# Cooling Fan Replacement Policy Evaluation Report with Worked Example

## Introduction

This report uses a discrete-event Monte Carlo simulation to compare two fan replacement policies. We simulate servicing (N=45) failures and track both total cost and total elapsed time. The **Current Policy** replaces one failed fan at a time; the **Proposed Policy** replaces all three fans upon any failure. Below we present the methodology, key formulas, and then a worked example with real numbers.

## Key Formulas

Let (n) be the number of fans replaced ((n=1) for Current, (n=3) for Proposed). Each failure event incurs three cost components:

1. **Replacement Cost** [ C\_{} = n c\_{}, ] where (c\_{} = 32).
2. **Downtime Cost** [ C\_{} = (D + T\_{}) c\_{}, ] where (D) is the sampled technician delay (in minutes), (T\_{}) is replacement time ((20) min for Current, (40) min for Proposed), and (c\_{} = 10) (dollars per minute).
3. **Labor Cost** [ C\_{} = c\_{}, ] where (c\_{} = 30) (dollars per hour).

The **total cost per event** is [ C\_{} = C\_{} + C\_{} + C\_{}. ]

Each event also contributes to elapsed time: [ T\_{} = t\_{} + (), ] where (t\_{}) is the operational interval until the next failure (sampled from the fan lifetime distribution) measured in hours.

## Worked Example with Real Numbers

Consider a single failure event with technician delay (D=20) minutes. We compute costs under both policies and then scale to (N=45) failures by assuming this same delay for every event.

### Current Policy ( (n=1), (T\_{}=20) min )

1. Replacement Cost: [ C\_{} = 1 = 32. ]
2. Downtime Cost: [ C\_{} = (20 + 20) = 40 = 400. ]
3. Labor Cost: [ C\_{} = = = 10. ]
4. Total Cost per Event: [ C\_{}^{} = 32 + 400 + 10 = 442. ]

Scaling to 45 failures: [ C\_{}^{} = 45 = 19{,}890. ]

### Proposed Policy ( (n=3), (T\_{}=40) min )

1. Replacement Cost: [ C\_{} = 3 = 96. ]
2. Downtime Cost: [ C\_{} = (20 + 40) = 60 = 600. ]
3. Labor Cost: [ C\_{} = = = 20. ]
4. Total Cost per Event: [ C\_{}^{} = 96 + 600 + 20 = 716. ]

Scaling to 45 failures: [ C\_{}^{} = 45 = 32{,}220. ]

## Interpretation

Using this fixed-delay example, the **Current Policy** costs $19,890 to service 45 failures, while the **Proposed Policy** costs $32,220. When delays vary randomly in the Monte Carlo simulation, these totals fluctuate around their averages, but this example illustrates how the formulas produce the event-by-event costs that the code sums automatically.

*This worked example validates the cost-calculation logic implemented in the Python simulation script.*