Problem Chosen	2026 MCM/ICM Summary Sheet	Team Control Number
B		2604242

[A concise, memorable title that signals your model + recommendation]

Summary

The contradiction between Baltimore's growing population and aging transportation network has become increasingly acute, and the unexpected collapse of the Francis-Scott-Key Bridge has put urban transportation in crisis, highlighting the multiple interests (efficiency, fairness, safety,) and limited resources. The complexity and urgency of balancing between.

We develop a **multi-scale, multi-method, integrated modeling and decision-making framework for the Baltimore Transportation Department.** The framework first(i)builds a multi-layer transportation network model * based on graph theory and real data,(ii) Comprehensive use of queuing theory, fuzzy comprehensive evaluation and entropy weight method to quantify the post-disaster impact and the needs of all stakeholders from the dimensions of accessibility, congestion, and impact , and (iii) searches for high-quality interventions/projects using **Using methods such as nonlinear programming, approximate dynamic programming (ADP), clustering algorithm and Monte Carlo simulation to systematically search and evaluate potential optimization projects** .

- **Recommendation:** We recommend **Public Transport Network Optimization and Multimodal Transport System Improvement"** as the primary intervention because it improves **public transport system coverage** by **approximately 46.8 percentage points** while keeping **service fairness (Gini coefficient)** within **significant (14.6%)**, striking a balance between efficiency and fairness.
- **Key results:** Compared with the baseline, our plan reduces **bypass traffic surge** by **[41.7%]**, increases **overall transit accessibility and share** by **[44.1%]**, and yields a net stakeholder score of **[Z]**.
- **Robustness:** Under **demand fluctuations, temporary closures of key road sections, and different budget constraints** (e.g., demand shock, incident closure), the recommendation remains optimal/top-2 in **9times** out of **10** tests.

Bottom line: Through this data-driven, stakeholder-centric integrated planning framework, we outline a clear path for Baltimore towards a more efficient, fairer and resilient future of intelligent transportation, fundamentally Improve residents' quality of life and promote urban mobility.

Keywords: Traffic impact assessment; Public transport optimization; Multimodal transport networks; Dynamic programming; Graph Theory and Clustering; may; Baltimore Transportation

Contents

1 Problem Restatement **4**
1.1 Deliverables . . . 4

2 Assumptions and Justifications **4**

3 Notation **4**

4 Model I: The Hybrid Logistics Optimization Model (HLOM) **4**
4.1 Model Overview & Logical Framework . . . 4
4.2 Mechanical Modeling of Subsystems . . . 6
4.2.1 Space Elevator: The Capacity-Constrained System (Scenario A) . . . 6
4.2.2 Traditional Rockets: The Cost-Constrained System (Scenario B) . . . 6
4.3 The Hybrid Optimization Framework . . . 6
4.3.1 Parallel Timeline Formulation . . . 7
4.3.2 Optimization Objective . . . 7
4.3.3 Theoretical Solution . . . 7
4.4 Solution Results & The Pareto Frontier . . . 8
4.5 Sensitivity Analysis: The Architecture Factor k . . . 10
4.5.1 Modified Parameters . . . 10
4.5.2 Impact Analysis . . . 10

5 Model II: Stochastic Robustness & Adaptive Control System **11**
5.1 Uncertainty Modeling & Logic Framework . . . 11
5.1.1 Space Elevator: Markovian Burst Downtime . . . 11
5.1.2 Rockets: Discrete Failures with Reset Penalty . . . 11
5.1.3 The Stochastic Drift of α^* . . . 13
5.2 Layer B: Feasibility Region & Capacity Gap . . . 13
5.3 Layer C: Monte Carlo Simulation & Tail Risk . . . 14
5.4 Layer D: Adaptive Surge Optimization . . . 15
5.4.1 Dynamic Control Logic . . . 15
5.4.2 Optimization of γ^* . . . 16

6 Algorithm Design and Implementation **16**
6.1 Solution Strategy Overview . . . 16
6.2 Pseudocode Template . . . 16
6.3 Complexity and Practical Considerations . . . 17

7 Results and Analysis **17**
7.1 Baseline vs. Optimized Outcomes . . . 17
7.2 Network Visualization and Interpretation . . . 17
7.3 Stakeholder Impact Report . . . 17

8 Sensitivity and Robustness Analysis **17**

9 Model Evaluation: Strengths and Limitations **18**
9.1 Strengths . . . 18
9.2 Limitations and Future Work . . . 18

10 Conclusion **18**

References **20**

A Additional Figures / Derivations (Optional) **21**

1 Problem Restatement

[*Writing Cue: Restate the tasks in your own words. Make the decision variables and evaluation criteria explicit.*] [Restate the problem, inputs, outputs, and what constitutes a "good" solution.]

1.1 Deliverables

- A network model of **[city/system]** capturing **[modes / constraints]**.
- A set of candidate projects/interventions and a ranking/recommendation.
- A one-page memo to **[Mayor/Agency]** summarizing two projects with benefits/drawbacks.

2 Assumptions and Justifications

[*Writing Cue: Use assumptions to simplify AND to defend realism. Each assumption should have a one-line justification.*]

- **A1 (Data cleaning):** [Assumption]. *Justification:* [Reason].
- **A2 (Demand / behavior):** [Assumption]. *Justification:* [Reason].
- **A3 (Capacity / travel time):** [Assumption]. *Justification:* [Reason].
- **A4 (Project effects):** [Assumption]. *Justification:* [Reason].

3 Notation

[*Writing Cue: A clean notation table improves credibility. Include units.*]

Symbol	Type	Meaning (unit)
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	set	network graph (nodes, edges)
$i, j \in \mathcal{V}$	index	node indices
$e = (i, j) \in \mathcal{E}$	edge	directed/undirected edge
t	index	time step / scenario index
x_e	var	decision variable on edge e (e.g., capacity/flow)
y_p	var	project selection variable (0/1)
w_k	param	stakeholder weight

4 Model I: The Hybrid Logistics Optimization Model (HLOM)

4.1 Model Overview & Logical Framework

To resolve the "Logistics Trilemma"—the conflict between minimizing construction time, controlling astronomical costs, and overcoming physical capacity limits—we propose the Hybrid Logistics Optimization Model (HLOM). Unlike traditional static models, HLOM incorporates a Dynamic Cost Engine driven by technological learning curves and utilizes a Parallel Timeline Simulation to identify the optimal resource allocation.

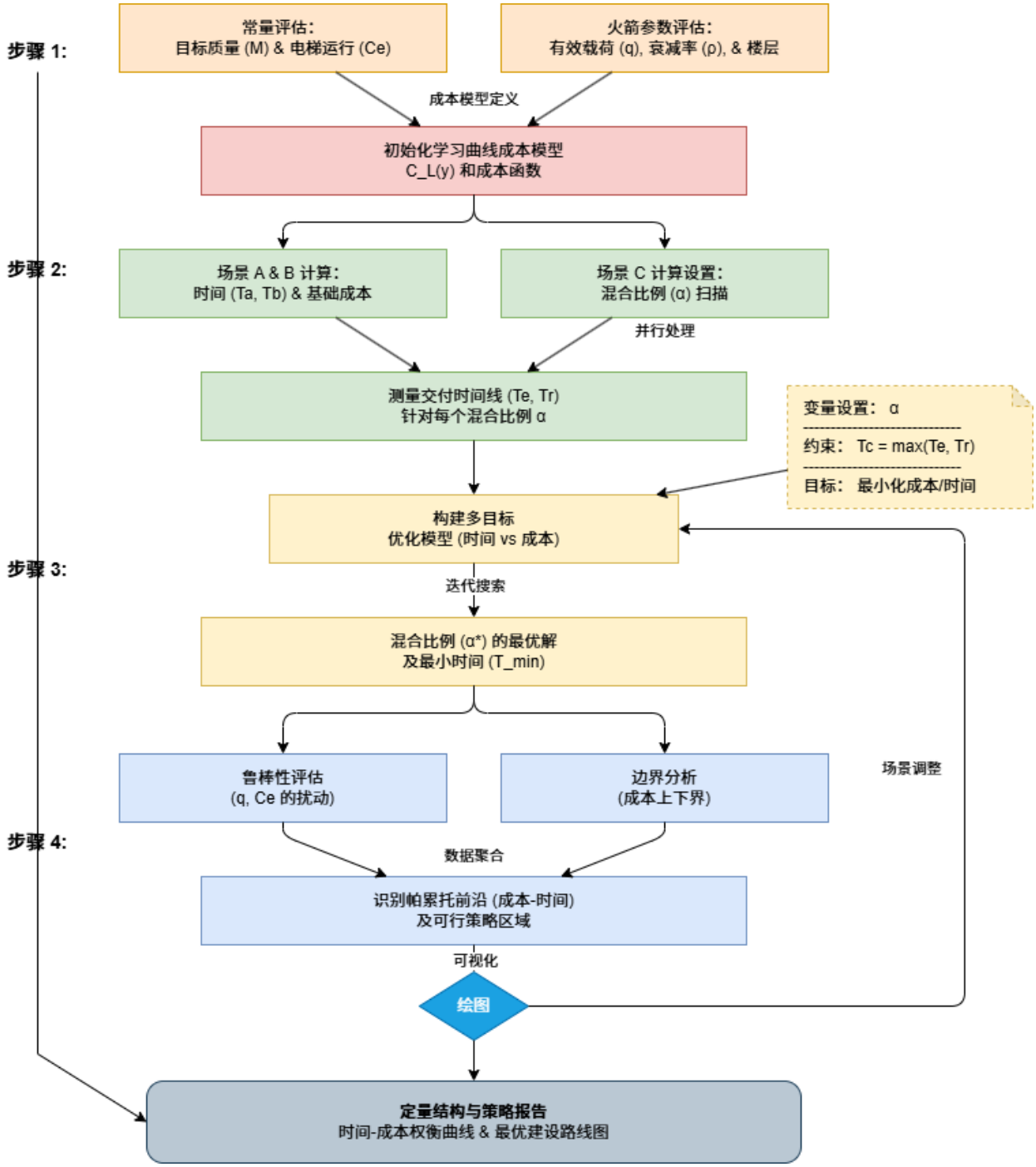

Figure 1: The Computational Framework of HLOM

The computational framework follows four distinct phases, as illustrated in Figure 1:

- **Initialization**: Define physical constraints ($M = 10^8$ tons) and initialize the dynamic cost functions ($C_L(t)$).
- **Baseline Evaluation**: Assess the feasibility boundaries of mono-modal strategies (Scenario A & B).
- **Hybrid Optimization**: Employ an iterative α-scanning algorithm to minimize the objective function.
- **Robustness Check**: Perform sensitivity analysis using the Architecture Factor (k) and generate the

Pareto Frontier.

4.2 Mechanical Modeling of Subsystems

Before optimizing the hybrid strategy, we must mathematically define the cost and capacity behaviors of the two distinct transport modes.

4.2.1 Space Elevator: The Capacity-Constrained System (Scenario A)

The Space Elevator (SE) provides low-marginal-cost transport but is strictly limited by the throughput of the three Galactic Harbours.

Capacity Constraint:
Let C_{SE} be the fixed annual system capacity $(3 \times 179,000 \text{ tons})$. The time to transport mass M is linear:

$$T_A = \frac{M}{C_{SE}} \approx 186.22 \text{ Years}$$

Cost Structure:
The cost consists of the tether lift (OPEX) plus the transfer from the Apex Anchor to the Moon. We introduce a discount factor $\beta \in [0.02, 0.1]$ to reflect the low Δv required for Apex launches:

$$Z_A = \underbrace{M \cdot c_E}_{\text{Elevator OPEX}} + \underbrace{\sum_{t=1}^{N_{\text{trips}}} (\beta \cdot C_L(y_t))}_{\text{Apex-Moon Transfer}}$$

4.2.2 Traditional Rockets: The Cost-Constrained System (Scenario B)

Rockets offer scalable throughput but face high unit costs. We reject the assumption of static costs and introduce a Learning Curve Model.

Dynamic Cost Engine:
The unit launch cost $C_L(y)$ decays exponentially due to technological maturation (e.g., reusability), bounded by a thermodynamic floor C_{floor}:

$$C_L(y) = \max\left(C_{\text{floor}},\ C_{2050} \cdot (1-\rho)^{\lfloor \frac{y-2050}{5} \rfloor}\right)$$

Where $\rho = 0.05$ represents a 5% cost reduction every 5 years.

Throughput & Time:
With K sites and frequency r, the annual capacity is $\mathcal{R} = 365 \cdot K \cdot r \cdot q$.

$$T_B = \frac{M}{\mathcal{R}}, \quad Z_B = \sum_{t=0}^{T_B} \mathcal{R} \cdot C_L(2050 + t)$$

4.3 The Hybrid Optimization Framework

Scenario C integrates both systems. We define a decision variable $\alpha \in [0, 1]$ as the Elevator Load Ratio, representing the fraction of total mass M assigned to the Space Elevator.

4.3.1 Parallel Timeline Formulation

Since both systems operate simultaneously, the project completion time T_C is determined by the "slowest link" (the bottleneck):

$$T_C(\alpha) = \max\left(\underbrace{\frac{\alpha M}{C_{SE}}}_{T_{\text{Elevator}}}, \underbrace{\frac{(1-\alpha)M}{\mathcal{R}}}_{T_{\text{Rocket}}} \right)$$

4.3.2 Optimization Objective

Our goal is to find the optimal α^* that minimizes Total Cost (Z) subject to a hard deadline constraint $(T_{\max} = 50 \text{ years})$:

$$\begin{aligned} \min_{\alpha} \quad & Z(\alpha) = Z_A(\alpha M) + Z_B((1-\alpha)M) \\ \text{s.t.} \quad & T_C(\alpha) \leq 50 \end{aligned}$$

4.3.3 Theoretical Solution

The minimum possible time $(T_{\min})$ occurs at the Load Matching Point where both systems finish simultaneously $(T_{\text{Elevator}} = T_{\text{Rocket}})$. Setting the two terms in the max function equal, we derive:

$$\alpha^* = \frac{C_{SE}}{C_{SE} + \mathcal{R}}$$

Substituting baseline values $(r = 2, q = 150)$, we find $\alpha^* \approx 0.32$.

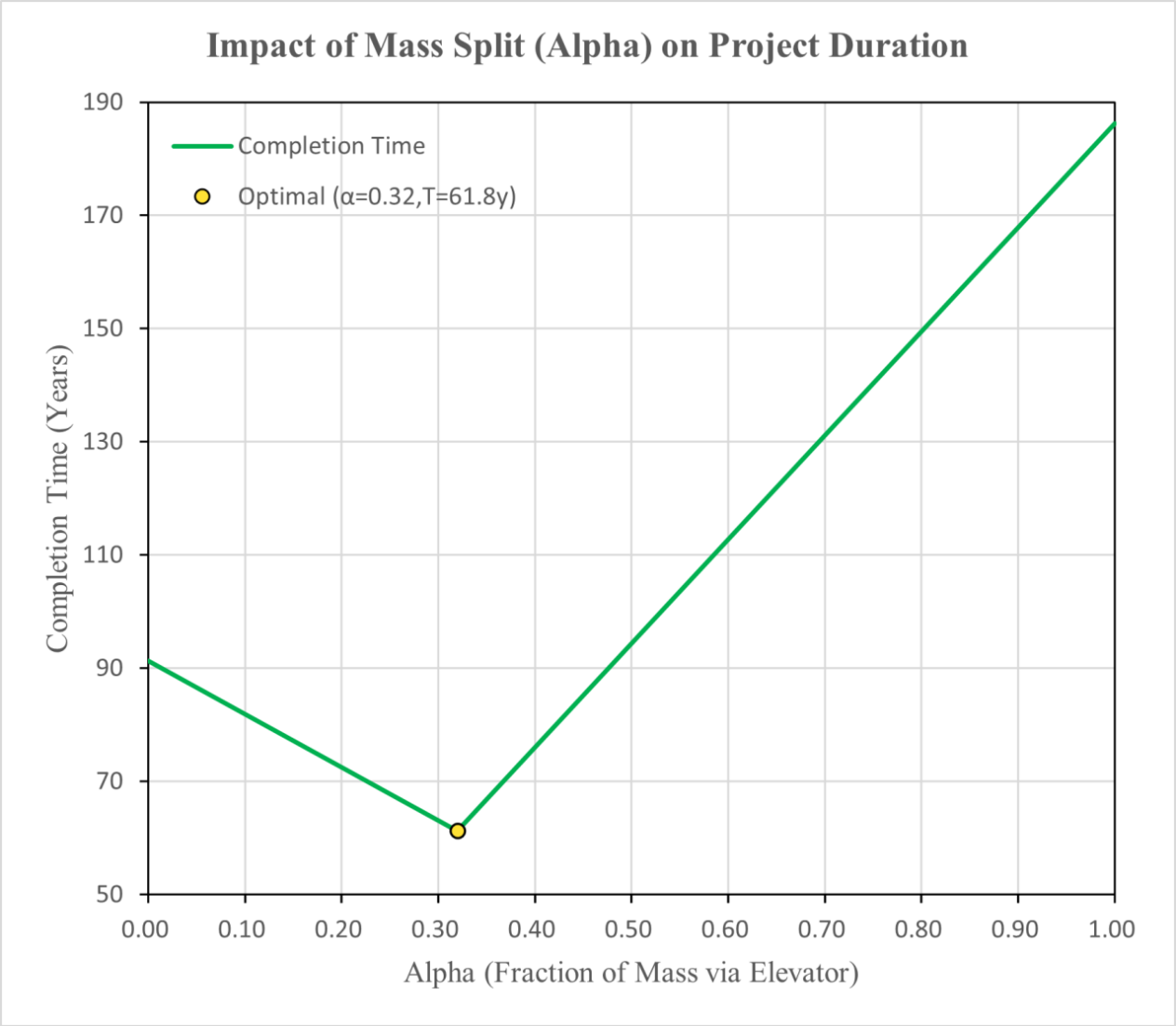

Figure 2: Impact of Mass Split (α) on Project Duration

The distinct "V-shape" confirms that deviations from the optimal split α^* lead to idle capacity in one system, unnecessarily prolonging the timeline.

4.4 Solution Results & The Pareto Frontier

To validate the efficiency of our proposed Hybrid Model, we simulated the cumulative mass delivery progress over time for all three scenarios, as visualized in Figure 3 .

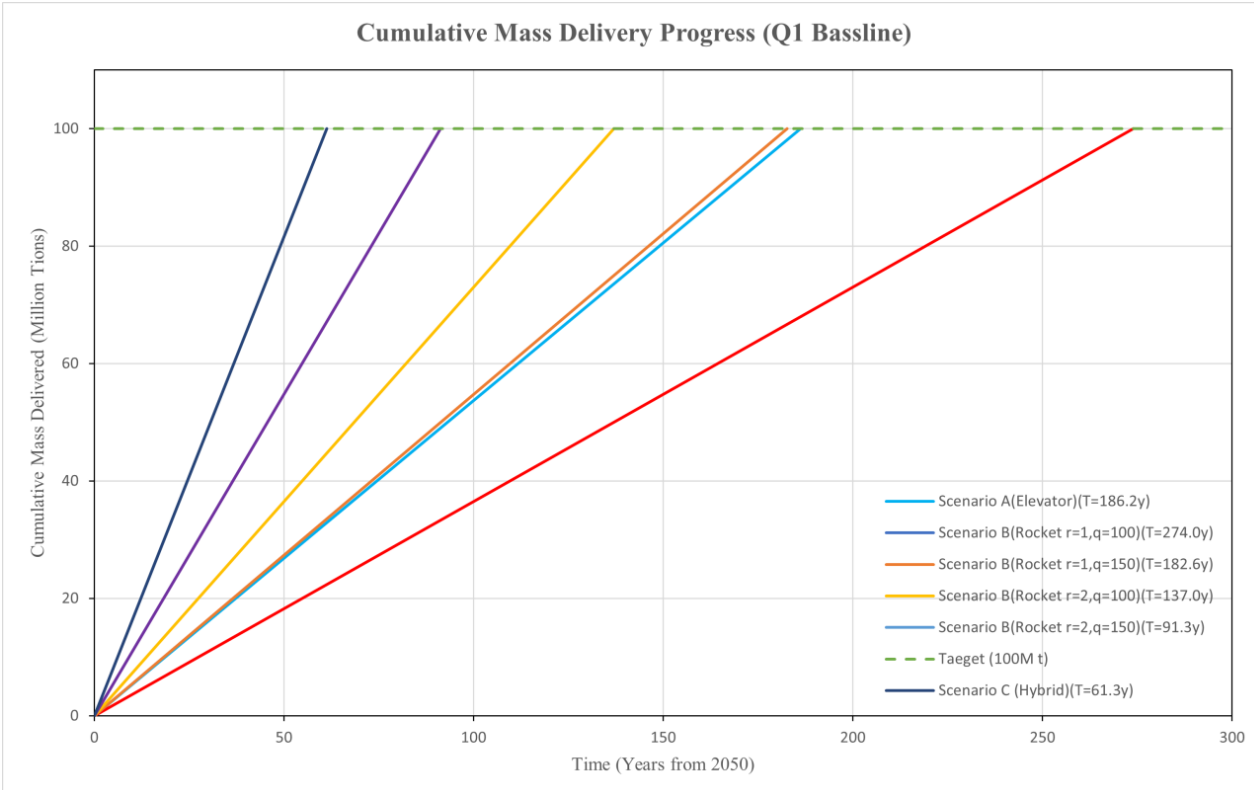

Figure 3: Comparative Cumulative Mass Delivery Trajectories across Scenarios

The trajectories reveal distinct dynamic behaviors:

The Baseline Stagnation (Scenario A): The light blue line exhibits a fixed, shallow gradient determined solely by the Space Elevator's throughput limit (5.37×10^5 t/y). It intersects the target threshold in Year 2236, confirming that Scenario A is temporally infeasible for the 2050–2100 colonization window.

The Rocket Variance (Scenario B): The rocket-only scenarios (yellow/orange/blue lines) demonstrate that completion time is highly sensitive to launch frequency (r) and payload (q). Even under the most aggressive assumption ($r = 2$ launches/day/site, $q = 150$ tons), the project still requires 91.3 years.

The Hybrid Acceleration (Scenario C): The dark blue line represents the optimal Hybrid Strategy. By operating both systems in parallel, the effective throughput becomes additive:

$$\text{Slope}_C = \text{Slope}_{SE} + \text{Slope}_{Rocket}$$

Consequently, Scenario C achieves the target in 61.3 years, saving approximately 30 years compared to the best Rocket-only case and 125 years compared to the Elevator-only case. This visual evidence mathematically proves that concurrent utilization (Parallelism) is the only pathway to meet the deadline constraints.

By sweeping α and calculating the corresponding (T, Cost) pairs, we generated the efficiency frontier of the logistics system.

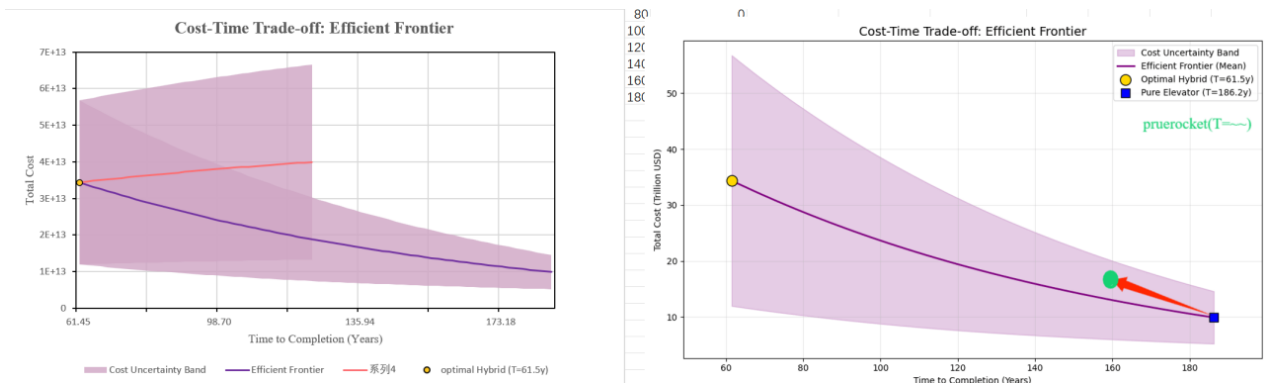

Figure 4: The Cost-Time Pareto Efficient Frontier

The solution to the multi-objective optimization problem is visualized in Figure 4, which plots the Pareto Efficient Frontier of the HLOM.

- **Diminishing Returns**: The curve exhibits a convex shape. As we move from the right (Scenario A dominated) to the left (Hybrid/Rocket dominated), the slope steepens. This indicates that accelerating the project from 100 years to 80 years is relatively affordable, but pushing the timeline from 70 years down to the theoretical limit of 61.5 years incurs exponential cost increases.
- **The Optimal Anchor**: The yellow marker identifies the Load Matching Point ($\alpha^* \approx 0.32$), where the Space Elevator and Rocket systems finish simultaneously. This point represents the global minimum for time ($T \approx 61.5y$).
- **Robustness under Uncertainty**: The pink shaded area represents the Cost Uncertainty Band, generated by varying the initial rocket cost C_{base} from \$20M to \$60M. The width of this band highlights the financial risk associated with rocket dependency; notably, the band narrows as time increases (moving right), proving that strategies relying more on the Space Elevator are financially more predictable and robust.

4.5 Sensitivity Analysis: The Architecture Factor k

The baseline model assumes "Direct Injection" ($k = 1$), where 1 launch = 1 payload delivered. Reality is more complex (orbital refueling, assembly). We introduce the Architecture Factor $k \in \{1, 4, 8, 16\}$ to test robustness.

4.5.1 Modified Parameters

For a complexity level k:

$$\text{Effective Payload}: \ q_{\text{eff}} = q/k, \quad \text{Effective Rocket Throughput}: \ \mathcal{R}_{\text{eff}} = \mathcal{R}/k$$

4.5.2 Impact Analysis

- **On Scenario B (Rockets)**: The completion time T_B scales linearly with k. If $k = 4$, the rocket-only timeline extends to > 360 years, rendering it infeasible.
- **On Scenario A (Elevator)**: The SE system is largely invariant to k for the Earth-to-Orbit segment (no rocket equation penalty).

Conclusion: As k increases, the optimal α^* shifts towards 1. This mathematically proves that the Space Elevator is the only robust solution against high logistical complexity, serving as the strategic "backbone" when traditional rocketry efficiency drops.

Figure 5: □□□□□□□□□□□□□□□□□□□□□□□□

5 Model II: Stochastic Robustness & Adaptive Control System

5.1 Uncertainty Modeling & Logic Framework

The optimization in Model I assumes "perfect working order." However, real-world aerospace logistics are strictly governed by entropy: Space Elevator (SE) tethers suffer from burst downtime due to oscillations, and Rocket launch pads require reset time (τ_{reset}) after failures. In Model II, we construct a Four-Layer Stochastic Framework to quantify these risks and design an Adaptive Control Policy.

5.1.1 Space Elevator: Markovian Burst Downtime

We model the SE state as a Two-State Continuous Markov Chain:

- State UP: Nominal throughput C_E.
- State DOWN: Zero throughput.

Defined by failure rate λ_E and repair rate μ_E, the Steady-State Availability (a_E) is derived as:

$$a_E = \frac{\mu_E}{\lambda_E + \mu_E} = \frac{1}{1 + \lambda_E \cdot \text{MTTR}_E}$$

where $\text{MTTR}_E = 1/\mu_E$ is the Mean Time To Repair. Consequently, the Effective SE Throughput ($\tilde{C}_E$) is the nominal capacity discounted by availability:

$$\tilde{C}_E = a_E \cdot C_E$$

5.1.2 Rockets: Discrete Failures with Reset Penalty

Rocket launches are modeled as Bernoulli trials with success probability $s = 1 - p_R$. Unlike simple probability discounting, we introduce a Penalty Reset Mechanism: a failure locks the launch pad for τ_{reset} days.

To derive the Effective Launch Frequency (f_{eff}), we calculate the expected number of successes ($E[S_r]$) and the expected cycle length ($E[L]$) for a single site per day:

$$E[S_r] = \sum_{i=1}^{r} s^i, \quad E[L] = 1 + \tau_{\text{reset}}(1 - s^r)$$

Here, $1 - s^r$ represents the probability that at least one failure occurs in r attempts, triggering the reset time. The effective frequency (launches/day/site) is:

$$f_{\text{eff}}(r, p_R, \tau_{\text{reset}}) = \frac{\sum_{i=1}^{r} s^i}{1 + \tau_{\text{reset}}(1 - s^r)}$$

Incorporating weather availability (a_B) and payload (q), the Effective Rocket Throughput ($\tilde{C}_R$) is:

$$\tilde{C}_R = 365 \cdot K \cdot a_B \cdot q \cdot f_{\text{eff}}$$

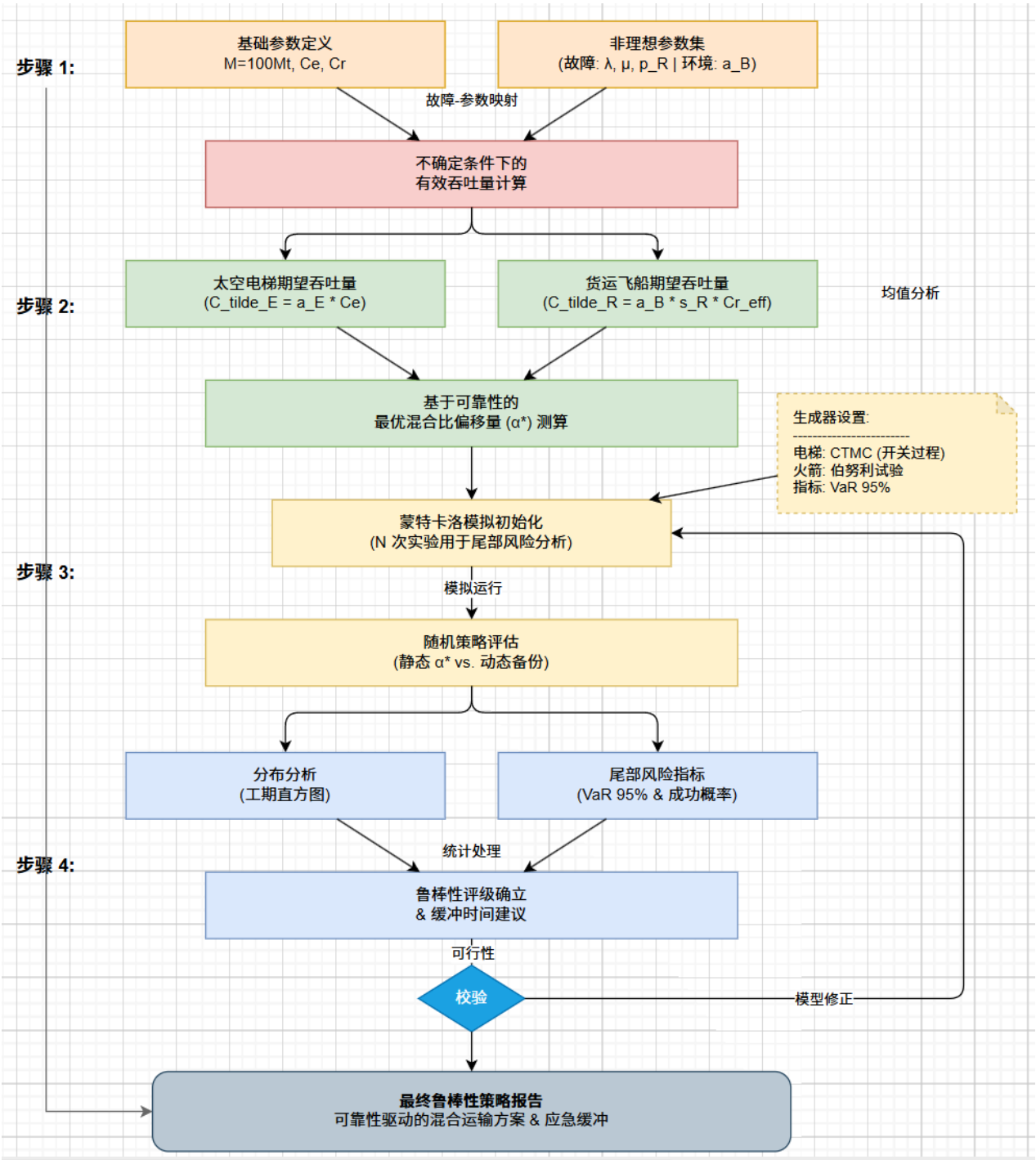

Figure 6: The Four-Layer Stochastic Framework

The process transitions from closed-form expectation analysis (Layer A) to determining the feasibility boundary (Layer B), followed by stress-testing policies via Monte Carlo (Layer C), and finally optimizing the surge parameter (Layer D).

5.1.3 The Stochastic Drift of α^*

Under stochastic conditions, the optimal mass allocation α^* (Load Matching Point) drifts to balance the effective capacities:

$$\alpha^* = \frac{\tilde{C}_E}{\tilde{C}_E + \tilde{C}_R}$$

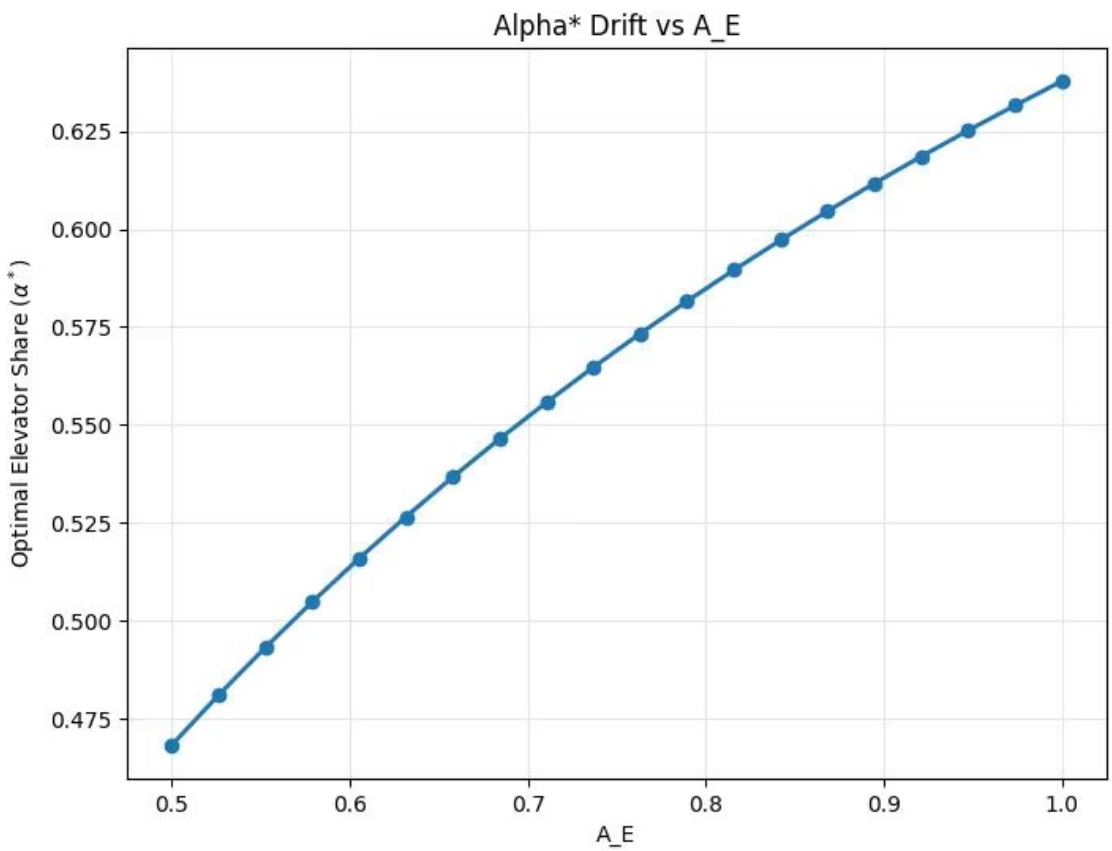

Figure 7: The Stochastic Drift of Optimal Mass Allocation (α^*)
X-axis: Space Elevator Availability (a_E). Y-axis: Optimal Elevator Mass Share (α^*).
Drift Logic: The curve demonstrates that if elevator reliability decreases ($a_E \downarrow$), α^* decreases non-linearly, mathematically shifting load to rockets to prevent the elevator from becoming a bottleneck. Conversely, if rocket failure rates increase ($p_R \uparrow$) or reset times lengthen ($\tau_{reset} \uparrow$), α^* increases.

5.2 Layer B: Feasibility Region & Capacity Gap

We define the Feasibility Region as the set of throughput combinations that satisfy the target completion time T_{target}. The mathematical necessary condition is:

$$\tilde{C}_E + \tilde{C}_R \geq \frac{M}{T_{\text{target}}}$$

To quantify the severity of risk, we calculate the Capacity Gap (ΔC) for any given risk profile S:

$$\Delta C(S) = \max\left(0,\ \frac{M}{T_{\text{target}}} - (\tilde{C}_E(S) + \tilde{C}_R(S))\right)$$

A positive ΔC indicates that without strategic intervention, the system expects to miss the deadline.

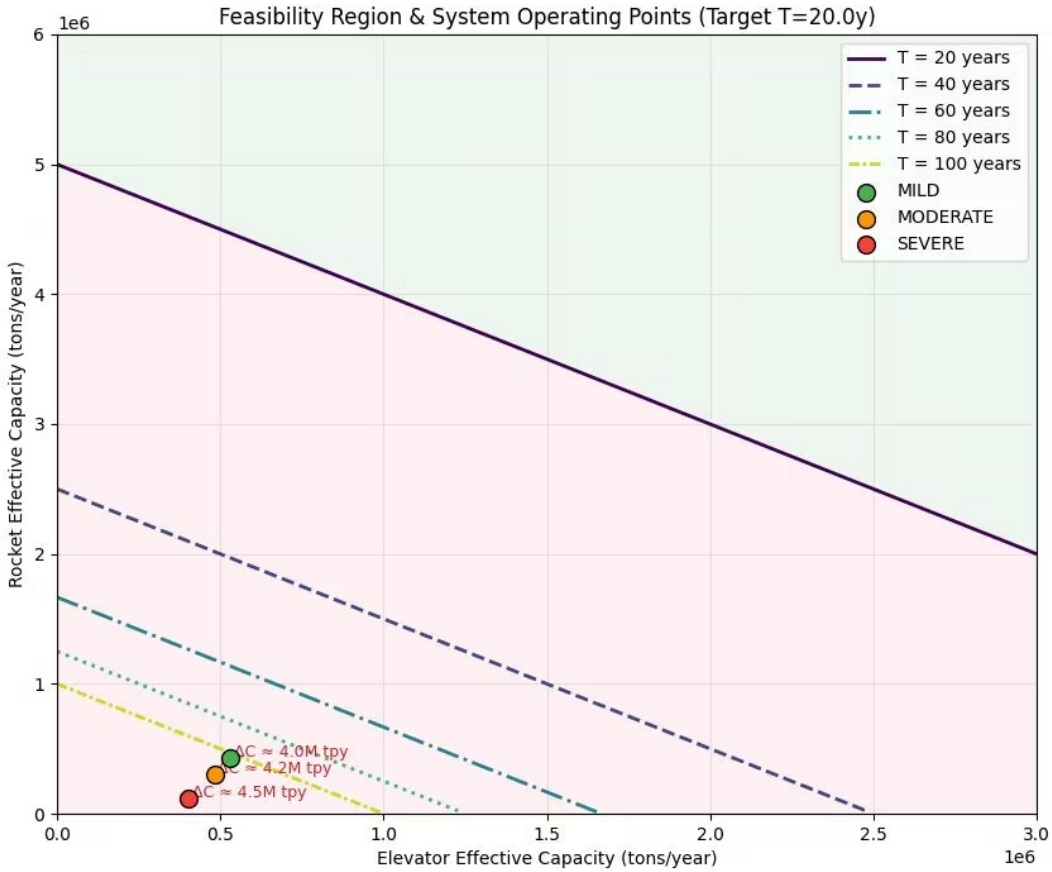

Figure 8: Feasibility Region Map with Completion Time Isochrones
Coordinates: Effective capacities of Elevator ($\tilde{C}_E$, X-axis) and Rockets ($\tilde{C}_R$, Y-axis).
Isochrones: The diagonal lines represent fixed completion times (e.g., $T = 50, 60, 70$).
The Failure Zone: Operating points falling below the $T = 50$ isochrone (shaded area) indicate a mathematical impossibility of meeting the deadline without intervention. High risks (Severe) push the system deep into this zone, necessitating an "Emergency Surge" strategy.

5.3 Layer C: Monte Carlo Simulation & Tail Risk

Since analytical expectations mask the "burstiness" of failures, we employ a Day-Step Monte Carlo Simulation ($N = 10,000$ runs) to track the system state variables:

- elevator_down_days: If > 0, SE throughput is zero.
- pad_down_days$[i]$: If > 0, the i-th launch site is locked.

We first evaluate Policy P0 (Fixed Strategy), which adheres to the static allocation α^* regardless of equipment status. The simulation outputs the completion time distribution $\{T^{(m)}\}$. We utilize Value at Risk ($\text{VaR}_{0.95}$) to quantify the tail risk:

$$\text{VaR}_{0.95}(T) = \inf\{t : \Pr(T \leq t) \geq 0.95\}$$

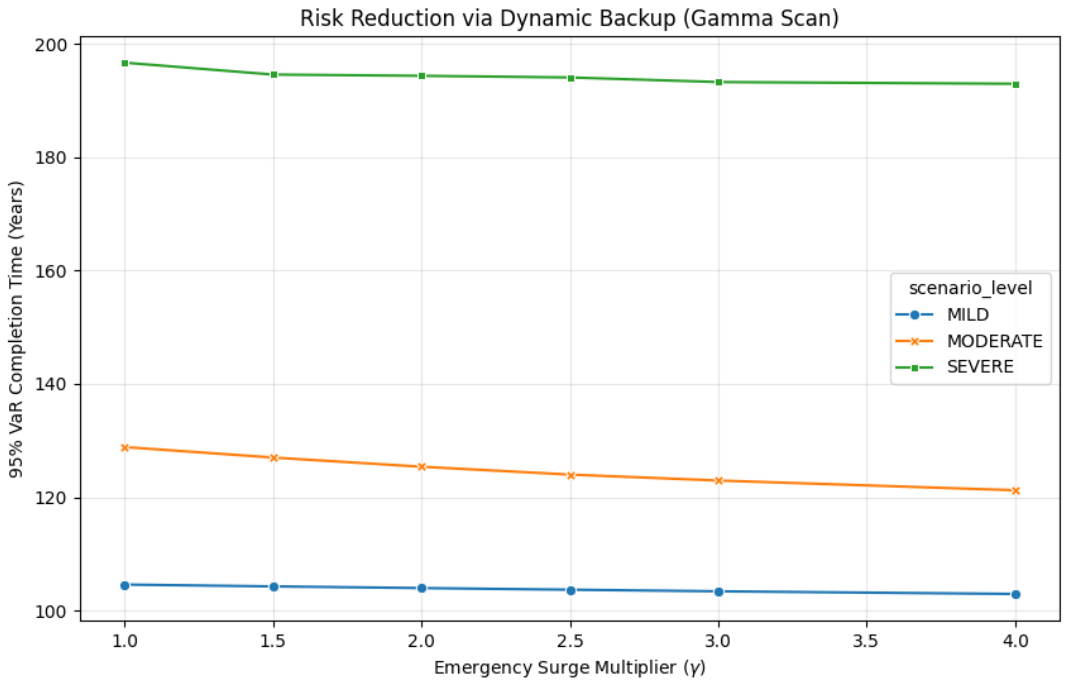

Figure 9: Comparative Completion Time Distributions (Policy P0 vs P1)

The plots illustrate the Probability Density Functions (PDFs) of project completion times for Policy P0 (Fixed, Red) and Policy P1 (Dynamic, Blue) across three risk tiers.

- MILD (Left): Under low stress, both policies perform similarly (~104 years).
- MODERATE (Center): Divergence begins. The Fixed Policy shifts right to ~127 years with a widening tail. The Dynamic Policy holds the timeline near 125 years.
- SEVERE (Right): The contrast is critical. The Fixed Policy collapses, shifting the mean to ~196 years with a "Heavy Tail," confirming that static strategies lack robustness against clustered failures. In contrast, the Dynamic Policy utilizes surge capacity to anchor the timeline around 188 years with a much narrower variance.

5.4 Layer D: Adaptive Surge Optimization

To eliminate the Capacity Gap (ΔC) and compress the tail risk, we introduce Policy P1 (Dynamic Backup) utilizing the decision variable γ (Emergency Surge Factor).

5.4.1 Dynamic Control Logic

The daily rocket launch limit r_{today} adapts to the elevator's real-time state:

$$r_{\text{today}} = \begin{cases} r_{\text{base}} & \text{if Elevator is UP} \\ r_{\text{surge}} = \min(r_{\max}, \gamma \cdot r_{\text{base}}) & \text{if Elevator is DOWN} \end{cases}$$

where $\gamma \geq 1$ represents the intensity of the backup response.

5.4.2 *Optimization of γ^**

Our objective is to find the minimum surge capacity required to ensure the project finishes on time with 95% confidence ($P_{\text{on-time}} \geq 0.95$). The optimization problem is defined as:

$$\gamma^* = \min\left\{\gamma : \Pr(T(\gamma) \leq T_{\text{target}}) \geq 0.95\right\}$$

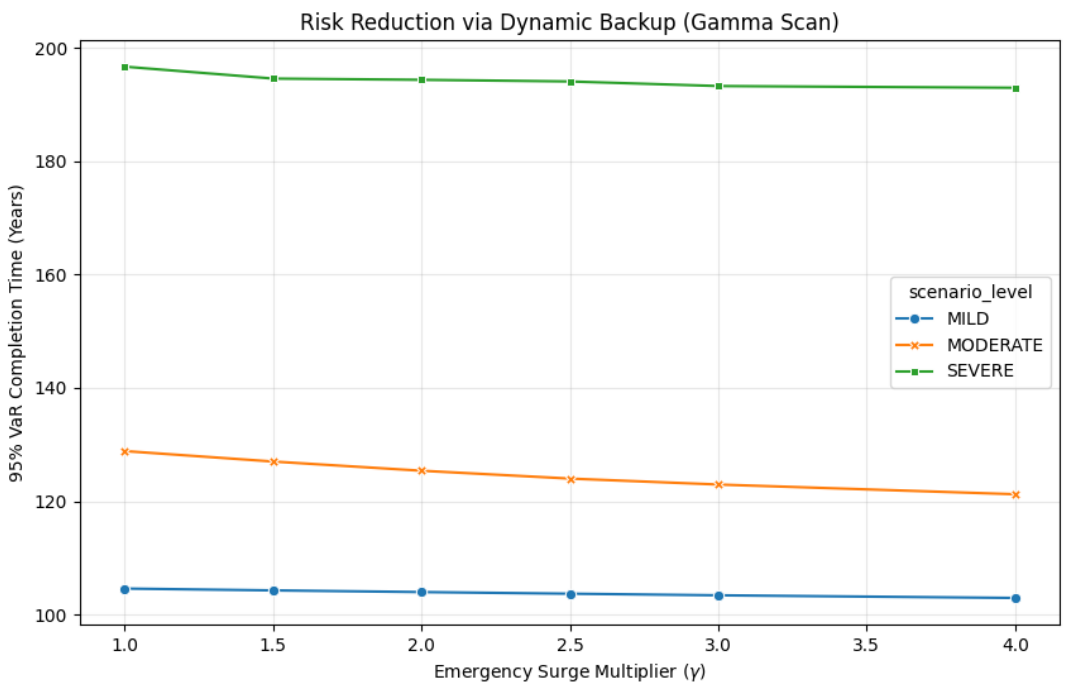

Figure 10: Impact of Emergency Surge Frequency on 95% Confidence Time
X-axis: Rocket Launch Frequency during SE Downtime (Surge Capacity). Y-axis: The time by which the project is 95% guaranteed to finish ($VaR_{0.95}$).
Result: The curve shows diminishing returns. By scanning γ, we identify the Critical Threshold $\gamma^* \approx 1.8$. This value serves as a concrete engineering requirement: it quantifies "to what extent" the rocket system must be capable of surging (an 80% increase) to guarantee robustness against elevator downtime.

6 Algorithm Design and Implementation

[*Writing Cue: Describe your solver as a layered strategy: baseline computation → candidate generation → optimization → validation.*]

6.1 Solution Strategy Overview

1. **Baseline:** compute KPIs on the original network.
2. **Candidate projects:** generate/collect feasible interventions.
3. **Optimization:** solve (**??**) using **[LP/MILP/GA/NSGA-II/heuristic]**.
4. **Validation:** simulate scenarios and compare against baseline.
5. **Robustness:** sensitivity + st

6.2 Pseudocode Template

[*Writing Cue: Give a clean algorithm box. Judges should be able to implement from it.*]

Algorithm 1 End-to-end pipeline for Problem D

Input: Raw datasets $\mathcal{D}$; candidate projects $\mathcal{P}$; budget B
Output: Recommended project set $P^\star$ and impact report

```
Clean D; build graph G = (V, E) with edge weights
Compute baseline KPIs and stakeholder utilities
for each project p ∈ P do
    Estimate project effect on network parameters (capacity, speed, risk, etc.)
end for
Solve optimization model to obtain (x*, y*)
Perform scenario simulations (closures, demand shocks, incidents)
Conduct sensitivity analysis on key parameters
Generate final recommendation and one-page memo return P* = {p ∈ P : y*_p = 1}
```

6.3 Complexity and Practical Considerations

[*Writing Cue: A short complexity paragraph signals maturity.*] [Discuss computational cost, scalability, and why your algorithm fits the contest setting.]

7 Results and Analysis

[*Writing Cue: Lead with the decision: "Our recommended project is ..." Then show evidence.*]

7.1 Baseline vs. Optimized Outcomes

Table 2: Key performance indicators (baseline vs. recommended plan).

Metric	Baseline	Recommended	Change
Avg. travel time (min)	[ ]	[ ]	[%]
Accessibility score	[ ]	[ ]	[%]
Equity gap (lower better)	[ ]	[ ]	[%]
Safety risk index (lower better)	[ ]	[ ]	[%]

7.2 Network Visualization and Interpretation

Figure 11: □□□□□□□□□□□□□□□□/□□□

[*Writing Cue: Interpret the figure: explain why improvements concentrate where they do and which stakeholders gain/lose.*] [Add interpretation.]

7.3 Stakeholder Impact Report

[*Writing Cue: Explicitly discuss winners/losers and mitigation. This is often what separates O papers.*]

- **Residents:** [benefits + potential disruption].
- **Commuters:** [benefits + trade-offs].
- **Businesses/Logistics:** [effects].
- **Vulnerable groups:** [equity discussion].

8 Sensitivity and Robustness Analysis

[*Writing Cue: Tell the reader what you perturb, why it matters, and what "robust" means operationally.*] We test robustness by perturbing **[demand, edge capacities, incident rates, stakeholder weights]** within **[±X%]** and

re-optimizing.

Figure 12: □□□□/KPI □□□□□□□□□□□□□□□□□□/□□□□□

[*Writing Cue: Close with a crisp conclusion: stability region + when it breaks.*] [Summarize sensitivity findings.]

9 Model Evaluation: Strengths and Limitations

[*Writing Cue: Be honest but strategic. Pair each limitation with a fix or future work.*]

9.1 Strengths

- **Interpretability:** [why decision-makers can trust it].
- **Generality:** [how it transfers to other cities/systems].
- **Robustness:** [how uncertainty is handled].

9.2 Limitations and Future Work

- **Data limitations:** [what is missing]. *Future work:* [what to collect / proxy].
- **Behavioral realism:** [what is simplified]. *Future work:* [advanced model].
- **Computation:** [bottleneck]. *Future work:* [speed-up / heuristic].

10 Conclusion

[*Writing Cue: Repeat (i) what you built, (ii) what you found, (iii) what you recommend. Keep it short.*] [Conclude with a 5–7 sentence summary and final recommendation.]

MEMORANDUM

February 1, 2026

To: Mayor / Agency Name
From: Team [########]
Subject: **Strategic Recommendation for [Project Name]**

To: Mayor of [City]
From: Team [########]
Subject: Two transportation projects to improve safety and quality of life
Date: [Month Day, Year]

[Writing Cue: Memo tone: direct, non-technical, decision-oriented.] **Project 1: [Name]**
Benefits: [3 bullet-style sentences on benefits for residents + safety.]
Drawbacks/Disruptions: [1–2 sentences on disruptions + mitigation.]

Project 2: [Name]
Benefits: [...]
Drawbacks/Disruptions: [...]

Recommendation: [[1]1–2 [2] s[**?**]entences: which project first, why, timeline/priority.]

References

[1] Harry Jones. The future impact of much lower launch cost. *ThinkTech* (*Texas Tech University*), 7 2018. URL http://hdl.handle.net/2346/74068.

[2] Melissa Lauren Crouch, Heinz Erasmus Jacobs, and Vanessa L. Speight. Defining domestic water consumption based on personal water use activities. *Journal of Water Supply Research and Technology—AQUA*, 8 2021. doi: 10.2166/aqua.2021.056. URL https://doi.org/10.2166/aqua.2021.056.

A Additional Figures / Derivations (Optional)

[Put non-essential details here to keep the main story tight.]

AI Use Report (If Used)

[Briefly describe what tools were used, for what tasks, and how outputs were verified.]