InfraRL: A Benchmark for Constrained Resource Allocation in Large-Scale Infrastructure Asset Management

Yantian Wang[1], Wenhao Li[1], Bo Jin[1,2]

[1]School of Computer Science and Technology, Tongji University, China

[2]Shanghai Research Institute for Intelligent Autonomous Systems

MAIL-TAII LAB github 请加微信联系

Correspondence: Wenhao Li (whli@tongji.edu.cn)

Bo Jin (bjin@tongji.edu.cn)

Why is InfraRL needed?

Goal: Make infrastructure maintenance a realistic offline constrained RL problem, rather than a toy simulator or unconstrained control task.

Benchmark: A real-world offline constrained RL benchmark showing when value learning, allocation, and heuristic guidance matter for infrastructure management.

Why existing RL benchmarks are insufficient

- **No safe online exploration:** maintenance actions are expensive, delayed, and cannot be tested by trial-and-error on real bridges.
- **Budgets couple many assets:** agencies must allocate limited annual resources across regions instead of optimizing one independent agent.
- **Historical logs are messy:** real administrative data need action typing and causal verification before they become valid RL trajectories.

Benchmark comparison

Feat.	InfraRL	IMP	Sust.	DSRL	SGym	D4RL
Offline	✔	✗	✗	✔	✗	✔
MA	✔	✔	✔	✗	✗	✗
Real logs	✔	✗	✔	Hyb.	✗	✗
Constraints	✔	✔	✔	✔	✔	✗

IMP=IMP-MARL; Sust.=SustainGym; SGym=Safety-Gym; Hyb.=Hybrid.

What gap InfraRL fills

- Real NBI bridge records → verified offline CMDP episodes.
- Shared constraints, allocators, and metrics for fair method comparison.
- Long-horizon evaluation of health gains, feasibility, and policy fidelity.

Benchmark gap

InfraRL jointly covers offline data, real logs, multi-agent allocation, and native constraints, which prior benchmarks cover only partially.

What makes the problem hard?

Hard-budget decision making

- **Actions:** No Action, Minor Repair, Major Repair, Replacement.
- **Constraint:** annual budget must satisfy $C(\mathbf{s}, \mathbf{a}) \leq B$.
- **Objective:** improve long-term bridge health under coupled allocation.

Offline constraint

Learn from fixed administrative logs; no trial-and-error interaction with infrastructure.

Algorithms must optimize value while respecting a shared annual budget at execution time.

How do we make NBI RL-ready?

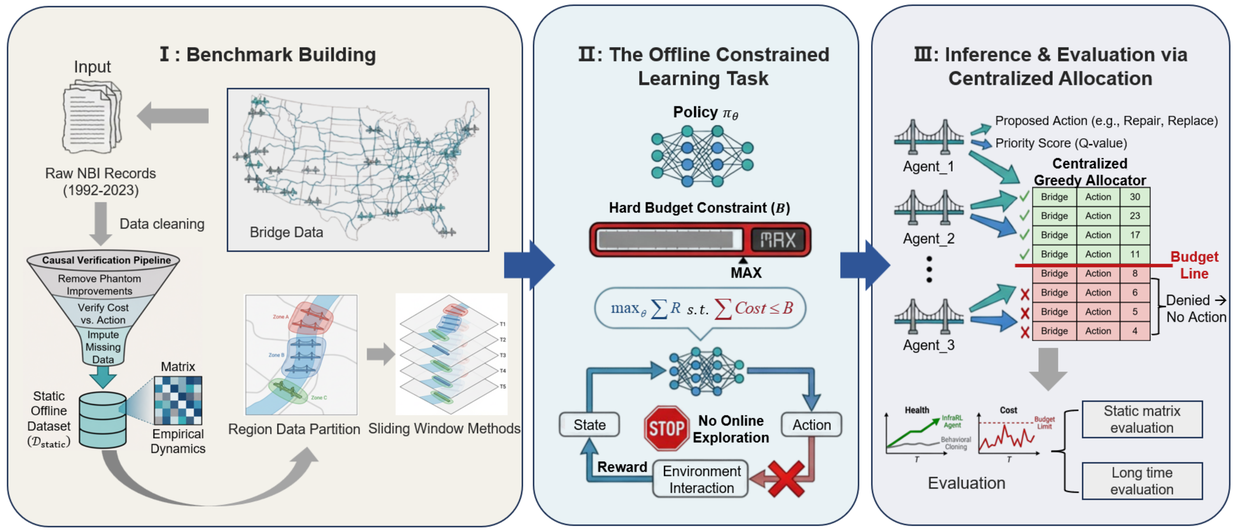
From records to episodes

- Raw NBI records → action typing.
- Execution verification → causal/logical filtering.
- Regional sampling → offline episodes.
- Reject phantom actions and preserve rare repair decisions.

Scope: CA highway bridges, 1992–2023; 400 regions; 2,000 episodes; 15-year windows / 5-year stride.

Evaluation: compare representative baselines by utility, feasibility, fidelity, FQE, and 100-year robustness.

How does InfraRL work?



Benchmark building: NBI logs → verified regional episodes.

Offline constrained RL: Learn from fixed buffers under annual budgets.

Safe execution: Allocator enforces feasible joint actions.

What did InfraRL reveal?

Evaluation: BC, CQL, CPQ, QMIX-CQL-MF, MPC+Forecasting, and CQL-Heuristic are compared by utility gain, feasibility, fidelity, FQE, and 100-year robustness.

Three empirical findings

- **Value learning matters:** CQL / CPQ improve health with low raw budget violations.
- **Imitation is insufficient:** BC stays close to logs, while QMIX variants often violate budgets.
- **Guidance helps:** CQL-Heuristic gives the best non-oracle learned FQE.

Representative algorithm performance

Method	Budget Ratio	Improve vs Hist.	Raw Violation	FQE
MultiTask-BC	0.824	-0.022	**0.0%**	-0.104
CQL	0.891	**0.430**	0.7%	-0.104
MultiTask-CPQ	0.920	0.394	1.7%	**-0.094**
IQL-CQL-MARL	0.829	-0.012	0.7%	-0.111
QMIX-CQL	0.907	-0.067	71.1%	-0.209
QMIX-CQL-MF	1.002	0.085	71.5%	-0.183
MPC+Forecasting	1.007	0.417	N/A	N/A
MPC-Oracle*	1.006	1.185	N/A	N/A

*Privileged planning reference; not a fair learned baseline.

Allocator sensitivity

Method	Greedy	Cost-ratio	Knapsack
MT-BC	-0.042	-0.027	-0.021
CQL	0.430	0.662	0.703
MT-CPQ	0.394	0.599	0.661
QMIX-MF	0.085	0.406	0.444
CQL-Heur	0.708	0.928	**1.000**

Metric: normalized Improve vs History. Cost-ratio and knapsack raise scores while preserving the ranking.

Key message: value learning improves health, allocator choice raises execution quality, and heuristic guidance gives the best non-oracle learned FQE.