintenance

▷ Advantages:

- ▷ Small state-space $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.
- ▷ Small action-space $\mathcal{A}^e = \{a_0, a_1, a_2, a_3, a_4\}$.

While it is easy to find an element-level optimal policy,

Element-Level Maintenance

- ▷ Advantages:

- ▷ Small state-space $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.
- ▷ Small action-space $\mathcal{A}^e = \{a_0, a_1, a_2, a_3, a_4\}$.

**While it is easy to find an element-level optimal policy,
it is not useful for decision-making.**

Deep Reinforcement Learning (DRL)

Source: Google Images

Deep Reinforcement Learning (DRL)

Source: Google Images

Deep Reinforcement Learning (DRL)

Bad decision (-1)

Source: Google Images

Deep Reinforcement Learning (DRL)

Bad decision (-1)

Source: Google Images

Deep Reinforcement Learning (DRL)

Bad decision (-1)

Good decision (+1)

Source: Google Images

Deep Reinforcement Learning (DRL)

Bad decision (-1)

▷ Agent: Mario.

Good decision (+1)

Source: Google Images

Deep Reinforcement Learning (DRL)

Bad decision (-1)

- ▷ Agent: Mario.
- ▷ Environnement: Mario World.

Good decision (+1)

Source: Google Images

Deep Reinforcement Learning (DRL)

Bad decision (-1)

- ▷ Agent: Mario.
- ▷ Environment: Mario World.
- ▷ Actions: Movement.

Good decision (+1)

Source: Google Images

Deep Reinforcement Learning (DRL)

Bad decision (-1)

- ▷ Agent: Mario.
- ▷ Environment: Mario World.
- ▷ Actions: Movement.
- ▷ Rewards: Win the game.

Good decision (+1)

Source: Google Images

Deep Reinforcement Learning (DRL)

Bad decision (-1)

- ▷ Agent: Mario.
- ▷ Environment: Mario World.
- ▷ Actions: Movement.
- ▷ Rewards: Win the game.

Good decision (+1)

RL Goal: learn a policy that maximizes the total expected discounted rewards.

Source: Google Images

Problem Formulation

Problem Formulation

- ▷ Problem: Find an optimal maintenance policy.

Problem Formulation

- ▷ Problem: Find an optimal maintenance policy.
- ▷ Environment: Element-level slabs $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.

Problem Formulation

- ▷ Problem: Find an optimal maintenance policy.
- ▷ Environment: Element-level slabs $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.
- ▷ Actions: $\mathcal{A}^e = \{ \underbrace{a_0}_{\text{(do nothing)}}, \underbrace{a_1}_{\text{(routine mainten.)}}, \underbrace{a_2}_{\text{(prev. mainten.)}}, \underbrace{a_3}_{\text{(repair)}}, \underbrace{a_4}_{\text{(replace)}} \}$.

Problem Formulation

- ▷ Problem: Find an optimal maintenance policy.
- ▷ Environment: Element-level slabs $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.
- ▷ Actions: $\mathcal{A}^e = \{ \underbrace{a_0}_{\text{(do nothing)}}, \underbrace{a_1}_{\text{(routine mainten.)}}, \underbrace{a_2}_{\text{(prev. mainten.)}}, \underbrace{a_3}_{\text{(repair)}}, \underbrace{a_4}_{\text{(replace)}} \}$.

Reinforcement Learning Setup:

Problem Formulation

- ▷ Problem: Find an optimal maintenance policy.
- ▷ Environment: Element-level slabs $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.
- ▷ Actions: $\mathcal{A}^e = \{ \underbrace{a_0}_{\text{(do nothing)}}, \underbrace{a_1}_{\text{(routine mainten.)}}, \underbrace{a_2}_{\text{(prev. mainten.)}}, \underbrace{a_3}_{\text{(repair)}}, \underbrace{a_4}_{\text{(replace)}} \}$.

Reinforcement Learning Setup:

- ▷ Planning duration $T = 100$ years.

Problem Formulation

- ▷ Problem: Find an optimal maintenance policy.
- ▷ Environment: Element-level slabs $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.
- ▷ Actions: $\mathcal{A}^e = \{ \underbrace{a_0}_{\text{(do nothing)}}, \underbrace{a_1}_{\text{(routine mainten.)}}, \underbrace{a_2}_{\text{(prev. mainten.)}}, \underbrace{a_3}_{\text{(repair)}}, \underbrace{a_4}_{\text{(replace)}} \}$.

Reinforcement Learning Setup:

- ▷ Planning duration $T = 100$ years.
- ▷ Infinite planning horizon, discount factor $\gamma = 0.97$.

Problem Formulation

- ▷ Problem: Find an optimal maintenance policy.
- ▷ Environment: Element-level slabs $s_t^e = [x_{t|t-1}, \dot{x}_{t|t-1}]$.
- ▷ Actions: $\mathcal{A}^e = \{ \underbrace{a_0}_{\text{(do nothing)}}, \underbrace{a_1}_{\text{(routine mainten.)}}, \underbrace{a_2}_{\text{(prev. mainten.)}}, \underbrace{a_3}_{\text{(repair)}}, \underbrace{a_4}_{\text{(replace)}} \}$.

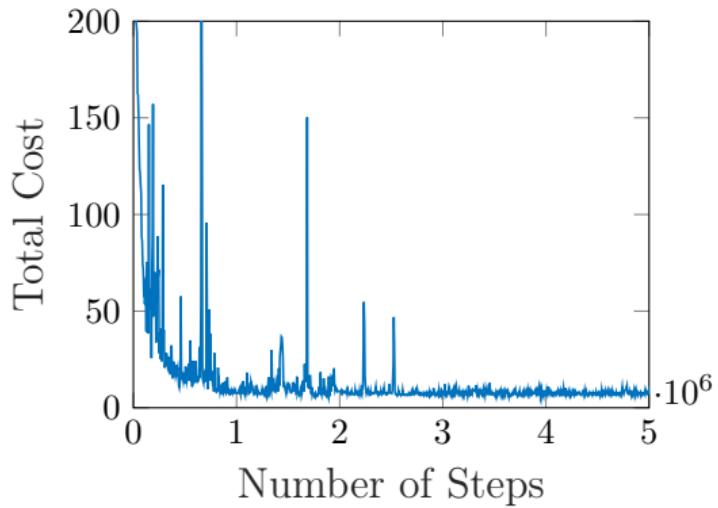
Reinforcement Learning Setup:

- ▷ Planning duration $T = 100$ years.
- ▷ Infinite planning horizon, discount factor $\gamma = 0.97$.
- ▷ Vectorized Environment with Deep Q-Network

Maintenance Policy for Slabs

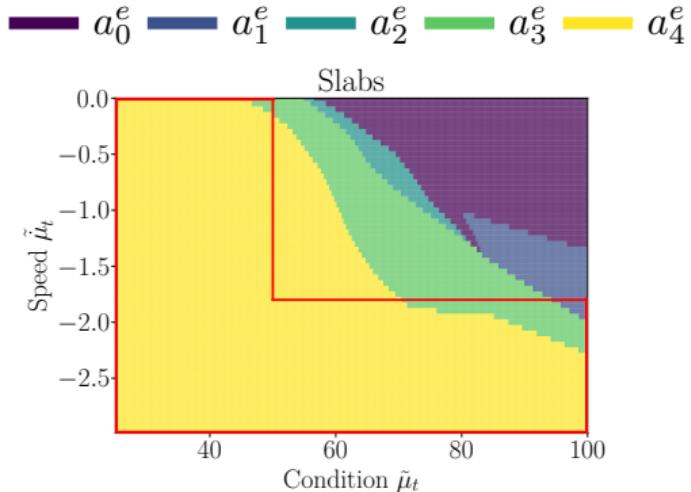
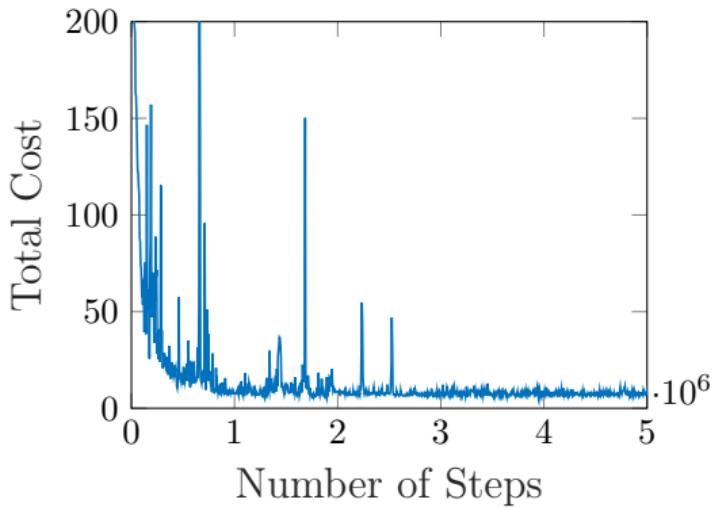
Maintenance Policy for Slabs

Maintenance Policy for Slabs



Maintenance Policy for Slabs

Maintenance Policy for Slabs



Deriving Expected Costs of Delays

$Q^*(s_t, a_t)$: optimal Q function associated with $a_t \sim \pi^*(s_t)$,

Deriving Expected Costs of Delays

$Q^*(s_t, a_t)$: optimal Q function associated with $a_t \sim \pi^*(s_t)$,

Consider,

\bar{a} : any action that is not optimal, $\bar{a} \neq a_t \sim \pi^*(s_t)$,

Deriving Expected Costs of Delays

$Q^*(s_t, a_t)$: optimal Q function associated with $a_t \sim \pi^*(s_t)$,

Consider,

\bar{a} : any action that is not optimal, $\bar{a} \neq a_t \sim \pi^*(s_t)$,

The incurred cost Δ of taking \bar{a} ,

Deriving Expected Costs of Delays

$Q^*(s_t, a_t)$: optimal Q function associated with $a_t \sim \pi^*(s_t)$,

Consider,

\bar{a} : any action that is not optimal, $\bar{a} \neq a_t \sim \pi^*(s_t)$,

The incurred cost Δ of taking \bar{a} ,

$$\Delta_t(s_t, \bar{a}) = Q(s_t, \bar{a}) - Q^*(s_t, a_t).$$

Deriving Expected Costs of Delays

$Q^*(s_t, a_t)$: optimal Q function associated with $a_t \sim \pi^*(s_t)$,

Consider,

\bar{a} : any action that is not optimal, $\bar{a} \neq a_t \sim \pi^*(s_t)$,

The incurred cost Δ of taking \bar{a} ,

$$\Delta_{t:T}(s_t, \bar{a}) = \sum_t \gamma^t [Q(s_t, \bar{a}) - Q^*(s_t, a_t)], \quad \forall t = 0, \dots, T.$$

Deriving Expected Costs of Delays

$Q^*(s_t, a_t)$: optimal Q function associated with $a_t \sim \pi^*(s_t)$,

Consider,

\bar{a} : any action that is not optimal, $\bar{a} \neq a_t \sim \pi^*(s_t)$,

The incurred cost Δ of taking \bar{a} ,

$$\Delta_{t:T}(s_t, \bar{a}) = \sum_t \gamma^t [Q(s_t, \bar{a}) - Q^*(s_t, a_t)], \quad \forall t = 0, \dots, T.$$

For a better interpretability,

Deriving Expected Costs of Delays

$Q^*(s_t, a_t)$: optimal Q function associated with $a_t \sim \pi^*(s_t)$,

Consider,

\bar{a} : any action that is not optimal, $\bar{a} \neq a_t \sim \pi^*(s_t)$,

The incurred cost Δ of taking \bar{a} ,

$$\Delta_{t:T}(s_t, \bar{a}) = \sum_t \gamma^t [Q(s_t, \bar{a}) - Q^*(s_t, a_t)], \quad \forall t = 0, \dots, T.$$

For a better interpretability,

$$\rho_{t:T} = \frac{Q^*(s_t, a_t) + \Delta_{t:T}(s_t, \bar{a})}{Q^*(s_t, a_t)}.$$

Deriving Expected Costs of Delays

$Q^*(s_t, a_t)$: optimal Q function associated with $a_t \sim \pi^*(s_t)$,

Consider,

\bar{a} : any action that is not optimal, $\bar{a} \neq a_t \sim \pi^*(s_t)$,

The incurred cost Δ of taking \bar{a} ,

$$\Delta_{t:T}(s_t, \bar{a}) = \sum_t \gamma^t [Q(s_t, \bar{a}) - Q^*(s_t, a_t)], \quad \forall t = 0, \dots, T.$$

For a better interpretability,

$$\rho_{t:T} = \frac{Q^*(s_t, a_t) + \Delta_{t:T}(s_t, \bar{a})}{Q^*(s_t, a_t)}.$$

$\rho_{t:T}$: cost ratio due to maintenance delays ($\rho_{t:T} \geq 1$).

Deriving Expected Costs of Delays

For a bridge \mathcal{B}_j with P_j elements,

Deriving Expected Costs of Delays

For a bridge \mathcal{B}_j with P_j elements,

$$\bar{\rho}_{t:T} = \frac{\sum_p^{P_j} \omega_p [Q_p^*(s_t, a_t) + \Delta_{t:T}(s_t, \bar{a})]}{\sum_p^{P_j} \omega_p \times Q_p^*(s_t, a_t)}.$$

Deriving Expected Costs of Delays

For a bridge \mathcal{B}_j with P_j elements,

$$\bar{\rho}_{t:T} = \frac{\sum_p^{P_j} \omega_p [Q_p^*(s_t, a_t) + \Delta_{t:T}(s_t, \bar{a})]}{\sum_p^{P_j} \omega_p \times Q_p^*(s_t, a_t)}.$$

where,

Deriving Expected Costs of Delays

For a bridge \mathcal{B}_j with P_j elements,

$$\bar{\rho}_{t:T} = \frac{\sum_p^{P_j} \omega_p [Q_p^*(s_t, a_t) + \Delta_{t:T}(s_t, \bar{a})]}{\sum_p^{P_j} \omega_p \times Q_p^*(s_t, a_t)}.$$

where,

ω_p : within-category size ratio of the p -th structural element.

Bridge Description

Bridge Description

Bridge Components:

Bridge Description

Bridge Components:

- ▷ Beams ×15.

Bridge Description

Bridge Components:

- ▷ Beams ×15.
- ▷ Front walls ×2.

Bridge Description

Bridge Components:

- ▷ Beams ×15.
- ▷ Front walls ×2.
- ▷ Slabs ×3.

Bridge Description

Bridge Components:

- ▷ Beams ×15.
- ▷ Front walls ×2.
- ▷ Slabs ×3.
- ▷ Guardrail ×2.

Bridge Description

Bridge Components:

- ▷ Beams ×15.
- ▷ Front walls ×2.
- ▷ Slabs ×3.
- ▷ Guardrail ×2.
- ▷ Wing walls ×4.

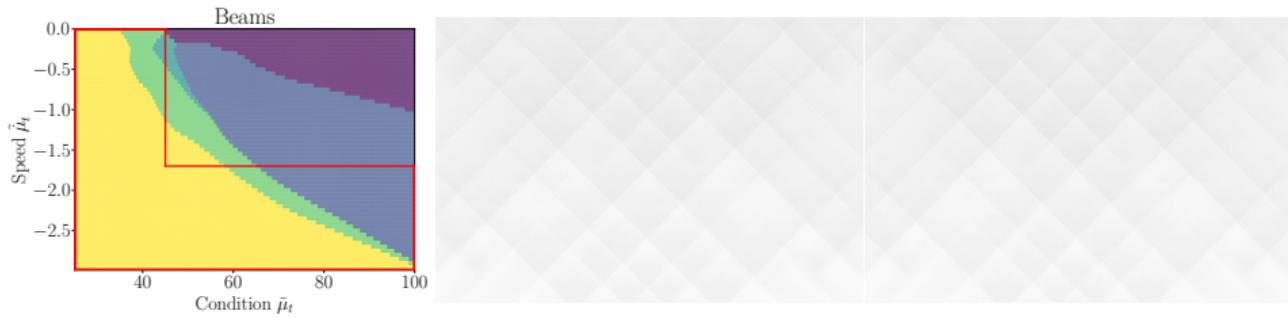
Bridge Description

Bridge Components:

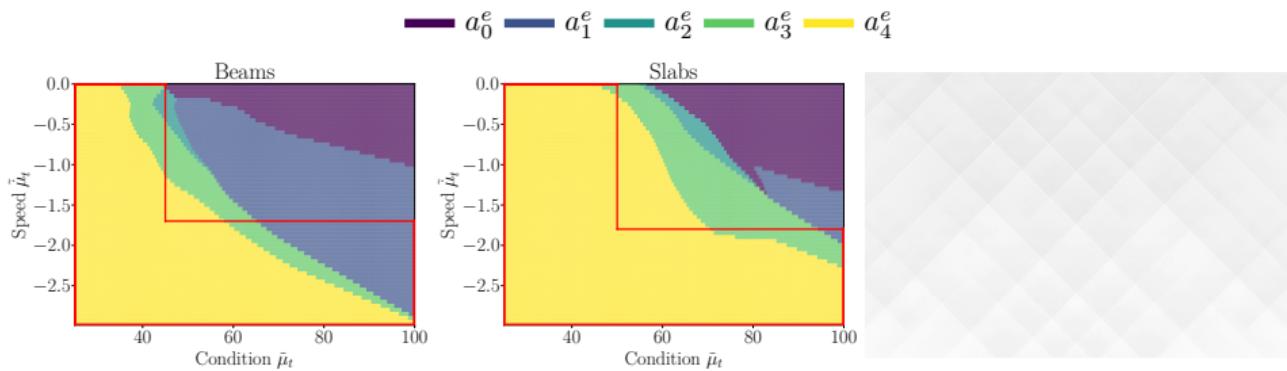
- ▷ Beams ×15.
- ▷ Front walls ×2.
- ▷ Slabs ×3.
- ▷ Guardrail ×2.
- ▷ Wing walls ×4.
- ▷ Pavements ×3.

Decision Boundaries for Maintenance Policies

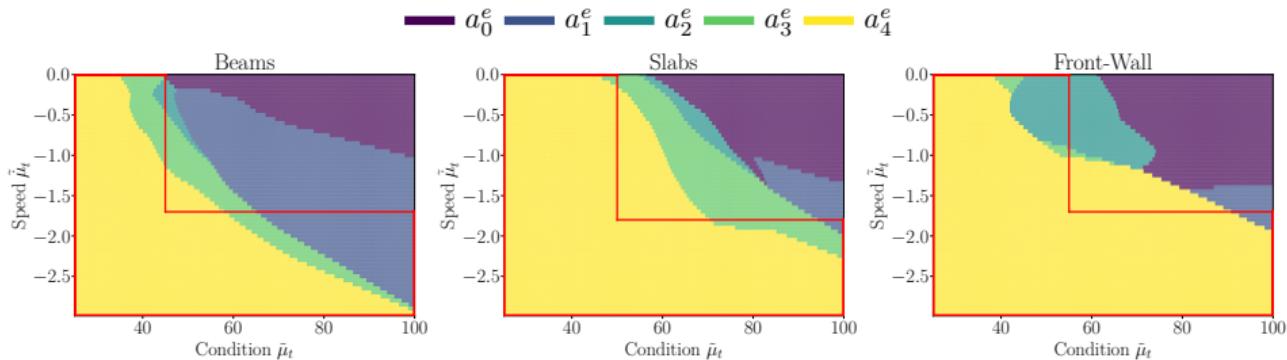
— a_0^e — a_1^e — a_2^e — a_3^e — a_4^e



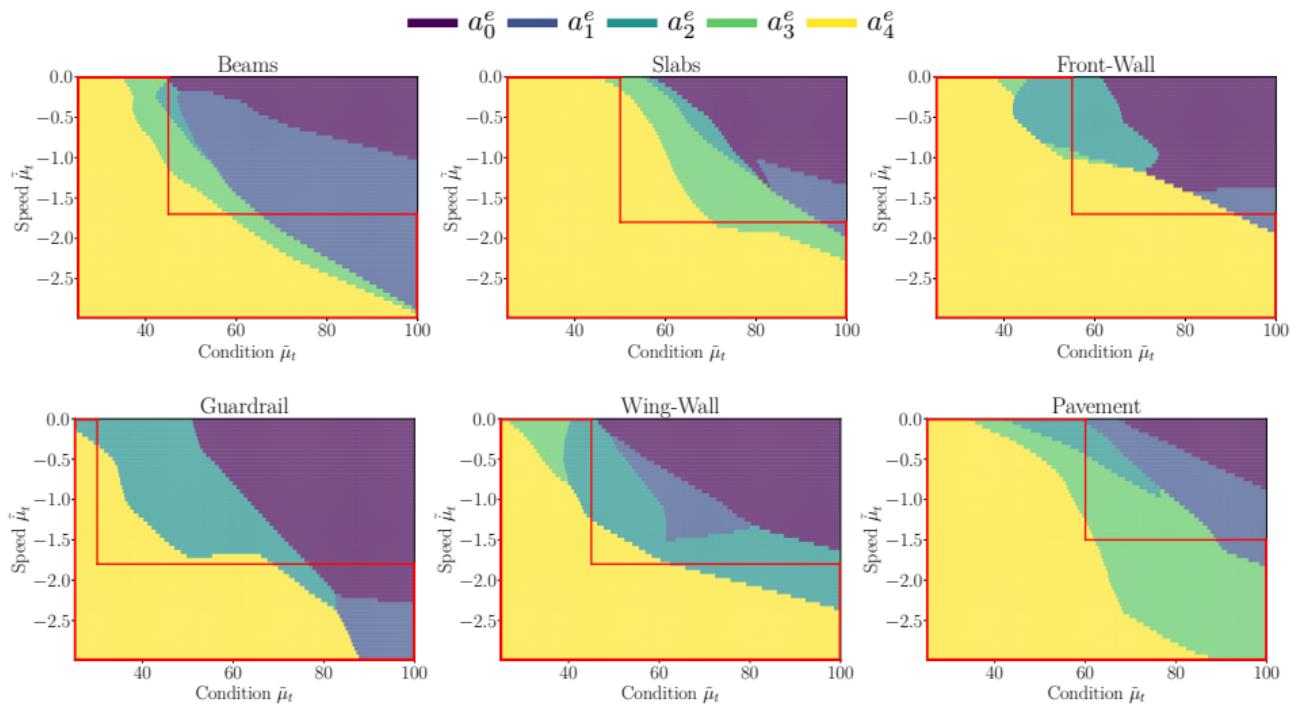
Decision Boundaries for Maintenance Policies



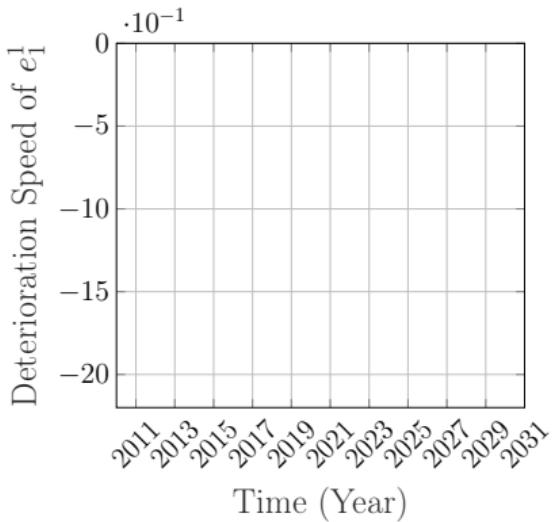
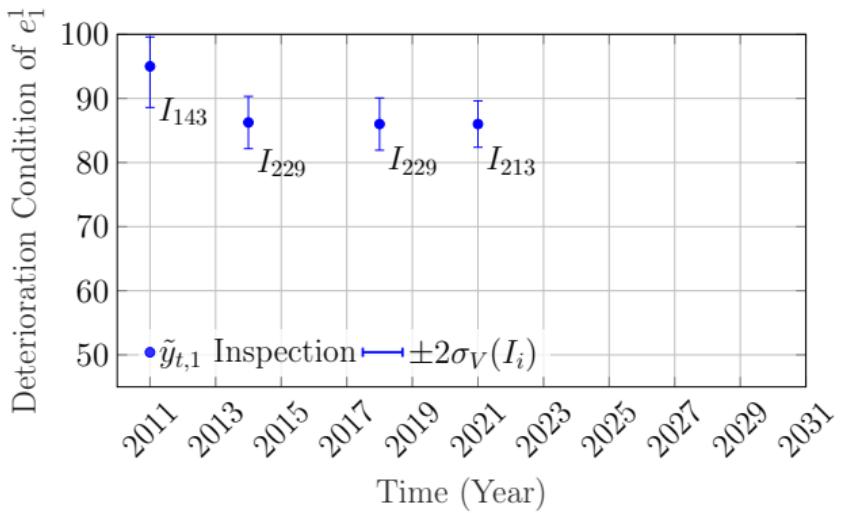
Decision Boundaries for Maintenance Policies



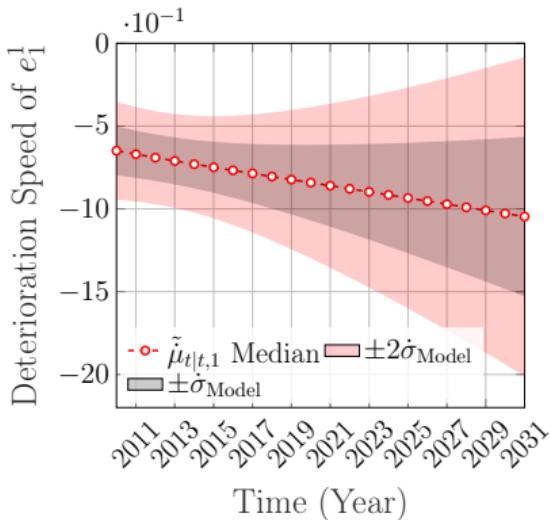
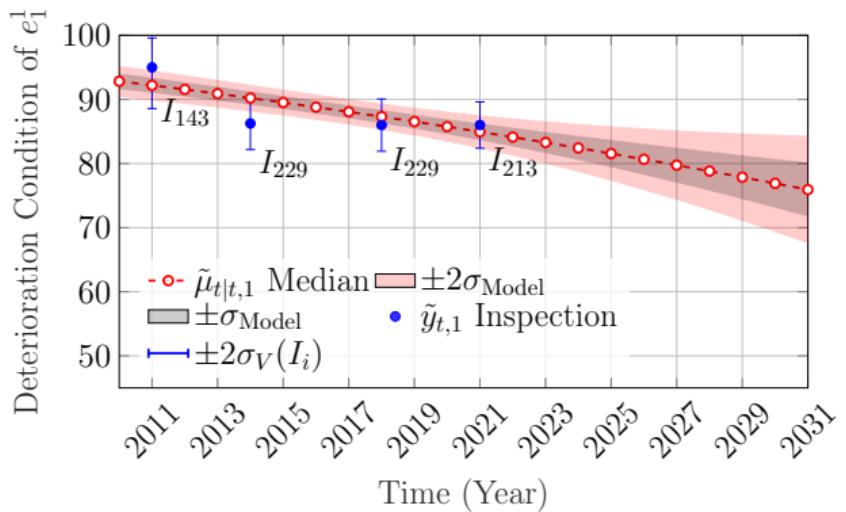
Decision Boundaries for Maintenance Policies



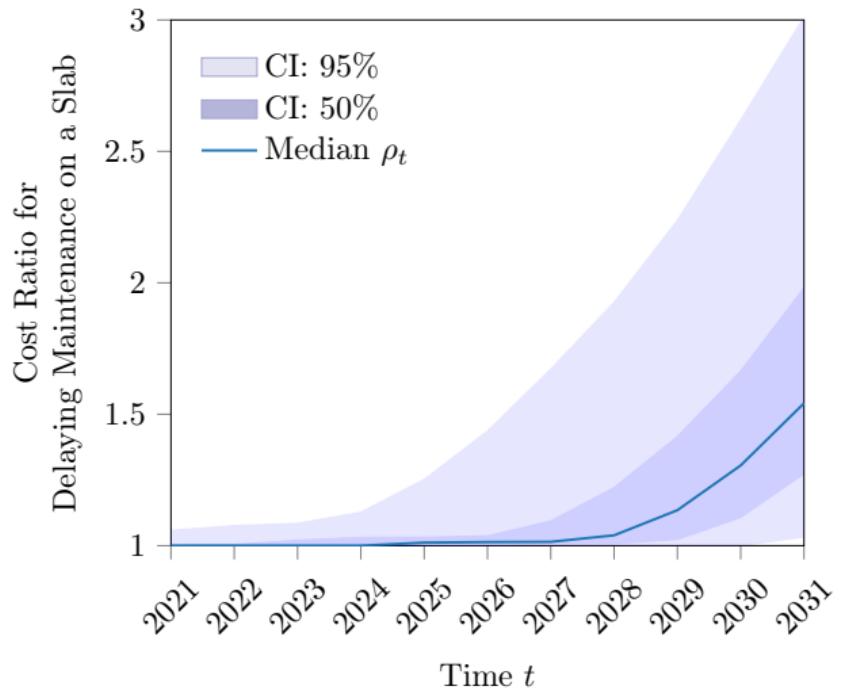
Slab Structural Element



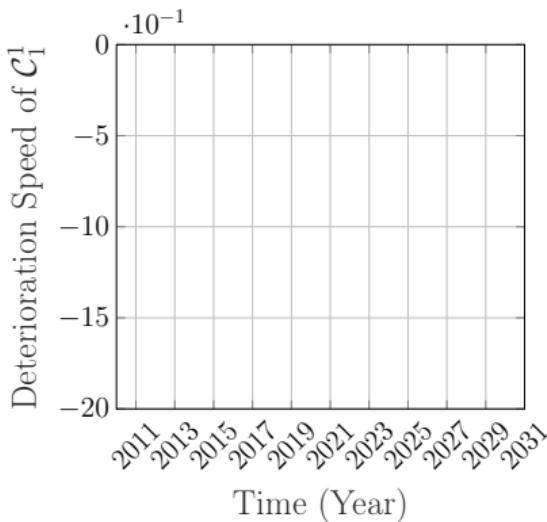
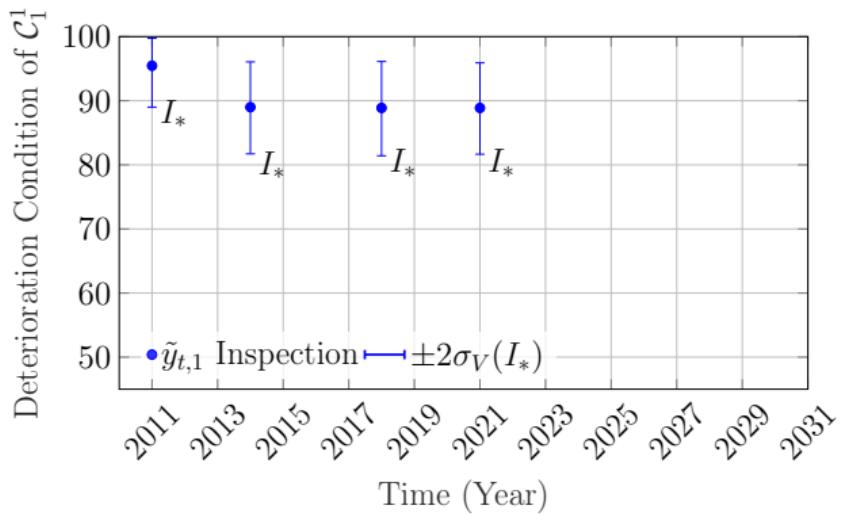
Slab Structural Element



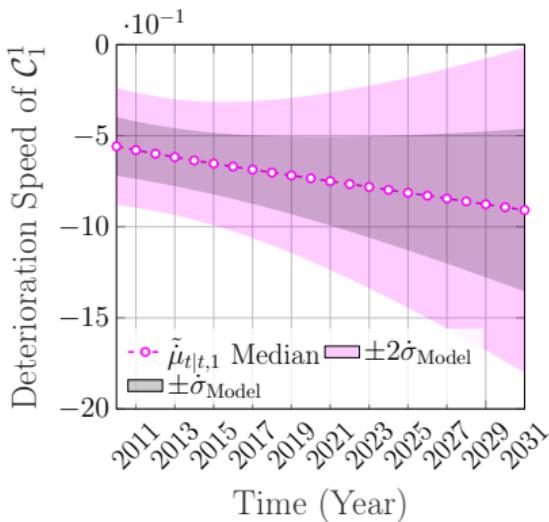
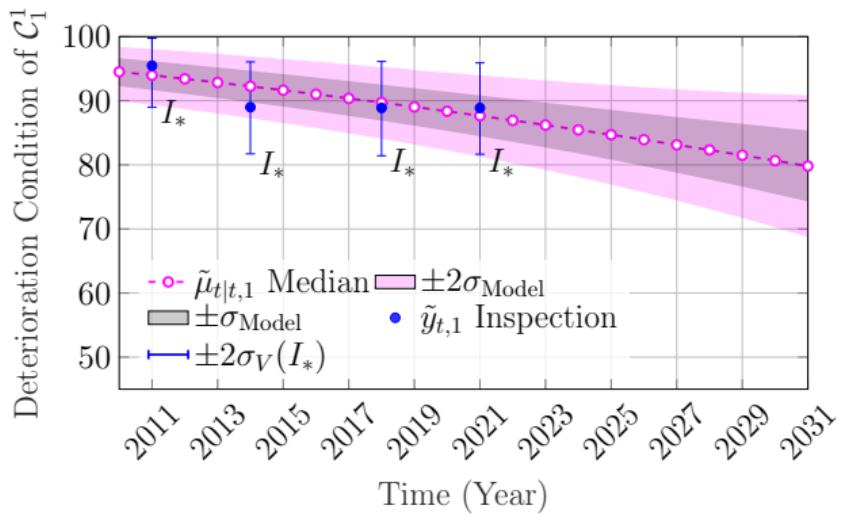
Slab Structural Element



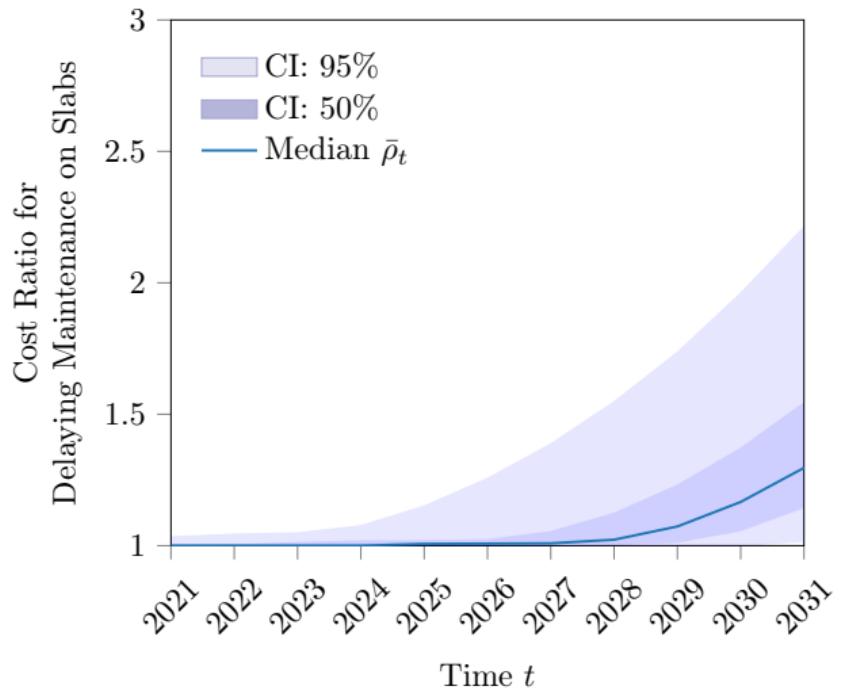
Slab Structural Category in the Bridge



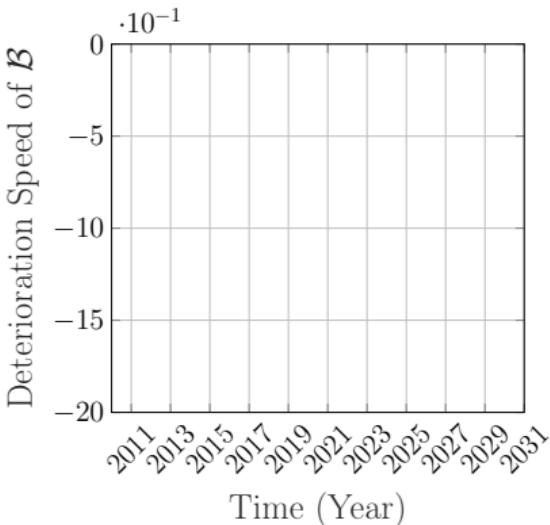
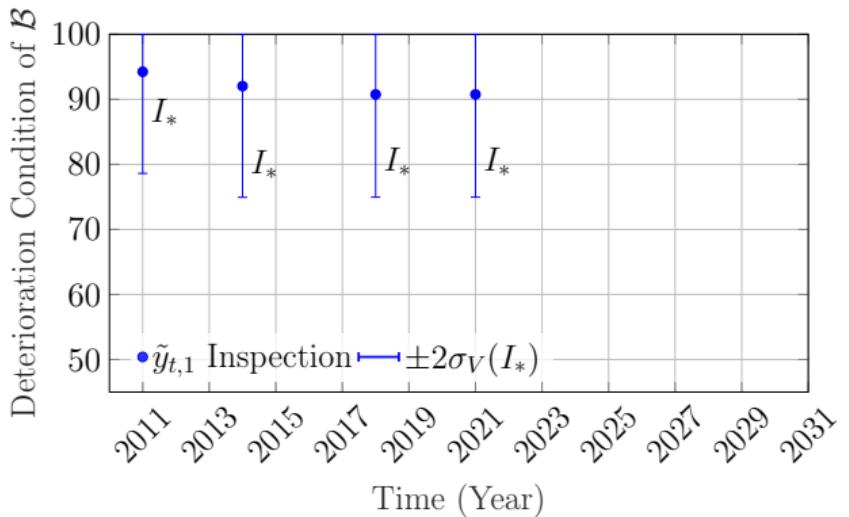
Slab Structural Category in the Bridge



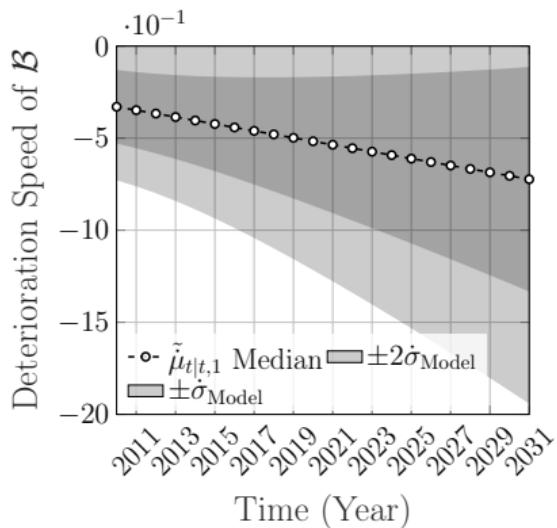
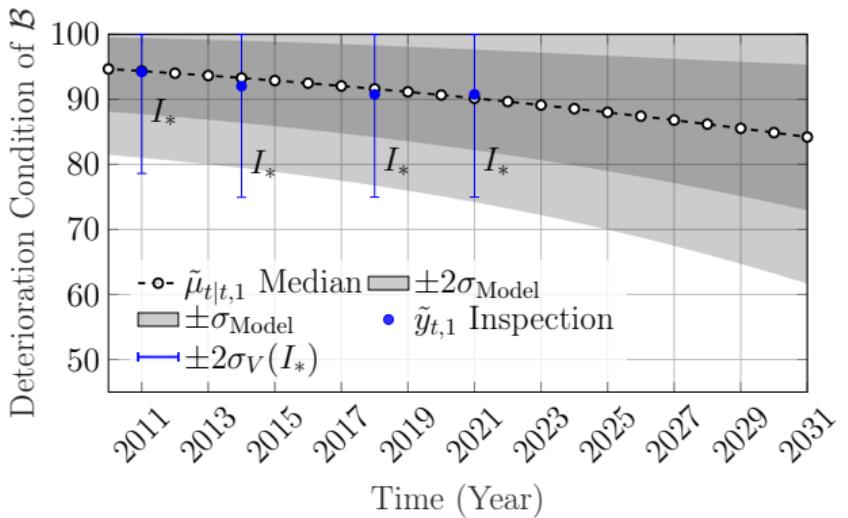
Slab Structural Category in the Bridge



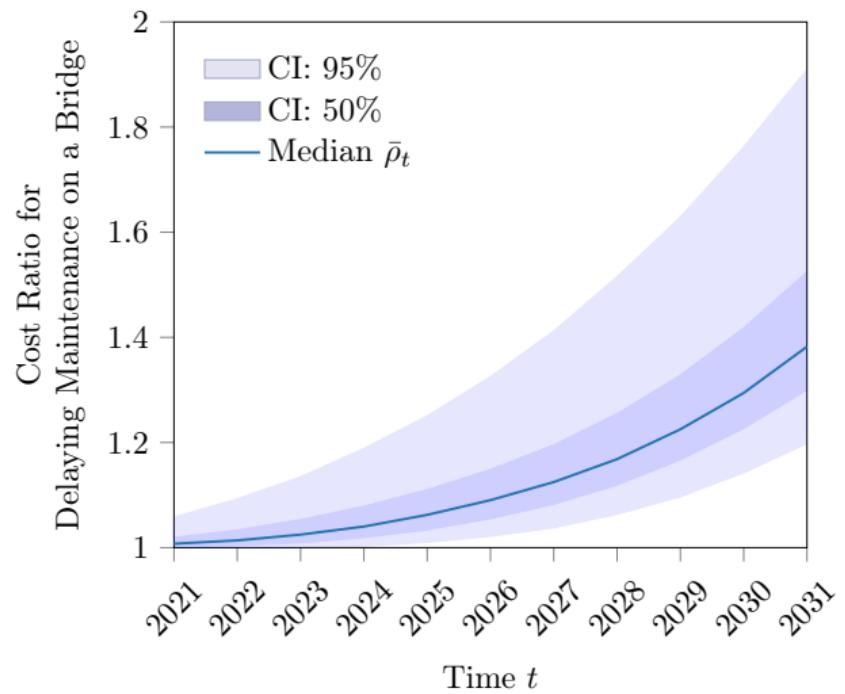
Bridge Analyses



Bridge Analyses



Bridge Analyses



Existing Limitations

Limitations

Limitations

- ▶ Results are dependent on the cost function:

Limitations

- ▶ Results are dependent on the cost function:
 - ▷ The relation between the cost of an action and the deterioration state.

Limitations

- ▶ Results are dependent on the cost function:
 - ▷ The relation between the cost of an action and the deterioration state.
 - ▷ Network connectivity & re-routing costs.

Limitations

- ▶ Results are dependent on the cost function:
 - ▷ The relation between the cost of an action and the deterioration state.
 - ▷ Network connectivity & re-routing costs.
- ▶ Unaddressed challenges:

Limitations

- ▶ Results are dependent on the cost function:
 - ▷ The relation between the cost of an action and the deterioration state.
 - ▷ Network connectivity & re-routing costs.
- ▶ Unaddressed challenges:
 - ▷ The RL agent acting in a POMDP environment instead of MDP.

Limitations

- ▶ Results are dependent on the cost function:
 - ▷ The relation between the cost of an action and the deterioration state.
 - ▷ Network connectivity & re-routing costs.
- ▶ Unaddressed challenges:
 - ▷ The RL agent acting in a POMDP environment instead of MDP.
 - ▷ Factoring duration of service stoppage as an additional cost.

Limitations

- ▶ Results are dependent on the cost function:
 - ▷ The relation between the cost of an action and the deterioration state.
 - ▷ Network connectivity & re-routing costs.
- ▶ Unaddressed challenges:
 - ▷ The RL agent acting in a POMDP environment instead of MDP.
 - ▷ Factoring duration of service stoppage as an additional cost.

Future Work

- ▶ Compare DQN vs. DDQN vs. PPO for a single policy.

Conclusions & Future Work

Future Work

- ▶ Compare DQN vs. DDQN vs. PPO for a single policy.

- ▶ Examine the extreme scenarios for the deterioration state.

