The Body Needs a Manager:
Operational World Models as the Control Plane for Hospitality Robotics

Part II: From Physical Intelligence to Operational Intelligence

A Research Agenda for Embodied AI in Service Environments

July 2026

Abstract

Advances in physical AI are producing robots with increasingly general capabilities in perception, navigation, manipulation, and language understanding. Yet these capabilities enable robots to ask only: "Can I move through this environment and perform this action?" In complex service environments like hotels and restaurants, this question is insufficient. Robots must also answer: "Should this action happen, for whom, at what time, under which constraints, and how does it affect the rest of the operation?"

This paper establishes the formal necessity of venue-level operational world models as the control plane for hospitality robotics. We distinguish three non-substitutable intelligences: spatial intelligence (navigation), physical intelligence (manipulation), and operational intelligence (context, priorities, policies, consequences). We formalize the hospitality venue as a Dec-POMDP with two coupled layers: an embodied world model handling geometry, motion, and immediate physical risk; and an operational world model handling people, permissions, intent, task dependencies, and service priorities. We develop the Venue-Robot Control Protocol, establishing how operational models assign goals and constraints to robots, how robots report observations back to the venue model, and how heterogeneous robots from different manufacturers can coordinate through a shared belief state.

We present six detailed operational scenarios demonstrating why optimizing robot task completion alone produces locally successful but operationally harmful behavior. We propose **VenueBench**, a benchmark evaluating robots on operational state inference, task prioritization, human-robot coordination, permission awareness, and safe escalation under uncertainty. We analyze safety, privacy, labor implications, and economic thresholds for deployment. We conclude with a five-phase roadmap and a falsifiable thesis: *The enduring robotics platform in hospitality may not be the company that builds every body. It may be the system that already understands the venue when those bodies arrive.*

1 Introduction

The robots are coming. Vision-Language-Action (VLA) models like RT-2 (Brohan et al., 2023), PaLM-E (Driess et al., 2023), and OpenVLA (Kim et al., 2024) demonstrate impressive generalization: the same model can follow natural language instructions to manipulate diverse objects, navigate unfamiliar spaces, and even transfer knowledge from Internet-scale vision-language pretraining to robotic control (Li et al., 2025). Quadruped robots traverse stairs and obstacles. Mobile manipulators grasp objects they have never seen before. The physical AI revolution promises to bring capable robots into hotels, restaurants, hospitals, and homes.

But a robot that *can* perform a task may still be the wrong robot for the job, at the wrong time, in the wrong way. Consider:

- A delivery robot that *can* open a guest room door, but enters while the guest is showering

- A cleaning robot that *can* vacuum the hallway, but blocks emergency egress during a fire drill
- A restocking robot that *can* navigate to the kitchen, but arrives during the lunch rush when corridors are congested
- A service robot that *can* deliver towels, but prioritizes Room 205 over Room 312 while ignoring that Room 312 has a crying infant and an urgent request

In each case, the robot succeeds at its local task but fails operationally. The failure is not in perception, navigation, or manipulation—the core capabilities that physical AI addresses. The failure is in *operational intelligence*: understanding context, priorities, dependencies, policies, and consequences.

1.1 The Central Distinction

We distinguish two fundamentally different questions:

1. **Embodied Question:** "Can I move through this environment and perform this action?"
2. **Operational Question:** "Should this action happen, for whom, at what time, under which constraints, and how does it affect the rest of the operation?"

Current robot foundation models address the embodied question. They learn world models of physics, geometry, and object affordances. But they lack the operational context that turns physical capability into appropriate action.

1.2 Contributions

This paper (Part II of our research agenda) makes the following contributions:

1. A **three-intelligence framework** distinguishing spatial, physical, and operational intelligence as non-substitutable capabilities required for service robotics.
2. A **formal Dec-POMDP framework** for hospitality venues as multi-agent systems with two coupled layers: embodied and operational.
3. The **Venue-Robot Control Protocol**, specifying how operational models assign goals and constraints to robots, and how robots report observations to update venue belief states.
4. Six **detailed operational scenarios** demonstrating the failure modes of robot-local optimization and the necessity of venue-level operational reasoning.
5. **VenueBench**, a proposed benchmark evaluating robots on operational capabilities beyond navigation and manipulation.
6. An analysis of **safety, privacy, labor implications**, and **economic thresholds** for deployment.
7. A **skeptical assessment** identifying assumptions, failure modes, and experiments that could falsify our thesis.

1.3 The Core Thesis

Our thesis is defensible and falsifiable:

> **The enduring robotics platform in hospitality may not be the company that builds every body. It may be the system that already understands the venue when those bodies arrive.**

Physical AI companies will produce increasingly capable robots. But these robots will arrive without understanding of the specific venue's operational rhythms, policies, and priorities. The venue that has spent years building an operational world model—learning from human operations, encoding institutional memory, establishing coordination protocols—will be ready to absorb and direct these robots effectively. The venue without such a model will struggle to deploy robots productively, regardless of the robots' physical capabilities.

2 Why Robot Foundation Models Alone Are Insufficient

We examine the capabilities and limitations of current robot foundation models, establishing why they cannot, by themselves, operate effectively in hospitality venues.

2.1 Capabilities of Robot Foundation Models

Recent advances in robot foundation models have been remarkable:

Vision-Language-Action Models: RT-2 (Brohan et al., 2023) trains a VLA model on Internet-scale vision-language data combined with robotic demonstration data, enabling zero-shot generalization to unseen objects and instructions. The model learns implicit world models of object affordances and physical dynamics. OpenVLA (Kim et al., 2024) demonstrates that open-source VLA models can achieve competitive performance with careful data curation and training protocols.

Embodied Multimodal Models: PaLM-E (Driess et al., 2023) integrates continuous sensor modalities (images, state estimates) into language models, enabling multimodal reasoning about physical scenes. The model can answer questions about object configurations, predict physical outcomes, and generate plans for manipulation tasks.

Generalist Manipulation: Recent surveys (Li et al., 2025; Shao et al., 2024) document rapid progress in generalist robot policies that can follow natural language instructions, generalize across object categories, and adapt to novel environments through in-context learning.

2.2 The Gap: Can vs. Should

Despite these capabilities, robot foundation models face fundamental limitations for hospitality operations:

Local vs. Global Optimization: Robot policies optimize task completion from the robot's perspective. They lack the venue-level view that would reveal when local success creates global problems. A robot that efficiently delivers towels to Room 205 while blocking hallway traffic may achieve its local objective while disrupting the broader operation.

Missing Operational State: Robot world models represent physical state—object positions, geometries, affordances. They do not represent operational state: which guest is a VIP, which employee is handling an incident, whether housekeeping is behind schedule, whether a task should be delayed for service quality reasons.

No Permission System: Robots lack understanding of venue policies governing who can enter which spaces, when guests should not be disturbed, which tasks require manager approval, and when physical capability does not imply authorization.

Static vs. Dynamic Priorities: Robot tasks are typically specified with fixed priorities. Hospitality operations require dynamic prioritization based on real-time operational context: the same delivery may be urgent during normal operations but deferred during an emergency.

2.3 Case Study: The Delivery Robot

Consider a towel-delivery task. A robot foundation model can:

- Navigate to the room (spatial intelligence)
- Grasp and carry towels (physical intelligence)
- Open doors and place items (manipulation)

But it cannot determine:

- Whether the room is currently occupied and the guest is sleeping
- Whether housekeeping is cleaning the room and entry would disrupt their work
- Whether the guest has a “Do Not Disturb” request in the PMS
- Whether an urgent VIP arrival in the adjacent room takes priority
- Whether the hallway is congested with luggage carts and timing should be adjusted
- Whether a maintenance issue in the room makes entry inappropriate

These determinations require access to operational systems (PMS, task management, scheduling), reasoning about service priorities, and understanding of venue policies. They cannot be inferred from the robot’s onboard sensors alone.

3 Three Intelligences: Spatial, Physical, and Operational

We propose a framework distinguishing three non-substitutable intelligences required for effective service robotics.

3.1 Spatial Intelligence

Definition: The ability to perceive, represent, and navigate through geometric space.

Capabilities:

- Mapping and localization (SLAM)
- Path planning and obstacle avoidance
- Spatial memory and place recognition
- Understanding of traversability and connectivity

Current State: Well-developed. LiDAR, visual SLAM, and learned navigation policies enable robust spatial intelligence in structured indoor environments.

Limitations: Spatial intelligence answers “where” but not “when,” “for whom,” or “under what constraints.” A robot may know it *can* navigate to a location without knowing if it *should*.

3.2 Physical Intelligence

Definition: The ability to manipulate objects, control physical interactions, and reason about dynamics and affordances.

Capabilities:

- Grasping and manipulation of diverse objects
- Force control and compliant motion
- Understanding of object affordances and physical properties
- Tool use and dexterous manipulation

Current State: Rapidly advancing. VLA models (Brohan et al., 2023; Kim et al., 2024) demonstrate generalization to novel objects and tasks. Learned dynamics models enable prediction of physical outcomes.

Limitations: Physical intelligence answers "how" but not "whether." A robot may know it *can* open a door without knowing if entry is permitted, appropriate, or safe given operational context.

3.3 Operational Intelligence

Definition: The ability to understand context, priorities, dependencies, policies, and consequences within an organizational operation.

Capabilities:

- Understanding of roles, responsibilities, and permissions
- Reasoning about task priorities and dependencies
- Awareness of service quality standards and guest expectations
- Knowledge of policies, procedures, and escalation protocols
- Ability to predict operational consequences of actions
- Coordination with human and artificial agents

Current State: Underdeveloped for robotics. Current systems lack the structured operational knowledge required for service environments.

Why It Cannot Be Subsumed: Operational intelligence cannot be reduced to spatial or physical intelligence. Knowing where a room is located (spatial) and how to open its door (physical) does not tell you whether to enter (operational). The operational layer requires integration with information systems, understanding of organizational processes, and reasoning about human preferences and priorities.

3.4 Integration of the Three Intelligences

Effective service robotics requires all three intelligences working together:

- **Spatial + Physical:** The robot can navigate to an object and manipulate it (embodied AI)
- **Physical + Operational:** The robot knows what actions are permitted and appropriate (policy compliance)

- **Operational + Spatial:** The robot knows where to go based on operational priorities (context-aware navigation)
- **All Three:** The robot can navigate to the right place, at the right time, and perform the right action in the right way

4 Formal Framework: Dec-POMDP for Hospitality Multi-Agent Systems

We formalize the hospitality venue as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP), capturing the multi-agent nature of the environment and the coupling between embodied and operational layers.

4.1 Dec-POMDP Definition

Definition 4.1 (Hospitality Dec-POMDP)**.** A hospitality venue with n robots and m human agents is a Dec-POMDP $\mathcal{M} = (\mathcal{S}, \{\mathcal{A}_i\}_{i=1}^{n+m}, \{\mathcal{O}_i\}_{i=1}^{n+m}, \mathcal{T}, \{\Omega_i\}_{i=1}^{n+m}, \mathcal{R}, \gamma)$ where:

- $\mathcal{S} = \mathcal{S}^{\text{embodied}} \times \mathcal{S}^{\text{operational}}$ is the global state space
- $\mathcal{A}_i$ is the action space for agent i
- $\mathcal{O}_i$ is the observation space for agent i
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition function
- $\Omega_i : \mathcal{S} \to \Delta(\mathcal{O}_i)$ is the observation function for agent i
- $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the shared reward function
- $\gamma \in [0, 1)$ is the discount factor

4.2 State Space Decomposition

The global state decomposes into embodied and operational components:

$$s = (s^{\text{embodied}}, s^{\text{operational}}) \tag{1}$$
$$s^{\text{embodied}} = (\{z_i\}_{i=1}^{n}, \text{geometry}, \text{objects}, \text{dynamics}) \tag{2}$$
$$s^{\text{operational}} = (\text{staff}, \text{guests}, \text{tasks}, \text{policies}, \text{history}) \tag{3}$$

where z_i is the local embodied state of robot i (pose, velocity, sensor readings).

4.3 Belief State Coupling

The venue maintains a shared belief state over the operational state:

$$b_t(s^{\text{operational}}) = P(S_t^{\text{operational}} = s^{\text{operational}} | o_{1:t}, a_{1:t-1}) \tag{4}$$

Each robot i maintains a local belief over its embodied state and receives the shared operational belief:

$$b_t^i = (b_t^{\text{local}}, b_t^{\text{operational}}) \tag{5}$$

4.4 Robot Policy Structure

The policy for robot i depends on both local embodied state and shared operational belief:

$$\pi_i(a_t^i | z_t^i, b_t, g_t^i, c_t^i) \tag{6}$$

where:

- z_t^i is the robot's local embodied state
- b_t is the venue's shared belief state
- g_t^i is the assigned goal from the operational layer
- c_t^i represents constraints (permissions, timing, safety bounds)

4.5 Shared Objective Function

The system optimizes a shared objective balancing service quality, task completion, and costs:

$$\max_{\pi} \mathbb{E}\left[\sum_t \gamma^t \left(R_{\text{service}} + R_{\text{completion}} - \lambda_1 C_{\text{safety}} - \lambda_2 C_{\text{disruption}} - \lambda_3 C_{\text{labor}} - \lambda_4 C_{\text{guest}}\right)\right] \tag{7}$$

Reward Components:

- R_{service}: Service quality metrics (guest satisfaction, response time)
- $R_{\text{completion}}$: Task completion rewards
- C_{safety}: Safety violations and near-misses
- $C_{\text{disruption}}$: Operational disruption costs
- C_{labor}: Labor coordination costs
- C_{guest}: Guest disturbance and inconvenience

4.6 Why Local Optimization Fails

Optimizing only robot task completion corresponds to maximizing $R_{\text{completion}}$ while ignoring other terms. This produces locally optimal but globally harmful behavior:

- High $R_{\text{completion}}$ but high C_{guest}: Robot completes delivery but wakes sleeping guest
- High $R_{\text{completion}}$ but high $C_{\text{disruption}}$: Robot navigates efficiently but blocks emergency egress
- High $R_{\text{completion}}$ but low R_{service}: Robot delivers to wrong room due to missing VIP context

The operational layer ensures optimization of the complete objective, not just local task success.

5 Two-Layer Architecture

We propose a two-layer architecture coupling embodied and operational intelligence. Figure 1 illustrates this architecture.

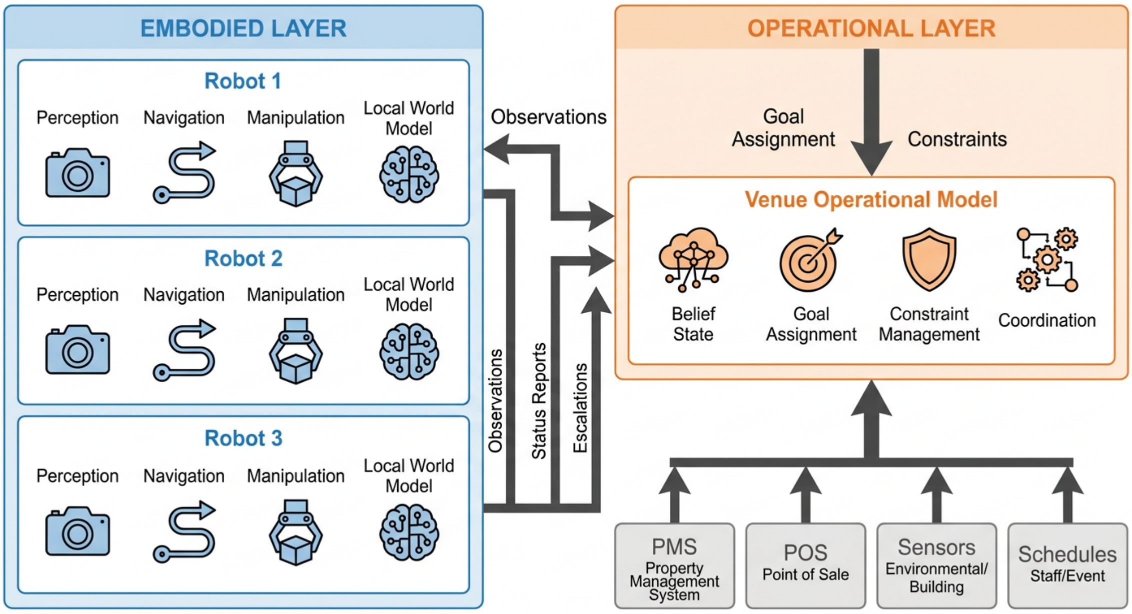

Figure 1: Two-layer architecture for hospitality robotics. The Embodied Layer handles perception, navigation, and manipulation. The Operational Layer maintains venue belief state and provides goal assignment and constraints. The Interface Protocol mediates communication between layers.

5.1 Embodied Layer

The embodied layer resides on each robot and handles:

Perception:

- Visual object detection and segmentation
- Depth estimation and 3D scene reconstruction
- Semantic understanding of spaces and objects
- Human detection and tracking

Navigation and Control:

- Localization and mapping
- Path planning and obstacle avoidance
- Trajectory generation and tracking
- Compliance with dynamics constraints

Manipulation:

- Grasp planning and execution
- Object manipulation and placement
- Force control and compliant motion
- Tool use and dexterous manipulation

Local World Model: Each robot maintains a local world model of its immediate physical environment, sufficient for collision-free navigation and manipulation but not encompassing the full venue operational state.

5.2 Operational Layer

The operational layer is a venue-level system that maintains:

Venue Belief State:

- Staff locations, availability, and task assignments
- Guest locations, preferences, and status
- Room/table status and occupancy
- Active tasks and their priorities
- Equipment status and maintenance needs
- Current operational mode (normal, rush, emergency)

Goal Assignment:

- Dynamic task allocation to robots
- Priority adjustment based on operational context
- Load balancing across robot fleet
- Coordination with human task assignments

Constraint Management:

- Permission enforcement (which robots can enter which spaces)
- Timing constraints (when tasks should/should not occur)
- Coordination constraints (avoiding conflicts, sequencing)
- Safety constraints (emergency stops, exclusion zones)

5.3 Interface Protocol

The interface between layers specifies communication protocols:

Downward Communication (Operational → Embodied):

- `assign_goal(robot_id, goal_spec, priority, constraints)`
- `update_constraints(robot_id, new_constraints)`
- `cancel_goal(robot_id, reason)`
- `query_status(robot_id)`

Upward Communication (Embodied → Operational):

- `report_observation(robot_id, observation_type, data)`
- `request_clarification(robot_id, uncertainty_type)`
- `report_completion(robot_id, goal_id, outcome)`
- `escalate(robot_id, situation_type, severity)`

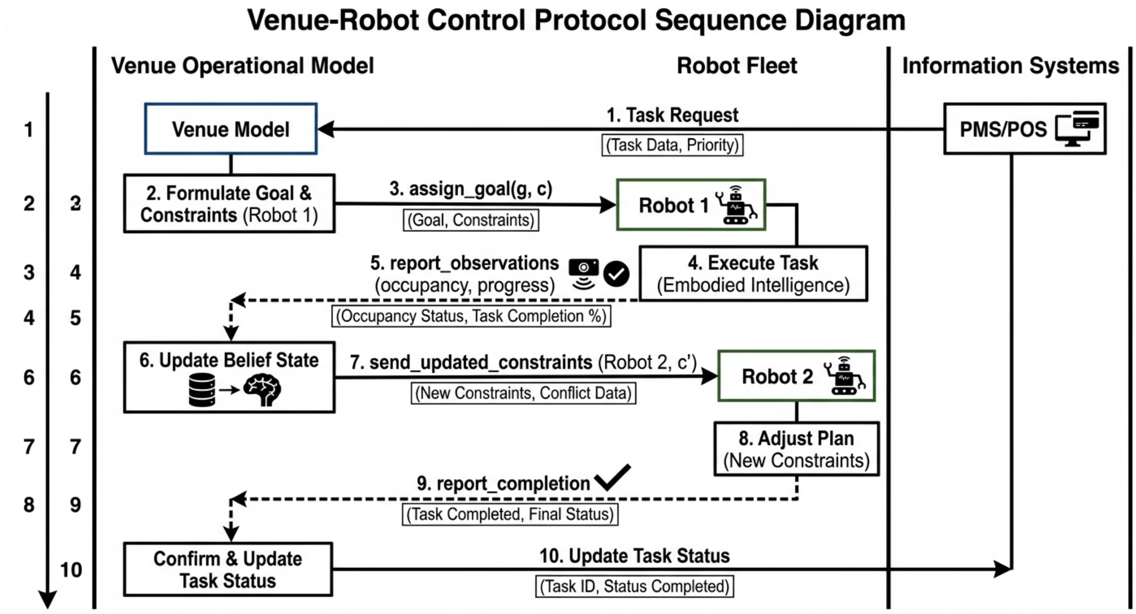

Figure 2: Venue-Robot Control Protocol sequence diagram showing goal assignment, execution, observation reporting, and belief update cycles.

6 The Venue-Robot Control Protocol

We specify the real-time protocol through which the venue operational model assigns goals and constraints to robots, robots execute actions and report observations, and the venue belief state updates. Figure 2 illustrates the protocol flow.

6.1 Goal Assignment

When the operational layer determines that a robot should perform a task:

1. **Select Robot:** Choose robot i based on capability, location, availability, and load balancing
2. **Formulate Goal:** Create goal specification $g_t^i = (\text{task\_type}, \text{target}, \text{requirements})$
3. **Determine Constraints:** Generate constraint set c_t^i including:
   - Spatial constraints (paths to avoid, zones to remain in)
   - Temporal constraints (earliest/latest start time, maximum duration)
   - Interaction constraints (spaces to avoid if occupied, escalation triggers)
   - Policy constraints (authorization levels, required approvals)
4. **Transmit Assignment:** Send (g_t^i, c_t^i) to robot i via the interface protocol

6.2 Constraint Propagation

Constraints from the operational layer are translated into robot-executable form:

Spatial Constraints:

- Inclusion zones: “Remain within corridors A, B, and C”
- Exclusion zones: “Do not enter the lobby between 7:00-9:00 AM”

- Dynamic exclusion: “Do not approach Room 205 (DND active)”

Temporal Constraints:

- Time windows: “Deliver between 2:00-4:00 PM”
- Relative timing: “Arrive after housekeeping completes Room 312”
- Duration limits: “Complete within 10 minutes”

Interaction Constraints:

- Yield rules: “Yield to housekeeping carts and luggage trolleys”
- Distance rules: “Maintain 2m distance from guests unless delivering”
- Escalation triggers: “Stop and escalate if guest approaches”

6.3 Observation Reporting

As robots execute tasks, they report observations that update the venue belief state:

Task Progress Observations:

- `departed(location)`: Robot has left starting location
- `arrived(location)`: Robot has reached destination
- `action_started(action)`: Robot has begun manipulation
- `action_completed(action, outcome)`: Robot has finished manipulation

Environmental Observations:

- `occupancy_detected(location, confidence)`: Robot detected people in space
- `obstacle_detected(location, type)`: Robot encountered unexpected obstacle
- `anomaly_detected(description)`: Robot detected unusual condition

State Verification Observations:

- `room_status_verified(room, status)`: Robot confirmed room state
- `guest_presence_verified(location, present)`: Robot confirmed guest presence
- `equipment_status_verified(equipment, status)`: Robot confirmed equipment state

6.4 Belief Update Cycle

The venue operational layer updates its belief state based on robot observations:

$$b_{t+1}(s^{\text{operational}}) \propto P(o_t^{\text{robot}}|s^{\text{operational}}) \cdot P(s^{\text{operational}}|s_{t-1}^{\text{operational}}, a_t^{\text{venue}}) \cdot b_t(s^{\text{operational}}) \tag{8}$$

Robot observations provide evidence about operational state:

- Robot reports room is occupied → Update room occupancy belief
- Robot detects water on floor → Update maintenance alert status
- Robot observes housekeeping cart → Update housekeeping location

6.5 Handling Heterogeneity

The protocol accommodates robots from different manufacturers with varying capabilities:

Capability Registration: Each robot registers its capabilities upon connection:

- Navigation: maximum speed, terrain types, stairs/elevators
- Manipulation: grasp types, payload limits, precision
- Sensing: camera, LiDAR, microphones, other sensors
- Communication: update frequency, bandwidth limits

Graceful Degradation: If a robot cannot handle a constraint type, the operational layer:

- Adjusts the task to match robot capabilities
- Assigns additional human oversight
- Routes the task to a more capable robot

7 Robots as Active Sensors

Robots can improve the venue world model through their physical presence, acting as mobile sensors that explore uncertain regions and detect anomalies.

7.1 Exploration of Uncertain Regions

When the venue belief state has high uncertainty about a region, robots can be directed to explore:

- **Uncertainty-Driven Routing:** Route robots through spaces with high state uncertainty to gather observations
- **Information-Gathering Tasks:** Explicitly assign robots to verify uncertain states ("Check if Room 412 is vacant")
- **Patrol Patterns:** Systematic exploration of low-traffic areas to maintain up-to-date state estimates

7.2 Anomaly Detection

Robots can detect anomalies that static sensors miss:

Physical Anomalies:

- Spills, leaks, or debris not visible to ceiling cameras
- Equipment failures indicated by unusual sounds or vibrations
- Temperature variations indicating HVAC issues

Behavioral Anomalies:

- Guests appearing lost or distressed
- Unusual congestion patterns
- Unauthorized access attempts

Operational Anomalies:

- Tasks taking significantly longer than expected
- Unexpected occupancy in supposedly vacant rooms
- Discrepancies between expected and observed equipment states

7.3 Updating Venue Belief Through Embodied Observation

When a robot detects an anomaly, the venue belief updates:

1. Robot reports observation o^{anomaly}
2. Operational layer infers likely state change: $P(s^{\text{new}}|o^{\text{anomaly}})$
3. Belief updates to incorporate new information
4. Operational layer may trigger responses:
   - Create maintenance task
   - Alert security
   - Adjust scheduling for affected areas
   - Escalate to human manager

8 Pre-Robot Data as Institutional Memory

Operational data gathered before robot arrival becomes training data and institutional memory for later embodied systems.

8.1 Learning Human Operational Patterns

The venue world model learns from years of human operations:

Task Patterns:

- Typical task sequences and dependencies
- Time-of-day and day-of-week patterns
- Seasonal variations in operations
- Response patterns to disruptions

Coordination Patterns:

- How staff coordinate during busy periods
- How priorities shift during disruptions
- How communication flows during incidents

Service Quality Patterns:

- Guest satisfaction correlates with operational factors
- Timing and sequencing that optimizes service quality
- Warning signs of service degradation

8.2 Training Robot Policies

This historical data enables:

Imitation Learning: Train robot policies to mimic successful human operational patterns

Dynamics Learning: Learn transition models for how venue state evolves under different conditions

Value Learning: Learn reward functions that capture operational priorities and service quality

Anomaly Detection: Identify when current operations deviate from learned patterns

8.3 Transfer from Human Operations to Robot Execution

When robots are introduced, the venue model provides:

- **Operational context:** "This is how we handle VIP arrivals"
- **Timing expectations:** "Deliveries to occupied rooms should occur between 2-4 PM"
- **Coordination protocols:** "Always yield to housekeeping during morning rush"
- **Escalation criteria:** "Escalate to manager if guest complaint detected"

The robots inherit the venue's institutional memory through the operational layer, rather than learning it from scratch.

9 Detailed Operational Scenarios

We present six detailed scenarios demonstrating the necessity of operational world models. For each scenario, we specify the initial state, robot capabilities, operational context required, decision process, and failure modes without the venue model.

9.1 Scenario 1: Towel Delivery with Uncertain Room State

Task: Deliver fresh towels to Room 312.

Robot Capabilities: Can navigate to room, open door with RFID, place towels, exit.

Operational Context Required:

- Is Room 312 occupied? If yes, is guest present or out?
- Is housekeeping currently cleaning Room 312?
- Does Room 312 have active DND (Do Not Disturb)?
- Is there a VIP arrival scheduled in adjacent Room 310 within next 30 minutes?
- Is there a maintenance issue in Room 312 (leak, electrical problem)?

Decision Process with Venue Model:

1. Venue model checks PMS: Room 312 occupied, guest out (checked out for dinner)
2. Venue model checks task system: Housekeeping completed Room 312 at 2:15 PM
3. Venue model checks guest preferences: No DND on file
4. Venue model checks schedule: VIP arrival in Room 310 at 3:00 PM
5. Decision: Robot can deliver now but must complete before 2:45 PM to avoid VIP arrival

6. Constraint assigned: Complete by 2:45 PM, minimize noise near Room 310

Failure Modes Without Venue Model:

- Robot enters during housekeeping → disrupts staff workflow
- Robot enters while guest showering → privacy violation, complaint
- Robot arrives during VIP arrival → congestion, poor first impression
- Robot places towels on wet counter from leak → towels ruined, guest complaint

9.2 Scenario 2: Elevator Competition During Check-In Rush

Task: Three delivery robots need to reach floors 5, 8, and 12 during 3:00-4:00 PM check-in rush.

Robot Capabilities: Can call elevators, enter, navigate to floors.
Operational Context Required:

- Current elevator demand from guests with luggage
- Expected check-in queue length (affects elevator wait times)
- Alternative routes (service elevators, stairwells for small robots)
- Task priorities (which delivery is most urgent?)
- Robot capabilities (which robots can use service elevators?)

Decision Process with Venue Model:

1. Venue model predicts elevator wait times: 5+ minutes for guest elevators
2. Venue model checks service elevator availability: Available for robots
3. Venue model checks task priorities: Floor 8 delivery is for VIP → highest priority
4. Venue model assigns routes:
   - Robot A (Floor 8 VIP): Service elevator, immediate departure
   - Robot B (Floor 5): Service elevator, depart after Robot A
   - Robot C (Floor 12): Guest elevator (least urgent), wait for demand lull

Failure Modes Without Venue Model:

- All robots compete for guest elevators → long waits, guest frustration
- VIP delivery delayed by congestion → service quality impact
- Robots block elevator lobbies while waiting → guest flow disruption
- Service elevators underutilized while guest elevators overwhelmed

9.3 Scenario 3: Restaurant Robot Reprioritization After Kitchen Delay

Task: Multiple food delivery robots serving tables during dinner service.

Operational Context Required:

- Kitchen status (on-time, delayed, backed up)
- Table status (just seated, waiting for food, ready for dessert)
- Food readiness (which orders are ready, which delayed)
- Server workload (which tables need attention)
- Guest urgency (VIP tables, guests with time constraints)

Decision Process with Venue Model:

1. Kitchen system reports 15-minute delay on entrees
2. Venue model reassesses priorities:
   - Tables waiting for appetizers → high priority (get them something)
   - Tables with entrees ready → immediate delivery
   - Tables with delayed entrees → inform server to communicate delay
   - Tables ready for dessert → offer dessert while waiting
3. Robot fleet reassigned: Some robots deliver ready food, others support server communication

Failure Modes Without Venue Model:

- Robots deliver to tables with delayed orders → guests confused about missing food
- Robots ignore tables with ready appetizers → food gets cold, quality suffers
- No communication of delay → guest frustration, complaints
- Servers manually coordinate robots during crisis → overwhelmed, service degrades

9.4 Scenario 4: Leak Discovery and Coordinated Response

Task: Cleaning robot discovers water leak in hallway.

Operational Context Required:

- Leak severity and source (minor spill vs. pipe burst)
- Affected areas (rooms, electrical, guest paths)
- Maintenance staff availability and location
- Guest impact (which rooms affected, relocation needed?)
- Safety implications (slip hazard, electrical risk)

Decision Process with Venue Model:

1. Robot detects leak, captures image/video
2. Robot reports to venue model: "Water detected, Hallway 3C, 2 gallons"

3. Venue model assesses:
    - Rooms 312-318 adjacent to leak
    - Guest in Room 315 (VIP) → prioritize
    - Maintenance staff Joe on Floor 2, available in 5 minutes
    - Housekeeping Maria nearby → can place warning signs
4. Venue model coordinates response:
    - Alert Joe (maintenance) with location and image
    - Alert Maria (housekeeping) to place warning signs
    - Alert manager for potential guest relocation decision
    - Robot remains as sentinel, warns approaching guests

Failure Modes Without Venue Model:

- Robot reports leak but no coordination → maintenance arrives without context
- No guest warning system → slip hazard until response arrives
- VIP guest discovers leak first → service quality disaster
- Multiple departments respond without coordination → confusion, delay

9.5 Scenario 5: Covering an Absent Employee's Shift

Task: Cover tasks normally performed by housekeeping staff member who called in sick.

Operational Context Required:

- Absent employee's assigned rooms and priorities
- Room statuses (which need cleaning, which are stay-overs vs. check-outs)
- VIP rooms requiring special attention
- Other staff availability and workload
- Robot capabilities (which tasks can robots perform?)

Decision Process with Venue Model:

1. Venue model identifies absent employee's room assignments
2. Venue model categorizes by urgency:
    - Check-outs (must be ready by 3 PM) → highest priority
    - VIP stay-overs → high priority
    - Regular stay-overs → medium priority
3. Venue model assigns tasks:
    - Robots: Deliver amenities, light tidying, towel exchange (robot-capable)
    - Human staff: Full cleaning, bed making, bathroom sanitization
    - Cross-train: Pull staff from less urgent areas
4. Venue model monitors progress and reassigns as needed

Failure Modes Without Venue Model:

- Robots assigned to tasks beyond capability → incomplete service
- High-priority check-outs delayed → guest check-in problems
- Human staff overloaded, robots underutilized → inefficiency, burnout
- No monitoring of coverage progress → some rooms missed entirely

9.6 Scenario 6: Refusing a Physically Possible Action

Task: Robot is instructed to enter Room 205 to deliver package.
Operational Context:

- Room 205 shows occupied in PMS
- Guest in Room 205 has medical emergency (paramedics en route)
- Guest's spouse at front desk explicitly requested no interruptions
- Package is non-urgent (guest's personal item they forgot in lobby)

Robot Capability: Can physically navigate to room and enter.
Decision Process with Venue Model:

1. Robot requests authorization to enter Room 205
2. Venue model checks operational state:
    - Medical emergency flag on Room 205
    - Guest request: "Do Not Disturb - Medical Emergency"
    - Package priority: Low (personal item, non-urgent)
3. Venue model decides: Entry would violate policy and guest request
4. Venue model responds to robot: `REFUSE` with reason: "Medical emergency, guest requested no interruptions"
5. Venue model creates alternative: Hold package at front desk, deliver after emergency resolved

Failure Modes Without Venue Model:

- Robot enters during medical emergency → interferes with paramedics, violates guest request
- Robot delivers package but creates distress → service quality disaster
- No record of guest request → robot acts on outdated information
- Robot cannot interpret urgency → treats all tasks as equally important

10 VenueBench: Beyond Navigation and Manipulation

We propose **VenueBench**, a benchmark evaluating robots on operational capabilities beyond traditional navigation and manipulation metrics.

10.1 Design Principles

VenueBench evaluates:

1. **Operational awareness, not just spatial awareness**
2. **Appropriate action, not just successful action**
3. **Coordination, not just individual performance**
4. **Service quality preservation, not just task completion**

10.2 Task Taxonomy

Table 1: VenueBench Task Taxonomy

Category	**Description**	**Metrics**
Operational State Inference	Infer hidden state from partial observations	Accuracy, calibration, early detection
Task Prioritization	Select appropriate task ordering under constraints	Service quality, completion rate, wait time
Human-Robot Coordination	Coordinate with human staff and guests	Fluency, safety, satisfaction
Multi-Robot Scheduling	Coordinate multiple robots without conflicts	Efficiency, conflict rate, coverage
Counterfactual Planning	Evaluate "what if" scenarios before acting	Prediction accuracy, decision quality
Permission Awareness	Respect access controls and guest preferences	Violation rate, policy compliance
Safe Escalation	Identify when to stop and request human help	Escalation appropriateness, safety
Recovery from Uncertainty	Handle incomplete or contradictory information	Robustness, recovery success

10.3 Evaluation Protocols

Simulation Environment: High-fidelity simulation of hotel/restaurant operations with:

- Realistic guest and staff behavior models
- Dynamic events (check-ins, requests, disruptions)
- Operational data streams (PMS, POS, task systems)
- Physics-based robot simulation

Scenario-Based Evaluation: Predefined scenarios representing common operational situations:

- Rush hour coordination
- Disruption response
- VIP service delivery

- Emergency handling
- Multi-robot coordination

Human-in-the-Loop: Human operators interact with simulated robots to evaluate:

- Task acceptance rates
- Intervention frequency
- Subjective assessment of appropriateness

10.4 Metrics

Primary Metrics:

- **Operational Appropriateness Score:** Human-rated assessment of action appropriateness
- **Service Quality Preservation:** Guest satisfaction outcomes in simulation
- **Policy Compliance Rate:** Percentage of actions respecting venue policies
- **Coordination Fluency:** Smoothness of human-robot and robot-robot coordination

Secondary Metrics:

- Task completion rate
- Response time
- Escalation appropriateness
- Recovery success rate after failures

11 Safety, Permissions, and Governance

We analyze critical safety, privacy, labor, and governance considerations.

11.1 Guest Privacy

Observation Limits:

- What can robots observe and record?
- Are guest rooms off-limits for recording?
- How long are observations retained?
- Who has access to robot sensor data?

Policy Recommendations:

- Robots should not record in guest rooms
- Observations should be anonymized when possible
- Data retention should be limited and policy-governed
- Guests should be informed of robot presence and capabilities

11.2 Labor Implications

Displacement vs. Augmentation:

- Are robots replacing workers or assisting them?
- What tasks remain human-only?
- How are workers trained to collaborate with robots?

Policy Recommendations:

- Prioritize augmentation over displacement
- Retain human roles requiring judgment, creativity, emotional intelligence
- Invest in worker training for human-robot collaboration
- Establish clear task boundaries between humans and robots

11.3 Liability Allocation

Questions:

- Who is liable when a robot causes harm?
- Robot manufacturer? Venue operator? Software provider?
- How does venue model guidance affect liability?

Policy Recommendations:

- Establish clear liability frameworks before deployment
- Require insurance coverage for robot operations
- Document decision trails for accountability
- Define escalation protocols for high-stakes decisions

11.4 Cybersecurity

Attack Surfaces:

- Compromised robots could disrupt operations
- Manipulated venue models could cause inappropriate robot actions
- Sensor spoofing could corrupt venue belief state

Security Measures:

- Encrypted communication between layers
- Authentication for robot-venue connections
- Anomaly detection for unusual robot behavior
- Graceful degradation under attack

11.5 Human Override

Requirements:

- Humans must be able to override robot actions at any time
- Emergency stop mechanisms must be accessible
- Override events should be logged and analyzed

12 Economic and Technical Thresholds

We analyze when robots become economically viable for hospitality venues.

12.1 Cost-Benefit Framework

Robot Costs:

- Capital cost (purchase/lease)
- Integration cost (connecting to venue systems)
- Maintenance and support
- Training and onboarding

Benefits:

- Labor cost reduction (or reallocation)
- Service quality improvements
- Operational efficiency gains
- 24/7 availability for certain tasks

12.2 Technical Prerequisites

For viable deployment, the following must be mature:

- **World model accuracy:** Venue belief state must be sufficiently accurate for reliable decision-making
- **Robot reliability:** Robots must achieve high uptime and low failure rates
- **Integration capability:** Robots must connect effectively to venue operational layer
- **Safety assurance:** Risk of harm to guests or staff must be acceptably low

12.3 Deployment Tipping Points

Hotels:

- Large properties (200+ rooms) with repetitive delivery tasks
- Properties with 24-hour operations
- High-labor-cost markets
- Properties with existing operational technology infrastructure

Restaurants:

- Large establishments with predictable traffic patterns
- Quick-service environments with standardized tasks
- Properties with severe labor shortages

13 Roadmap

We outline a five-phase roadmap for realizing the vision of operational world models controlling hospitality robotics.

13.1 Phase 1: World Model Learns from Human Operations (Current)

Focus: Build operational world model using existing human operations data.

Activities:

- Integrate PMS, POS, scheduling, task management systems
- Learn operational patterns from historical data
- Validate belief state accuracy against ground truth
- Establish baseline operational performance metrics

Outcome: Venue has functioning operational world model with no robots yet deployed.

13.2 Phase 2: Software Agents Take Bounded Actions

Focus: Deploy software agents (no physical robots) that use the operational model.

Activities:

- Automated task assignment and scheduling
- Proactive alerts and recommendations to human staff
- Limited autonomous actions (notifications, low-risk adjustments)
- Validate decision quality and safety

Outcome: Venue operational model proven effective for decision support.

13.3 Phase 3: Specialized Robots Plug Into Shared Model

Focus: Introduce specialized robots (delivery, cleaning) connected to operational model.

Activities:

- Deploy single-purpose robots for well-defined tasks
- Connect robots to venue operational layer
- Train staff on human-robot collaboration
- Monitor and refine coordination protocols

Outcome: Robots operating effectively under operational model direction.

13.4 Phase 4: Heterogeneous Human-Robot Teams Coordinate

Focus: Scale to diverse robot types and full human-robot team coordination.

Activities:

- Deploy multiple robot types from different manufacturers
- Establish coordination protocols for mixed teams
- Optimize task allocation across humans and robots
- Develop advanced operational capabilities (predictive scheduling, dynamic prioritization)

Outcome: Fully integrated human-robot teams coordinated through operational model.

13.5 Phase 5: Persistent Intelligence Layer Across Locations

Focus: Operational model becomes portable across venues and persistent over time.

Activities:

- Transfer learning: deploy learned models to new venues
- Multi-venue coordination for hospitality chains
- Continuous learning and model improvement
- Integration with broader hospitality ecosystem

Outcome: Operational world model as enduring platform, independent of specific robots or venues.

14 Limitations and Falsification

We identify assumptions, failure modes, and experiments that could falsify our thesis.

14.1 Technical Assumptions That Could Fail

Assumption 1: Operational world models can achieve sufficient accuracy for reliable robot control.

Falsification: If venue belief state accuracy remains below 70% after 2 years of data collection, the approach may be infeasible.

Assumption 2: Robots can reliably connect to and follow guidance from external operational models.

Falsification: If network latency or robot computational limitations prevent real-time coordination, the architecture may be impractical.

Assumption 3: Human staff will accept and effectively collaborate with robot-directed operations.

Falsification: If staff acceptance remains below 50% after appropriate training, the human-robot coordination model may require fundamental revision.

14.2 Conditions Under Which the Thesis Breaks Down

Condition 1: Robot foundation models develop sufficient operational reasoning internally.

If future VLA models can learn operational context from pretraining (e.g., reading hotel operations manuals, watching training videos), the need for separate operational models may diminish.

Condition 2: Hospitality operations become so standardized that operational context is unnecessary.

If all hotels adopt identical operational procedures and robots can be pre-trained on these standards, venue-specific models may be unnecessary.

Condition 3: Guest and staff behavior is too unpredictable for model-based coordination.

If hospitality operations are inherently too chaotic for any model to capture effectively, the operational world model approach may fail.

14.3 Experiments That Could Falsify Key Claims

Experiment 1: Compare robot performance with and without operational model guidance.

Prediction: Robots with operational model guidance achieve higher service quality scores. *Falsification:* No significant difference in service quality.

Experiment 2: Test whether venue-specific operational knowledge transfers across properties.

Prediction: Models trained on one venue improve performance at similar venues. *Falsification:* No transfer; each venue requires complete retraining.

Experiment 3: Measure staff acceptance and collaboration effectiveness.

Prediction: Staff acceptance increases with exposure and training. *Falsification:* Staff acceptance decreases or remains low regardless of exposure.

14.4 Skeptical Assessment of Timelines

We acknowledge significant uncertainty in deployment timelines:

Near-term (2-5 years): Software agents using operational models are feasible. Limited robot deployment in controlled environments is possible.

Medium-term (5-10 years): Heterogeneous robot coordination remains challenging. Technical hurdles (reliability, safety certification) may delay widespread deployment.

Long-term (10+ years): Fully autonomous human-robot teams depend on advances in both robot hardware and AI reasoning. Timelines are speculative.

15 Conclusion

We have established the formal necessity of operational world models as the control plane for hospitality robotics. The central arguments are:

1. **Physical AI is insufficient:** Robot foundation models can answer "Can I perform this action?" but not "Should I perform this action?" The gap between physical capability and operational appropriateness is fundamental.

2. **Three intelligences are required:** Spatial intelligence (navigation), physical intelligence (manipulation), and operational intelligence (context, priorities, policies) are non-substitutable. Current advances address the first two; the third requires venue-level operational models.

3. **Formal framework enables rigorous analysis:** The Dec-POMDP framework with coupled embodied and operational layers provides mathematical structure for analyzing robot-venue coordination.

4. **Operational scenarios demonstrate necessity:** The six detailed scenarios show how optimizing robot task completion alone produces locally successful but operationally harmful behavior.

5. **VenueBench defines the evaluation challenge:** Traditional robotics benchmarks are insufficient. VenueBench evaluates operational awareness, appropriate action, and service quality preservation.

15.1 The Defensible Thesis

We conclude with the thesis stated at the outset, now grounded in the arguments developed:

> **The enduring robotics platform in hospitality may not be the company that builds every body. It may be the system that already understands the venue when those bodies arrive.**

Physical AI companies will produce increasingly capable robots. But these robots will arrive as capable bodies without operational minds. The hotel or restaurant that has invested years in building an operational world model—learning from human operations, encoding institutional memory, establishing coordination protocols—will be ready to absorb and direct these robots effectively.

The venue without such a model will struggle to deploy robots productively, regardless of the robots' physical capabilities. The robot that can navigate hallways and grasp objects will still fail if dispatched to the wrong room at the wrong time for the wrong reason.

The body needs a manager. The operational world model is that manager.

15.2 A Call for Research

We call on researchers in robotics, AI, operations research, and hospitality management to:

- Develop the technical foundations for operational world models
- Create VenueBench and establish evaluation standards
- Study human-robot collaboration in service environments
- Address safety, privacy, and labor implications
- Build the institutional memory systems that will enable future robotics

The robots are coming. The question is not whether they will arrive, but whether we will be ready to manage them when they do.

Acknowledgments

This paper (Part II) builds directly on the foundations established in "World Models for Hospitality: A Research Agenda for Agentic Venue Operations" (Part I). Together, these papers present a comprehensive research agenda for intelligent hospitality operations spanning both software-only and embodied AI systems.

References

Brohan, A., Brown, N., Carbajal, J., et al. (2023). RT-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*.

Driess, D., Xia, F., Sajjadi, M. S. M., et al. (2023). PaLM-E: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*.

Kim, M. J., Pertsch, K., Karamcheti, S., et al. (2024). OpenVLA: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*.

Li, H., Chen, Y., Cui, W., et al. (2025). Large VLM-based vision-language-action models for robotic manipulation: A survey. *arXiv preprint arXiv:2508.13073*.

Shao, R., Li, W., Zhang, L., et al. (2024). Survey of vision-language-action models for embodied manipulation. *arXiv preprint arXiv:2508.15201*.

Tessera, K., Hinckeldey, L., Zamboni, R., et al. (2026). Probing Dec-POMDP reasoning in cooperative MARL. *arXiv preprint arXiv:2602.20804*.

You, Y., Schutz, A., Li, Z., et al. (2025). Scalable solution methods for Dec-POMDPs with deterministic dynamics. *arXiv preprint arXiv:2508.21595*.

Lauri, M., Pajarinen, J., and Peters, J. (2019). Information gathering in decentralized POMDPs by policy graph improvement. *arXiv preprint arXiv:1902.09840*.

Zhu, C., Dastani, M., and Wang, S. (2024). A survey of multi-agent deep reinforcement learning with communication. *arXiv preprint arXiv:2203.08975*.

Ardon, L., Furelos-Blanco, D., and Russo, A. (2023). Learning reward machines in cooperative multi-agent tasks. *arXiv preprint arXiv:2303.14061*.

Colan, J., Davila, A., Yamada, Y., et al. (2025). Human-robot collaboration in surgery: Advances and challenges towards autonomous surgical assistants. *arXiv preprint arXiv:2507.11460*.

Gregory, J. M., Reardon, C., Lee, K., et al. (2019). Enabling intuitive human-robot teaming using augmented reality and gesture control. *arXiv preprint arXiv:1909.06415*.

Reimann, M. M., Hindriks, K. V., Kunneman, F. A., et al. (2025). What can you say to a robot? Capability communication leads to more natural conversations. *arXiv preprint arXiv:2502.01448*.

Pineda, K. T., Brown, E., and Huang, C. M. (2025). "See you later, alligator": Impacts of robot small talk on task, rapport, and interaction dynamics in human-robot collaboration. *arXiv preprint arXiv:2501.13233*.

Kim, M. J., Finn, C., and Liang, P. (2025). Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*.

Li, Q., Liang, Y., Wang, Z., et al. (2024). CogACT: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*.

Jeong, J., Zhu, E., Lin, J., et al. (2026). Your vision-language-action model already has attention heads for path deviation detection. *arXiv preprint arXiv:2603.13782*.

Zhong, L., Liu, Y., Wei, Y., et al. (2026). ACoT-VLA: Action chain-of-thought for vision-language-action models. *arXiv preprint arXiv:2601.11404*.

Wang, C., Bao, W., Gao, S., et al. (2026). VLA-Thinker: Boosting vision-language-action models through thinking-with-image reasoning. *arXiv preprint arXiv:2603.14523*.