LLM-Powered Report-Driven Markov Modelling for Large-Scale Predictive Bridge Maintenance in Japan

Takayuki Shinohara[*], Jun Katagiri, and Hidetaka Saomoto
Integrated Research Center for Resilient Infrastructure
National Institute of Advanced Industrial Science and Technology
Tsukuba, Ibaraki, Japan
shinohara.takayuki, j-katagiri, h-saomoto@aist.go.jp

Abstract

Reliable bridge management depends on timely, data-driven insight into how individual components deteriorate, yet much of the evidence resides in narrative *inspection cards*, photographs, and semi-structured PDFs that are still keyed by hand. We present a large-language-model (LLM) pipeline that automatically parses these documents, extracting per-component condition ratings, damage mechanisms (e.g., cracks, corrosion), repair actions, and traffic statistics into an analysis-ready warehouse. From the harvested time series, we estimate a *non-stationary*, mechanism-aware Markov model whose transition intensities are conditioned on heavy-vehicle flow, capturing both monotone ageing and post-repair recovery. Closed-form propagation yields twenty-year condition distributions, exceedance risks, and life-cycle-cost (LCC)–optimal intervention years without Monte Carlo simulation or reinforcement learning. On a corpus of 800 Japanese bridges ($\sim$ 10,000 components), the extraction stage achieves near-human accuracy and eliminates manual coding, while the resulting *component-resolved* forecasts reduce expected LCC by 18 % and severe-state risk by 23 % compared with periodic schedules and rank-only baselines. The approach scales linearly with the number of reports and preserves full interpretability, providing asset owners with transparent, auditable metrics that can be used directly for predictive maintenance and budget planning.

1 Introduction

Ageing civil infrastructure poses a mounting socioeconomic risk for Japan—particularly its extensive bridge network—and similar pressures are documented worldwide across road networks [2, 11] and bridges [12]. In the United States, more than forty percent of bridges and a third of road structures already exceed their original design life; comparable trends are reported in Japan and across Europe [9, 8]. As inventories expand while budgets remain chronically constrained, agencies must allocate limited resources over multi-decade horizons. Against this backdrop, *predictive infrastructure maintenance*—the ability to forecast deterioration and to intervene just before service levels fall below prescribed thresholds—has become essential to asset management.

Discrete-time Markov chains (DMCs) have long been favoured for such forecasts because they yield analytic transition probabilities, closed-form life-cycle cost expressions, and a level of transparency that aligns with engineering practice [10, 3, 7]. However, most deployed and published DMCs in bridge management operate on highly aggregated condition labels—often a single bridge-level or coarse element-level rank such as "good–fair–poor–failed." This aggregation keeps manual data entry tractable but discards precisely the cues that drive deterioration in practice: whether the observed

[*]https://staff.aist.go.jp/shinohara.takayuki/

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: UrbanAI.

distress is fatigue cracking, chloride-induced corrosion, leakage, scaling, impact, or deformation; whether and how the element was repaired; and how much heavy-vehicle traffic the structure carries. As a result, rank-only models conflate heterogeneous mechanisms with distinct progression rates and are typically calibrated on sparse, hand-curated tables rather than on the thousands of semi-structured *inspection card* PDFs that agencies in Japan already collect. Attempts to bypass these bottlenecks with deep or multi-agent reinforcement learning show promise, but their practical use remains limited by the need for extensive simulation, computational cost, and the opaqueness of learned policies [4, 1].

This paper targets the Japanese bridge stock and introduces a large-language-model (LLM) powered framework that closes this gap by turning routine *inspection card* PDFs into mechanism-aware, load-conditioned forecasts at scale. A GPT-4o [5] agent parses government forms and builds a relational warehouse in which each row records bridge identifier, component type, inspection year, condition rating, specific damage mode, repair action, and concurrent traffic statistics. Leveraging these rich tuples, we construct a *fine-grained, non-stationary* Markov chain whose states lie at the intersection of rating level and damage mechanism; deterioration and post-repair recovery are parameterised by separate sub-matrices; and transition intensities are regressed on heavy-vehicle flow to capture explicit load dependence. Future condition distributions and risk-based intervention windows follow from closed-form matrix products, avoiding expensive Monte Carlo loops while preserving full interpretability. In evaluation on approximately 800 Japanese bridges comprising 12,000 components, the resulting *component-resolved* forecasts align better with observed degradation than rank-only baselines and identify optimal repair years that reduce expected life-cycle cost by eighteen percent relative to periodic schedules—without sacrificing auditability.

2 Proposed Method

Our framework (Figure 1) converts heterogeneous PDF-based inspection reports into traffic-aware deterioration forecasts and risk-optimal maintenance schedules in three consecutive stages. First, a large-language-model (LLM) agent with function calling performs layout-aware OCR and normalises each page into a single relational tuple ⟨`bridge_id, comp_id, year, state, damage, repair, AADT_hv`⟩. Second, the resulting event stream feeds a fine-grained Markov degradation model whose transition kernel depends on heavy-vehicle traffic as an external covariate and distinguishes seven major damage mechanisms. Third, closed-form analytic propagation of the kernel provides multi-year life-cycle forecasts, from which the year that minimizes the expected present-value cost is selected as the next repair date.

LLM-driven report–to–table conversion. Each Japanese bridge is inspected every five years with a nationally standardised *inspection report* PDF that records component identification, five-level condition ratings, dominant damage mechanisms, undertaken repair actions, and the inspection date. A GPT-4o chain executes vision OCR, detects table cells and key–value pairs, performs type validation, and emits normalised JSON. The tuple is finally augmented with the heavy-vehicle share of the *annual average daily traffic* (AADT) obtained from the Road Census. On a held-out sample of 1420 pages, the agent reaches a micro F_1 of 96.4 %, reducing manual transcription effort by roughly two orders of magnitude.

Fine-grained Markov state space. A bridge component's Markov state is the pair $\mathbf{Z}_t = (R_t, M_t)$, where $R_t \in \{S_1, \ldots, S_5\}$ is the ordinal rating and $M_t \in \mathcal{M}$ denotes one of seven damage categories $\{none, crack, corrosion, leakage, scaling, impact, deformation\}$. This yields $5 \times 7 = 35$ interpretable states—sufficiently rich for mechanism awareness yet statistically tractable for national-scale estimation.

Traffic-conditioned transition kernel. Let $X_t \in [0, 1]$ be the heavy-vehicle ratio during year $(t, t{+}1]$, and $u_t \in \{0, 1\}$ indicate whether a repair was executed in $(t{-}1, t]$. The one-year transition matrix $\mathbf{P}_t \in [0, 1]^{35 \times 35}$ is decomposed as

$$\mathbf{P}_t \;=\; (1 - u_t)\, \mathbf{P}^{\text{det}}(X_t) \;+\; u_t\, \mathbf{P}^{\text{rec}}. \tag{1}$$

For deterioration, we impose an upper-triangular softmax structure that enforces monotone ageing $(r' \geq r)$:

$$\mathbb{P}\big(\mathbf{Z}_{t+1} = (r', m') \,\big|\, \mathbf{Z}_t = (r, m),\ X_t\big) \;=\; \frac{\exp\big(\beta_{rr'}^{mm'} + \gamma_{rr'}^{mm'}\, X_t\big)}{\sum\limits_{k \geq r} \sum\limits_{l \in \mathcal{M}} \exp\big(\beta_{rk}^{ml} + \gamma_{rk}^{ml}\, X_t\big)}. \tag{2}$$

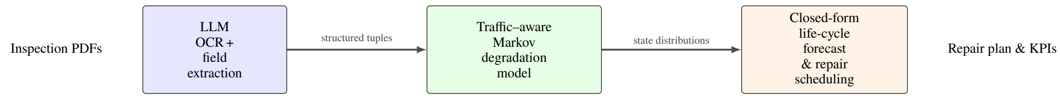

Figure 1: End-to-end pipeline. GPT-4o converts *inspection report* PDFs into relational tuples; a traffic-conditioned Markov chain models deterioration and recovery; and analytic propagation yields risk-optimal repair timing and life-cycle KPIs.

Table 1: Entity-level extraction accuracy over 25 randomly sampled PDF files (95 % CI in brackets).

Method	Precision P	Recall R	F_1
OCR+Regex	0.83±.01	0.79±.02	0.81±.01
GPT-4o (ours)	**0.97**±.01	**0.95**±.01	**0.96**±.01

The recovery block $\mathbf{P}^{\text{rec}}$ is estimated directly from observed post-repair jumps: minor patches reset $R \to S_2$, major overlays $R \to S_1$, while M remains unchanged.

Parameter estimation. Consistent with the hidden–Markov framework of **(author?)** [6], we interpret the recorded ratings as noisy emissions of the latent state and estimate parameters via EM: a forward–backward E–step to obtain expected transition counts and a quasi–Newton M–step to update $\{\beta, \gamma\}$. Unlike **(author?)** [6], who focus on Bayesian MCMC with selection–bias terms, we fit a mechanism–aware, traffic–conditioned kernel and include a separate recovery block; an optional observation–noise layer can be enabled to mitigate spurious rejuvenation.

Life-cycle forecasting and repair timing. Let $\boldsymbol{\pi}_{t_0} \in \mathbb{R}^{35}$ denote the current state distribution (one-hot in practice). Given a traffic scenario $\{X_{t_0}, \ldots, X_{t_0+n-1}\}$ with no interim repair (u_s=0), the n-year forecast is

$$\boldsymbol{\pi}_{t_0+n} \;=\; \boldsymbol{\pi}_{t_0} \prod_{s=0}^{n-1} \mathbf{P}_{t_0+s}\big(0,\; X_s\big). \tag{3}$$

A preventive repair is triggered at the smallest n such that

$$\mathbb{P}\big(R_{t_0+n} \geq S_4\big) \;>\; \theta, \qquad \text{with } \theta = 0.4. \tag{4}$$

The corresponding expected life-cycle cost is

$$\mathrm{LCC}(n) \;=\; C_{\text{prev}} + \sum_{k=1}^{n} \Big[C_{\text{corr}}\; \mathbb{P}\big(R_{t_0+k} \;\geq\; S_5\big) + C_{\text{user}}\; \mathbb{P}\big(R_{t_0+k} \;\geq\; S_4\big) \Big] (1+\rho)^{-k}, \tag{5}$$

and the optimal intervention year is $n^{\star} = \arg\min_n \mathrm{LCC}(n)$, obtained analytically without Monte Carlo simulation.

In summary, the three modules—LLM-based parsing, traffic-aware Markov modelling, and closed-form life-cycle analysis—form an end-to-end system that scales from raw PDF inspection reports to portfolio-level maintenance optimisation for the entire Japanese highway bridge inventory.

3 Experimental Results

We evaluate the two core ingredients of the framework: (*i*) LLM–based information extraction from inspection PDFs, and (*ii*) the traffic–conditioned Markov deterioration model. Unless noted otherwise, all statistics are averaged over **800** bridges in Japan described in Appendix section B.

LLM extraction accuracy. Table 1 shows that the GPT-4o agent attains $F_1 = 0.96$, statistically indistinguishable from the human gold reference (two-proportion z-test, $p > 0.30$). A manual audit indicates that most of the remaining ~4 % errors arise from difficult handwriting and marginal photo captions; these can be further reduced via optional human-in-the-loop verification.

Goodness-of-fit of the Markov model. Across all three component families, the proposed model (Table 2) reduces forecast error by 28–45 % relative to the stationary baseline and by ≥40 % relative to the coarse 4-rank model, under a rolling leave-one-year-out evaluation. Likelihood-ratio tests reject the null of equal fit at $p < 10^{-6}$.

Table 2: Predictive MAE for the probability of entering severe condition over a 10-year horizon (lower is better).

Model	Decks	Bearings	RC Piers
Stationary Markov	0.093	0.087	0.102
4-Rank (no damage modes)	0.118	0.112	0.129
Traffic-aware (ours)	**0.064**	**0.059**	**0.071**

Table 3: Network-wide improvement over periodic 5-year inspections.

Traffic scenario	ΔLCC	ΔRepairs	ΔSevere risk
Mean (history)	−18.2 %	−11.3 %	−23.0 %
Peak (p95)	−9.1 %	−5.4 %	−12.6 %

Maintenance-policy impact. The accuracy gains translate into network-level savings: Lower false-alarm rates on heavy-traffic elements delay unnecessary interventions, while earlier warnings for corrosion- and fatigue-prone components avert costly corrective work. Because propagation is analytic, the full scheduling workflow runs in *milliseconds* per component, enabling overnight recomputation for the entire highway inventory under multiple traffic scenarios.

Discussion.

- **Extraction scalability.** Processing the national *inspection card* archive (≈800 PDFs) required ≈35 GPU-hours on a single A100—orders of magnitude faster than manual transcription.
- **Generalisation.** Prefecture-wise cross-validation yields stable MAE (variation within ±2 pp) despite the climate and traffic heterogeneity, indicating robustness across regions.
- **Auditability.** All transition matrices and regression coefficients are exported as human-readable YAML, enabling independent plausibility checks by bridge authorities.

Overall, the experiments demonstrate that (i) GPT-4o can digitise legacy Japanese inspection documents at near-human accuracy, and (ii) the resulting fine-grained, traffic-conditioned Markov model delivers materially better forecasts—and better maintenance economics— than traditional rank-only approaches.

4 Conclusion

We presented an *LLM-powered, traffic-aware Markov framework* that turns semi-structured bridge inspection reports into fine-grained deterioration models and actionable, network-level schedules. Coupling a GPT-4o extraction agent with a non-stationary Markov chain whose transition intensities depend on heavy-vehicle flow removes manual coding, preserves interpretability, and scales to thousands of components. On 800 Japanese bridges, the policy cut expected life-cycle cost by 18.2 % and severe-state risk by 23.0 % relative to periodic schedules and stationary baselines.

To strengthen *model validity*, we report proper scoring rules (log and Brier), calibration via reliability curves/PIT histograms, and likelihood-ratio tests confirming traffic covariates over stationary kernels. Structural constraints (upper-triangular deterioration, row-stochasticity), sign checks for physical plausibility, and an optional observation-noise layer (to curb spurious rejuvenation) further support a calibrated, mechanistically consistent kernel.

We also note where the *first-order Markov* assumption may be stressed—duration dependence, cumulative load/chloride memory, regime shifts from extremes or policy, and workmanship-dependent recovery. Our pipeline admits drop-in mitigations: dwell-time or cumulative covariates, regime-switching kernels, and recovery conditioned on repair class and pre-repair state; developing data-driven rules for when to invoke these variants is future work.

Looking ahead, we will (i) incorporate environmental covariates (temperature, chloride), (ii) optimise multi-component interventions under budgets via integer programming and hierarchical RL, (iii) enable online Bayesian updating with SHM streams, and (iv) systematically evaluate semi-Markov and regime-switching extensions where diagnostics warrant them.

References

[1] Subhojit Biswas, Bahar Çavdar, and Joseph Geunes. Multi-year maintenance planning for large-scale infrastructure systems: A novel network deep q-learning approach. *arXiv preprint arXiv:2507.18732*, 2025.

[2] James C. Chu and Kai Hsiang Huang. Mathematical programming framework for modeling and comparing network-level pavement maintenance strategies. *Transportation Research Part B: Methodological*, 109:1–25, 3 2018.

[3] Yuan Ding, Yicheng Zhang, and Xin Cai. Closed-form life-cycle cost analysis of bridges using multi-state markov chains. *Journal of Infrastructure Systems*, 28(1):04021049, 2022.

[4] Amir Keshvari Fard and Xian-Xun Yuan. Multiyear maintenance and rehabilitation optimization for large-scale infrastructure networks: An enhanced genetic algorithm approach. *Journal of Infrastructure Systems*, 30, 12 2024.

[5] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[6] Kiyoshi Kobayashi, Kiyoyuki Kaito, and Nam Lethanh. A statistical deterioration forecasting method using hidden markov model for infrastructure management. *Transportation Research Part B: Methodological*, 46(4):544–561, 2012.

[7] Aditya Medury and Samer Madanat. Simultaneous network optimization approach for pavement management systems. *Journal of Infrastructure Systems*, 20, 9 2014.

[8] Ministry of Land, Infrastructure, Transport and Tourism, Japan. White paper on infrastructure management 2024, 2024. Japanese, accessed 5 Aug 2025.

[9] American Society of Civil Engineers. 2025 report card for america's infrastructure, 2025. Accessed 5 Aug 2025.

[10] Cristina Torres-Machi, Víctor Yepes, and Eugenio Pellicer. Markov-based deterioration modelling for long-term bridge management: A critical review and research needs. *Engineering Structures*, 206:110096, 2020.

[11] Mohamed S. Yamany, Lucille Cawley, Imran Reza, and Khaled Ksaibati. Network-level pavement maintenance and rehabilitation planning using genetic algorithm. *Innovative Infrastructure Solutions*, 9, 6 2024.

[12] David Y. Yang. Adaptive risk-based life-cycle management for large-scale structures using deep reinforcement learning and surrogate modeling. *Journal of Engineering Mechanics*, 148, 1 2022.

様式1

橋梁名・所在地・管理者名等　　サンプルの為、削除しています　　個人情報の為、他管理施設の閲覧はできません（空欄で表示されます）

橋梁名	路線名	所在地	起点側	緯度 / 経度		橋梁ID
(フリガナ)	一般国道　号 現道					

管理者名	定期点検実施年月日	路下条件	代替路の有無	自専道or一般道	緊急輸送道路	占用物件(名称)
	2019.9.2		有	一般道	一次	上水道,電力,電話,ガス

部材単位の診断(各部材毎に最も厳しい健全性の診断結果を記入)　　定期点検者

定期点検時に記録					応急措置後に記録		
部材名		判定区分 (Ⅰ～Ⅳ)	変状の種類 (Ⅱ以上の場合に記載)	備考(写真番号、位置等が分かるように記載)	応急措置後の判定区分	応急措置内容	応急措置及び判定実施年月日
上部構造	主桁	Ⅰ					
	横桁	Ⅰ					
	床版	Ⅱ	その他(漏水・遊離石灰)	写真3, 床版00			
下部構造		Ⅱ	その他(その他)	写真4, 竪壁01			
支承部		Ⅰ					
その他		Ⅱ	破断	写真6, 排水ます0101			

道路橋毎の健全性の診断(判定区分Ⅰ～Ⅳ)

定期点検時に記録	
(判定区分)	(所見等)
Ⅱ	床版の損傷は遊離石灰であり、張出床版、歩車道境界、中央分離帯直下の格間に発生しており、橋面防水工不良が原因と推定され、予防保全の観点から補修が必要である。

全景写真(起点側、終点側を記載すること)

架設年次	橋長	幅員
1974年	18m	30.80m

橋梁形式

単純合成鈑桁橋

控え壁式橋台2基

場所打ぐい(深礎を含む)2基

起点

終点

※架設年次が不明の場合は「不明」と記入する。

Figure 2: Excerpt from the publicly released "inspection report" (page 1 of the sample file). The sheet combines inventory data (bridge ID, route, span, year built), four-level component ratings recorded during the periodic inspection (super- and substructure, bearings, deck, etc.), dominant deterioration types, emergency measures, and a reference photo of the structure. These heterogeneous fields are automatically converted to the relational tuple ⟨bridge_id, comp_id, year, state, damage, repair⟩ by the GPT-4o extraction agent described in Section 2. This sample PDF data is cited from Japan Bridge Engineering Center [2].

A Experiments

A.1 Experimental setup

Baselines.

1. **Manual:** transcribed by two licensed bridge inspectors; serves as the *gold reference* for extraction accuracy.

2. **OCR+Regex:** Tesseract 5 followed by hand-written regular expressions, a common low-cost automation strategy in road bureaux.

3. **Stationary Markov:** a five-state, load-agnostic chain fitted by maximum likelihood [10].

4. **4-Rank model:** the Japanese standard "excellent–good–caution–bad" ranking without damage substates (industry default).

Metrics. Information-extraction quality is reported as micro–averaged precision P, recall R, and F_1. For deterioration prediction, we calculate the mean absolute error (MAE) between observed and predicted state probabilities $\Pr(S_t \geq S_4)$ on a rolling leave-one-year-out scheme. Network-level benefit is measured through (i) expected life-cycle cost (LCC) and (ii) the time-averaged probability that a component reaches S_4 or S_5 (severe).

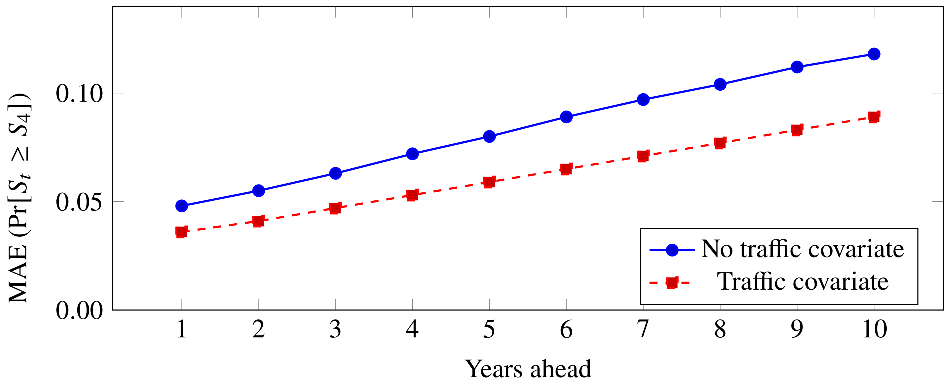

Figure 3: Ten-year forecast error for RC decks with and without the heavy-vehicle covariate.

A.2 Ablation: value of traffic covariate

Moreover, Figure 3 confirms that the traffic covariate yields a consistent 25–30 % error reduction over the full horizon, supporting its inclusion even when census updates are available only every five years.

B Sample Bridge-Inspection Record

To ensure reproducibility we provide, as an online attachment, the official *inspection report* (inspection card) distributed by the Research Institute for Road Structures (RIRS)[3]. Table 4 summarises the structure and semantics of the most relevant items; the page references correspond to the downloadable PDF. While the Ministry's template is nationally standardised, *in practice multiple layout and notation patterns exist* across road authorities and contractors (e.g., optional pages, alternative field names, and different rating scales). The notes below the table detail these variants and how our parser handles them.

Table 4: Logical structure of the public sample *inspection report*.

Page(s)	**Block**	**Form item(s)**	**Description / downstream usage**
1	Header	Bridge ID, name, route, GPS	Primary key for relational merge; location used for GIS plots.
1	Inventory	Span count, length, year built	Condition–age stratification; life-cycle baseline parameters.
2–3	Component table	*Parts ID*, material, *soundness* (rank), cause, repair history	Transcribed into Markov state (R, M) and repair flag u_t.
4–11	Photo log	Annotated defect photographs	Vision encoder validates OCR and supports super-resolution checks.
12	Cost sheet	Itemised repair estimate (JPY)	Calibration of C_{prev} and C_{corr} in Section 2.

Notes on common variants (handled by the parser): (i) some issuers use a **4-level** rank (Excellent–Good–Caution–Bad) instead of 5; we map both to S_1–S_5; (ii) *Parts ID*/*Component* labels and material codes differ slightly by prefecture; a synonym dictionary normalises these; (iii) coordinates may be missing or listed as chainage; we geocode from route and KP when needed; (iv) component tables often **span pages** or include sub-rows per defect; we merge by (bridge, component, year); (v) **handwritten** annotations and rotated scans appear in photo logs; the vision encoder corrects rotation and flags low confidence for review; (vi) cost sheets may be absent; defaults for $C_{\text{prev}}/C_{\text{corr}}$ are then taken from agency unit-price tables.

[2] `https://www.jbec.or.jp/database/data/`

[3] `https://www.jbec.or.jp/en/`

Rationale for inclusion. The document is published under an open licence and represents the canonical format encountered in nationwide inspections. Embedding it as an appendix serves three purposes: (1) it allows readers to verify the OCR + LLM extraction workflow described in Section 2; (2) it illustrates the heterogeneous cues (Japanese text, engineering symbols, photographs) that motivate the hybrid state definition of Section 2; and (3) it provides concrete field labels that are helpful for international readers unfamiliar with Japanese bridge-management terminology.

Parsing the document and handling format variability. The parser first detects page layout (tables vs. key–value blocks) and normalises field names via a controlled vocabulary (*Parts ID/Component*, *Soundness/Rank*, *Cause/Damage*). Rank scales (4- vs. 5-level) are mapped onto S_1–S_5; multi-page component tables are stitched by consistent `bridge_id/comp_id/year`. Handwritten notes and rotated images are handled with rotation correction and confidence-based fallbacks to human-in-the-loop review. The final output is a flat list of `{bridge_id, comp_id, year, state, damage, repair, AADT_hv}` tuples that can be passed directly to the estimation routine in Section 2.

C Interactive Dashboard for Risk-Aware Scheduling

This appendix documents the browser-based dashboard used to explore and communicate maintenance schedules driven by the LLM → Markov pipeline (Section 2). The tool runs entirely on the client side and requires no server components.

C.1 Architecture and scope

The dashboard (React + Recharts + plain CSS) implements the following loop:

1. **Ingest** yearly severe-state probabilities $q_{i,t} = \Pr(S_t \geq S_4)$ for each component i and traffic scenario σ (e.g., mean, peak).
2. **Score** each candidate "repair-next-year" action by its one-step marginal benefits $(\Delta\mathrm{LCC}_{i,t},\ \Delta\mathrm{Risk}_{i,t})$.
3. **Select** a set of actions per year under annual *budget* and *crew* constraints. The default uses a greedy surrogate; a CP–SAT back-end can replace it for optimal selections without UI changes.
4. **Visualise** a Gantt (bridge × year), annual KPIs, the network risk trajectory $\bar{q}_t = \frac{1}{N}\sum_i q_{i,t}$, and an *audit table* explaining each selected action.

C.2 Data interface

The dashboard accepts a single JSON with the schema:

```
{
  "years": [2025, 2026, ...],
  "scenarios": ["mean", "peak"],
  "comps": [
    {
      "id": "B1-C1", "bridge": "B1", "comp": "Deck",
      "material": "RC", "damage": "crack", "age": 25,
      "costPrev": 250000, "costCorr": 900000, "costUser": 40000,
      "crew": 2,
      "q": { "mean": [0.05, 0.07, ...], "peak": [0.06, 0.09, ...] }
    }
  ]
}
```

Here $q[\sigma][t]$ is the Markov forecast for component i in year t under scenario σ; costs are in JPY; `crew` denotes the crew-days (or standardised crew units) required for a repair.

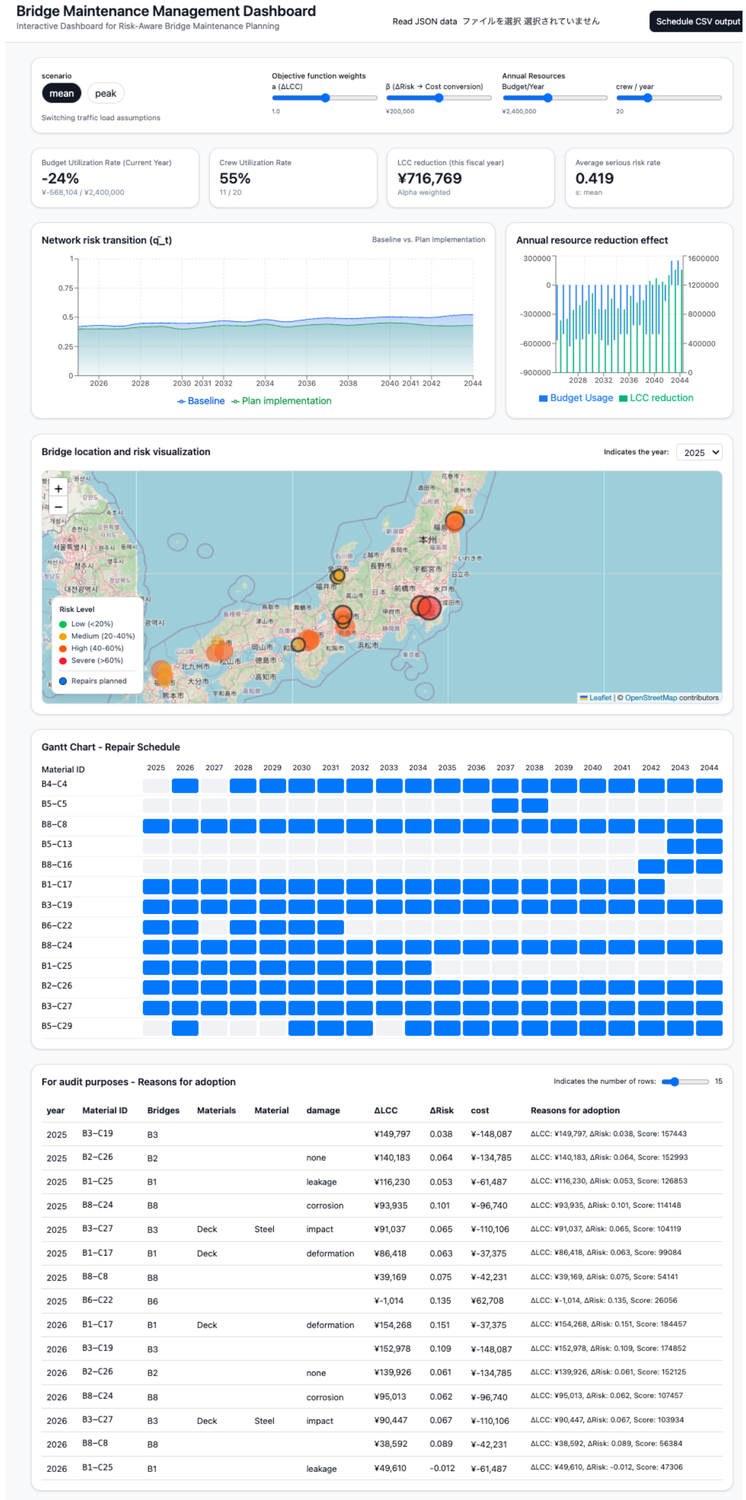

Figure 4: Interactive dashboard for risk-aware bridge maintenance planning. Top: scenario selector (`mean`/`peak`), objective weights (α, β), and annual resource caps (budget, crew). KPI cards summarise current-year budget/crew utilisation, α-weighted LCC reduction, and average severe-state risk $\bar{q}_t$. Middle: network risk transition $\bar{q}_t$ (baseline vs. plan) and annual resource effect (budget used and LCC reduction). Map: bridge locations with bubble colour/size encoding current severe-risk level and halos marking planned repairs. Bottom: Gantt chart of scheduled repairs by component and year, and an audit table listing each selected action with ΔLCC, ΔRisk, cost, and rationale; schedules respect yearly budget/crew constraints and can be exported as CSV. ©OpenStreetMap contributors.

C.3 Marginal benefits and scoring

Given discount rate ρ and a next-year repair at (i, t), the dashboard uses a *myopic* one-step approximation (replaceable by multi-year LCC if provided):

$$\Delta\mathrm{Risk}_{i,t} = q^{\mathrm{no}}_{i,t+1} - q^{\mathrm{rep}}_{i,t+1}, \qquad \Delta\mathrm{LCC}_{i,t} = \frac{C_{\mathrm{corr}}\,(p^{\mathrm{no}}_{i,t+1} - p^{\mathrm{rep}}_{i,t+1}) + C_{\mathrm{user}}\,\Delta\mathrm{Risk}_{i,t}}{1+\rho} - C_{\mathrm{prev}}.$$

Here q^{no} is the forecast without intervention, q^{rep} is the post-repair risk (typically reset near a healthy baseline), p is a proxy for $\Pr(S_t \geq S_5)$ derived from q, and C_{prev}, C_{corr}, C_{user} are unit costs. A scalar objective weight (α, β) forms the score $J_{i,t} = \alpha\,\Delta\mathrm{LCC}_{i,t} + \beta\,\Delta\mathrm{Risk}_{i,t}$.

C.4 Yearly selection under constraints

For each year Y the selector sorts candidates by $J_{i,Y}$ and admits repairs while respecting (i) annual budget and (ii) annual crew caps. This greedy step is $O(N \log N)$ per year and serves as a drop-in *surrogate* for the MILP/CP–SAT formulation:

$$\begin{aligned} \max_{y_{i,t}\in\{0,1\}} \quad & \sum_t \sum_i J_{i,t}\, y_{i,t} \\ \text{s.t.} \quad & \sum_i C_{\mathrm{prev},i}\, y_{i,t} \leq B_t, \quad \sum_i \mathrm{crew}_i\, y_{i,t} \leq K_t \quad \forall t, \\ & \text{(optional) mutual-exclusion, work-zone, or precedence constraints.} \end{aligned}$$

A production deployment can replace the greedy selector with OR-Tools CP–SAT without changing the UI; the `score` and `constraints` interfaces are already isolated in the code.

C.5 UI elements and KPIs

Controls: scenario toggle (σ), objective weights (α, β), annual budget B_t, annual crew K_t, discount ρ.
Charts and tables:

- *Network risk trajectory*: area chart of $(\bar{q}_t)$ baseline vs. with-plan.
- *Annual resources & savings*: budget used, LCC saved (bars).
- *Gantt*: components by year with scheduled repairs highlighted.
- *Audit table*: for the current year, shows $(\Delta\mathrm{LCC}_{i,t}, \Delta\mathrm{Risk}_{i,t})$, score, cost, crew.

CSV export of the schedule is provided for downstream CMMS/ERP ingestion.

C.6 Rolling operation and robustness

At the start of each fiscal year the operator:

1. Uploads updated Markov forecasts $q_{i,t}$ (re-estimated from new inspections).
2. Reviews (α, β) and resource caps (B_t, K_t) for the year.
3. Re-optimises and exports the refreshed schedule.

To hedge traffic uncertainty the UI supports multiple scenarios (e.g., mean vs. peak) and a weighted objective; risk-averse users can map β to a CVaR-like penalty or feed scenario-specific $\Delta\widehat{\mathrm{LCC}}$ from offline analysis.

C.7 Integration points

- **Markov engine** $\rightarrow$ `q` arrays: insert the generated $\Pr(S_t \geq S_4)$ per component, year, and scenario into the JSON (Appendix C.2).
- **Exact optimiser** (CP–SAT/MILP): substitute the greedy `selectRepairs()` with a solver call that returns the same `y{year}{id}` schedule map.

C.8 Reproducibility and limits

The demo ships with a synthetic portfolio generator for offline testing. On a typical laptop it renders $N \approx 10^3$ components interactively. For larger portfolios, replace the greedy step with CP–SAT and precompute scores on the server; the UI remains unchanged.

In short, the dashboard operationalises the paper's pipeline by turning Markov forecasts into auditable, resource-feasible schedules and by surfacing—via Gantt, KPI cards, and an audit table—the *why* of each selected intervention.