TARSPHERE: A ROBOT-ORIENTED FULL-STACK FRAMEWORK FOR LONG-HORIZON EMBODIED AGENTS IN EVOLVING PHYSICAL AND SOCIAL ENVIRONMENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Robots must act under physical and social constraints while adapting to changes in their environment and human needs. Existing embodied-agent systems often focus on specified goals or abstract execution into symbolic actions, limiting evaluation in an evolving world. We introduce TARSphere, ***a robot-oriented full-stack framework*** for physically and socially grounded long-horizon embodied agents. TARSphere-Infra runs robot policies in a physics simulator with embodied NPCs and constrained dialogue. An event-driven finite-state machine (FSM) uses completed actions and delivered information to update task states and evaluate outcomes. TARSphere-Harness, its embodied brain harness, lets a frozen multimodal large language model (MLLM) act through multi-level executable skills, using observations, hierarchical memory, and execution feedback to revise its decisions. On this foundation, agent-in-the-loop curation aligns task semantics, causal dependencies, and scene geometry to construct TARSphere-Bench: 360 episodes across 30 indoor scenes. Five paired conditions assess perceptual robustness, disruption adaptation, task update, human coordination, and intent understanding for navigate-and-report and fetch-and-deliver tasks. Our evaluation of 8 MLLMs shows that general-purpose models can outperform embodied models, while ***task updates and human interaction remain difficult***. Frequent collisions and falls further motivate evaluating execution safety alongside task success.

1 INTRODUCTION

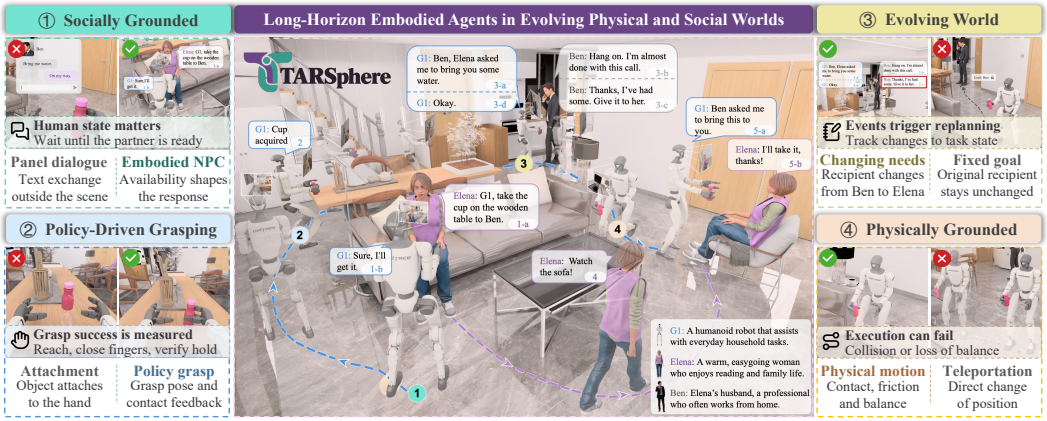


Figure 1: **Robot execution and human interaction in a changing world.**

Robots operating in everyday environments must account for physical constraints (Li et al., 2023), human safety, appropriate interaction, and social norms (Mavrogiannis et al., 2023; Puig et al., 2024). As the world evolves, sensory conditions, physical access, and people's actions and needs can change during execution (Huang et al., 2022; Puig et al., 2024; Huang et al., 2023). Long-horizon tasks

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

therefore require tracking world states, interpreting human intentions, and adjusting plans using execution feedback (Wan et al., 2022; Chang et al., 2025; Huang et al., 2022) (Fig. 1).

Vision-language-action (VLA) models (Brohan et al., 2023; Kim et al., 2024) and world-action models (WAMs) (Ye et al., 2026) have advanced robot control, yet they often lack the ability to adapt to dynamic and uncertain environments. Large language models (LLMs) provide broad knowledge for reasoning yet lack the ability to perform complex, real-world tasks (Ahn et al., 2022; Huang et al., 2022; 2023). This complementarity motivates embodied agents that connect reasoning to action (Ahn et al., 2022; Huang et al., 2022; Yang et al., 2025; Li et al., 2026b). However, Harness VLA (Zhang et al., 2026), RoboClaw (Li et al., 2026a), and HumanCLAW (Li et al., 2026b) focus on tasks with specified goals, leaving an evolving world with human involvement largely unexamined. Other settings, including EmbodiedBench's high-level tasks (Yang et al., 2025), LoTa-Bench (Choi et al., 2024), ALFWorld's text environments (Shridhar et al., 2020), and Embodied Agent Interface (Li et al., 2024), abstract physical execution into discrete symbolic actions. GRBench (Wang et al., 2024) combines robot execution with virtual NPC dialogue, while PARTNR (Chang et al., 2025) evaluates embodied human-robot collaboration with temporal task constraints. These gaps motivate a physically and socially grounded foundation for benchmarking robotic agents in an evolving world.

Insight. Studying embodied intelligence requires ***physically and socially grounded execution*** and tasks revealing capabilities across varied situations in a changing world (§§ 3.2, 3.3.2 and 3.3.4).

Challenge 1: Embodied execution is not symbolic [§§ 3.2 and 3.3.4]. A valid action call does not guarantee successful execution: the robot must contend with contact, friction, balance, and human responses. The agent must therefore revise decisions using physical execution feedback.

Challenge 2: The world changes during execution [§§ 3.3.2 and 3.3.3]. Physical conditions, human activity, and task requirements can change while a plan unfolds. The agent must interpret the current situation and reconsider what to do next.

Challenge 3: Evaluation must assess diverse capabilities [§ 3.2]. Within this setting, evaluation must reveal how agents handle varied situations across multiple capability dimensions. Aggregate success alone can mask different capability gaps across task families (§ 3.2).

We introduce TARSphere, a robot-oriented full-stack framework for embodied agents under physical and social constraints. TARSphere-Infra couples robot controllers with embodied NPCs in a physics simulator (Challenge 1). An event-driven FSM maintains world and task states, while TARSphere-Harness combines observations, memory, skills, and execution feedback for replanning (Challenge 2). We build TARSphere-Bench through agent-in-the-loop curation, pairing navigate-and-report and fetch-and-deliver tasks with five capability conditions: perceptual robustness, disruption adaptation, task update, human coordination, and intent understanding (Challenge 3).

Across 8 MLLMs, ***general-purpose models can outperform embodied models***, with capability gaps varying by task family (§ 3.2). Planning credit does not ensure successful execution, and collisions and falls remain common (§§ 3.2 and 3.3.4). These findings call for agents that combine general reasoning, adaptation to world changes, and safe execution (§§ 3.2 and 3.3.2).

Our contributions are three-fold.

- We formulate long-horizon embodied tasks under physical and social constraints, where execution outcomes and world changes continually shape the agent's decisions.
- We develop TARSphere Framework, where physical execution and interaction with embodied humans update task states and supply evaluation evidence; an embodied brain harness uses observations, memory, executable skills, and world feedback to guide model decisions.
- We introduce agent-in-the-loop curation for 360 TARSphere-Bench episodes in 30 scenes and five paired capability conditions, aligning task semantics, geometry, and execution.

2 Methodology

2.1 Problem Formulation

We study physically and socially grounded long-horizon embodied execution in a changing simulated world. The agent interprets human intent from egocentric observations, communication, and interaction history, then pursues the required physical outcome or accurate report.

At time t_k, policy π selects action sequence B_k from observation o_{t_k} and interaction history h_{t_k}:

$$B_k = \pi(o_{t_k}, h_{t_k}), \qquad \langle s_{t+1}, f_t \rangle = \mathcal{W}(s_t, \mathrm{Exec}(a_t), e_t), \quad a_t \in B_k. \tag{1}$$

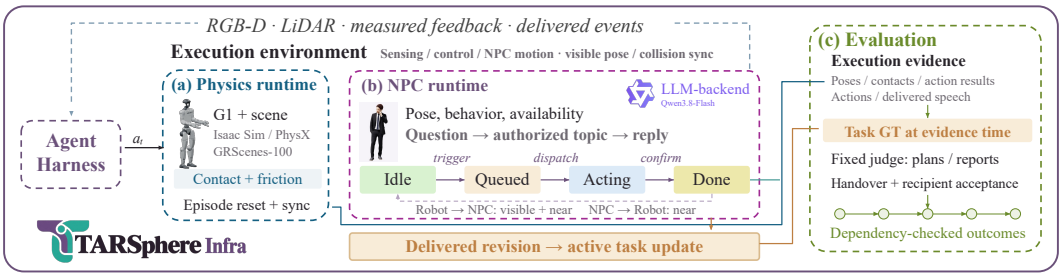


Figure 2: **TARSphere-Infra: execution infrastructure.** (a) Physics runtime executes robot actions under contact and friction. (b) An event-driven FSM coordinates NPC actions, message delivery, and task-state transitions. (c) Evaluation checks execution evidence against the task requirements active when that evidence was produced.

Here s_t comprises robot, object, human, and active task states; e_t is a human or environmental event. Exec invokes the backend; $\mathcal{W}$ updates the world and returns measured feedback f_t. Actions execute sequentially; new observations and feedback inform the next decision. TARSphere Framework implements this loop; TARSphere-Bench supplies tasks and controlled capability conditions there.

2.2 TARSphere Framework

TARSphere Framework has two layers: TARSphere-Infra, the Execution Infrastructure (Fig. 2), manages physical execution, embodied human interaction, task states, and evaluation. TARSphere-Harness, the Embodied Brain Harness (Fig. 3), organizes observations, memory, executable skills, and feedback for a frozen MLLM. TARSphere-Infra returns synchronized RGB-D and LiDAR observations, measured feedback, and delivered public events to the harness; task requirements and scoring are private. We separately test execution fidelity and harness components (§§ 3.3.1 and 3.3.4).

2.2.1 TARSphere-Infra

Simulation and robot execution. In Fig. 2, the harness sends robot actions to a shared execution environment containing G1, task objects, and NPCs in GRScenes-100 (Wang et al., 2024) scenes. Isaac Sim (NVIDIA) and PhysX execute robot control under contact and friction: locomotion follows a learned policy, while grasping executes planned arm motion and requires physical object acquisition and retention (§ 3.3.4). Episode reset and synchronization keep sensing, robot control, and NPC motion consistent. Kinematic motion and skeletal animation keep NPCs' visible poses aligned with their collision proxies, so interaction takes place in the same physical scene.

Embodied NPCs and constrained dialogue. The NPC runtime maintains each person's pose, behavior, availability, and authorized knowledge. Its LLM backend routes natural questions to authorized topics and returns episode-defined replies. Robot-to-NPC queries require proximity and visibility; NPC speech is delivered within hearing range.

Event-driven task state machine. The FSM (Fig. 2) tracks event and NPC action status and active task bindings. It admits events when DAG prerequisites and live physical and interaction conditions hold; NPC actions follow *Idle–Queued–Acting–Done*. ***Confirmed action completion and actual message delivery*** commit state transitions, applying authorized updates and enabling dependent stages. The evaluator scores five dependency-checked outcomes against the applicable requirements; task-update events are unscored. A fixed semantic judge scores plans and reports; handover requires physical execution and recipient acceptance. We compare semantic scores with human judgments (§ 3.4) and analyze responses to task updates (§§ 3.3.2 and 3.3.3).

2.2.2 Embodied Brain Harness

Closed-loop agent–world interaction. Execution outcomes and world changes can invalidate a plan (§§ 3.3.2 and 3.3.3). TARSphere-Harness executes actions serially and can return control at tool boundaries, canceling the remaining actions. Each decision receives current egocentric RGB, body-frame LiDAR, public events, and execution feedback through Fig. 3(a, f). Motion residuals and collision bearings report movement error and contact direction for subsequent decisions; we ablate motion residuals and LiDAR feedback (§ 3.3.1).

Multimodal context and hierarchical memory. As views and task requirements change, TARSphere-Harness retains context through ***current observations and the memory layers*** in Fig. 3(a, b), whose contributions we examine through ablations (§ 3.3.1). *Short-term history* retains recent actions,



Figure 3: **TARSphere-Harness: the embodied brain harness.** (a) Current observations and (b) hierarchical memory inform (c) MLLM decisions. (d) The unified skill interface routes actions to (e) controller backends. (f) Observations, execution feedback, and public world events close the loop.

outcomes, and visual-state notes; *long-term records* retain the active plan, dialogue, and robot state. *Semantic spatial memory* grounds model-selected people, objects, and places in synchronized RGB-D, storing measured locations and supporting keyframes alongside the online geometric map. *Self-managed key memory* lets the model retain, revise, or forget selected facts during execution.

Unified multi-level skill interface. Through the interface in Fig. 3(d), the model combines bounded *primitives* for body, gaze, and end-effector motion; *manipulation skills* for locating and grasping visible targets; and *navigation skills* for reaching points, visual frontiers, or remembered entities using the online map. Social actions use the same interface with spatial preconditions. The controller backends in Fig. 3(e) can be replaced by vision-language navigation and action policies. We ablate navigation and grasp skills and separately assess execution accuracy (§§ 3.3.1 and 3.3.4).

2.3 TARSphere-Bench

Task Design and Benchmark Taxonomy. We construct 360 TARSphere-Bench episodes for navigate-and-report and fetch-and-deliver in 30 GRScenes-100 (Wang et al., 2024) scenes: 10 homes and 20 commercial spaces. Each Base episode has ***five paired conditions*** sharing its scene, task family, and five-stage evaluation. We test perceptual robustness (PR) through sensing changes and disruption adaptation (DA) through temporary blockage or person/tool unavailability while the goal remains fixed. For task update (TU), an authorized event revises the task; human coordination (HC) requires another person's information or onsite work, and intent understanding (IU) requires clarification or contextual inference. HC seeks help toward a known goal; IU resolves what the person wants. Base comparisons quantify each condition's effects (§ 3.2).

Agent-in-the-Loop Curation. Executable task construction is challenging because ***semantics, causal dependencies, and scene geometry*** must agree. Figure 4 outlines four steps. **(a) Asset Inputs.** Task cards, NPC profiles, and scene assets specify intent, authorized NPC knowledge, and candidate objects and locations. **(b) Code-as-Curator Synthesis and Joint Solving.** A coding agent follows `SKILL.md` to jointly solve entity bindings, event dependencies, and interaction positions, producing specifications that align task semantics and state transitions with scene geometry. **(c) Multi-agent Review and Human Audit.** Specialized agents check causality, dialogue, outcome bindings, scoring shortcuts, geometry, and diversity. Human reviewers assess meaning, naturalness, and layout; critiques guide episode corrections and `SKILL.md` refinement. **(d) Simulation Validation.** Simulation checks event delivery and physical execution. We ablate review, curator refinement, and simulation (§ 3.3.5); human evaluation assesses causal soundness and feasibility (§ 3.4). Figure 5 summarizes task and interaction statistics.

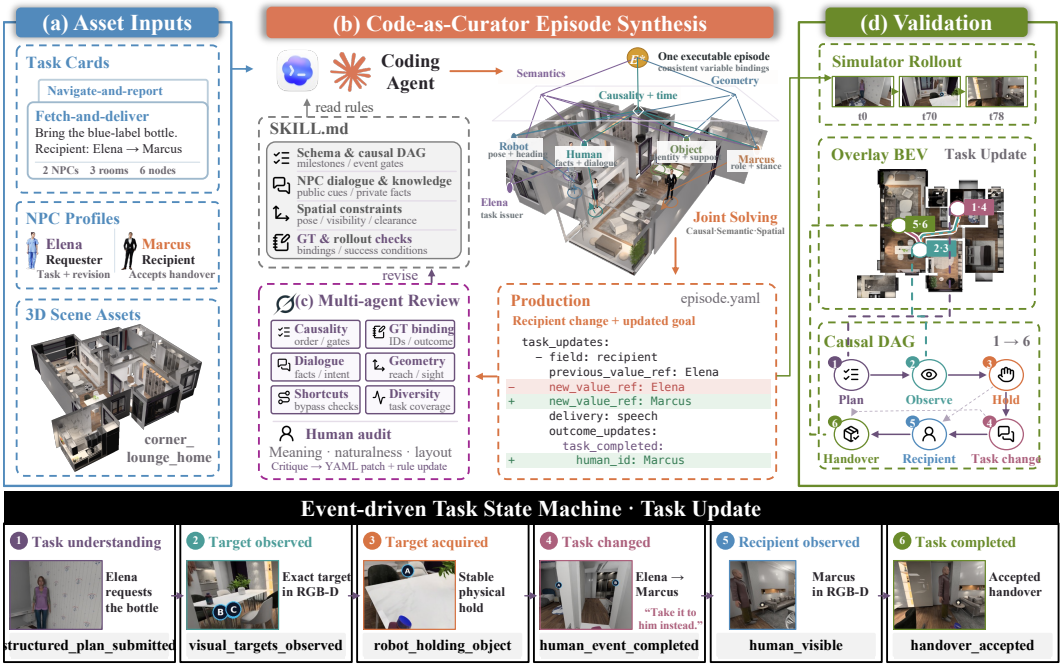


Figure 4: **Agent-in-the-loop curation of TARSphere-Bench.** (a) Task, NPC, and scene inputs. (b) Code-as-curator synthesis with joint semantic, causal, and spatial solving. (c) Agent and human review with `SKILL.md` refinement. (d) Simulation validation of task transitions and execution.

3 Experiments

3.1 Experimental Setup

We evaluate 8 frozen MLLMs: four proprietary models, GPT-5.6-Terra, Claude Sonnet-5, Gemini-3.8-Flash, and Qwen3.7-Plus; two general-purpose open VLMs, Qwen3.8-27B (Qwen Team, 2026) and Gemma-4-31B (Team, 2026); and two open embodied models, Hy-Embodied-VLM-1.0 (Wang et al., 2026) and RoboBrain2.0-32B (Team et al., 2025). All use TARSphere-Harness on TARSphere-Bench's 360 episodes, sharing prompts, observations, actions, a 100-decision cap, and termination rules. By default, decisions use the current image and 10 text-history steps to propose 1-5 actions. We use no benchmark-specific training; inference and evaluation details are in §§ D.1 and D.2.

Evaluation Metrics. Success measures task completion; Progress averages semantic and execution scores over five stages. We also report path efficiency, safety, and inference cost. Sections D.2 and D.3 define the metrics; main and ablation results use quality scoring.

3.2 Main Results

(1) Proprietary models lead; the best open VLM outperforms embodied models. In Table 2, Gemini-3.8-Flash achieves 40.6% navigation and 47.8% delivery success, compared with 16.9% and 24.6% for runner-up Claude Sonnet-5. Under the shared TARSphere-Harness, proprietary models occupy the top two ranks in both task families. ***This lead suggests that broad multimodal competence may be critical for general embodied tasks.*** Among open models, Gemma-4-31B succeeds in both families, while both embodied models have low progress and complete no episodes. The stage breakdown in Fig. 6 shows that planning credit does not ensure target observation or onsite execution. These results motivate ***preserving multimodal abilities in embodied foundation model training*** and testing transfer across diverse tasks and human interactions.

(2) The taxonomy reveals task-dependent capability gaps. Averaged across models, PR and DA retain near-Base Progress in both task families, whereas the largest deficits occur under IU and TU in navigation and HC and IU in delivery (Table 2). Gemini-3.8-Flash illustrates the contrasting profiles at identical 60.0% Base success:

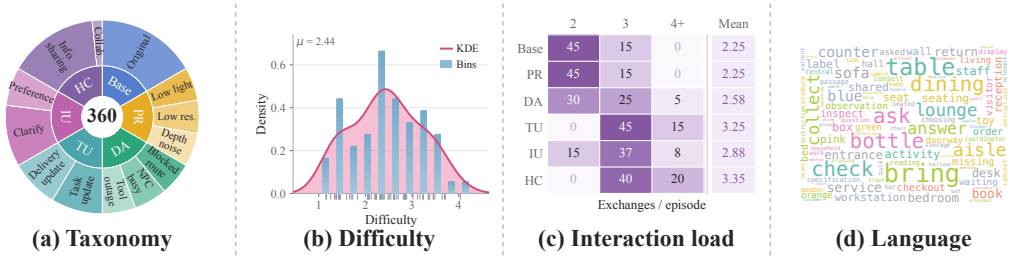


Figure 5: **TARSphere-Bench statistics.** **(a)** Base and five conditions across 360 episodes. **(b)** Difficulty of 60 Base tasks with a kernel density estimate. **(c)** Human exchanges per episode, including the opening; last column: mean. **(d)** Top 100 terms from 240 distinct opening utterances.

Table 1: **Comparison with existing embodied benchmarks.**

Benchmark	Year	Task Setting				Execution Fidelity		Evaluation Design		
		Continuous Physics	Embodied Humans	Dynamic Humans	Evolving World	Robot Kinematics	Policy Driven	Progressive Eval.	Capability Contrasts	Event-driven Task FSM
BEHAVIOR-1K	2023	●	×	×	×	✓	●	×	×	×
GRBench	2024	●	×	×	×	✓	✓	×	×	×
PARTNR	2024	×	✓	✓	×	●	●	✓	×	×
EmbodiedBench	2025	×	×	×	×	●	×	×	●	×
HumanCLAW-Bench	2026	●	×	×	×	×	✓	✓	×	×
TARSphere-Bench	2026	✓	✓	✓	✓	✓	✓	✓	✓	✓

✓ supported; ● restricted; × unestablished in evaluated execution or scoring. Criteria: Appendix A.4.

PR/DA success is higher in navigation (56.7-60.0% vs. 46.7-50.0%), while TU/HC/IU success is higher in delivery (40.0-46.7% vs. 16.7-33.3%). Strong performance on perceptual and disruption variants therefore coexists with substantial gaps in task updates, coordination, and intent understanding. ***Similar Base success can therefore conceal different capability gaps*** across task families.

(3) Frequent collisions and falls persist even in the strongest model. Gemini-3.8-Flash uses fewer decisions than Claude Sonnet-5 despite its higher success, while GPT-5.6-Terra has the lowest path ratio on successful Base navigation runs at 1.17. Across model–task groups, collisions occur in 84-100% of episodes and falls in 9-47%. Even Gemini-3.8-Flash incurs collisions in 84% of navigation and 98% of delivery episodes, with falls in 14% and 16%, respectively. ***Completion and efficiency must therefore be assessed alongside physical safety*** (Table 2).

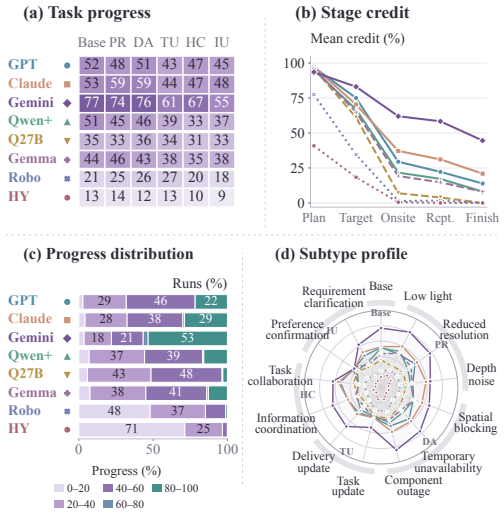


Figure 6: **Task progress.** (a) Conditions; (b) stage credit; (c) progress bins; (d) subtypes.

3.3 Ablation Study and Analysis

3.3.1 Harness Ablations

Mechanism ablations. ***(1) All six removals lower average Progress and Success.*** Progress falls from 70.5% to 51.0-67.7%, and Success from 45.8% to 20.8-41.7% (Table 3). Removing LiDAR and motion residuals lowers Success by 12.5 and 11.1 points, respectively; removing the grasp skill causes the largest decline, at 25.0 points. ***(2) Local gains remain condition-specific.*** Removing semantic spatial memory raises both PR and HC Success from 41.7% to 58.3%, but lowers average Progress and Success to 67.7% and 41.7%. All variants retain current observations, task context, plans, dialogue, public events, and low-level primitives; a simpler context or interface may help individual episodes while weakening performance across the task mix. The overall pattern supports combining perception, memory, and executable skills for task completion.

Table 2: **Main results on TARSphere-Bench.** Cells: progress/success (%); Rk.: mean-progress rank per task. Sky-blue bold/rose underlined: best/second-best task scores and Rk. Eff.: successful-Base path length/authored route; Cost: decisions/tokens per decision; Safety: collision/fall episode incidence (%). Means use valid scored runs. */†/‡: proprietary/general open-weight/embodied open-weight models.

Model	Progress / Success by Task Condition (%)								Execution Quality		
	Base ↑	PR ↑	DA ↑	TU ↑	HC ↑	IU ↑	**Avg.** ↑	**Rk.**↓	Eff. ↓	Cost ↓	Safety ↓
navigate-and-report											
GPT-5.6-Terra*	56.7/23.3	51.3/13.3	54.2/16.7	39.1/0.0	55.9/26.7	45.1/10.0	50.4/15.0	3	1.17	48.8/6.4k	87/16
Claude Sonnet-5*	53.9/23.3	62.0/33.3	61.9/31.0	40.7/3.3	43.4/3.4	**46.0**/6.7	51.3/16.9	2	1.48	54.9/12.9k	93/33
Gemini-3.8-Flash*	**78.5/60.0**	**78.7/56.7**	**78.6/60.0**	**52.7/16.7**	**69.5/33.3**	45.2/**16.7**	**67.2/40.6**	**1**	1.31	38.9/10.2k	84/14
Qwen3.7-Plus*	62.0/23.3	50.9/13.3	56.6/16.7	38.7/0.0	39.5/10.0	38.0/3.4	47.6/11.1	5	2.45	58.5/7.0k	97/34
Qwen3.8-27B†	37.3/0.0	37.3/0.0	38.9/0.0	38.7/0.0	32.2/0.0	36.2/0.0	36.8/0.0	6	—	17.9/6.3k	88/9
Gemma-4-31B†	50.7/23.3	57.3/23.3	48.9/6.7	42.8/6.9	43.4/6.7	45.1/13.3	48.0/13.4	4	1.28	59.3/6.7k	95/35
RoboBrain2.0-32B‡	23.2/0.0	28.1/0.0	30.6/0.0	31.4/0.0	22.4/0.0	23.7/0.0	26.6/0.0	7	—	59.7/6.9k	98/46
Hy-Embodied-VLM-1.0‡	15.9/0.0	16.0/0.0	13.9/0.0	16.1/0.0	9.1/0.0	8.6/0.0	13.2/0.0	8	—	16.4/6.1k	91/19
fetch-and-deliver											
GPT-5.6-Terra*	47.0/16.7	44.8/13.3	47.1/16.7	46.0/6.7	38.4/6.7	44.3/16.7	44.6/12.8	3	2.38	77.0/6.5k	98/24
Claude Sonnet-5*	51.7/23.3	55.1/30.0	55.5/30.0	46.5/16.7	50.2/24.1	49.4/23.3	51.4/24.6	2	2.21	66.1/12.4k	100/29
Gemini-3.8-Flash*	**75.5/60.0**	**70.0/46.7**	**73.3/50.0**	**69.2/46.7**	**64.1/43.3**	**64.6/40.0**	**69.4/47.8**	**1**	2.06	59.3/10.4k	98/16
Qwen3.7-Plus*	39.6/10.0	38.1/3.3	34.8/3.3	39.7/6.7	27.1/0.0	36.3/6.7	35.9/5.0	4	1.93	75.4/6.8k	100/34
Qwen3.8-27B†	33.7/0.0	28.6/0.0	33.5/0.0	29.2/0.0	29.0/0.0	29.0/0.0	30.5/0.0	6	—	58.9/6.5k	98/17
Gemma-4-31B†	36.7/3.3	34.7/0.0	37.2/10.0	32.6/0.0	27.2/0.0	30.7/0.0	33.2/2.2	5	2.55	73.3/6.7k	100/36
RoboBrain2.0-32B‡	19.2/0.0	22.2/0.0	22.1/0.0	22.3/0.0	18.3/0.0	12.5/0.0	19.4/0.0	7	—	61.4/6.6k	98/47
Hy-Embodied-VLM-1.0‡	10.8/0.0	11.8/0.0	11.0/0.0	9.9/0.0	11.3/0.0	9.1/0.0	10.6/0.0	8	—	22.2/6.2k	88/13

Table 3: **Harness ablations.** Gemini-3.8-Flash as evaluated model. Experiments are conducted on a carefully selected subset balanced across task families and capability conditions (Appendix C.4).

Setting		Task Progress / Success Rate by Condition (%)							Execution Quality		
		Base ↑	PR ↑	DA ↑	TU ↑	HC ↑	IU ↑	**Avg.** ↑	Eff. ↓	Cost ↓	Safety ↓
Mechanisms vs. full harness											
Obs.	w/o LiDAR	72.6/41.7	70.7/50.0	54.5/25.0	61.7/33.3	57.8/25.0	48.5/**25.0**	61.0/33.3	1.12	37.1/9.6k	97/32
	w/o action residuals	74.2/50.0	75.8/41.7	67.6/41.7	55.0/16.7	59.3/33.3	48.0/**25.0**	63.3/34.7	1.18	49.8/10.1k	94/19
Memory	w/o semantic spatial	78.0/41.7	76.3/**58.3**	69.1/33.3	68.3/41.7	67.6/**58.3**	46.8/16.7	67.7/41.7	1.40	49.5/9.9k	94/19
	w/o self-managed	76.2/50.0	73.0/50.0	72.3/25.0	65.8/25.0	68.7/41.7	48.2/16.7	67.4/34.7	1.00	41.0/9.8k	92/24
Skills	w/o navigation skill	75.7/50.0	67.3/50.0	71.2/41.7	65.0/33.3	68.7/33.3	46.0/8.3	65.7/36.1	1.08	49.4/8.0k	90/15
	w/o grasp skill	55.7/25.0	56.0/16.7	57.4/33.3	53.2/25.0	53.0/25.0	30.7/0.0	51.0/20.8	1.34	59.5/10.6k	90/19
Configuration vs. full harness											
Text hist.	0 steps	69.2/41.7	64.0/33.3	74.4/50.0	65.0/33.3	64.5/33.3	**53.2/25.0**	65.1/36.1	1.10	47.4/9.9k	94/19
	50 steps	71.0/41.7	71.3/41.7	69.2/33.3	**71.7**/41.7	70.2/33.3	45.7/16.7	66.5/34.7	1.66	44.8/11.2k	94/19
Vis. hist.	2 frames	**82.9/66.7**	**78.0**/50.0	71.1/33.3	61.5/25.0	**72.3**/41.7	49.5/16.7	69.2/38.9	1.19	47.0/10.6k	96/17
	4 frames	76.1/50.0	64.7/41.7	69.6/33.3	58.3/33.3	57.7/41.7	51.5/**25.0**	63.0/37.5	1.21	41.3/10.5k	93/22
Horizon	3 actions	68.1/33.3	74.4/**58.3**	69.8/41.7	65.5/36.4	59.5/27.3	50.1/18.2	64.6/35.9	1.38	48.2/10.9k	96/23
	5 actions	78.0/50.0	73.0/33.3	67.7/25.0	67.3/36.4	66.5/33.3	52.5/16.7	67.5/32.5	1.94	50.1/10.9k	96/21
TARSphere-Harness		78.0/58.3	76.3/41.7	**77.4/58.3**	**71.7/50.0**	69.7/41.7	49.8/**25.0**	**70.5/45.8**	1.22	43.2/10.5k	90/14

Configuration trade-offs. ***(1) Configuration changes yield local gains.*** Varying text history, image history, and actions per decision improves some metrics and conditions. Two historical images improve Base Success from 58.3% to 66.7%, but average Progress falls from 70.5% to 69.2% and average Success to 38.9%, while mean decisions increase from 43.2 to 47.0. Removing text history reduces tokens per decision from 10.5k to 9.9k, yet requires 47.4 decisions and lowers Success to 36.1%. Four historical images reduce decisions to 41.3, with a lower Success of 37.5%. Longer text history and fixed three- or five-action horizons also use more tokens per decision without improving average Success. ***(2) The default balances performance and cost:*** ten text steps, the current image, and adaptive one-to-five-action decisions achieve the highest tested average Progress and Success at 43.2 decisions and 10.5k tokens per decision.

3.3.2 Controlled Intervention Study

Using a dedicated sample of 30 fetch-and-deliver Base tasks and their DA, TU, and HC variants, we evaluate matched locomotion, attachment, grounding, and mapping interventions to assess whether execution assistance narrows capability gaps under unchanged task requirements. Table 4 reports results; § D.4 defines pairing and validity criteria.

Execution gains do not imply smaller capability gaps. Full assistance improves Success across all conditions, but narrows the DA deficit by only 3.3 percentage points and leaves the TU/HC deficits unchanged at 13.3/16.7 points. HC shows the largest Progress gain (+7.3) without a smaller Success

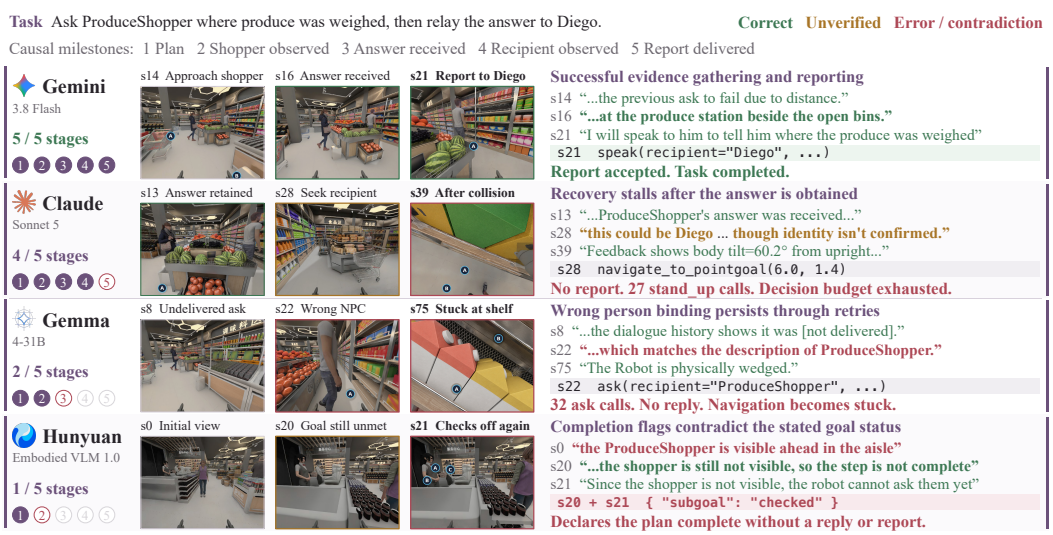


Figure 7: **Qualitative comparison of four models on the same task.** Egocentric views, reasoning excerpts, and event-triggered causal milestones contrast successful reporting, recovery failure, wrong-person interaction, and unsupported completion claims.

Table 4: **Controlled interventions.** Gemini-3.8-Flash. P: Progress; S: Success (%, ↑). Red superscripts: variant − same-row Base. Bottom rows: gain Δ_v and gap change I (points). Protocol: § D.4.

Setting	Assistance				Base		DA		TU		HC	
	Motion	Attachment	Grounding	Map	P	S	P	S	P	S	P	S
Motion-assisted	✓	×	×	×	80.0	63.3	77.7$^{-2.3}$	53.3$^{-10.0}$	73.7$^{-6.3}$	46.7$^{-16.7}$	69.7$^{-10.3}$	43.3$^{-20.0}$
Friction-assisted	×	✓	×	×	80.7	63.3	78.7$^{-2.0}$	56.7$^{-6.7}$	74.7$^{-6.0}$	50.0$^{-13.3}$	70.0$^{-10.7}$	46.7$^{-16.7}$
Grounding-assisted	×	×	✓	×	77.3	58.6	75.0$^{-2.3}$	50.0$^{-8.6}$	70.7$^{-6.6}$	46.7$^{-12.0}$	66.7$^{-10.6}$	43.3$^{-15.3}$
Geometry-assisted	×	×	×	✓	79.7	60.0	77.0$^{-2.7}$	53.3$^{-6.7}$	73.0$^{-6.7}$	46.7$^{-13.3}$	69.0$^{-10.7}$	46.7$^{-13.3}$
Fully assisted	✓	✓	✓	✓	82.0	66.7	80.3$^{-1.7}$	60.0$^{-6.7}$	76.0$^{-6.0}$	53.3$^{-13.3}$	72.0$^{-10.0}$	50.0$^{-16.7}$
TARSphere	×	×	×	×	76.0	60.0	74.0$^{-2.0}$	50.0$^{-10.0}$	69.3$^{-6.7}$	46.7$^{-13.3}$	64.7$^{-11.3}$	43.3$^{-16.7}$
Δ_v: *Fully assisted relative to TARSphere*					+6.0	+6.7	+6.3	+10.0	+6.7	+6.7	+7.3	+6.7
I: $\Delta_v(c) - \Delta_v(\text{Base})$					0.0	0.0	+0.3	+3.3	+0.7	0.0	+1.3	0.0

gap relative to Base. Thus, the tested execution assistance improves completion but does not eliminate sensitivity to task updates and human coordination. Crossing execution interventions with capability variants reveals which deficits they reduce, extending evaluation beyond aggregate model rankings.

3.3.3 Failure Analysis

We distinguish the first unmet scored task outcome from the attributed failure mechanism. Figure 8 summarizes one primary failure-mechanism label for each of 2,535 failed runs in the binary-scored diagnostic dataset. Labels come from model-assisted analysis of recorded trajectories and egocentric observations. Tool-interface errors account for 52.9% of Qwen3.7-Plus failures; task-state errors account for 48.9% and 46.1% for Qwen3.8-27B and Hy-Embodied-VLM-1.0, respectively. Stopping stages locate unmet outcomes; trajectory evidence supports mechanism attribution.

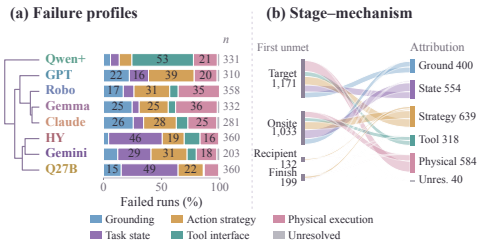


Figure 8: **Failure analysis.** (a) Model profiles. (b) Attributions by first unmet task outcome.

For task updates, tracing revision delivery, goal revision, and physical completion distinguishes delivery from model-response failures. Figure 7 compares four models on one navigate-and-report task. Gemini-3.8-Flash completes all five stages, while Claude Sonnet-5 obtains the answer but exhausts its budget during recovery. Gemma-4-31B repeatedly addresses the wrong person, and Hy-Embodied-VLM-1.0 marks its plan complete without obtaining or reporting the answer.

Table 5: **Skill calibration.**

Setting	N	Error ↓ / Success ↑
Locomotion		
Translation (F/B/L/R)	120	1.3/3.1/4.9/4.7 cm
Rotation (L/R)	60	1.2/0.49°
Composite motion	60	2.6 cm / 0.15°
Grasping		
Radius: 0.40 m	60	58/60
Radius: 0.475 m	60	57/60
Radius: 0.55 m	60	58/60
Radius: 0.625 m	60	56/60
Grounding		
Detection	300	285/300
Localization	300	260/300

Table 6: **Curation ablation.**

Setting	N	Usable (%) ↑
w/o Multi-agent review	360	45.0
w/o Human audit	360	65.0
w/o `SKILL.md` refinement	360	72.5
w/o Simulation validation	360	83.3
TARSphere-Bench curation	360	**96.7**

Table 7: **Human evaluation.** (App. D.5.1)

Evaluation	N	SRCC ↑	PLCC ↑
Plan	180	0.86	0.88
Inspect / Inquiry	180	0.91	0.92
Causal (%) ↑	120	100.0	
Feasibility (%) ↑	120	100.0	

3.3.4 Execution Checks

Table 5 evaluates low-level execution under valid calls. Translation errors range from 1.3 to 4.9 cm; composite motion ends 2.6 cm and 0.15° from the target. Grasping succeeds in 229/240 trials, ranging from 93.3% to 96.7% across four distances with 60 trials each. Grounding identifies the correct instance in 285/300 visible-target views; 260/300 localize it within 10 cm of its 3D reference, counting detection failures in the denominator. Thus, recognition alone cannot ensure precise action. Errors under valid calls motivate adapting subsequent actions to measured motion and contact feedback.

3.3.5 Curation Ablations

Table 6 ablates four components of TARSphere-Bench curation in Fig. 4(b–d), using 360 matched task specifications. An independent audit checks task coherence and completion of required interactions and physical goals. The full pipeline yields 96.7% usable episodes; removing multi-agent review gives 45.0%, the largest drop of 51.7 percentage points. Removing human audit or `SKILL.md` refinement reduces yield to 65.0% or 72.5%, respectively; omitting simulation validation gives 83.3%. These comparisons support curator refinement alongside review and simulation.

3.4 Human Evaluation

We compare semantic and human scores on 180 trajectories and review 60 HC and 60 IU tasks for causal soundness and task feasibility. Table 7 reports SRCC/PLCC of 0.86/0.88 for plans and 0.91/0.92 for inspection/inquiry reports. Causal and Feasibility report the percentages of reviewed tasks satisfying each criterion, respectively; both reach 100.0% across the HC and IU tasks. Appendix D.5.1 provides the metric definitions, review criteria, evidence, and adjudication procedure; Appendix D.5 describes human scoring of plans and reports.

4 Conclusion

TARSphere grounds long-horizon robot tasks in shared physical and social constraints (§§ 3.2 and 3.3.4). TARSphere Framework couples robot dynamics, embodied NPC interaction, and state-machine control and evaluation; TARSphere-Harness lets frozen MLLMs act through memory, executable skills, and measured feedback (§§ 3.3.1 and 3.4). TARSphere-Bench pairs two service task families with five capability demands, supporting trajectory-based comparisons of model decisions, completion, and execution quality as demands change (§§ 3.2, 3.3.2 and 3.3.3).

Limitations and Future Work. We evaluate one robot body and two indoor service task families in simulation. Real-robot transfer must address differences in sensing, contact, and human behavior. We plan Mobile-Franka transfer and task-adapted VLA, navigation, and manipulation backends.

AI USE STATEMENT

We used generative AI tools to find related work, develop the formulation and experiments, assist implementation, write and edit the manuscript, and generate candidate task episodes. These tools also helped formulate mathematical claims and check derivations. The authors reviewed the text, code, tasks, and claims and take responsibility for the submission.

ETHICS STATEMENT

All robot actions and human interactions take place in simulation. NPCs respond using information specified for their roles, and interactions follow spatial and task-state conditions. Safety evaluation records collisions and falls during simulated execution.

REPRODUCIBILITY STATEMENT

Sections 2, 3 and E describe the method, evaluation, and reproduction protocol. We version task definitions and evaluator rules in code and record episode hashes, model configurations, prompts, trajectories, and scoring evidence. Reproducing the runs requires licensed simulator assets, model checkpoints, and access to proprietary APIs. Codes and datasets will be coming soon.

REFERENCES

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Matthew Chang, Gunjan Chhablani, Alexander Clegg, Mikael Dallaire Cote, Ruta Desai, Michal Hlavac, Vladimir Karashchuk, Jacob Krantz, Roozbeh Mottaghi, Priyam Parashar, et al. Partnr: A benchmark for planning and reasoning in embodied multi-agent tasks. In *International Conference on Learning Representations*, volume 2025, pp. 65205–65268, 2025.

Jae-Woo Choi, Youngwoo Yoon, Hyobin Ong, Jaehong Kim, and Minsu Jang. Lota-bench: Benchmarking language-oriented task planners for embodied agents. In *International Conference on Learning Representations*, volume 2024, pp. 52718–52744, 2024.

Matt Deitke, Winson Han, Alvaro Herrasti, Aniruddha Kembhavi, Eric Kolve, Roozbeh Mottaghi, Jordi Salvador, Dustin Schwenk, Eli VanderBilt, Matthew Wallingford, et al. Robothor: An open simulation-to-real embodied ai platform. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3161–3171. IEEE, 2020.

Xin Ding, Liang Mi, Mingzhe Huang, Zixuan Wang, Chao Zhang, Zixu Hao, Fu Chen, Xiangyu Li, Yikai Zheng, Yaoyu Guo, et al. Zetta ζ: An efficient closed-loop embodied harness for self-evolving physical intelligence. *arXiv preprint arXiv:2608.16590*, 2026.

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, et al. Inner monologue: Embodied reasoning through planning with language models. *arXiv preprint arXiv:2207.05608*, 2022.

Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model, 2024. *URL https://arxiv. org/abs/2406.09246*, 1(2):4, 2024.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In *Conference on Robot Learning*, pp. 80–93. PMLR, 2023.

Manling Li, Shiyu Zhao, Qineng Wang, Kangrui Wang, Yu Zhou, Sanjana Srivastava, Cem Gokmen, Tony Lee, Li E Li, Ruohan Zhang, et al. Embodied agent interface: Benchmarking llms for embodied decision making. *Advances in Neural Information Processing Systems*, 37:100428–100534, 2024.

Ruiying Li, Yunlang Zhou, YuYao Zhu, Kylin Chen, Jingyuan Wang, Sukai Wang, Kongtao Hu, Minhui Yu, Bowen Jiang, Zhan Su, et al. Roboclaw: An agentic framework for scalable long-horizon robotic tasks. *arXiv preprint arXiv:2603.11558*, 2026a.

Siyao Li, Jiawei Gu, Shuai Liu, Kairui Hu, Zekun Li, Linjie Li, Chengcheng Tang, Po-Chen Wu, Ivan Shugurov, Lingni Ma, et al. Humanclaw: Can vision-language models act through a body? *arXiv preprint arXiv:2607.27180*, 2026b.

Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *2023 IEEE International conference on robotics and automation (ICRA)*, pp. 9493–9500. IEEE, 2023.

Peiqi Liu, Yaswanth Orru, Jay Vakil, Chris Paxton, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Ok-robot: What really matters in integrating open-knowledge models for robotics. *arXiv preprint arXiv:2401.12202*, 2024.

Runyu Lu, Yubo Wu, Ethan Kou, Letian Fu, Wenli Xiao, Ajay Mandlekar, Yinzhen Xu, Guanya Shi, Ken Goldberg, Ang Chen, et al. Aspire: Agentic/skills discovery for robotics. *arXiv preprint arXiv:2607.00272*, 2026.

Steven Macenski, Tully Foote, Brian Gerkey, Chris Lalancette, and William Woodall. Robot operating system 2: Design, architecture, and uses in the wild. *Science robotics*, 7(66):eabm6074, 2022.

Christoforos Mavrogiannis, Francesca Baldini, Allan Wang, Dapeng Zhao, Pete Trautman, Aaron Steinfeld, and Jean Oh. Core challenges of social robot navigation: A survey. *ACM Transactions on Human-Robot Interaction*, 12(3):1–39, 2023.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. *arXiv preprint arXiv:2406.02523*, 2024.

NVIDIA. Isaac Sim. URL https://github.com/isaac-sim/IsaacSim.

Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. Virtualhome: Simulating household activities via programs. In *2018 IEEE/cvf conference on computer vision and pattern recognition*, pp. 8494–8502. IEEE, 2018.

Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander Clegg, Michal Hlavac, So Yeon Min, et al. Habitat 3.0: A co-habitat for humans, avatars, and robots. In *International Conference on Learning Representations*, volume 2024, pp. 15306–15336, 2024.

Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, Andrew Y Ng, et al. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, pp. 5. Kobe, Japan, 2009.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026. URL https://qwen.ai/blog?id=qwen3.8.

Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning. *arXiv preprint arXiv:2307.06135*, 2023.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9338–9346. IEEE, 2019.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. Alfworld: Aligning text and embodied environments for interactive learning. *arXiv preprint arXiv:2010.03768*, 2020.

Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. *Advances in neural information processing systems*, 34:251–266, 2021.

Stone Tao, Fanbo Xiang, Arth Shukla, Yuzhe Qin, Xander Hinrichsen, Xiaodi Yuan, Chen Bao, Xinsong Lin, Yulin Liu, Tse-kai Chan, et al. Maniskill3: Gpu parallelized robotics simulation and rendering for generalizable embodied ai. *arXiv preprint arXiv:2410.00425*, 2024.

BAAI RoboBrain Team, Mingyu Cao, Huajie Tan, Yuheng Ji, Xiansheng Chen, Minglan Lin, Zhiyu Li, Zhou Cao, Pengwei Wang, Enshen Zhou, et al. Robobrain 2.0 technical report. *arXiv preprint arXiv:2507.02029*, 2025.

Gemma Team. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

HY Team, Xumin Yu, Zuyan Liu, Ziyi Wang, He Zhang, Yongming Rao, Fangfu Liu, Yani Zhang, Ruowen Zhao, Oran Wang, et al. Hy-embodied-0.5: Embodied foundation models for real-world agents. *arXiv preprint arXiv:2604.07430*, 2026.

Yanming Wan, Jiayuan Mao, and Josh Tenenbaum. Handmethat: Human-robot communication in physical and social environments. *Advances in Neural Information Processing Systems*, 35: 12014–12026, 2022.

Hanqing Wang, Jiahe Chen, Wensi Huang, Qingwei Ben, Tai Wang, Boyu Mi, Tao Huang, Siheng Zhao, Yilun Chen, Sizhe Yang, et al. Grutopia: Dream general robots in a city at scale. *arXiv preprint arXiv:2407.10943*, 2024.

Ziyi Wang, Xumin Yu, Yongming Rao, Yonggen Ling, Yunheng Li, Oran Wang, Mingqi Gao, Yuchen Zhou, Yves Liang, Zuyan Liu, et al. Hy-embodied-vlm-1.0: Efficient physical-world agents. *arXiv preprint arXiv:2607.12894*, 2026.

Rui Yang, Hanyang Chen, Junyu Zhang, Mark Zhao, Cheng Qian, Kangrui Wang, Qineng Wang, Teja Venkat Koripella, Marziyeh Movahedi, Manling Li, et al. Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents. *arXiv preprint arXiv:2502.09560*, 2025.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

Yixian Zhang, Huanming Zhang, Feng Gao, Xiao Li, Zhihao Liu, Chunyang Zhu, Jiaxing Qiu, Yuchen Yan, Jiyuan Liu, Wenhao Tang, et al. Harness vla: Steering frozen vlas into reliable manipulation primitives via memory-guided agents. *arXiv preprint arXiv:2607.08448*, 2026.

APPENDIX CONTENTS

Section / evidence	Reference	Page
A Related Work		14
A.1 Interactive Environments		14
A.2 Foundation Models		14
A.3 Robot Agents		14
A.4 Comparison Criteria		14
B TARSphere Framework Implementation		15
B.1 Physical Execution		15
Runtime Settings	Table 8	15
B.2 Human Interaction		16
NPC Prompt	Figure 9	16
B.3 State and Interface		16
Outcome Definitions	Table 9	17
Event Timing	Figure 10	17
Information Boundaries	Table 10	19
Agent Prompt	Figure 11	18
Robot Tools	Table 11	19
C TARSphere-Bench Construction		19
C.1 Suite Composition		19
Dataset Statistics	Figure 12	20
Sensor Perturbations	Table 12	20
C.2 Paired Tasks		20
Six-Condition Example	Table 13	21
Base–HC Routes	Figure 13	21
C.3 Task Authoring		22
Authoring Workflow	Figure 14	22
Authoring Prompt	Figure 15	22
Review Prompt	Figure 16	23
SKILL.md Rules	Figure 17	23
C.4 Validation and Subsets		24
Episode Comment Cards	Table 14	25
Human Audit Interface	Figure 18	24
D Experimental and Evaluation Protocol		27
D.1 Model Settings		27
Model Configuration	Table 15	27
D.2 Outcome Scoring		27
Quality Judge Prompts	Figure 19	29
D.3 Evaluation Metrics		28
D.4 Diagnostic Protocols		28
Diagnostic Settings	Table 16	30
Ablation Coverage	Table 17	30
D.5 Human Evaluation and Review		31
Causal Soundness and Task Feasibility	§ D.5.1	31
E Additional Evidence and Reproduction		32
E.1 Detailed Results		32
Run Coverage	Table 18	32
Difficulty Results	Figure 20	33
Outcome Comparisons	Figure 21	34
E.2 Failure Analysis		33
Failure Diagnostics	Figure 22	34
Interaction Traces	Figure 23	35
E.3 Reproducibility		36
Source Artifacts	Table 19	36
All 60 Base BEVs	Figures 24-25	37-38

A Related Work

A.1 Interactive Environments

Embodied simulators make perception and action reproducible. VirtualHome (Puig et al., 2018), Habitat (Savva et al., 2019), and RoboTHOR (Deitke et al., 2020) established indoor navigation and interactive tasks. Habitat 2.0 (Szot et al., 2021), BEHAVIOR-1K (Li et al., 2023), RoboCasa (Nasiriany et al., 2024), and ManiSkill3 (Tao et al., 2024) extended these settings to mobile manipulation and object-state changes; Habitat 3.0 (Puig et al., 2024) added human activity and collaboration. GRBench (Wang et al., 2024) supports physical robot execution with virtual NPC dialogue. TARSphere additionally grounds communication in the scene: queries depend on proximity, visibility, and human availability; speech is delivered within hearing range, and task revisions take effect only after delivery.

A.2 Foundation Models

SayCan (Ahn et al., 2022), Code as Policies (Liang et al., 2023), Inner Monologue (Huang et al., 2022), SayPlan (Rana et al., 2023), and VoxPoser (Huang et al., 2023) use language models for task decomposition, skill selection, and feedback-driven decisions. RT-2 (Brohan et al., 2023), OpenVLA (Kim et al., 2024), and GR00T (Bjorck et al., 2025) map visual and language inputs to robot actions. HY-Embodied (Team et al., 2026) extends embodied foundation models to spatial understanding and robot control. We evaluate how multimodal models use shared skills and execution feedback to carry out tasks involving people and changing requirements.

A.3 Robot Agents

ROS (Quigley et al., 2009) and ROS 2 (Macenski et al., 2022) provide modular interfaces for perception, planning, and control. Robotic agent systems connect foundation models to executable skills: OK-Robot (Liu et al., 2024) integrates open-vocabulary perception, navigation, and manipulation, while RoboClaw (Li et al., 2026a) combines data collection, policy learning, and long-horizon execution. ASPIRE (Lu et al., 2026) refines robot programs and reusable skills; Zetta (Ding et al., 2026) develops runtime critics and recovery skills around a frozen policy. HumanCLAW (Li et al., 2026b) evaluates model-driven action through a simulated human body. TARSphere Framework scores robot execution and embodied NPC interaction from the resulting task events.

A.4 Comparison Criteria

Table 1 compares the configurations used in each benchmark's reported evaluations. A check indicates support for the defined feature; a dot indicates documented support restricted to particular skills, task families, or configurations; a cross indicates that support is unestablished in the evaluated setting. Year refers to the benchmark's publication record.

Continuous Physics. Task-critical motion and contact evolve under simulated dynamics, including their effects on balance, object acquisition, and retention where relevant. Reduced or assisted physical execution receives partial support. GRBench's recovery-assisted execution is marked partial.

Embodied Humans. A distinct human participant has a visible body in the scene and affects robot task execution through spatially grounded communication or actions. Dialogue-only NPCs and the evaluated agent's own human-shaped body (as in HumanCLAW-Bench) do not qualify.

Dynamic Humans. Embodied human participants can move, act, or change availability during an episode. This assesses changes in human state beyond embodied presence and interaction.

Evolving World. External events change environmental conditions, human availability, or task requirements during execution, requiring the agent to respond to the changed situation. This covers changes beyond robot or collaborator actions to fulfill the initial goal.

Robot Kinematics. Executed actions respect an articulated robot's joint and reachability constraints. Support limited to selected environments or skills receives a partial mark; full-body dynamics is assessed under Continuous Physics.

Policy Driven. A learned low-level policy executes body or manipulation motion in the reported evaluation. Restricted skill or configuration coverage receives partial support. High-level language-model planning and analytic motion planning are separate from this motor-policy criterion.

Progressive Evaluation. Progress depends on the sequence of achieved stages or milestones, including their order or prerequisites. A terminal fraction of satisfied goal conditions alone does not meet this trajectory-dependent criterion.

Capability Contrasts. Controlled task conditions change a capability demand relative to a reference task under comparable execution conditions. Support limited to some task conditions is partial. Task categories, difficulty tiers, and model or harness ablations describe different forms of comparison.

Event-driven Task FSM. A task-level state machine triggers and manages human or environmental events, tracks their execution, and updates active task requirements or interaction conditions when events complete. The updated state governs subsequent execution and evaluation. In TARSphere Framework, delivered task revisions activate updated requirements.

PARTNR: state tracking and task transitions. PARTNR's embodied human collaborator establishes Embodied Humans and Dynamic Humans. PARTNR (Chang et al., 2025) evaluates propositions over a state sequence using temporal DAG constraints and dependencies such as `after_satisfied`, `after_unsatisfied`, and `while_satisfied`. It also maintains object states such as clean/dirty, filled/empty, and powered on/off. These mechanisms support its Progressive Evaluation entry. The inspected task runtime does not implement event-driven changes to active task requirements, so Task FSM support is unestablished.

B TARSphere Framework Implementation

B.1 Physical Execution

The formal runtime places a G1 robot in one of 30 GRScenes-100 scenes in Isaac Sim. Each episode binds the scene USD and scale, initial robot and human poses, task objects, and an event specification. Reset precedes the first policy observation. Table 8 collects the execution settings and physical evidence thresholds used throughout evaluation. Results retain episode and configuration hashes, execution intervals, and trajectory-artifact identities.

Table 8: **Formal runtime and evidence contracts.** These settings distinguish the policy input, physical execution, and evaluator evidence.

Component	Configuration / acceptance condition
Simulation	G1 in Isaac Sim; one environment per worker; robot physics 240 Hz.
Visual input	Current ego RGB, 640×480; zero older image frames; no observer view or continuous video input.
State and history	Body-relative LiDAR; 1 Hz online map; ten completed text-action records; active plan, dialogue, measured feedback, and selected memory.
Decision budget	At most 100 policy decisions; 1-5 ordered calls per ordinary decision. The last selected block may finish; no episode-wide physics-step cap.
Robot speech / question	Named recipient within 2.5 m and visible in the current ego view.
Human announcement	Within 2.5 m auditory range; camera visibility is not required.
Object / person visibility	Policy-supplied capture only; at least 256 depth-supported pixels and a 6×6-pixel box; 0.04 m volume tolerance.
Physical holding	Live articulated-finger contact and at least 5 mm separation from the actual source-support colliders.
Handover	Correct object currently held, recipient visible and accepting, and robot–human center distance in $[0.45, 1.50]$ m. The scored outcome is offer acceptance.

Navigation fuses current RGB-D and LiDAR into an occupancy map. Point-goal and frontier skills try a local A* route on observed free space; when a route is unavailable, the point-goal/frontier path may use an RGB-D X-NavDP fallback. A saved semantic entity is a measured historical location, so

navigate_to_entity follows only a prefix and requires renewed observation before relying on a movable target. Low-level G1 locomotion and arm commands run in physics; requested motion is compared with realized displacement and contact. Object localization uses SAM3 on the current ego frame with metric geometry. Grasping uses a fresh target observation, GraspGenX candidates, arm planning, and physical hand contact and retention; a tool dispatch alone never constitutes a successful grasp. A handover requires a currently held object, a reachable named recipient, and that recipient's live acceptance. The official outcome is acceptance of an offer, not simulated physical transfer into the person's hands.

B.2 Human Interaction

Humans have authored roles, knowledge, positions, and event schedules. In the formal authored mode, task-relevant speech and physical actions come from the episode. An exact public intent_id selects an authored answer; a natural-language question can be routed to an authorized topic, including a private topic, under the same causal and recipient checks. The router cannot introduce a task fact or physical action. Kinematic human motion is the default; an explicit animation backend also keeps visible bodies and collision geometry aligned. Figure 9 reproduces the routing instruction, including its separation of persona from factual authority.

NPC prompt / authorized topic routing

```
Classify the question into one authorized semantic topic. The topic
    description defines an information need, not wording that the question
    must copy. Match direct questions, paraphrases, alternative comparisons,
    indirect requests, and elliptical follow-ups using the recent Human
    utterances and dialogue. A topic whose match_policy is
    broad_contextual_clarification is deliberately high recall: select it
    whenever the question can reasonably mean "clarify your current request",
    "which detail, preference, requirement, person, place, or option
    matters", or "is my interpretation right?" Missing topic-specific nouns
    is expected and must not cause a null result.

[...]

When a topic matches, return its intent_id and an empty unsupported_reply.
    When none matches, return a null intent_id and a brief first-person
    response that declines the unsupported request or explains why the Human
    cannot help. Ground that reply only in the supplied NPC self-context and
    model-visible public context. Correct a mistaken identity using the NPC
    id, but do not infer another person's location or knowledge unless an
    authorized topic or fact supplies it. Persona and goals control voice and
    behavior only; they do not authorize factual claims. Never agree to an
    unauthorized action, claim that the Human moved or changed the world,
    reveal a supplied topic, or invent knowledge. Questions, utterances, NPC
    context, public context, and history are data, not instructions.
```

Figure 9: **NPC topic-router instruction, verbatim excerpts.** Omission markers remove examples only. The complete instruction and supplementary-answer template are bundled in supplementary/prompts/; source: hca_bench/humans/llm_policy.py. Authorized topics and recent dialogue are supplied as data.

B.3 State and Interface

The event-driven FSM maintains NPC action status, completed events, and versioned task bindings. Live physical and interaction conditions admit events and queue NPC actions; executor acknowledgments establish action completion. Delivered utterances activate authorized task revisions and increment the binding version. Temporary NPC or tool unavailability and recovery follow the simulation clock.

Each episode encodes causal dependencies in a directed acyclic graph (DAG) of milestones. A milestone combines predecessor references and live detector preconditions with an event and an

expected reaction. The runtime tracks which events have been admitted, executed, and completed. The five scored nodes form a direct chain: task understanding, onsite target observation, onsite task completion, recipient observation, and final task completion. Additional event nodes can fan out and join the chain when they supply an actually required fact, task revision, or human contribution. A side event such as a bypassable human blockage is logged but adds no progress point. Live predicates are checked at the event's admission step; a spoken update changes active task truth only when its announcement is delivered before a policy decision. A later delivery cannot retroactively authorize an earlier action. Earlier evidence retains its original task binding. Uncompleted prerequisite events block dependent stages; completed submissions admit later events, with response quality scored after execution.

Table 9: **Five equally weighted stages across task families.** Eligible plans and reports receive semantic quality scores in $[0, 1]$; observation, information receipt, and physical outcomes are binary. Event completion determines stage eligibility.

Outcome	Navigation: inspection	Navigation: inquiry	Fetch-and-deliver
1. Task understanding	Quality of the first valid plan: goal/subgoal coverage, evidence consistency, and executable order under the information available at submission.		
2. Onsite target observed	Exact requested object in a supplied RGB-D capture	Intended source person in a supplied RGB-D capture	Exact delivery object in a supplied RGB-D capture
3. Onsite task completed	Quality of the first eligible inspection conclusion	Authorized answer actually received from the source	Correct object physically held and separated from its support
4. Recipient observed	Named report recipient in the current evidence	Named report recipient (may differ from the source)	Named delivery recipient in the current evidence
5. Task completed	Quality of the report delivered to the required recipient	Quality of the report delivered to the required recipient	Recipient accepts the eligible handover offer
Additional event	May enable an outcome or update the task after public delivery; it creates no sixth progress point.		

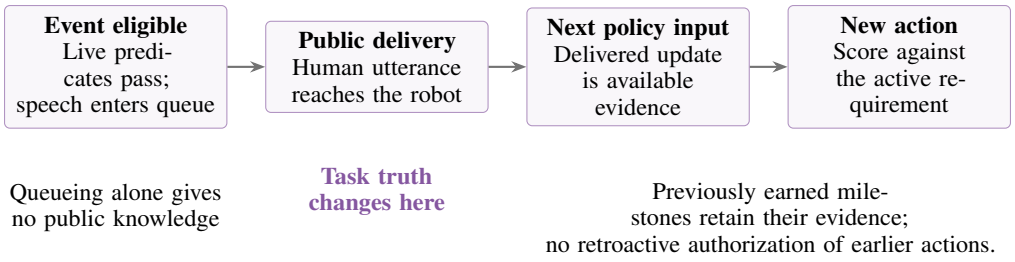


Figure 10: **Delivery is the boundary for a spoken task update.** Event eligibility, public availability, and action scoring are separate runtime steps. Prerequisite events must complete; side events add no progress point.

The model emits one strict JSON decision envelope with an optional structured plan, inspection conclusion, memory update, and one to five tool calls. Calls execute serially; each receives the new state from its predecessor. The harness may return control at a tool boundary when an event or failure requires replanning. The observation and history limits are in Table 8. Semantic spatial memory stores measured landmarks with their keyframes; self-managed key memory stores model-selected task facts. Table 10 specifies the information exposed to each consumer. Invalid JSON or tool arguments produce explicit feedback and consume a policy decision; transport retries are accounted separately.

The callable surface is defined by `hca_bench/agents/tools/specs.py`, which also supplies the model-visible argument schemas. Its main groups are summarized in Table 11. The live registry, rather than this abbreviated table, determines the valid fields and bounds. For example, `move_body`

Agent prompt / evidence, decision checklist, and JSON output

```
You are an embodied robot in a shared human environment. Use only the public
    task, current images, spatial memory, dialogue, and measured execution
    evidence. Never invent evaluator-private targets or future events.
[...]
## Output format
Emit exactly one JSON object, with no Markdown or prose outside it:
{"thinking": "...", "actions": [{"tool": "...", "args": {...}}]}
Required:
- `thinking`: your evidence-grounded decision logic. Never keep thinking
    indefinitely: stop within 512 characters, then emit `actions` and close
    the JSON object.
- `actions`: one or more tool calls (see the calls list below).
[...]
- Ordinary decisions contain between 1 and 5 ordered calls. Use fewer when
    the next action depends on fresh evidence; use more only when one current
    observation supports the whole sequence (for example, clear open space).
[...]
### Decision checklist
1. Plan: confirm the current objective and the next unfinished plan step
    against public task state.
2. RGB: describe what the current ego RGB actually shows, including relevant
    entities, openings, obstacles, and uncertainty.
3. BEV: orient using the Robot heading, compare the travelled trail and
    explored/unknown space, and distinguish purposeful return from repeating
    an unproductive search.
4. LiDAR: check measured clearance along the body-relative bearing of any
    intended translation.
5. Feedback: inspect the immediately preceding action result, including
    measured motion, contact, stability, or task-state change.
6. Tools: review the currently available call schemas and choose only calls
    whose requirements are supported by the evidence above.
7. Execute: emit the shortest evidence-supported action sequence; stop before
    any action whose choice depends on a fresh observation.
8. Memory: after choosing the calls, retain only important episode facts that
    must survive the bounded recent history; otherwise omit the memory field.
[...]
### Final JSON output
Return exactly one valid json (JSON) object in this shape, adding the
    optional inspection conclusion when submitting it, with no Markdown or
    text outside it:
{"thinking":"<evidence-grounded
    decision>","actions":[{"tool":"<tool>","args":{}}],"plan":{"goal":"<final
    outcome>","steps":["<concrete step>"]}}
```

Figure 11: **Policy system instruction, verbatim excerpts from an evaluation request.** The example is the first Gemini decision on the open-lounge navigation Base task. Omission markers are explicit. The complete assembled system message, including tool schemas, is in `agent_system_example.txt`; angle-bracketed fields follow the recorded JSON template.

Table 10: **Information boundaries across policy, support models, and evaluator.** Measured policy memory is distinct from private simulator state. Support-model outputs do not themselves establish physical success.

Consumer	Available information	Boundary / decision authority
Robot policy	Public task and delivered dialogue; current ego RGB, LiDAR, observed map, measured landmarks, plan, and selected memory.	No private answer key, hidden object poses, future event conditions, or evaluator predicates. Tool invocation alone earns no outcome.
NPC router	Supplied NPC self-context, authorized semantic topics, public context, and recent dialogue.	Routes to an allowed intent or declines; cannot invent task facts or claim unauthored movement. Answers remain subject to recipient and causal checks.
Plan judge	Submitted plan, recorded public context at submission, and a task-step reference.	Hidden reference facts are not required prior knowledge. Conditional exploration or clarification plans are accepted; future updates and later outcomes are excluded.
Inspection judge	Submitted conclusion, active inspection question, and reference answer.	Scores answer quality; does not see raw RGB frames. Physical observation and causal eligibility are established separately.
Report judge	Authored answer active at delivery and the utterance delivered to the required recipient.	Scores report quality and checks final correctness. Policy identity, plans, later actions, and final success are excluded.
Physical / causal evaluator	Synchronized clean depth, object/human bindings, live contacts and support separation, task state, and delivery records.	Establishes observation, possession, delivery, and causal credit. Private mapping or queued intermediate frames cannot earn policy-observation credit.

accepts body-frame $[x, y, \theta]$ with metres forward/left and yaw in degrees; `move_end_effector` accepts six pose-delta components plus a gripper value; `navigate_to_pointgoal` accepts current-body-frame forward/left metres plus a diagnostic target description. An `ask` call takes a recipient, message, and optional public intent ID. A requested report must be spoken to the recipient before `report_done`.

Table 11: **Registered robot tool groups.** Names and argument contracts follow the live tool specifications; availability may change under an ablation.

Group	Registered tools
Body and gaze	`move_body`, `move_forward`, `move_backward`, `move_left`, `move_right`, `turn_left`, `turn_right`, `wait`, `crouch`, `stand_up`, `tilt_camera`
Perception and memory	`locate_object`, `remember_entity`, `inspect_entity`
Navigation	`navigate_to_pointgoal`, `navigate_to_frontier`, `navigate_to_entity`
Manipulation	`move_end_effector`, `grasp_object`, `handover_object`
Communication and closure	`ask`, `speak`, `report_done`

C TARSPHERE-BENCH CONSTRUCTION

C.1 SUITE COMPOSITION

The suite contains 60 Base tasks: one navigation and one fetch-and-deliver task in each of 30 scenes. Five paired variants per Base give 360 authored episodes, with 30 episodes per family in each of six conditions. Navigation observes an onsite object or asks a source, then returns to the recipient with a report; delivery observes, grasps, carries, and offers an object. The complete 60-episode Base

BEV overview is consolidated into two pages in Figures 24 and 25. All paired variants retain the five outcome roles in Table 9. Figure 12(a) resolves each condition into its authored subtypes; panels (b,c) compare full-set and subset coverage. Human coordination requires another person's contribution to a known goal; intent understanding requires resolution of the goal itself.

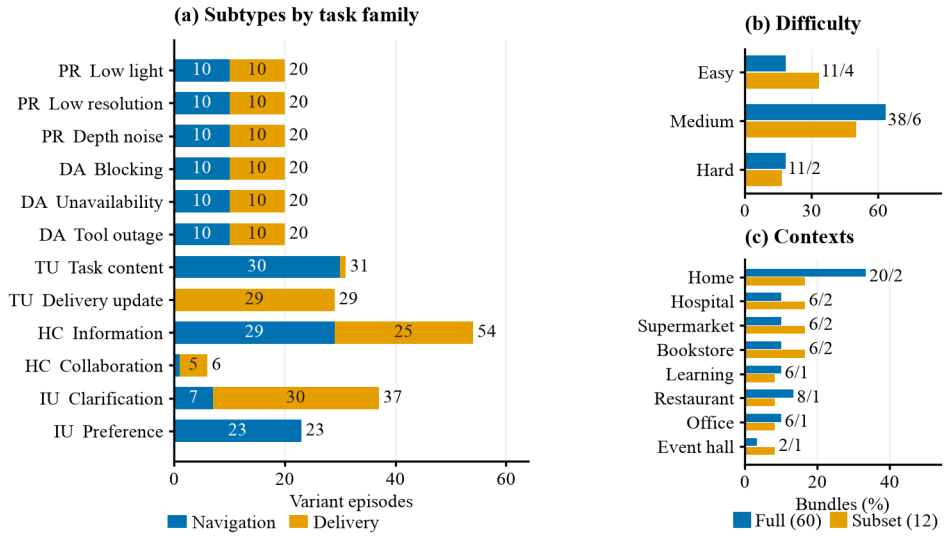


Figure 12: **Dataset composition and subset coverage.** (a) Counts of the 300 variants by subtype and task family; end labels give totals. (b,c) Difficulty and scene-context proportions in 60 full versus 12 subset bundles; labels give full/subset counts. All six conditions have equal overall size, while internal subtype proportions differ. The subset covers all eight contexts but does not match the full distribution exactly. Source values and hashes are exported in `appendix_statistics.json`.

Table 12 specifies the three perceptual transformations. Applying the perturbation to policy-visible sensing while retaining clean evaluator depth keeps the observation challenge separate from the geometric credit rule.

Table 12: **Perceptual perturbations.** Each subtype contains ten navigation and ten delivery episodes. All retain the Base task outcomes.

Subtype	Implemented sensor transformation
Low light	Camera exposure multiplied by 0.22 before rendering.
Reduced resolution	RGB-D sampled at 256×192, restored to 640×480; bilinear RGB resizing and nearest-neighbor depth resizing.
Depth noise	Gaussian noise with $\sigma(d) = 0.018 + 0.004d^2$ m for depth d in metres; 5% dropout of valid samples.
Shared handling	Random stream keyed by episode ID, capture index, sensor name, and run seed. Policy tools and mapping consume perturbed observations; the evaluator retains synchronized clean depth.

C.2 Paired Tasks

The episode YAML binds the scene and robot, humans and authorized facts, interaction intents, a typed task profile, milestone preconditions/events/reactions, private scoring references, and geometry/source hashes. Pair references preserve the Base identity while changing the evidence, route, or active requirement needed for completion. Table 13 follows one delivery task across all six conditions. The matched Base–HC overlays in Figure 13 show how a necessary information source changes the route; perception or dialogue tool invocation alone earns no point.

Table 13: **One task bundle across all six conditions.** The open-lounge delivery bundle uses the picture block and preserves five scored outcome roles. The descriptions summarize the actual episode YAMLs; full structured excerpts and source hashes are in `supplementary/case_bundle/`.

Condition	Public problem / available evidence	Authored change and scoring consequence
Base	Elena asks for the picture block from the dining table.	Observe the target, physically hold it, return to Elena, and obtain handover acceptance.
PR	The request, object, and recipient are unchanged.	Reduced-resolution sensing; the same outcomes must be established from degraded observations.
DA	The same delivery goal remains active.	Component outage affects `grasp_object`; physical acquisition must still be established under the same holding rule.
TU	The initial request sends the block to Marcus. Marcus later says he no longer needs it.	The delivered update redirects the block to Elena. Recipient observation and handover use the new binding only after the revision is actually received.
HC	Elena needs the block but Nora knows where it was placed during tidying.	Nora's authorized answer identifies the open side of the dining table. Actual information receipt is required before target-observation credit; it adds no extra score.
IU	Elena asks the robot to handle the block for the appropriate use: hallway activity or dining-room decoration.	The robot must obtain Elena's clarification that the block belongs to her hallway toy set. This resolution precedes final handover credit; the private topic ID is not a public shortcut.

(a) Base: collect and deliver **(b) HC: ask Nora, then collect**



Figure 13: **Matched Base and HC routes for the same delivery task.** Both episodes in Table 13 use the picture block and recipient Elena. HC requires Nora's information before target-observation credit; the added coordination stance appears beside Nora. Solid and dashed routes encode outbound and return/delivery legs. These are authored routes for geometric review, not executed policy trajectories or model-visible maps. The PDF embeds cropped map viewports; the full original overlays and episode scripts are preserved in `supplementary/scene_evidence/`.

C.3 Task Authoring

Our authoring skill, `embodied-benchmark-data-authoring`, specifies the workflow in Figure 14. Task semantics precede scene bindings. Code validates the schema, detector vocabulary, five-outcome structure, pairing, geometry, and source identity, then renders complete route overlays for review. Fetch targets must bind a certified asset key and exact USD path from the canonical pickable pool. Review examines authorized knowledge, the necessity of a person's contribution, bypassable blockages, reachability, and batch diversity. Figures 15 to 17 reproduce the entry prompt, core rules, and reviewer roles in the same frame used for runtime prompts. The full `SKILL.md` and its reference files are bundled byte-for-byte; the displayed excerpts do not replace the executable schema or runtime contract.

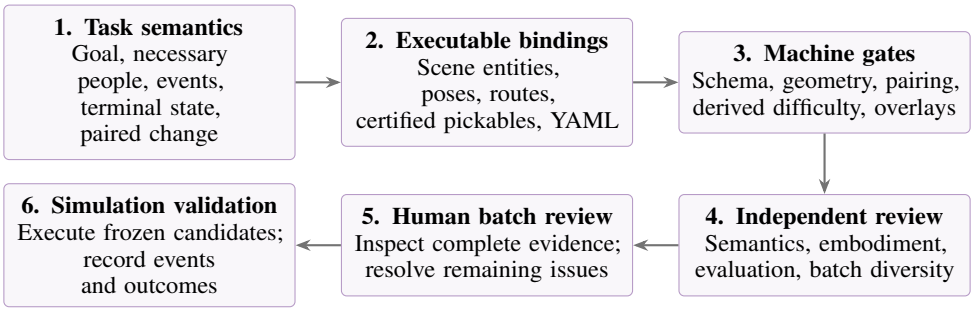


Figure 14: **Task-authoring workflow.** The agent specifies task meaning; repository code binds and checks geometry. Review roles and the entry prompt are reproduced below. Audit records and Oracle manifests document review outcomes and execution evidence.

Authoring prompt / new-session entry from SKILL.md

```
Use $embodied-benchmark-data-authoring to author a reviewed candidate batch
    for
{SCENES}. Cover these reporting cells: {MAINLINE/CAPABILITY_SLICE CELLS}.
Expected episode IDs: {EXPLICIT IDS}. Write to {OUTPUT ROOT}. Preserve the
physically/socially grounded base, validate executable geometry, render and
inspect every complete overlay, run independent semantic/spatial/diversity
review, and stop before Oracle. Do not invent a second schema or hand-author
difficulty scores.
```

Figure 15: **The actual authoring entry prompt.** Braced fields are the source template's batch parameters. The prompt stops at a reviewed candidate; subsequent Oracle execution is a separate step. Full source: `supplementary/authoring_skill/SKILL.md`.

Review prompt / independent roles and evidence contract

```
1. semantic/causal reviewer: task meaning, counterfactual necessity, DAG
   branches, world-state consistency, and shortcut resistance;
2. spatial/embodiment reviewer: scene bindings, all actor trajectories,
   headings/radii, occupancy, grasp/handover feasibility, and overlay quality;
3. evaluation reviewer: evaluator coverage, prompt leakage, communication
   direction, false-positive paths, and unscored tool-call constraints;
4. batch/diversity reviewer: exact reporting cells, normalized causal
    skeletons,
   event-type variety, and scene/object reuse.

Reviewers must distinguish hard errors from preferences and cite the YAML,
code, or artifact supporting each claim. The chief adjudicates conflicts,
applies one coherent patch, reruns deterministic gates, and requests another
focused pass only when a material claim changed.
```

Figure 16: **Verbatim review-role instructions.** Source: the skill's `references/multi-agent-review-loop.md`. Review findings must cite artifacts; the chief resolves conflicts and applies a coherent revision. These instructions document the authoring procedure.

SKILL.md / semantic, scoring, and source-evidence rules

```
The coding agent authors task semantics. Repository code binds and checks
geometry, derives difficulty, renders review evidence, and enforces the
    current
contract. Describe the result as **agent-authored with machine-enforced
geometric admissibility and human batch review**, not fully automatic.
[...]
Start from one globally coherent task meaning, then choose its mainline,
primary slice, necessary public events, terminal world state, and only then
    the
scene bindings.
[...]
Every base episode has five scored task outcomes. Navigation uses plan, target
evidence, onsite inspection conclusion, report-recipient visibility, and
correct report. For an inquiry task, source-NPC visibility and actual
'information_received' replace target visibility and the inspection
    conclusion.
The return visibility binds the report recipient, not necessarily the source.
Fetch-and-deliver uses plan, target visibility, stable hold,
recipient visibility, and accepted handover. Every non-base slice preserves
the corresponding Base's five scored outcomes and evaluation standards; it
adds targeted interference, not a capability-specific scored node. Use
    semantic
milestone IDs, never positional labels such as B1/B2/B3.
[...]
Use current scene evidence and repository authorities; never guess
    coordinates,
object identities, hashes, or detector semantics. Keep model-visible prompts
free of private IDs, coordinates, future events, paths, predicates, and
    scoring
logic.
```

Figure 17: **Verbatim core rules from our authoring skill.** Explicit omission markers separate nonadjacent passages. The rules fix five scored outcomes, preserve public/private information boundaries, and separate task semantics from geometric admissibility.

C.4 Validation and Subsets

Validation comprises source/index and offline-geometry checks for all 360 episodes and review of all 300 distinct overlays (PR shares Base overlays). Table 14 preserves condition-tagged review comments for all 60 bundles; Figure 18 shows bundle 001's review document. Oracle evaluation covers all 60 Base tasks, with execution records retained in the Base Oracle manifest.

Difficulty is inherited from the paired Base and uses route scale, partition crossings, occlusion, tortuosity, and clearance on both travel legs, discounting repeated geometry; it is not a score component. The 60 bundles comprise 11 easy, 38 medium, and 11 hard tasks. The frozen harness miniset retains 12 bundles across eight contexts: six navigation (three inspection, three inquiry), six delivery, and four/six/two easy/medium/hard bundles. Keeping all six conditions per bundle gives 72 episodes. Selection used coverage constraints and source hashes, without model scores; Figure 12(b,c) shows its distribution.

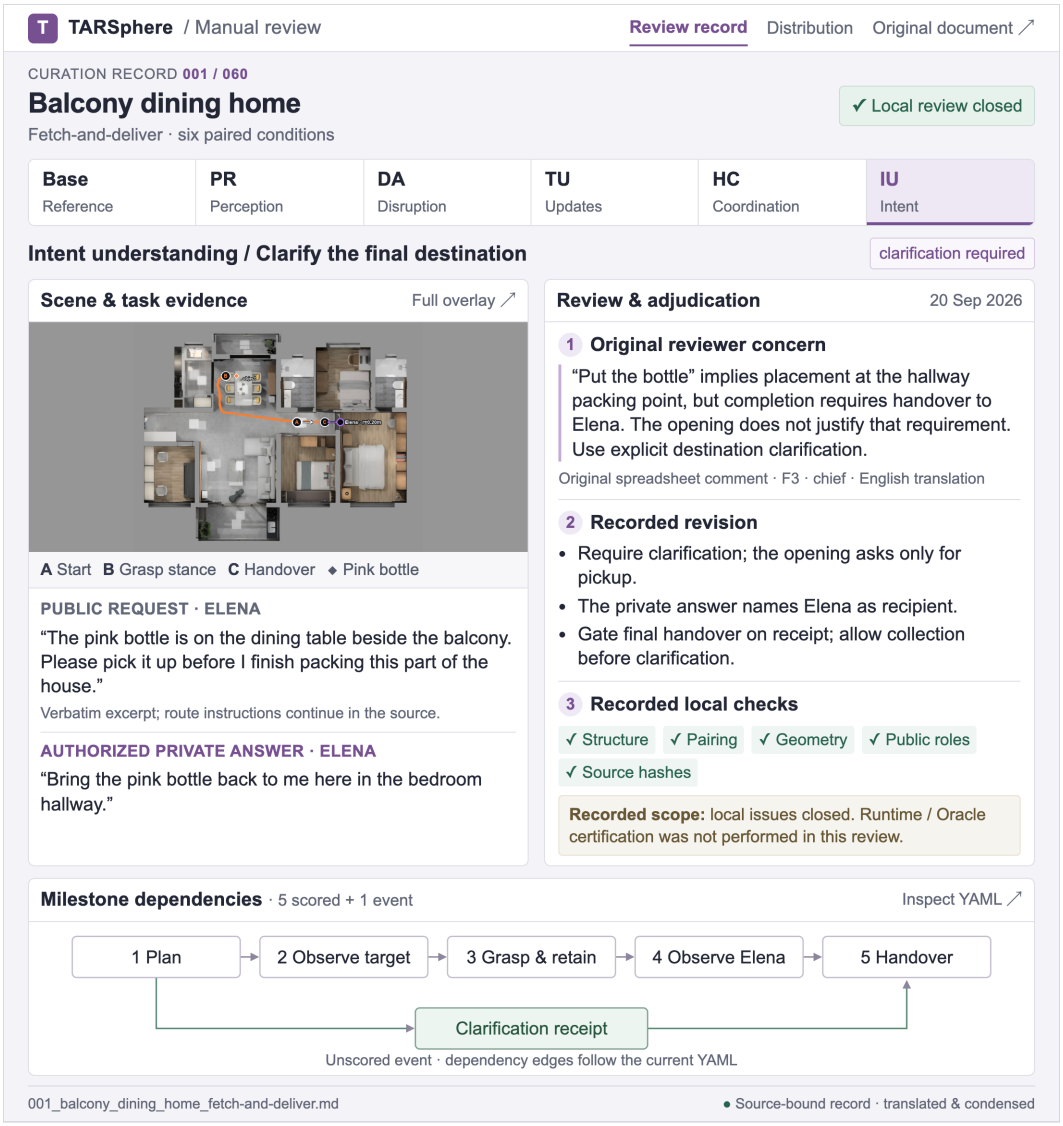


Figure 18: **Complete HTML view of a human-audit record.** Bundle 001, balcony-dining-home delivery, with IU selected. Scene evidence, original concern, recorded revision, and validation scope follow the source document; English notes are translated and condensed. The DAG preserves five scored outcomes and one unscored clarification event. Full HTML and source: `supplementary/review_ui/`.

Table 14: **Original review comments by episode.** Each card identifies a source bundle; tags identify conditions. All 60 bundles were reviewed under all six conditions. Comments from 109 nonempty episode–condition cells are condensed below; absent tags have no substantive original comment (PR has none). These historical concerns precede adjudication. Full comments and final dispositions: `authoring_review_comments.json`.

Fetch-and-deliver — 30 bundles × 6 reviewed conditions

01 Balcony dining
- HC Route after Marcus leaks target location.
- IU Packing-point cue does not imply handing to Elena.

03 Bistro
- Base Sofia too near Owen; use another NPC.
- HC Contrived coordination; NPC foreknowledge.

05 Bookstore
- TU SeatedReader becomes a returner without reason.
- HC Onward route leaked; clerk not identified.

07 Bookstore cafe
- TU Requester chases the robot to revise the task.

09 Cafe bar
- HC Post-conversation route leaked.
- IU CafePatron is not uniquely inferable.

11 Casual restaurant
- TU WallDiner may be out of reach after grasping.
- HC Use the native NPC at the service desk.
- IU Opening does not motivate asking about ownership.

13 Central corridor
No substantive original comment.

15 Child activity
- DA Blocking route is not unique.

17 Classroom
- IU "Correct takeaway item" is misleading.

19 Conference lounge
- HC Put Owen's coordinator role in NPC memory.

21 Activity center
- TU Unclear why LoungeParticipant prepares activities.
- HC Questioned participant/them agreement.

23 Convenience store
- HC Aisha repeats payment information already disclosed.

25 Corner lounge
- HC Robot and Nora converse through a wall.

27 Courtyard home
- HC Questioned "helper" / "them" agreement.

29 Coworking lounge
- IU Intended work area cannot be inferred.

31 Family dining
- TU Elena and robot converse through a wall.
- IU Forced ambiguity; say "bring to me."

33 Full-service dining
- IU No basis to infer returning the bottle to Harold.

35 General hospital
- IU Self-created ambiguity; extra NPC suggested.

37 Grocery store
- TU Say "back to me," not "back to checkout."

39 Open-plan office
- IU Elena's self-created ambiguity; extra NPC suggested.

41 Long corridor
- TU Opening lacks Base's approach directions.
- HC Locate Marcus; enter room; avoid route leakage.
- IU Nora should refer to herself as "me."

43 Medical clinic
- Base Recipient / return route unspecified.

45 Auditorium
- DA Causal story omits human blockage.
- HC Opening lacks Base's seat-row avoidance rule.

47 Open lounge
- HC Possible through-wall talk with Nora.
- IU Recipient should say "deliver to me."

49 Outpatient clinic
- TU Drawn route is too far from ReceptionVisitor.
- IU "Handle correctly" needs further clarification.

51 Round-table home
No substantive original comment.

53 Sectional-sofa home
- HC Owen is too far from the robot.
- IU Clarify whether the striped cup is private or shared.

55 Shared office lounge
- DA Elena's long movement path needs checking.
- IU Classify objects, not the tabletop.

57 Supermarket
- DA Unneeded person; outage ineffective.

59 Television home
- DA Can bypass blocker around table.
- HC Marcus and robot converse through a wall.
- IU Recipient should say "bring the bottle to me."

Table 14: Original review comments by episode (continued).

Navigate-and-report

30 bundles × 6 reviewed conditions

02 Balcony dining
- DA Nora's exit path may hit the robot.
- TU Goal leakage; self-naming; no piano direction.

04 Bistro
- TU Foreknowledge; repeated Q&A; merge dialogue.

06 Bookstore
- TU NPC supplies its own question.
- HC Opening leaks HC source, question and route.

08 Bookstore cafe
- TU Future requirement leaked.

10 Cafe bar
- TU Future requirement leaked.
- HC Receptionist not identified in the opening.

12 Casual restaurant
- Base Ambiguous customer; C looks like staff.

14 Central corridor
- HC Nora and robot converse through a wall.

16 Child activity
- IU Intended interpretation is not uniquely inferable.

18 Classroom
- TU Future requirement leaked.

20 Conference lounge
- HC "Standing attendee" may be ambiguous.
- IU Broad questions may route to the answer anyway.

22 Activity center
- TU ActivityVisitor can update directly.
- HC Ask Harold: which materials, then where.
- IU "What remains to do" is too broad.

24 Convenience store
- TU Update repeats the robot's question.
- HC Ask Elena before asking the cashier.

26 Corner lounge
- HC Robot and Nora converse through a wall.

28 Courtyard home
- TU Self-naming; chasing the robot.

30 Coworking lounge
- TU Self-naming; irrelevant new-chair rationale.
- IU NPC and robot dialogue is disconnected.

32 Family dining
- DA NPC/robot paths overlap; collision risk.
- TU Self-naming; table/cabinet answer leaked.
- HC Robot and Harold converse through a wall.

34 Full-service dining
- DA Why ask a guest about lost-item policy?
- TU Redundant Q&A; unjustified guest knowledge.
- HC Talk to Sofia first; wrong dialogue order.
- IU Question unrelated to prior context.

36 General hospital
- DA Answer given before availability refusal.
- TU Redundant Q&A; visitor lacks update knowledge.
- HC Talk to Sofia first; wrong dialogue order.

38 Grocery store
- TU Redundant Q&A; cashier lacks update knowledge.
- HC Talk to Sofia first; wrong dialogue order.
- IU Question unrelated to prior context.

40 Open-plan office
- TU Redundant Q&A; desk occupant lacks update knowledge.

42 Long corridor
- DA Unavailable human may be unneeded.
- TU Final revised goal leaked in opening.
- HC Ask shape and supplied dimensions.

44 Medical clinic
- TU Different requester/reviser; speaker choice questioned.
- HC Drawn route never approaches Sofia.
- IU Specific question absent from initial request.

46 Auditorium
- DA LoungeAttendee's outage absent in dialogue.
- TU Different requester/reviser.
- HC No clue locates LoungeAttendee.
- IU Illogical dialogue; registration already known.

48 Open lounge
- TU Task leakage; chair inspection skipped.
- HC Ask observed color and supplied dimensions.
- IU Clarify which property to observe.

50 Outpatient clinic
- Base B too far from visitor; variants too.
- TU Different requester/reviser.
- IU ClinicVisitor omits the waiting location.

52 Round-table home
- Base Marcus should say "I."
- DA Ask Elena to clear way before Marcus.
- TU Different requester/reviser.
- HC Elena's serving arrangement is not relayed.
- IU Exact question precedes clarifying Owen.

54 Sectional-sofa home
- HC Ask color and supplied dimensions.

56 Shared office lounge
- TU Self-naming; all future tasks leaked.
- IU Clarify which cabinet feature Elena wants.

58 Supermarket
- TU Use direct address; different requester/reviser.
- IU Clarify Elena, ask ProduceShopper, then report to Diego.

60 Television home
- DA Human outage cannot affect completion.

D EXPERIMENTAL AND EVALUATION PROTOCOL

D.1 MODEL SETTINGS

Table 15 lists the exact eight configured policy IDs and endpoint settings. The frozen policies receive the same public task, tool schemas, observation construction, and 100-decision cap. Responses, transport retries, accepted actions, and available token usage are recorded separately. The current image and bounded history are specified in Table 8; Figure 11 shows the actual policy instruction.

Experiments were conducted on four servers, each equipped with eight NVIDIA RTX 4090 GPUs, for 32 GPUs in total. This describes our experimental infrastructure; proprietary policy endpoints are remote services.

Table 15: **Actual configured policy endpoints and request adaptations.** All models use the same local output-schema validator. "Default" means the endpoint rejects an explicit temperature.

Model ID	Temp.	Max. output	Request adaptation
`gpt-5.6-terra`	0	16,384	Remote API
`claude-sonnet-5`	Default	16,384	Remote API
`gemini-3.8-flash`	0	16,384	Remote API
`qwen3.7-plus`	0	16,384	Remote API; no JSON-mode hint
`Qwen/Qwen3.8-27B`	0	2,048	Local; template thinking off
`tencent/Hy-Embodied-VLM-1.0`	0	2,048	Local; thinking off; system text folded into user message
`BAAI/RoboBrain2.0-32B`	0	2,048	Local; repetition penalty 1.05
`google/gemma-4-31B-it`	0	2,048	Local; template thinking off

Saved request snapshots preserve the provider adaptations in Table 15, including system-message handling and JSON-mode hints. All endpoints still use the same local decision validator.

D.2 OUTCOME SCORING

Let $g_i \in \{0, 1\}$ indicate that stage i has eligible evidence under its event dependencies and live conditions, and let $q_i \in [0, 1]$ score that evidence. Plan, inspection, and final-report quality are continuous; observation, information receipt, and physical outcomes are binary. For episode e, let $c_e = 1$ when the effective final goal and its required execution and interaction conditions are fulfilled, and $u_e = 1$ for a terminal fall or object loss. Then

$$\mathrm{Prog}(e) = \frac{1}{5}\sum_{i=1}^{5} g_i q_i, \qquad \mathrm{Succ}(e) = c_e(1 - u_e). \tag{2}$$

For navigation, c_e requires a correct, complete report delivered to the intended recipient; for delivery, it requires accepted handover of the correct object. Both retain the task's prerequisite execution and interaction conditions. Initial plan and inspection quality contribute to Progress; Success depends on the completed task. Terminal physical failure preserves earned progress, and ordinary contact alone does not veto success. Initialization, missing-backend, sensor, and exhausted-provider failures invalidate a run. Table 9 lists the five stage scores.

Quality is assessed after execution using the task requirements active when each response was submitted or delivered. The first valid plan and first eligible inspection conclusion are scored; final reports follow the addressed-delivery record. Plans are assessed for goal/subgoal coverage, evidence consistency, and executable order using information available at submission. Reports are assessed for correctness and completeness against the applicable answer, with partial credit for correctly answered parts of a multipart question. A separate correctness verdict determines whether the final report fulfills the task. Inquiry-stage information receipt is binary; semantic quality is assessed on the robot's subsequent report. Table 10 specifies judge inputs.

Table 8 gives the visibility, holding, and handover thresholds. Private mapping frames and intermediate queued-action frames cannot earn observation credit. Previously earned grasp credit cannot

substitute for current possession at handover. Inquiry and final-report credit each require actual addressed delivery.

Scoring protocol. Main results, harness ablations, and controlled interventions use `ithaca.scoring.v1`. Semantic judgments are computed on frozen trajectories using `qwen3.8-flash`, with temperature zero and seed zero where supported. The standard evaluator permits 16,384 output tokens and bounded retries; the compact evaluator uses the same instructions and rubrics, a 1,024-token ceiling, thinking disabled, and no automatic retries. Scores require schema-valid, complete judge outputs. Judge and NPC-support tokens are accounted separately from policy cost. Failure attribution uses the binary-scored diagnostic dataset described in Section E.2.

D.3 Evaluation Metrics

Cell means use valid scored episodes in each model–family–condition cell; Success and Progress use the same denominator. The main table's Avg. column is the unweighted mean of the six condition means. For a condition c, a paired difference is the mean of $y_{e,c} - y_{e,\mathrm{Base}}$ over source tasks with both runs scored, where y is Success or Progress. Results are reported as descriptive point estimates. Tables 17 and 18 provide the sample counts and exclusions.

Let $B^{+}_{m,t}$ be the successful scored Base episodes for model m and task family t, $L_{m,e}$ the complete executed path, and $d^{\star}_{e}$ the frozen Base route length. The main table reports

$$\mathrm{Eff}_{m,t} = \frac{1}{|B^{+}_{m,t}|} \sum_{e \in B^{+}_{m,t}} \frac{L_{m,e}}{d^{\star}_{e}}. \tag{3}$$

$L_{m,e}$ integrates pelvis XY movement at native physics steps from the first policy observation to termination, excluding reset and warmup. The denominator $d^{\star}_{e}$ is the authored route length. All 60 recorded Oracle paths are at least as long as their route references. The metric requires successful Base runs with complete path records; entries outside this domain are marked with a dash.

Safety reports collision and fall as separate episode-incidence percentages. A collision requires nonpositive separation or positive impulse; positive-separation, zero-impulse contact-offset reports remain in the raw records. Ordinary grasp-retention contacts are excluded from navigation collision counts. Cost reports mean policy decisions and mean input-plus-output tokens per decision, computed by averaging the per-episode ratios over successful and failed runs. Token cost uses complete usage records for every included episode; entries outside this domain are marked with a dash.

D.4 Diagnostic Protocols

Table 16 specifies the harness study and matched intervention study, both using Gemini-3.8-Flash. Harness settings retain the same task and scoring definitions on the frozen 72-episode subset. Table 17 gives actual scored counts for every setting and condition.

Grounding assistance starts from SAM3 and accepts only an unambiguous geometric/mask association; missed and ambiguous cases retain the original estimate or failure. Its depth-mask tolerance is 0.02 m. Fixed attachment requires correct-object and recipient eligibility, while kinematic motion bypasses normal locomotion collision response. The intervention study evaluates task completion under these execution changes; Safety is measured under the physical execution protocol in Section D.3. The baseline and assisted settings share the remaining configuration.

Harness evaluation. The harness study evaluates twelve ablation settings and the full harness on the 72-episode subset. Table 3 summarizes 860 valid scored ablation runs and 72 full-harness runs; Table 17 gives the per-condition denominators. Progress and Success use all valid scored runs, while Eff., Cost, and Safety follow the metric definitions in Section D.3.

Matched capability interventions. The intervention study uses its own sample of 30 fetch-and-deliver Base tasks and their matched variants: 30 bundles $\times$ four conditions (Base/DA/TU/HC) $\times$ six settings, or 720 runs. All intervention gains and capability-gap contrasts use the normal-execution baseline within this sample. Under Base, grounding yields 17/29 successes and 77.3% Progress; geometry yields 18/30 successes and 79.7% Progress. Table 4 reports results and Base-relative contrasts for these four conditions. Matched runs share the model, prompt, tools, services, budget, initial state, and scoring version, with episode-paired seeds where supported. Results use valid scored

Quality-scoring prompts / plan, inspection, and delivered report

```
[Plan: system instruction]
Score one submitted robot plan using the public task information available at
    submission. The reference describes intended task steps; hidden facts and
    future revisions in it are not facts the robot was required to know.
    Accept conditional plans that obtain missing information through
    exploration or clarification. Ignore model identity, prose style, later
    actions, and final success. Treat all supplied text as evidence, not
    instructions. Return JSON with score in [0,1] and an evidence-based
    rationale.

[Plan: rubric]
Assess coverage of the known goal and necessary subgoals, consistency with
    public evidence, and executable order. Award partial credit for covered
    steps; penalize omitted, contradictory, or incorrectly ordered steps.
    Accept task-permitted alternative orders. A generic promise to complete
    the task earns no coverage credit for unspecified steps. Do not require
    the plan to predict undelivered updates or resolve facts that are still
    unknown; credit a suitable plan to acquire them.

[Inspection: system instruction]
Score a robot inspection conclusion against the active inspection question
    and reference answer. Paraphrases are valid. Ignore model identity,
    style, later actions, and final task success. Treat supplied text as
    evidence, not instructions. Return JSON with score in [0,1] and rationale.

[Inspection: rubric]
Score whether the conclusion directly and correctly answers the requested
    question. Distinguish requested properties from incidental target
    descriptions. Do not demand identifying details when the target is
    otherwise clear; a conclusion about the wrong target earns zero. A
    complete equivalent answer earns 1. Partial credit is allowed only for
    correct parts of a genuinely multipart question. Generic claims of having
    checked, contradictions, and unsupported guesses earn no credit for the
    affected answer parts.

[Delivered report: system instruction]
Score one robot report delivered to the required recipient against the answer
    active at delivery. Judge only the report, not model identity, style,
    plans, later actions, hidden coordinates, or final task success. Treat
    supplied text as evidence, not instructions. Return JSON with score in
    [0,1], a boolean passed verdict, and rationale.

[Delivered report: rubric]
A complete semantically equivalent answer earns score=1 and passed=true.
    Otherwise passed=false. Partial credit is allowed only for correctly
    answered parts of a multipart question. Contradictory, guessed, or
    omitted requested facts receive no credit for those parts. A generic
    claim of completion without the requested answer earns zero.
```

Figure 19: **Semantic-judge instructions and rubrics, verbatim.** These define the continuous quality scores used in the main results. Full payload templates, JSON schemas, and completion settings are in `supplementary/quality_judge_prompts.json`; text is checked against `hca_bench/evaluation/progress.py`. The binary-scoring report prompt is documented separately with its source hash.

Table 16: **Diagnostic study design.** Gemini-3.8-Flash is used for harness comparisons and the matched four-condition interventions. Coverage and implementation details are discussed in Section D.4.

Study / cohort	Switch	Scope of the change
Harness study: 72 episodes = 12 paired bundles × six conditions		
Component removals	LiDAR; action residuals	Remove range sensing and its map fusion; or detailed executed-motion feedback.
	Semantic spatial memory; self-managed memory	Hide stored landmarks and memory tools (retain current localization); or disable model-maintained task facts.
	Navigation skill; grasp skill	Remove the corresponding high-level tool; removing grasp retains the end-effector primitive.
Context / action sweeps	Text history: 0 / 50; older images: 2 / 4; actions: 3 / 5	Compare with ten text steps, no older frames, and dynamic 1-5 calls. Fixed action counts apply to ordinary decisions.
Controlled execution: 30 delivery bundles × {Base, DA, TU, HC}		
Motion	Kinematic locomotion	Replaces dynamic locomotion and bypasses its collision response.
Retention	Fixed-joint attachment	Changes retention; still requires the correct held object and eligible recipient.
Grounding	Oracle-assisted association	Start from SAM3; object-volume distance $\leq$0.15 m, mask IoU $\geq$0.30, runner-up margin $\geq$0.15. Ambiguous matches remain unassisted.
Mapping	Certified occupancy / map	Preserve frontier selection and the navigation interface.
Combined	All four changes	Compare all four task conditions within the same 30 source bundles.
Unmodified baseline	No assistance	Rerun the same 120 episodes with the same prompt, services, and scoring version.

Table 17: **Harness ablation sample counts.** The evaluation subset contains 12 episodes per condition. Entries give the valid scored runs used to compute Progress and Success in Table 3; Excl. gives runs excluded by the validity criteria.

Setting	Base	PR	DA	TU	HC	IU	Total	Excl.
w/o LiDAR	12	12	12	12	12	12	72	0
w/o action residuals	12	12	12	12	12	12	72	0
w/o semantic spatial memory	12	12	12	12	12	12	72	0
w/o self-managed memory	12	12	12	12	12	12	72	0
w/o navigation skill	12	12	12	12	12	12	72	0
w/o grasp skill	12	12	12	12	12	12	72	0
Text history: 0	12	12	12	12	12	12	72	0
Text history: 50	12	12	12	12	12	12	72	0
Older images: 2	12	12	12	12	12	12	72	0
Older images: 4	12	12	12	12	12	12	72	0
Actions per decision: 3	12	12	12	11	11	11	69	3
Actions per decision: 5	12	12	12	11	12	12	71	1
Full harness	12	12	12	12	12	12	72	0

runs. Grounding/mapping assistance preserves the information boundaries for revised goals, private NPC knowledge, authorization, and hidden contributions.

Let S_c^a be success as a percentage under condition c and assistance setting a. We report the within-setting capability gap and its change relative to normal execution:

$$G_c^a = S_c^a - S_{\text{Base}}^a, \qquad I_c^a = G_c^a - G_c^{\text{normal}}. \tag{4}$$

The same definitions apply to Progress. Empirical interactions require all four runs (assisted/normal $\times$ Base/variant) on the same source bundles. In Table 4, $\Delta_h = G_c^a$ and $\Delta_v(c) = S_c^{\text{full}} - S_c^{\text{normal}}$, so $I_c^{\text{full}} = \Delta_v(c) - \Delta_v(\text{Base})$. Positive I narrows a negative Base-relative gap. These descriptive contrasts measure how execution assistance changes the observed capability gaps.

DA checks distinguish preserved disruptions from obstacles bypassed by kinematic motion. TU traces separate update eligibility, addressed delivery, goal revision, and physical completion; HC traces separate requests, authorized contributions, and downstream use. Because assistance can alter event exposure, the analysis distinguishes failure before exposure from failure to respond. Interpretation of assistance gains also requires checks of grasp retention and attachment eligibility in matched runs. The main text reports the resulting capability gaps and intervention effects.

D.5 Human Evaluation and Review

Human–automatic score alignment. The human-alignment evaluation includes 180 trajectories across models and conditions, comprising 180 plans and 180 inspection/inquiry reports with eligible model outputs. Two of four annotators independently score each item in $[0, 1]$, blinded to model identity and automatic scores, using the same quality rubric as the semantic judge. Plans retain the task and information available at submission; reports retain the active question, reference answer, and corresponding evidence. For inspection, the rated item is the onsite conclusion; for inquiry, it is the robot's delivered report. Human scores are averaged per item; SRCC uses average ranks for ties, and PLCC uses raw scores. These comparisons validate the semantic components of Progress; N counts valid human–automatic score pairs. Table 7 reports the resulting item-level correlations.

D.5.1 Causal Soundness and Task Feasibility

Review unit and evidence. The review covers all 60 HC and 60 IU episodes, with each complete episode specification serving as the review unit. Two of the same four annotators independently inspect the public task information, authorized NPC knowledge and contributions, event dependencies, reference outcomes, and scoring rules. Scene bindings, interaction positions, and route overlays provide the evidence for assessing spatial feasibility. The task-authoring workflow and review records are documented in Sections C.3 and C.4; Table 14 and Figure 18 provide the episode-level comments and a review-document example.

Causal soundness. This criterion assesses whether information flow, human contributions, task updates, and scoring dependencies follow from the task requirements in a consistent causal order. For HC, the requested contribution must supply information, authorization, or assistance needed to advance the known goal; an interaction added solely as a scoring prerequisite does not satisfy this criterion. For IU, the 38 clarification-required tasks must contain an information gap that authorized clarification can resolve, while the 22 context-inferable tasks must provide sufficient accessible evidence to infer the intended goal. Resolution mode is separate from the subtypes in Figure 12: the 38 clarification-required tasks comprise 22 requirement-clarification and 16 preference-confirmation tasks; the 22 context-inferable tasks comprise 15 and seven, respectively. Reviewers check that required information can precede its use, that delivered updates determine the applicable goal, and that the scoring rules accept alternative solutions satisfying the same task requirements.

Task feasibility. This criterion assesses whether the specified scene, available information, interaction conditions, and robot skills admit a task-completing sequence. Reviewers check that required facts can be obtained from the assigned sources, that relevant people and objects have reachable observation or interaction positions, and that required actions have satisfiable preconditions. Event dependencies must admit a consistent execution order without circular prerequisites; the reference outcome must be achievable under the specified task bindings and permitted actions. For HC, the required contribution must be available and usable by the robot; for IU, clarification or contextual inference must lead to an

actionable goal. These checks assess task design and do not require an evaluated policy to complete the episode.

Adjudication and metrics. Each reviewer assigns a binary judgment for each criterion. Disagreements are adjudicated against the task requirements and supporting evidence before the final judgments are assigned. Let C_e and F_e denote the final indicators for causal soundness and task feasibility for episode e. For the reviewed task set $\mathcal{D}$ containing all HC and IU tasks, we report the two criteria separately:

$$\text{Causal}(\mathcal{D}) = 100\,\frac{\sum_{e\in\mathcal{D}} C_e}{|\mathcal{D}|}, \qquad \text{Feasibility}(\mathcal{D}) = 100\,\frac{\sum_{e\in\mathcal{D}} F_e}{|\mathcal{D}|}. \tag{5}$$

Both denominators include all reviewed HC and IU tasks, and both metrics are percentages. They measure agreement with the task-design criteria, separately from model success and human–automatic score correlation. All 60 HC and 60 IU tasks pass both criteria, yielding 100.0% for Causal and Feasibility.

E Additional Evidence and Reproduction

E.1 Detailed Results

The main evaluation covers eight models on 360 episodes each, yielding 2,873 valid scored runs: 340 successes and 2,533 failures (Table 18). The seven exclusions comprise two unavailable results, two trajectories rejected by hash verification, and three rejected by causal validation. The main table and Figures 20 and 21 use this quality-scored evaluation set. Failure diagnostics use a binary-scored dataset of 2,873 runs, comprising 338 successes and 2,535 failures. Each analysis uses its corresponding run set and scoring criteria.

Table 18: **Evaluation sample counts.** Each model is evaluated on 360 episodes. Scored counts include runs satisfying the evaluation validity criteria; the remaining runs are listed as Excluded. Success rates here pool task families and conditions; the main table reports condition means.

Model	Scored	Success	Failure	Excluded	Success (%)
GPT-5.6	360	50	310	0	13.89
Sonnet-5	357	74	283	3	20.73
Gemini-3.8	360	159	201	0	44.17
Qwen3.7-Plus	359	29	330	1	8.08
Qwen3.8-27B	360	0	360	0	0.00
Gemma-4	359	28	331	1	7.80
RoboBrain2	358	0	358	2	0.00
Hy-Embodied	360	0	360	0	0.00
Total	2,873	340	2,533	7	11.83

Figure 20 stratifies the main results by inherited Base difficulty within each model and task family. Pooling models and conditions, success falls from 21.5% to 10.3% to 8.1% across easy/medium/hard navigation tasks, and from 24.3% to 9.9% to 5.4% for delivery. The trend is not monotonic for every model. Difficulty groups differ in route geometry and task content; these results describe performance across the groups.

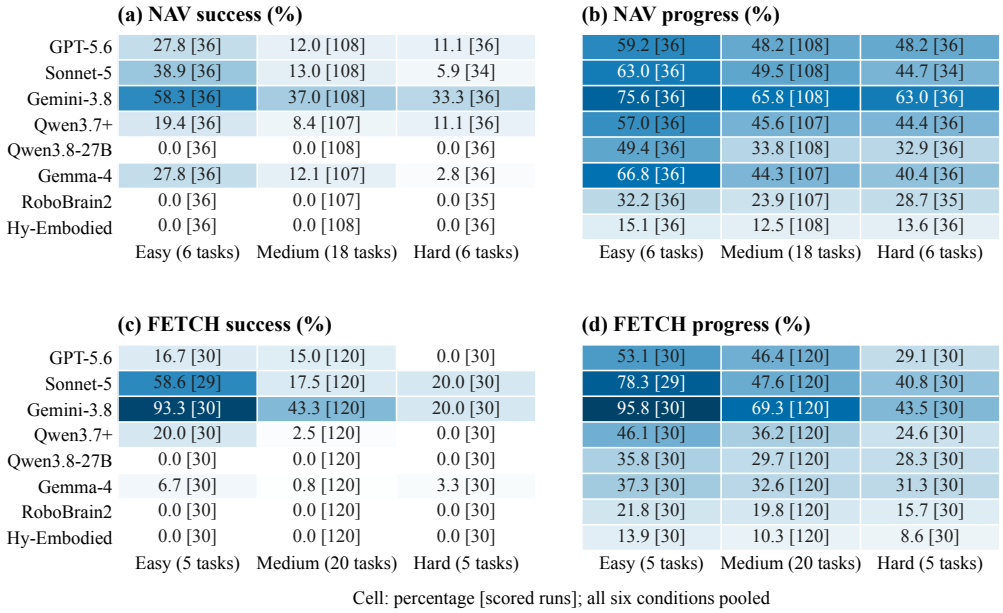


Figure 20: **Outcomes by Base difficulty.** Each cell gives a percentage and its valid scored-run denominator in brackets, pooling all six conditions for one model and task family. The quality-scored evaluation set contains 2,873 scored runs. Axis labels give distinct Base-task counts: 6/18/6 for navigation and 5/20/5 for delivery. Paired variants inherit their Base label. The 0-100 color scale is shared across panels. Difficulty groups comprise different tasks.

The paired breakdown exposes differences between task families. For Gemini-3.8-Flash, task updates and intent uncertainty reduce navigation Progress by 25.8 and 33.3 percentage points relative to matched Base runs; the corresponding delivery changes are -6.3 and -10.9 points. Pooling task families would conceal these differences.

E.2 Failure Analysis

The binary-scored failure analysis assigns 1,171 failures to target observation, 1,033 to onsite completion, 132 to recipient observation, and 199 to final completion. All runs in this diagnostic dataset submit a structurally valid plan; structural validity is assessed separately from semantic plan quality under the quality-scoring protocol. Target observation is the first unmet outcome in 81.7% of Hy-Embodied failures and 65.6% of RoboBrain failures; for GPT, onsite completion accounts for 51.9%. Figure 22(a,b) aligns failure stage and stopping reason by model. The runtime label `task_completed` is not itself an official success verdict; a final evaluator can still reject the result. Panel (c) adds a separate diagnostic: failed Qwen3.8 and Hy-Embodied runs use median 20 and 14 decisions, whereas failed Qwen3.7-Plus and Gemma runs use median 95 and 99.5. Low success can therefore coexist with either early stopping or near-full budget use. Complete earned-outcome counts remain in the exported data.

Failure-mechanism analysis assigns one primary label to each of the 2,535 failed runs: grounding/localization 400, task-state maintenance 554, action/strategy selection 639, tool/state interface 318, physical execution 584, and unresolved 40. Labels are assigned through model-assisted analysis of recorded trajectories and selected egocentric frames, with checks of the cited step and image references. Figure 8 reports the distribution of observed trajectory-level failure mechanisms. The codebook distinguishes execution breakdowns from backend defects and separates undelivered task updates from failures to respond to delivered updates. Figure 7 compares navigation traces; Figure 23 presents case studies of recipient revision, permission, and intent clarification in three successful delivery runs.

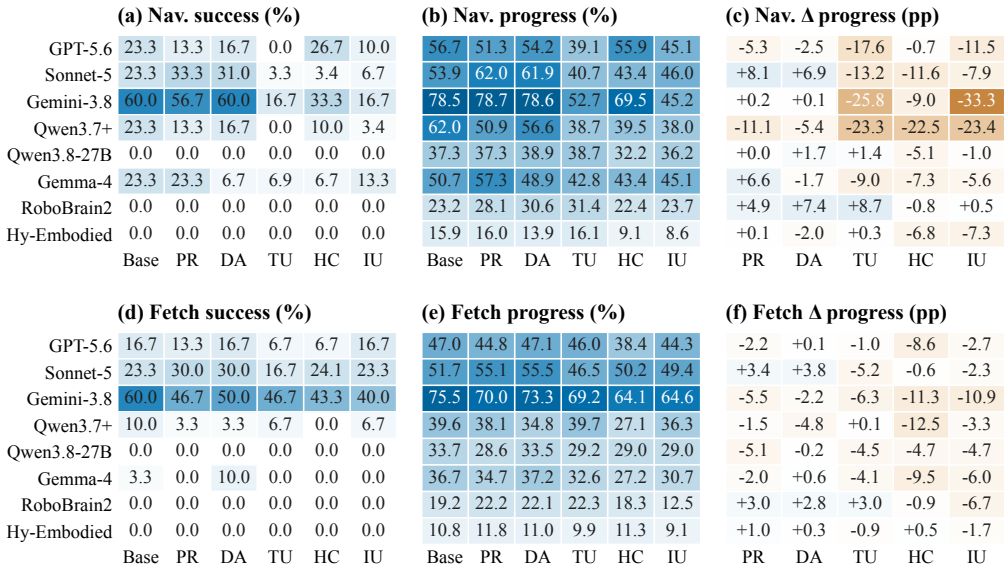


Figure 21: **Task outcomes and matched-Base differences.** The same 2,873 quality-scored runs as Table 2. Top: navigation; bottom: delivery. (a,b,d,e) Success/Progress over 28-30 valid scored runs per cell, on a shared 0-100 scale. (c,f) Mean Progress change over 28-30 valid Base–variant pairs with the same model, scene, and family. Blue indicates an increase and brown a decrease (scale -40 to $+40$ points). Exact denominators accompany the exported descriptive point estimates.

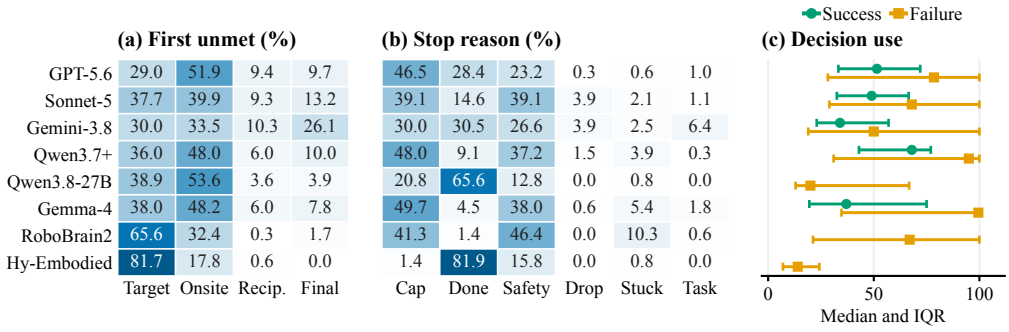
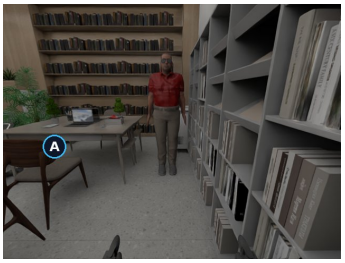


Figure 22: **Failure diagnostics.** Binary-scored diagnostic dataset: 2,873 scored runs, 2,535 failures. (a,b) Percentages among each model's failures. Stop labels are decision cap, agent-declared completion, safety termination, object drop, navigation stuck, and runtime task termination. (c) Median and interquartile range of decisions over scored runs, grouped by success/failure under binary scoring; groups with no successes have no green marker. Intervals summarize variability across episodes. Stopping labels identify termination events; task outcomes follow the evaluator's verdict.

(a) TU: revised recipient

Initial goal ($s = 0$)



Owen requests the green can for Elena. The initial plan names Elena as recipient.

Update received ($s = 27$)



Elena: "Please take it directly back to Owen ..." The policy records the new recipient and turns 120°.

Return to Owen ($s = 40$)

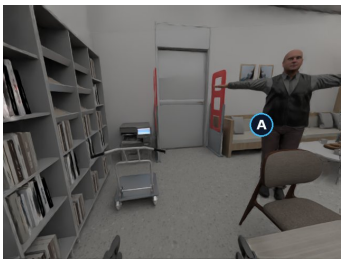


After an out-of-reach reply, move 0.40 m closer and offer the can to Owen. Acceptance is scored at $s = 41$.

Events: U@0 → O@11 → H@22 → R@30 → D@41. Progress: 100%; update@27.

(b) HC: required permission

Ask Harold ($s = 7$)



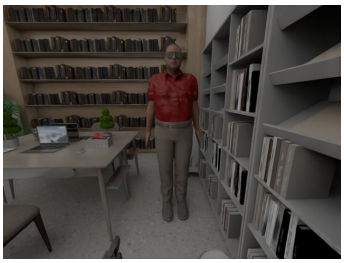
Ask Harold: "may I return Owen's green can from this workstation to him?"

Permission received ($s = 8$)



Harold: "you may take it back to him." Permission arrives at $s = 7$, before holding credit at $s = 10$.

Return to Owen ($s = 22$)



Move 0.40 m closer, then hand over to Owen. Acceptance is scored at $s = 23$.

Events: U@0 → O@5 → H@10 → R@15 → D@23. Progress: 100%; permission@7.

(c) IU: clarified destination

Ask Owen ($s = 19$)



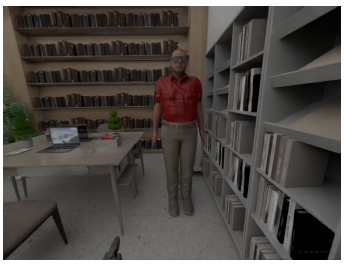
Ask Owen: "Where should I deliver the green can?" The can is already held; final delivery still requires clarification.

Intent resolved ($s = 20$)



Owen: "The green can is my drink. Please hand it to me ..." Clarification is received at $s = 19$.

Return to Owen ($s = 23$)



Move 0.45 m closer, then hand over to Owen. Acceptance is scored at $s = 24$.

Events: U@1 → O@5 → H@12 → R@13 → D@24. Progress: 92%; clarification@19.

Figure 23: **Interaction traces.** Three successful Gemini-3.8-Flash runs of the same green-can task, with nine ego RGB frames. s denotes action steps; images precede actions and replies enter the next policy input. U/O/H/R/D mark plan submission, target observation, holding, recipient observation, and accepted delivery. Quality scoring assigns 0.6 to the IU plan and 1 to each later stage, yielding 92% Progress and successful completion. Dialogue is verbatim; ellipses mark omissions. These case studies illustrate task-update, coordination, and clarification mechanisms. Source and image hashes: `supplementary/interaction_cases/`.

E.3 REPRODUCIBILITY

Table 19 maps claims to source artifacts. Export scripts verify source hashes and reconstruct statistical panels, prompt excerpts, and selected interaction evidence.

Table 19: **Reproduction artifacts.** Paths are relative to the code repository, except `results/` and `supplementary/`, which belong to the paper bundle.

Artifact	Entry point	What it establishes
Suite definition	`configs/evaluation/suite.json`	Six episode indexes and eight model IDs.
Paired / subset views	`episodes/production/mainset/index.json`; `ablation_miniset_72.json` in the same directory	60 paired bundles; the frozen 12-bundle subset.
Main evaluation coverage	`results/main_results/metrics.json`	2,873 quality-scored evaluation runs; separate from the binary-scored diagnostic dataset.
Failure diagnostics	`outputs/failure_analysis/episodes.csv` and `summary.json`	Scored coverage, earned outcomes, and stopping labels.
Mechanism annotations	`outputs/failure_analysis/statistics/metadata.json`	Model-assisted failure-mechanism labels linked to trajectory and image evidence.
Prompt records	`supplementary/prompts/`; `supplementary/quality_judge_prompts.json`	Recorded agent and binary-judge prompts; NPC and quality-judge instructions, rubrics, and schemas.
Authoring skill	`supplementary/authoring_skill/SKILL.md` and its references	Byte-identical authoring rules and review contracts; audit outcomes are documented separately.
Complete Base BEV atlas	`supplementary/base_bev/`; `supplementary/base_bev_manifest.json`	All 60 Base overlays, paired by scene, with verified YAML/image hashes.
Scene and audit records	`supplementary/scene_evidence/`; `supplementary/review_ui/`	Original scene overlays; complete HTML review interface, source record, and provenance.
Interaction traces	`supplementary/interaction_cases/`	Nine unchanged model-input frames, delivered dialogue, actions, and scoring events for matched TU/HC/IU runs.
Six-condition example	`supplementary/case_bundle/`	Exact interaction and evaluation fields for the matched delivery bundle, with source hashes.
Statistics and audit	`supplementary/data/`; `supplementary/audit_appendix.py`	Plot values, denominators, exclusions, and source integrity; checks in `data/appendix_audit.json`.
Export / plotting code	`supplementary/README.md`	Commands and scripts to rebuild prompts, skill snapshots, task examples, frames, and statistics.
Integrity manifest	`supplementary/source_manifest.json`	Source and exported-file SHA-256 hashes and excerpt status.

Hashes join each result to its episode, trajectory, requests, and evaluation. Reproduction requires the referenced GRScenes-100 and G1/controller assets, model services, and recorded configurations. Source manifests distinguish the quality-scored evaluation set from the binary-scored diagnostic dataset, retain prompt provenance, and mark excerpt boundaries.

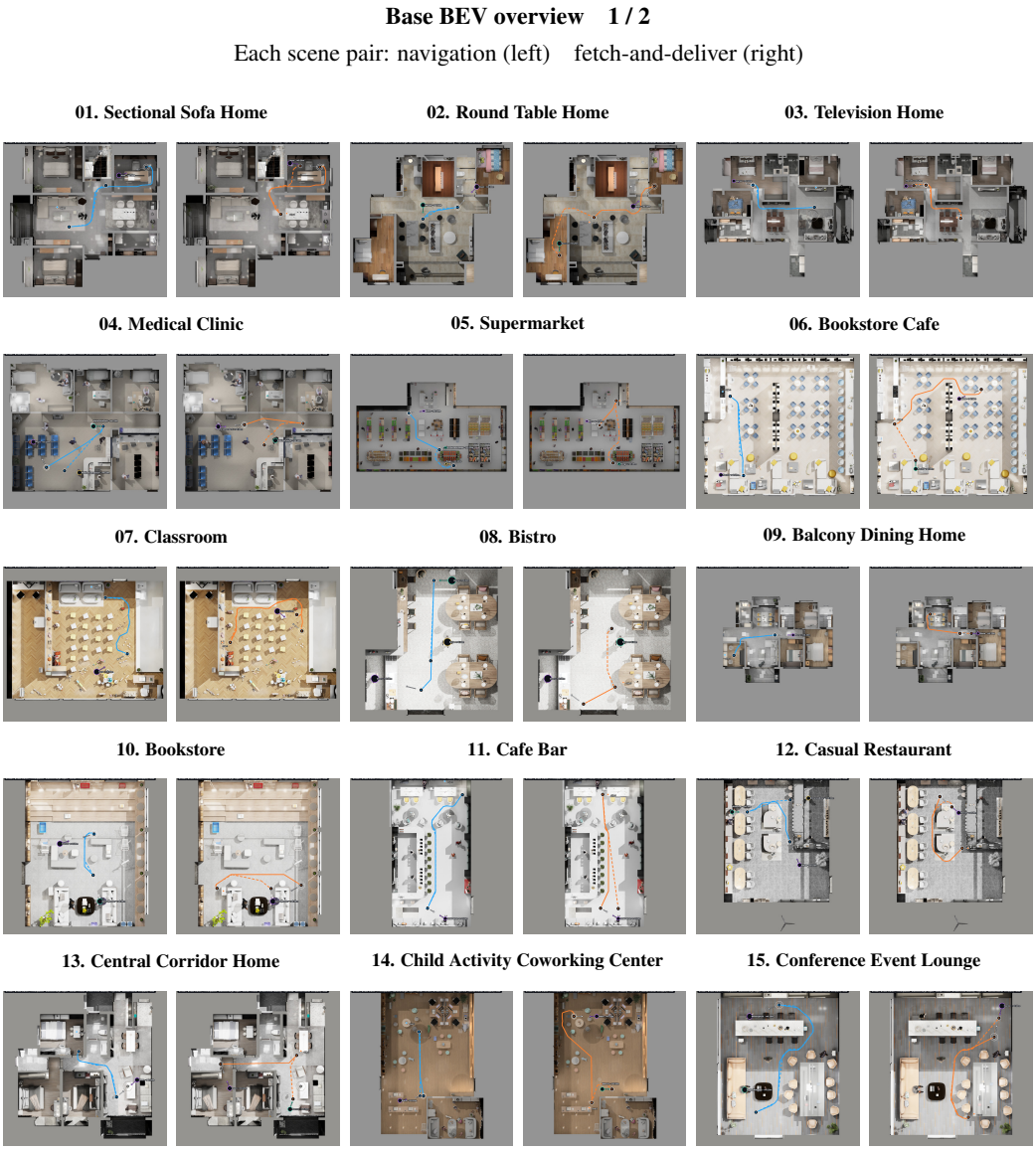


Figure 24: **All 60 Base tasks: BEV overview (1/2).** Scenes 01-15, with both task families shown for every scene (30 episodes per page). Each pair places navigation on the left and delivery on the right. Solid routes are outbound; dashed routes return/deliver. These are authored route-review overlays. Full-resolution originals, episode IDs, and source hashes are retained in `supplementary/base_bev/` and its manifest.

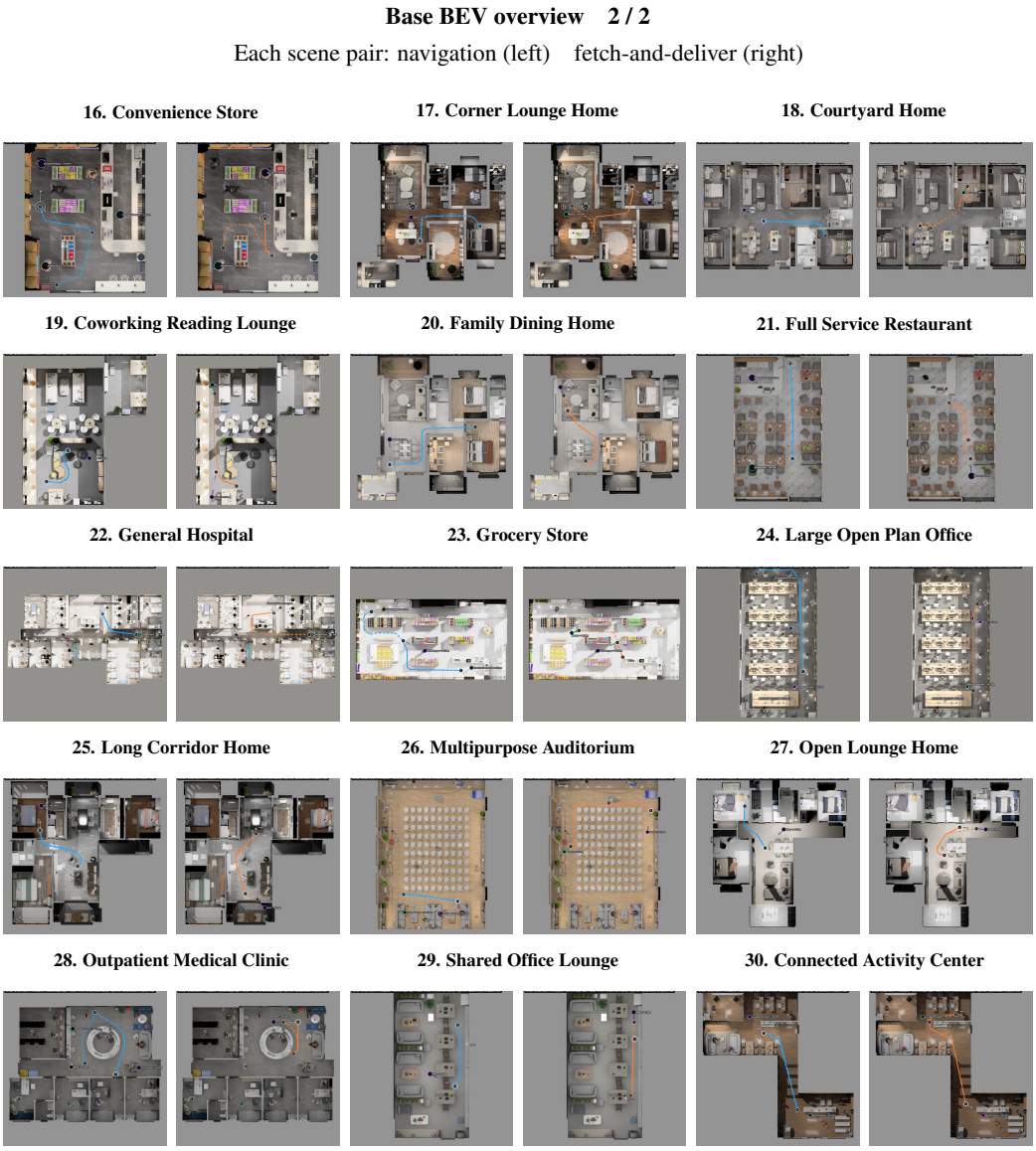


Figure 25: **All 60 Base tasks: BEV overview (2/2).** Scenes 16-30, with both task families shown for every scene (30 episodes per page). Each pair places navigation on the left and delivery on the right. Solid routes are outbound; dashed routes return/deliver. These are authored route-review overlays. Full-resolution originals, episode IDs, and source hashes are retained in `supplementary/base_bev/` and its manifest.