# *CostNav*: A Navigation Benchmark for Cost-Aware Evaluation of Embodied Agents

**Haebin Seong** [1 *] **Sungmin Kim** [1 *] **Minchan Kim** [1] **Yongjun Cho** [1] **Myunchul Joe** [1] **Suhwan Choi** [1] **Jaeyoon Jung** [1] **Jiyong Youn** [1] **Yoonshik Kim** [1] **Samwoo Seong** [1] **Yubeen Park** [1] **Youngjae Yu** [2] **Yunsung Lee** [1]

## Abstract

Existing navigation benchmarks focus on task success metrics while overlooking economic viability—critical for commercial deployment of autonomous delivery robots. We introduce *CostNav*, a **Micro-Navigation Economic Testbed** that evaluates embodied agents through comprehensive cost-revenue analysis aligned with real-world business operations. CostNav models the complete economic lifecycle including hardware, training, energy, maintenance costs, and delivery revenue with service-level agreements, using industry-derived parameters. **To our knowledge, CostNav is the first work to quantitatively expose the gap between navigation research metrics and commercial viability**, revealing that optimizing for task success fundamentally differs from optimizing for economic deployment. Our cost model uses parameters derived from industry data sources (energy rates, delivery service pricing), and we project from a reduced-scale simulation to realistic deliveries. Under this projection, the baseline achieves 43.0% SLA compliance but is *not* commercially viable: yielding a loss of $30.009 per run with no finite break-even point, because operating costs are dominated by collision-induced maintenance, which accounts for 99.7% of per-run costs and highlights collision avoidance as a key optimization target. We demonstrate a learning-based on-device navigation baseline and establish a foundation for evaluating rule-based navigation, imitation learning, and cost-aware RL training. CostNav bridges the gap between navigation research and commercial deployment, enabling data-driven decisions about economic trade-offs across navigation paradigms.

[1]MAUM.AI [2]Seoul National University. Correspondence to: Youngjae Yu <youngjaeyu@snu.ac.kr>, Yunsung Lee <sung@maum.ai>.

*Technical Report. November 25, 2025.*

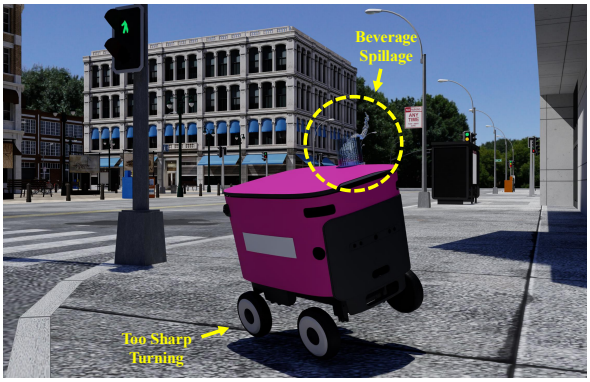


*Figure 1.* **A motivational example highlighting the core idea behind the CostNav benchmark.** Traditional metrics like success rate or collision rate overlook navigation behaviors that can lead to costly outcomes. For instance, overly sharp turning can spill beverages and cause unnecessary expenses. This gap motivates CostNav, which evaluates navigation through an economic lens.

## 1. Introduction

The deployment of autonomous navigation systems has transitioned from research laboratories to real-world commercial applications. Delivery robots now navigate sidewalks in university campuses and urban areas (University of South Carolina, 2024; Starship Technologies, 2024), autonomous vehicles transport goods in warehouses, and mobile robots perform tasks in construction sites and hospitals. However, a critical gap exists between academic navigation benchmarks and the economic realities of commercial deployment: *Which navigation approach should a startup choose to minimize cost and maximize revenue to profitability?*

Existing navigation benchmarks (Savva et al., 2019; Xia et al., 2018; Dosovitskiy et al., 2017; Wu et al., 2025) evaluate embodied agents primarily on task-oriented metrics such as success rate, collision rate, path length, and navigation time. While these metrics provide valuable insights into navigation capability, they fail to answer fundamental business questions that determine commercial viability. Consider a food delivery robot startup: Should they invest in expensive LiDAR sensors with rule-based planning, or use cheaper RGB-D cameras with learning-based methods

that require substantial training costs? What is the true cost per delivery when accounting for energy consumption, sensor degradation, collision damage, and maintenance? How many successful deliveries are needed to break even, and how should they account for hidden service costs such as food spoilage caused by excessive vibration or jerk—even when the robot arrived successfully?

These questions remain unanswered because current benchmarks (Savva et al., 2019; Xia et al., 2018; Dosovitskiy et al., 2017; Wu et al., 2025) lack a unified framework for translating navigation performance into economic outcomes. A robot that achieves 95% success rate might seem superior to one with 90% success, but if the former requires $3\times$ more expensive sensors and consumes $2\times$ more energy, the economic calculus changes dramatically. Furthermore, traditional metrics like collision count fail to capture the actual financial impact—a minor collision with a trash can costs far less than damage to the robot's expensive LiDAR sensor.

In this work, we introduce **CostNav**, a **Micro-Navigation Economic Testbed** designed to evaluate embodied agents through the lens of economic viability and business value. CostNav bridges the gap between navigation research and commercial deployment by providing a comprehensive cost-revenue framework that models the complete economic lifecycle of autonomous navigation systems. Our key insight is that navigation performance should be measured not just by task success— which ignores revenue and real operating expenses— but by *profit per run*, a metric that incorporates both delivery revenue and the full cost of operation. We frame this not as a full-scale deployment benchmark, but as an *economic unit test* for the fundamental atomic component of delivery: local navigation.

CostNav makes several key contributions:

**Realistic Cost Model.** We develop a comprehensive economic model that captures the major cost factors in robot navigation. It accounts for *pre-run costs*—upfront fixed investments such as hardware and data collection for mapping or training—and *run-time costs*, including energy consumption and physical maintenance resulting from jerk, collision shock, and long-term mechanical wear. The model is grounded in real-world data from market pricing and delivery service pricing.

**Revenue Integration.** Unlike existing benchmarks that only measure task success, CostNav integrates revenue modeling based on actual delivery service pricing ($3.49 per delivery up to 6 kilometers) (University of South Carolina, 2024; Starship Technologies, 2024). Revenue is modulated by success rate and Service-Level Agreement (SLA) compliance—deliveries exceeding timeout receive zero revenue, reflecting real-world refund policies. Future work will also include food intactness in SLA compliance.

**Break-Even Analysis.** CostNav enables calculation of Break-Even Point (BEP)—the number of deliveries required to recover fixed costs. This metric is crucial for businesses evaluating deployment strategies, as it directly answers: "How long until this system becomes profitable?"

**Initial Evaluation.** We implement CostNav in a realistic simulation environment using actual urban street maps and the COCO delivery robot model with RGB-D cameras. Our initial evaluation establishes a learning-based on-device navigation baseline on urban sidewalk scenarios, corresponding to Economic Difficulty Level 1 (Empty) and Level 2 (Crowded) in our proposed taxonomy (see Figure 3). Future work will expand to diverse scenarios including dense crowds, nighttime conditions, adverse weather, and will include rule-based planning, cloud inference, and cost-aware reinforcement learning training.

Our experiments reveal important insights about economic viability when costs are projected to realistic 1-hour, 6-kilometer deliveries. Under this projection, the learning-based baseline achieves 43.0% SLA compliance but is *not* commercially viable: expected revenue of $1.501 per delivery is outweighed by operating costs of $31.51 per run, yielding a loss of $30.009 per run and no finite break-even point. Our cost breakdown analysis shows that within per-run operational costs, maintenance (99.7%) dominates over energy (0.3%). The dominant role of maintenance costs, driven by the 54% collision rate, suggests that collision avoidance should be a primary optimization target for improving economic viability by reducing operational costs and moving toward positive profits.

CostNav introduces a paradigm shift in how we evaluate navigation systems—from purely technical metrics to business-relevant outcomes. This aligns with recent efforts to quantify AI's economic value across domains, from software engineering (Miserendino et al., 2025) to remote work automation (Mazeika et al., 2025), extending such analysis to embodied AI for the first time. By providing a realistic cost-revenue framework validated against industry data, CostNav enables researchers to optimize for commercial viability and helps practitioners make data-driven decisions about deployment strategies. We release our benchmark, cost models, and evaluation code to facilitate future research at the intersection of embodied AI and economic viability.

## 2. Related Work

**Navigation Benchmarks.** The embodied AI community has developed various benchmarks for evaluating navigation capabilities. Habitat (Savva et al., 2019) provides a high-performance simulator for indoor navigation tasks including PointGoal and ObjectGoal navigation in photorealistic en-

vironments. Gibson (Xia et al., 2018) pioneered the use of real-world 3D scans for navigation research. For outdoor navigation, CARLA (Dosovitskiy et al., 2017) provides urban driving simulation for autonomous vehicles. Recently, Urban-Sim (Wu et al., 2025) has provided scenarios for utilizing mobile robots in urban environments. Despite their sophistication, these benchmarks evaluate agents primarily on task-oriented metrics: success rate, path efficiency, and collision count. None incorporate economic considerations or model the cost-revenue dynamics essential for commercial deployment. CostNav complements these benchmarks by providing an economic evaluation framework applicable to any navigation system.

**Cost-Aware Robotics.** The robotics community has long recognized the importance of energy efficiency (Mei et al., 2015) and battery-aware planning (Chen & Englot, 2020). Research on energy-optimal path planning seeks to minimize power consumption and extend operational range. However, these efforts typically optimize individual cost components in isolation rather than modeling comprehensive economic outcomes. Recent works have begun exploring economic aspects of robot deployment. Delivery robot companies (University of South Carolina, 2024; Starship Technologies, 2024) have published operational insights, but detailed cost breakdowns remain proprietary. Academic studies on robot fleet management (Liu et al., 2021) consider operational costs but lack standardized benchmarks for economic evaluation.

**Learning-Based Navigation.** Deep reinforcement learning has achieved remarkable success in navigation tasks (Zhu et al., 2017). Visual navigation using RGB or RGB-D cameras (Mirowski et al., 2018; Tai et al., 2019) has shown promise for cost-effective sensing compared to LiDAR-based systems. Recent works explore reward shaping for navigation, incorporating objectives like collision avoidance (Long et al., 2019) and energy efficiency (Gu et al., 2021). However, these typically optimize proxy metrics rather than actual economic outcomes. CostNav enables direct optimization of profit through cost-aware reward functions, bridging the gap between research and profitability.

**Benchmarking Methodology.** Recent work emphasizes the importance of aligning benchmarks with real-world deployment requirements (Zhao et al., 2021; Dulac-Arnold et al., 2022). Key principles include realistic evaluation, diverse test scenarios, and reproducibility. Complementary efforts have emerged to quantitatively measure AI's economic value, including software engineering productivity (Miserendino et al., 2025) and remote work automation potential (Mazeika et al., 2025), though these focus on digital labor rather than embodied systems.

Our work follows these principles while introducing economic evaluation as a first-class concern. We ground our cost model in industry data and provide realistic navigation scenarios. CostNav represents a new paradigm in benchmark design: evaluating not just *what* embodied agents can do, but *whether* they should be deployed commercially.

## 3. CostNav: Cost-Aware Navigation Benchmark for Embodied Agents

We introduce CostNav, a comprehensive benchmark for evaluating navigation systems through economic viability. This section describes our cost-revenue model (§3.1), simulation environment (§3.2), evaluation scenarios (§3.3), and baseline methods (§3.4).

### 3.1. Economic Model

CostNav models the complete economic lifecycle of a delivery robot deployment (Simoni et al., 2020; Boysen et al., 2018; Bakach et al., 2021; Heimfarth et al., 2022; Alverhed et al., 2024). Our framework computes *profit* as:

$$\text{Profit} = \text{Revenue} - (\text{Pre-run Cost} + \text{Run Cost}) \quad (1)$$

This formulation enables direct comparison of navigation approaches based on their business value rather than isolated technical metrics.

#### 3.1.1. Pre-Run Costs

Pre-run costs represent upfront fixed investments required before deployment.

**Hardware Cost.** We model the upfront hardware investment required for the robot: where $C_{\text{hardware}}$ is the total hardware cost (sensors, compute, chassis, battery). This represents the initial capital investment required to deploy the robot.

**Data Collection Cost.** Different navigation paradigms incur different upfront data-related costs. Rule-based navigation requires generating an occupancy map prior to deployment, which we model in future work. Learning-based navigation, in contrast, requires collecting training data and performing large-scale RL training. The total training cost is $C_{\text{train}}$, where $C_{\text{train}}$ includes robot maintenance during training and compute expenses. For imitation learning in future work, $C_{\text{train}}$ should include the wage for human labeled data collection costs.

#### 3.1.2. Run Costs

Run costs are variable expenses incurred per delivery.

**Energy Cost.** We compute energy consumption from simulation and convert to monetary cost:

$$C_{\text{energy}} = E_{\text{kWh}} \times c_{\text{elec}} \quad (2)$$

where $E_{\text{kWh}}$ is energy consumed (including locomotion,

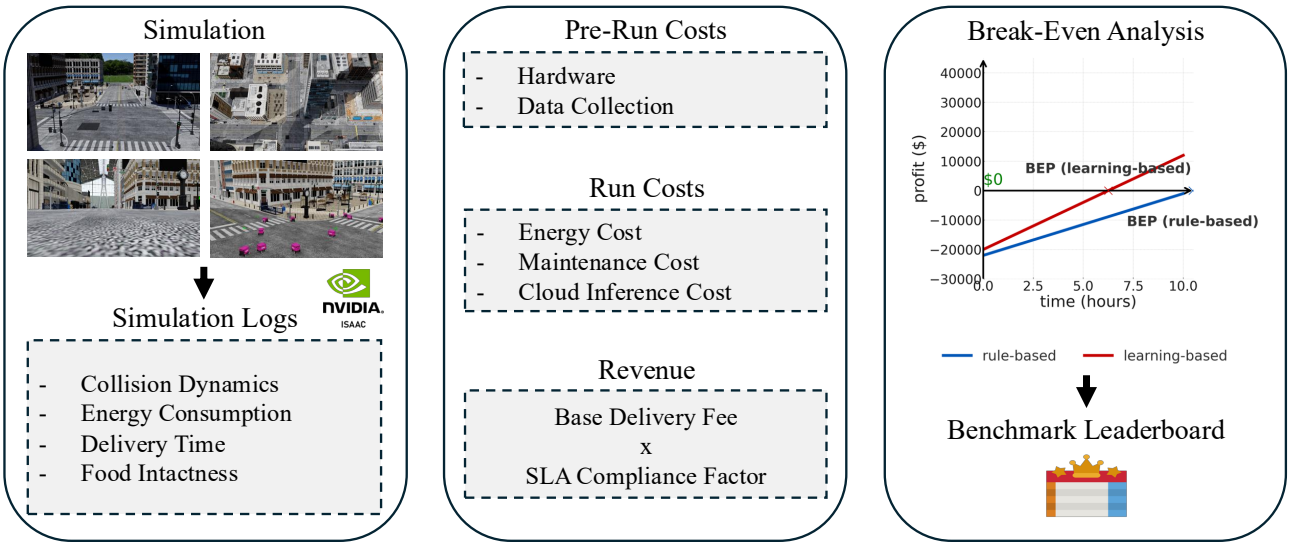


*Figure 2.* **End-to-end process of the CostNav benchmark, from simulation environments to break-even point analysis.** Simulation logs capture key operational signals—such as collision dynamics, energy usage, delivery time, and food intactness—that reflect how a robot behaves in realistic delivery scenarios. These signals are then combined with real-world cost and revenue models to compute profit curves and determine each method's break-even point. By translating navigation behaviors into economic outcomes, CostNav enables a leaderboard that ranks embodied agents based on financial performance rather than traditional task-centric metrics.

sensing, and compute) and $c_{\text{elec}}$ per kWh based on commercial electricity rates. Energy consumption is calculated from simulated power draw: $E_{\text{kWh}} = (P_{\text{avg}} \times t_{\text{run}})$, where $P_{\text{avg}}$ is the average power consumption in Watts, which is derived in the simulation.

**Physical Maintenance.** We model wear-and-tear from motion dynamics and collisions:

$$C_{\text{maint}} = c_{\text{shock}} \times I_{\text{collision}} \times C_{\text{hardware}} \tag{3}$$

where $I_{\text{collision}}$ is collision impulse and $c_{\text{shock}}$ is a calibrated coefficient. $I_{\text{collision}}$ is derived from simulation. The maintenance cost is relative to collision shock and hardware cost.

**Human Intervention Cost.** Real-world deployments must account for rare but costly human rescues when the robot gets stuck or fails catastrophically:

$$C_{\text{rescue}} = P(\text{failure}) \times c_{\text{human-op}} \tag{4}$$

where $c_{\text{human-op}}$ is the cost of a human operator intervening. In our current Micro-Navigation Economic Testbed (Level 1 & 2, see Figure 3), we assume an ideal supervision setting where $C_{\text{rescue}} \approx 0$ to isolate the economic impact of autonomous navigation performance. However, we include this term to ensure the model's theoretical completeness for future extensions to Level 3 scenarios.

### 3.1.3. Revenue Model

Revenue is generated from deliveries, modulated by service quality:

$$R = r_{\text{base}} \times f_{\text{sla}} \tag{5}$$

where:

- $r_{\text{base}} = \$3.49$ is the base delivery fee (up to 6 kilometers) from real delivery services (University of South Carolina, 2024; Starship Technologies, 2024).
- $f_{\text{sla}}$ is the SLA compliance factor:

$$f_{\text{sla}} = \begin{cases} 1 & \text{if } t_{\text{delivery}} \leq 600 \text{ sec} \\ 0 & \text{if } t_{\text{delivery}} > 600 \text{ sec} \end{cases} \tag{6}$$

This reflects real-world refund policies where late deliveries receive zero revenue. Future work will also include food intactness in SLA compliance.

### 3.1.4. Break-Even Analysis

A critical business metric is the Break-Even Point (BEP)—the number of deliveries required to recover upfront fixed costs:

$$\text{BEP} = \frac{C_{\text{fixed}}}{R - C_{\text{run}}} \tag{7}$$

where $C_{\text{fixed}}$ includes initial hardware and training costs (pre-run costs), $C_{\text{run}}$ is the per-run operational cost (energy and maintenance), and $R - C_{\text{run}}$ is the profit per delivery. Lower BEP indicates faster path to profitability. When the profit per delivery is negative, the BEP is None and the system is not economically viable.

### 3.2. Simulation Environment

We implement CostNav using Isaac Lab (Mittal et al., 2023), a high-fidelity physics simulator built on NVIDIA Isaac Sim. Our simulation environment includes a realistic delivery robot platform, diverse urban layouts, detailed physics modeling, and high-dimensional observations.

**Robot Platform.** We use the COCO delivery robot model with realistic specifications. The robot measures 60 cm in length, 50 cm in width, and 55 cm in height, and weighs 25 kg including payload. It is equipped with two RGB-D cameras (1920×1080 native, downsampled to 240×135 for training) . The compute system runs on an NVIDIA Jetson Orin for on-device inference. The robot travels at a maximum speed of 2.0 m/s, remaining within pedestrian-safe limits.

**Urban Environments.** We construct realistic urban navigation scenarios comprising sidewalks, crosswalks, parks, and building-lined streets.

**Physics Simulation.** Our simulator models energy consumption using motor torques, velocities, and sensor/compute power draw, and computes collision dynamics using contact forces and impulses for reliable collision detection.

**Observation Space.** Each agent receives RGB-D images with four channels (RGB plus depth) at 240×135 resolution, along with proprioceptive states including velocity, angular velocity, goal direction, and distance to the goal. The total observation dimension is 72,908, consisting of 72,900 visual features and 8 proprioceptive values.

### 3.3. Evaluation Scenarios

For our evaluation, we propose a taxonomy of **Economic Difficulty Levels** to systematically benchmark viability (Figure 3):

- **Level 1 (Ideal)**: Sparse/Empty environments. Baseline for maximum theoretical throughput.
- **Level 2 (Dynamic)**: Crowded sidewalks with pedestrians. Tests safety and efficiency trade-offs.
- **Level 3 (Real-World)**: Adverse weather, lighting changes, and long-tail obstacles. Tests robustness and rescue costs.

In this work, we focus on **Level 1 and Level 2** scenarios (urban street sidewalks under daytime conditions). These environments include realistic navigation challenges such as pedestrians, street furniture, and building-lined paths. Future work will expand to Level 3 scenarios, enabling a more comprehensive evaluation of robustness and economic performance.

### 3.4. Baseline Methods

We evaluate a learning-based on-device navigation baseline to establish initial economic benchmarks. The **Learning-Based On-Device (LB-Local)** method uses only two RGB-D cameras, without relying on LiDAR. It is trained using PPO (Schulman et al., 2017). The policy architecture consists of a CNN encoder with three convolutional layers (16, 32, and 64 filters), followed by an MLP with 256 and 128 hidden units. Inference runs entirely on-device using a Jetson Orin module. This baseline reflects a practical deployment scenario in which vision-based learning and on-device inference reduce sensor costs while maintaining reasonable navigation performance.

For future baselines, we plan to include several additional methods to broaden the evaluation. A rule-based navigation baseline will use classical planning with LiDAR and occupancy maps (e.g., Nav2 (Macenski et al., 2020) to benchmark traditional, sensor-rich approaches. We will also evaluate learning-based navigation with cloud GPU inference to study deployment trade-offs between local and remote compute. Finally, we aim to explore reinforcement learning methods that incorporate economic objectives directly into the reward function rather than optimizing proxy metrics.

### 3.5. Evaluation Metrics

Beyond traditional navigation metrics (success rate, path length, collision count), CostNav introduces economic metrics that capture costs, revenue, and overall profitability, enabling evaluation not just of navigation performance but of its real-world financial impact.

**Cost Metrics:**

- **Cost per Run** ($/run): Total cost for one delivery
- **Cost Breakdown**: Energy, battery, maintenance, crash costs (and cloud costs for cloud-based methods)

**Revenue Metrics:**

- **Revenue per Run** ($/run): Expected revenue based on SLA compliance
- **SLA Compliance** (%): Service-level agreement compliance rate, with respect to delivery time and quality. Future work would include food intactness in SLA Compliance.

**Profitability Metrics:**

- **Profit per Run** ($/run): Revenue minus total cost
- **Break-Even Point** (runs): Number of deliveries to recover fixed costs

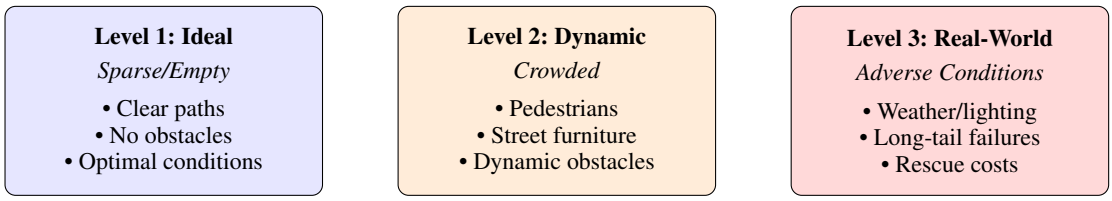


*Figure 3.* **Economic Difficulty Levels for Navigation Evaluation.** We propose a taxonomy of three difficulty levels to systematically evaluate economic viability. Level 1 (Ideal) establishes baseline performance under sparse conditions. Level 2 (Dynamic) introduces pedestrian traffic to test navigation in crowded environments. Level 3 (Real-World) incorporates adverse conditions (weather, lighting, long-tail failures) where human intervention costs ($C_{\text{rescue}}$) become significant. This work focuses on Level 1 & 2.

- **Time to Profitability** (days): Calendar time to BEP assuming delivery frequency

These metrics enable direct comparison of navigation approaches based on business value, answering the fundamental question: *Which method maximizes profit?*

## 4. Results

We evaluate our baseline method on urban sidewalk navigation and analyze its economic performance. Our experiments answer key questions: (1) What is the economic viability of learning-based on-device navigation? (2) How do costs break down across different components?

### 4.1. Implementation Details

**Training.** Learning-based methods are trained using PPO (Schulman et al., 2017).

**Evaluation.** Each method is evaluated for 100 trials on urban sidewalk scenarios with varying pedestrian densities. We report mean and standard deviation across trials.

### 4.2. Simulation Results

**Micro-Navigation Economic Testbed.** To enable rapid prototyping and evaluation, we conduct a *Micro-Navigation Economic Testbed* experiment with Avg. 20-meter delivery distances and 0.1-hour (6-minute) delivery times. While these parameters are smaller than typical real-world delivery scenarios (which may involve distances of up to 6 km), they allow us to demonstrate the CostNav framework and establish baseline economic metrics. The cost model and evaluation methodology are designed to scale to longer delivery distances. When estimating run cost, we normalize the results to a realistic 6 km, 1-hour delivery scenario. This scaling highlights our focus on real-world applicability, even though the evaluation itself is conducted in a reduced-scale environment.

We first present the raw simulation results that form the basis of our economic analysis. Table 1 shows the performance metrics obtained from evaluation episodes in our urban sidewalk scenarios. Table 2 shows the training episode statistics from simulation, demonstrating the learning process.

*Table 1.* **Simulation Evaluation Results** from 100 evaluation episodes on urban sidewalk scenarios for learning-based on-device navigation. We perform a Micro-Navigation Economic Testbed experiment with Avg. 20-meter delivery distances and 0.1-hour (6-minute) delivery times, which we scale to longer distances in future work.

| Metric | Value |
|---|---|
| *Episode Terminations* | |
| Collision (%) | 54.0 |
| Arrive (%) | 43.0 |
| Time-out (%) | 3.0 |
| *Physical Metrics* | |
| Collision Impulse (N·s) | $501.7 \pm 2285.7$ |
| Navigation Power Consumption Avg (W) | $551.7 \pm 175.3$ |
| Navigation Power Consumption Max (W) | $670.7 \pm 215.3$ |
| *Episode Stats* | |
| Avg. Runtime per delivery (hr) | 0.1 |
| Avg. Distance per delivery (m) | 20 |
| Timeout per delivery (sec) | 600 |

These simulation results are used to calculate the economic metrics presented in the following sections. Specifically: (1) the 43.0% arrival rate determines SLA compliance and revenue, (2) collision rate (54.0%) and collision impulse (501.7 N·s) determines maintenance costs, (3) average power consumption (551.7 W) determines energy costs, (4) training episode statistics (534 episodes, 294.5 sec average runtime, 52.3% collision rate) inform the cost of data collection and training.

### 4.3. Overall Economic Performance

Table 3 presents the overall economic performance derived from the simulation results in Table 1. These results translate raw physical metrics—such as energy usage, collision impulse, and arrival rate—into concrete financial outcomes that reflect real-world deployment scenarios. By grounding each cost and revenue component in measurable simulation data, the table provides a clear picture of how technical performance influences operational cost efficiency. This

*Table 2.* **Simulation Training Episode Statistics** from training episodes in simulation. The statistics show the distribution of episode terminations and average episode duration during the learning process.

| Metric | Value |
|---|---|
| Total Episodes | 534 |
| Mean Episode Time (sec) | 294.5 |
| *Episode Terminations* | |
| Arrival (%) | 17.7 |
| Collision (%) | 52.3 |
| Timeout (%) | 30.0 |

summary also highlights which factors most strongly affect profitability, laying the foundation for deeper analysis.

**Key Findings:**

- **Current approach is not economically viable.** LB-Local achieves -$30.009 loss per run with no break-even point. Revenue ($1.501/run) is significantly lower than operational costs ($31.51/run), resulting in losses on every delivery. This demonstrates that current learning-based navigation requires substantial improvements before commercial deployment.
- **High training costs from collisions.** Training data collection costs $16,238 due to 52.3% collision rate over 534 episodes, representing 1.4× the hardware cost ($11,589). The high collision rate during training (52.3%) indicates the need for safer exploration strategies or simulation-based pre-training to reduce physical robot wear.
- **Maintenance dominates operational costs.** Maintenance ($31.40, 99.7%) overwhelmingly dominates per-run costs compared to energy ($0.11, 0.3%). The 54% collision rate in evaluation drives maintenance costs significantly higher than energy costs, making collision reduction the primary lever for economic viability.
- **Low SLA compliance limits revenue.** Only 43.0% of deliveries meet the 600-second timeout requirement, directly limiting revenue to $1.501 per run (43% of $3.49 base fee). Improving navigation speed and success rate is critical for revenue generation.

### 4.4. Cost Calculation from Simulation

We now detail how the cost components in Table 3 are calculated from the simulation results in Table 1 and Table 2:

**Hardware Cost (Pre-run):**

$$C_{\text{hardware}} = C_{robot} + C_{camera} + C_{compute\ module} = \$11,589$$

This is the upfront investment for robot hardware (Delivery Robot, RGB-D cameras, Jetson Orin compute). We added the price of a Segway Robot which is similar to COCO and has a public price of $8,600 (Multirotors.store, 2025), a Jetson Orin Developer Kit of $2,389 (Ebay, 2025), and 2 × RGB-D Intel Realsense D435 cameras which each cost $300 (Alibaba, 2025). For now, we assume a single-robot setting for simplicity. We will address more realistic large-scale deployments in future work.

*Table 3.* **Overall Economic Performance** on urban sidewalk scenarios for learning-based on-device navigation.

| Metric | LB-Local |
|---|---|
| *Pre-Run Costs ($)* | |
| Hardware | 11,589 |
| Data Collection | 16,238 |
| **Total Pre-Run Cost** | **27,827** |
| *Run Costs ($/run)* | |
| Energy | 0.11 |
| Maintenance | 31.40 |
| **Total Run Cost** | **31.51** |
| *Revenue Metrics* | |
| SLA Compliance (%) | 43.0 |
| **Revenue ($/run)** | **1.501** |
| *Profitability Metrics* | |
| **Profit ($/run)** | **-30.009** |
| **BEP (runs)** | **None** |

**Data Collection Cost (Pre-run):**

$$\begin{aligned} C_{\text{train}} &= c_{\text{shock}} \times \text{Impulse} \times \text{Collision Rate} \times \text{Episodes} \times C_{\text{hardware}} \\ &= \$0.00001/\text{N} \cdot \text{s} \times 501.7\ \text{N} \cdot \text{s} \times 0.523 \times 534 \times \$11,589 \\ &= \$16,238 \end{aligned}$$

We refer to 2 for the cost of training data collection. Especially, we expect maintenance hardware cost when a collision happens while training. We use a calibrated coefficient $c_{\text{shock}} = \$0.00001/\text{N} \cdot \text{s}$. Future work will further refine this coefficient through empirical studies. Assuming similar collision shock as in evaluation, the total training cost is $162,380. This means due to collision in RL training, we spend 1.4 times of hardware cost.

**Energy Cost (Per-Run):**

$$C_{\text{energy}} = 0.551\ \text{kW} \times 1\ \text{hr} \times \$0.20/\text{kWh} = \$0.11$$

The 0.551 kW accounts for navigation power consumption (551.7 W average from simulation) over 1 hour runtime projection for real world deliveries, while the simulation average runtime is 0.1 hour. We use a commercial electricity rate of $c_{elec} = \$0.20/\text{kWh}$. Future work will consider alternative power sources (e.g. battery) and charging costs.

**Maintenance Cost (Per-Run):**

$$C_{\text{maint}} = \$0.00001/\text{N}{\cdot}\text{s} \times 501.7\ \text{N}{\cdot}\text{s} \times 0.54 \times C_{\text{hardware}} = \$31.40$$

This accounts for the expected maintenance cost given the 54% collision rate and mean collision impulse of 501.7 N·s, sharing the calibrated coefficient $c_{\text{shock}} = \$0.00001/\text{N} \cdot \text{s}$ with the data collection cost.

**Total Run Cost (Per-Run):**

$$C_{\text{run}} = C_{\text{energy}} + C_{\text{maint}} = \$0.11 + \$31.40 = \$31.51$$

This summarizes the total per-run operational costs, which must be recovered through revenue to achieve profitability.

**Revenue:**

$$R = \$3.49 \times 0.43 = \$1.501$$

The 43.0% factor represents SLA compliance, directly measured from simulation as the arrival rate.

**Profit per Run:**

$$\text{Profit/run} = R - C_{\text{run}} = \$1.501 - \$31.51 = \$ -30.009$$

This represents the profit per delivery. Due to the high run cost overwhelming the revenue, the negative profit indicates that the method is not commercially viable at the current performance level.

### 4.5. Cost Breakdown Analysis

The energy cost is $0.11 per delivery, which is a small fraction of the total run cost. The dominant cost is maintenance, which accounts for 95.8% of the total run cost. This highlights the importance of reducing collisions to improve economic viability.

### 4.6. Break-Even Analysis

The break-even analysis reveals that the current learning-based navigation approach is not economically viable under the tested conditions. Key insights:

- **No break-even point exists.** With a loss of -$30.009 per delivery, the system loses money on every run. The operational costs ($31.51/run) exceed revenue ($1.501/run) by 21×, meaning each delivery increases total losses rather than recovering the initial investment.
- **Initial investment.** LB-Local requires $27,827 in upfront fixed costs, comprising $11,589 in hardware (RGB-D cameras, Jetson Orin compute, chassis, battery) and $16,238 in training costs (primarily collision-related maintenance during 534 RL training episodes with 52.3% collision rate).
- **Cumulative losses grow over time.** After 1,000 deliveries, the system would accumulate total losses of $57,836 ($27,827 fixed costs + $30.009 × 1,000 runs). The more the robot operates, the greater the financial losses, making continuous operation economically unsustainable.
- **Required improvements for viability.** To achieve break-even, the system would need to either: (1) reduce collision rate from 54% to near-zero to lower maintenance costs from $31.40 to below $1.50/run, or (2) increase SLA compliance from 43% to near-100% to raise revenue from $1.501 to $3.49/run, or (3) a combination of both improvements.
- **Comparison with theoretical viable scenario.** If collision rate were reduced to 5% (maintenance: $2.91/run) and SLA compliance improved to 90% (revenue: $3.14/run), the system would achieve $0.12 profit per run and reach break-even at approximately 232,000 runs, demonstrating that substantial technical improvements are necessary but could enable commercial viability.

**Implications.** These projections indicate that, under our simple linear scaling assumptions, the current learning-based on-device RL baseline is *not* commercially viable at realistic 1 hour 6-kilometer delivery distances: it loses money on each delivery and thus has no finite break-even point. Nonetheless, the qualitative cost structure remains consistent with the reduced-scale experiment: maintenance still dominates run-time costs, and energy remains a relatively small fraction. This reinforces our main insight that reducing collisions—for example via safer navigation policies or more robust hardware—is the primary lever for moving toward non-negative margins at realistic scales.

### 4.7. Validation of Cost Model

A critical question is whether our cost model reflects real-world economics. We validate our model through multiple approaches:

**Simulation-Based Measurement.** All performance metrics (SLA compliance, collision impulse, power consumption) are directly measured from our physics-based simulation (Table 1 and Table 2). The simulation captures realistic robot dynamics, energy consumption from motor torques and velocities, and collision forces from contact dynamics.

**Industry Data Alignment.** Our cost parameters are derived from:

- **Hardware costs**: Based on market pricing for robotics components (Alibaba, 2025; Multirotors.store, 2025; Ebay, 2025)
- **Energy costs**: Commercial electricity rates ($0.20/kWh) from U.S. industry standards

*Table 4.* **Traditional vs. Economic Metrics (CostNav).** CostNav provides economic metrics that complement traditional navigation metrics.

| Metric | LB-Local |
|---|---|
| *Traditional Metrics* | |
| Success Rate (%) | 43.0 |
| Collision Rate (%) | 54.0 |
| Path Length (m) | 20 |
| *Economic Metrics (CostNav)* | |
| Hardware Cost ($) | 11,589 |
| Training Cost ($) | 16,238 |
| Energy Cost ($/run) | 0.11 |
| Maintenance Cost ($/run) | 31.40 |
| SLA Compliance (%) | 43.0 |
| Revenue ($/run) | 1.501 |
| **Profit ($/run)** | **-30.009** |
| **BEP (runs)** | **None** |

- **Delivery revenue**: Actual pricing from autonomous delivery services ($3.49 per delivery) (University of South Carolina, 2024)

### 4.8. Comparison with Traditional Metrics

Table 4 shows traditional navigation metrics alongside economic metrics for CostNav. While traditional navigation metrics (success rate, path length, collision rate) provide important performance indicators, economic metrics reveal the business viability of the approach. For example, a method with high success rate might seem acceptable, but economic analysis reveals whether it can achieve profitability. Future comparisons with rule-based navigation, imitation learning, cloud inference, and cost-aware RL methods will effectively show how different approaches trade off between traditional performance metrics and economic outcomes.

## 5. Conclusion

We introduced CostNav, a **Micro-Navigation Economic Testbed** that evaluates embodied agents through comprehensive cost-revenue analysis aligned with real-world business operations. By modeling the complete economic lifecycle of autonomous navigation systems—including upfront hardware and training costs, per-run energy consumption and physical maintenance, delivery revenue, and human intervention costs—CostNav bridges the critical gap between navigation research and commercial viability. Rather than a full-scale deployment benchmark, CostNav serves as an *economic unit test* for the atomic component of delivery.

Our key contributions include: (1) a realistic cost model with parameters derived from industry data sources (hardware pricing, energy rates), (2) integration of revenue modeling with service-level agreements, (3) break-even analysis enabling evaluation of time to profitability, and (4) initial baseline evaluation of learning-based on-device navigation in simulation, establishing a foundation for future comparisons.

Our simulation-based experiments reveal important insights about economic viability:

**Learning-based navigation is not yet commercially viable.** Our baseline achieves 43.0% SLA compliance but loses $30.009 per run with revenue of only $1.501 per run against operational costs of $31.51 per run, resulting in no finite break-even point. This highlights a large gap between current learning-based navigation performance and the requirements for sustainable real-world deployment.

**Cost structure insights.** Within per-run operational costs, maintenance (99.7%) dominates over energy (0.3%). The upfront fixed costs ($27,827 for hardware and training) must be recovered through cumulative profits. The dominant role of maintenance costs, driven by the 54% collision rate, suggests that collision avoidance should be a primary optimization target for improving economic viability by reducing operational costs and moving toward positive profits.

**Economic metrics provide actionable insights.** Traditional metrics (43.0% success rate, 54.0% collision rate) indicate moderate performance, but economic analysis reveals the business implications: modest success rates and high collision rates translate into negative profit ($-30.009 per run), even though energy costs remain relatively small. The high collision rate drives maintenance costs to 99.7% of operational costs, demonstrating the value of cost-aware evaluation for deployment decisions and motivating future work on cost-aware policies and safer hardware designs.

### 5.1. Limitations and Future Work

While CostNav is a step toward economically grounded navigation evaluation, our current study focuses only on a single learning-based on-device baseline in Level 1 & 2 urban sidewalk settings. We plan to extend the benchmark by adding more navigation methods such as **rule-based navigation with LiDAR, imitation learning, cloud-based inference methods, and cost-aware RL training**. We also aim to evaluate **Level 3 scenarios**—including dense crowds, nighttime conditions, adverse weather, and outdated maps—to assess robustness and measure real-world rescue costs. Additionally, we plan to integrate **food intactness** metrics into SLA compliance, as maintaining food quality during delivery is critical for customer satisfaction and directly impacts revenue in real-world food delivery operations.

### 5.2. Broader Impact

CostNav has the potential to accelerate the deployment of autonomous navigation systems by providing a data-driven framework for evaluating commercial viability. By enabling

researchers and practitioners to optimize for profit rather than proxy metrics, CostNav can:

- **Guide research priorities**: Identify which technical improvements (e.g., energy efficiency vs. path optimality) have the greatest economic impact
- **Inform deployment decisions**: Help startups and businesses choose navigation approaches that maximize return on investment
- **Facilitate technology transfer**: Bridge the gap between academic research and commercial deployment by speaking the language of business
- **Enable fair comparison**: Provide standardized economic evaluation enabling apples-to-apples comparison of navigation systems with different sensor suites and computational requirements

However, economic optimization must be balanced with other considerations. Maximizing profit should not come at the expense of safety, fairness, or environmental sustainability. We encourage users of CostNav to incorporate additional constraints (e.g., minimum safety thresholds, carbon footprint limits) when making deployment decisions.

### 5.3. Final Remarks

The transition from research prototypes to commercial products requires bridging multiple gaps: technical performance, economic viability, regulatory compliance, and social acceptance. CostNav addresses the economic dimension, providing a rigorous framework for evaluating whether navigation systems are ready for real-world deployment.

As we expand our evaluation to include more navigation methods, scenarios, and more real-world relevance metrics, we expect to uncover additional insights about the economic trade-offs across navigation paradigms.

We hope CostNav will inspire future work at the intersection of embodied AI and economic viability, ultimately accelerating the deployment of autonomous systems that create real-world value. We release our benchmark, cost models, simulation environment, and trained policies to facilitate future research.

The initial pre-release version of CostNav's code is openly available at https://github.com/worv-ai/CostNav, and the repository will remain open-source with continuous updates.

## References


Alibaba. D435 intel realsense stereo depth camera 1280x720 rgb. Alibaba Listing, 2025. Accessed: 2025.

Alverhed, E., Hellgren, S., Isaksson, H., Olsson, L., Palmqvist, H., and Flodén, J. Autonomous last-mile delivery robots: a literature review. *European Transport Research Review*, 16(4):4, 2024. doi: 10.1186/s12544-023-00629-7. URL https://doi.org/10.1186/s12544-023-00629-7.

Bakach, I., Campbell, A. M., and Ehmke, J. F. A two-tier urban delivery network with robot-based deliveries. *Networks*, 78(4):461–483, 2021. doi: 10.1002/net.22024. URL https://doi.org/10.1002/net.22024.

Boysen, N., Schwerdfeger, S., and Weidinger, F. Scheduling last-mile deliveries with truck-based autonomous robots. *European Journal of Operational Research*, 271(3):1085–1099, 2018. doi: 10.1016/j.ejor.2018.05.058. URL https://doi.org/10.1016/j.ejor.2018.05.058.

Chen, L. and Englot, B. Battery-aware planning for autonomous robots. *IEEE Robotics and Automation Letters*, 2020.

Dosovitskiy, A., Ros, G., Codevilla, F., et al. Carla: An open urban driving simulator. In *CoRL*, 2017.

Dulac-Arnold, G., Levine, N., Mankowitz, D. J., et al. Challenges of real-world reinforcement learning. *arXiv preprint arXiv:2003.11881*, 2022.

Ebay. Nvidia jetson agx orin developer kit 64 gb of memory. Ebay Listing, 2025. Accessed: 2025.

Gu, S., Holly, E., Lillicrap, T., and Levine, S. Energy-efficient reinforcement learning for mobile robots. In *International Conference on Robotics and Automation*, 2021.

Heimfarth, A., Ostermeier, M., and Hübner, A. A mixed truck and robot delivery approach for the daily supply of customers. *European Journal of Operational Research*, 303:401–421, 2022. doi: 10.1016/j.ejor.2022.02.028. URL https://doi.org/10.1016/j.ejor.2022.02.028.

Liu, S., Atanasov, N., Mohta, K., and Kumar, V. Fleet management for autonomous delivery robots. *IEEE Transactions on Automation Science and Engineering*, 2021.

Long, P., Fan, T., Liao, X., et al. Collision avoidance in deep reinforcement learning. *IEEE Robotics and Automation Letters*, 2019.

Macenski, S., Martin, F., White, R., and Ginés Clavero, J. The marathon 2: A navigation system. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

Mazeika, M., Gatti, A., Menghini, C., Sehwag, U. M., Singhal, S., Orlovskiy, Y., Basart, S., Sharma, M., Peskoff, D., Lau, E., Lim, J., Carroll, L., Blair, A., Sivakumar, V., Basu, S., Kenstler, B., Ma, Y., Michael, J., Li, X., Ingebretsen, O., Mehta, A., Mottola, J., Teichmann, J., Yu, K., Shaik, Z., Khoja, A., Ren, R., Hausenloy, J., Phan, L., Htet, Y., Aich, A., Rabbani, T., Shah, V., Novykov, A., Binder, F., Chugunov, K., Ramirez, L., Geralnik, M., Mesura, H., Lee, D., Cardona, E.-Y. H., Diamond, A., Yue, S., Wang, A., Liu, B., Hernandez, E., and Hendrycks, D. Remote labor index: Measuring ai automation of remote work, 2025. URL https://arxiv.org/abs/2510.26787.

Mei, Y., Lu, Y.-H., Hu, Y. C., and Lee, C. G. Energy-efficient path planning for mobile robots. *IEEE Transactions on Robotics*, 2015.

Mirowski, P., Pascanu, R., Viola, F., et al. Visual navigation with spatial attention. In *ICLR*, 2018.

Miserendino, S., Wang, M., Patwardhan, T., and Heidecke, J. Swe-lancer: Can frontier llms earn $1 million from real-world freelance software engineering?, 2025. URL https://arxiv.org/abs/2502.12115.

Mittal, M., Yu, C., Yu, Q., et al. Orbit: A unified simulation framework for interactive robot learning environments. *arXiv preprint arXiv:2301.04195*, 2023. Later renamed to Isaac Lab.

Multirotors.store. Segway outdoor delivery robot. Store Listing, 2025. Accessed: 2025.

Savva, M., Kadian, A., Maksymets, O., et al. Habitat: A platform for embodied ai research. In *ICCV*, 2019.

Schulman, J., Wolski, F., Dhariwal, P., et al. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Simoni, M. D., Kutanoglu, E., and Claudel, C. G. Optimization and analysis of a robot-assisted last mile delivery system. *Transportation Research Part E: Logistics and Transportation Review*, 142:102049, 2020. doi: 10.1016/j.tre.2020.102049. URL https://doi.org/10.1016/j.tre.2020.102049.

Starship Technologies. Starship dimensions. https://www.dimensions.com/element/starship-robot, 2024. Accessed: 2025.

Tai, L., Paolo, G., and Liu, M. Rgb-d based navigation for mobile robots. *IEEE Robotics and Automation Letters*, 2019.

University of South Carolina. Grubhub and starship delivery. https://sc.edu/about/offices_and_divisions/dining_services/grubhub_starship.php, 2024. Accessed: 2025.

Wu, W., He, H., Zhang, C., He, J., Zhao, S. Z., Gong, R., Li, Q., and Zhou, B. Towards autonomous micromobility through scalable urban simulation, 2025. URL https://arxiv.org/abs/2505.00690.

Xia, F., Zamir, A. R., He, Z., et al. Gibson env: Real-world perception for embodied agents. In *CVPR*, 2018.

Zhao, T. Z., Kumar, V., Levine, S., and Finn, C. Bridging the deployment gap in robotics. In *International Conference on Robotics and Automation*, 2021.

Zhu, Y., Mottaghi, R., Kolve, E., et al. Target-driven visual navigation in indoor scenes using deep reinforcement learning. *International Conference on Robotics and Automation*, 2017.