

The City College of New York

CSC 59867-E: Senior Project II

AI Agents for Decision Making in the Real World

Supply Chain Logistics and Orchestration

By Saptarashmi Bandyopadhyay
Email: sbandyopadhyay@ccny.cuny.edu
Assistant Professor of Computer Science
City College of New York and Graduate Center at the City University of New York

Saptarashmi Bandyopadhyay

October 7, 2026 CSC 59867 - E

Lecture Agenda

Dynamic Programming Foundations

01

Modeling replenishment as stochastic optimal control under non-zero lead times and state augmentation.

Automated Paradigms & Micro-Logistics

02

Contrasting Person-to-Goods (PTG) with Goods-to-Person (GTP) facilities and evaluating the systemic limits of rigid industry heuristic

MARL for Warehouse Fleets

03

Structuring high-density order picking as a Partially Observable Stochastic Game (POSG) with invalid action maskin

Scalable Fleet Coordination

04

Decoupling routing from goal assignment via manager-worker hierarchies.

Saptarshi Bandyopadhyay

Physical Flow Networks vs. Financial Markets

Physical inventory systems **conserve physical mass and incur non-zero transport lead times**, unlike financial asset exchanges that clear trades in sub-millisecond digital books.

Delayed state observability dominates physical supply chains because **goods in transit remain completely unavailable for downstream demand** until arrival.

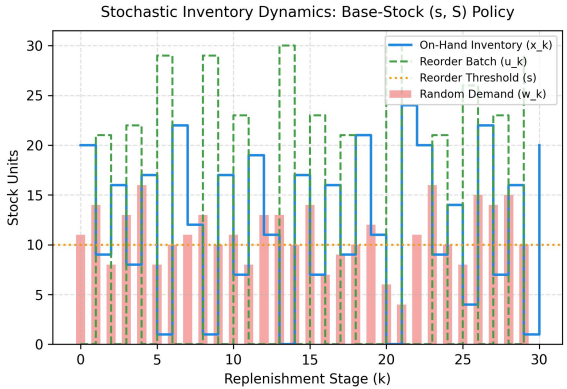
Reversal of bad decisions is fundamentally asymmetric because **unmet customer demand causes permanent lost sales or backlog penalties**, while excess stock incurs continuous physical depreciation and holding fees.

Large-scale shipping centers require coordinating heterogeneous workers, as **autonomous mobile robots and human coordinators must physically meet at matching spatial coordinates** to transfer items.

Saptarashmi Bandyopadhyay

Stochastic Inventory Control as an MDP

- Dynamic inventory replenishment evolves according to **the stock conservation equation driven by random customer demand.**
- The state variable x□ denotes current on-hand inventory at period k, the control u□ denotes the replenishment ordered, and w□ represents the stochastic demand disturbance. c is the unit purchase/procurement cost. S is the target order-up-to level.
- Unmet demand is turned away or backlogged, causing **the next state transition to clip non-linearly at zero**.
- Single-stage cost functions penalize capital expenditure and storage imbalances by **combining linear purchasing expenses with asymmetric piecewise-linear holding and stockout penalties**.



$$x_{k+1} = \max(0,\ x_k + u_k - w_k)$$

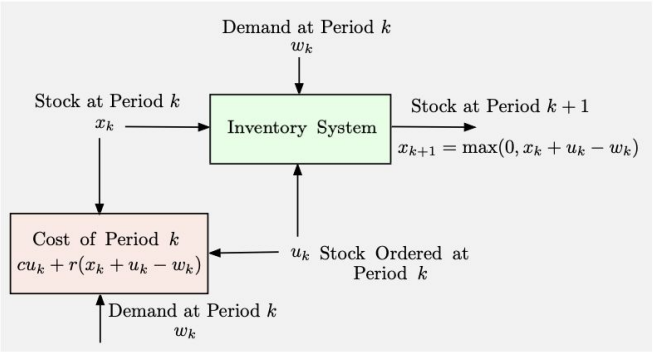
$$g_k(x_k, u_k, w_k) = cu_k + r(x_k + u_k - w_k), \quad r(z) = \beta^+ \max(0, z) + \beta^- \max(0, -z)$$

β+ is the holding cost for excess stock (z > 0), and β- is the shortage penalty for unmet demand (z < 0)

Sarbartha Mukherjee Bandyopadhyay

Lead Times and State Augmentation

- Delivery delays break Markov assumptions because **orders placed at stage k only reach inventory after transit delay L**.
- The system restores Markovian structure through state augmentation by **expanding the state vector to track the entire active pipeline of in-transit orders**.
- The augmented state variable $\tilde{x}_k$ bundles on-hand inventory with historical unfulfilled orders (x_k, u_{k-1}, ..., u_{k-L}).
- Dimensionality grows linearly with replenishment latency, which causes **exact dynamic programming to suffer from an exponential state-space explosion**.



$$x_{k+1} = f_k(x_k,\, u_k,\, u_{k-1},\, \ldots,\, u_{k-L},\, w_k) \quad \tilde{x}_k = \begin{pmatrix} x_k \\ u_{k-1} \\ \vdots \\ u_{k-L} \end{pmatrix} \implies \tilde{x}_{k+1} = \begin{pmatrix} f_k(x_k, u_k, u_{k-1}, \ldots, u_{k-L}, w_k) \\ u_k \\ \vdots \\ u_{k-L+1} \end{pmatrix}$$

Sankarshan Bandyopadhyay

Multi-Agent Rollout and Combinatorial Scaling

Decentralized logistics facilities assign decisions to m distinct local dispatch agents, causing **the joint control space to scale exponentially as n^m for n local options**.

Standard dynamic programming lookahead breaks down because **evaluating all joint Q-factors requires intractable compute between operational stages**.

Multi-agent rollout replaces joint enumeration with **sequential agent-by-agent minimization that incorporates previous decisions into subsequent choices**.

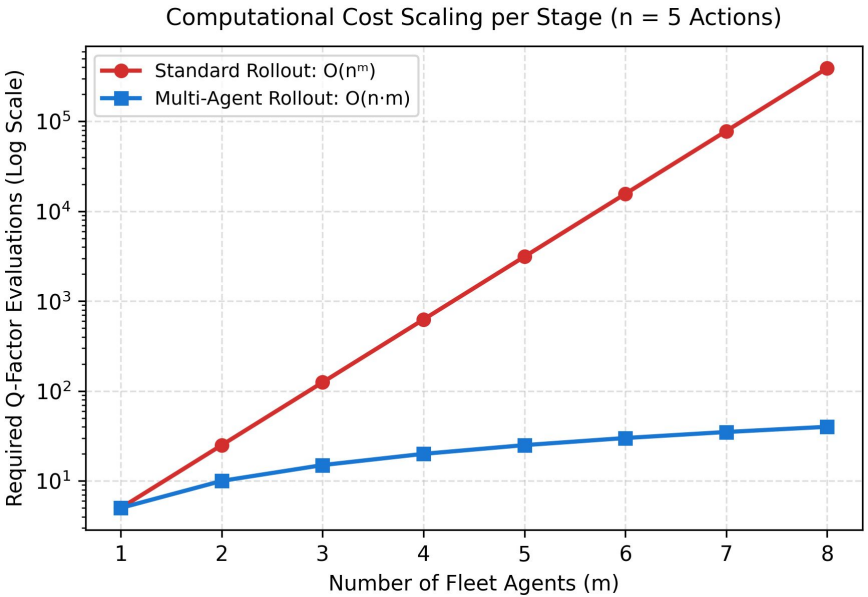
Factorizing the joint action space reduces complexity from n^m to nm while **strictly preserving the fundamental policy cost improvement property**.

$$\tilde{\mu}_l(x) \in \arg \min_{u_l \in U_l(x)} \mathbb{E}_w \left\{ g(x, \tilde{\mu}_1, \dots, u_l, \dots, \mu_m, w) + \alpha J_\mu(f(x, \tilde{\mu}_1, \dots, u_l, \dots, \mu_m, w)) \right\}$$

$$\mathcal{O}(n^m) \text{ Joint Q-Factors} \xrightarrow{\text{Agent-by-Agent Unfolding}} \mathcal{O}(n \cdot m) \text{ Sequential Q-Factors}$$

Sapta Rashmi Bandyopadhyay

Simulated Example: Single vs. Multi-agent rollout



MARL for Supply Chain Logistics

Saptarashmi Bandyopadhyay

Warehouse Paradigms

Modern distribution centers optimize the order-picking problem to **maximize the operational pick rate measured in completed order-lines per hour**.

Person-to-Goods (PTG) environments augment human coordinators with Autonomous Guided Vehicles (AGVs) that **transport consolidated orders prepared by humans**.

Goods-to-Person (GTP) facilities employ mobile robots to **carry entire storage shelving pods directly to stationary pick stations**.

Coordination failures cause severe bottlenecks, as **idle delays accumulate whenever transport vehicles and humans arrive out of sync**.

Sabtarshmi Bandyopadhyay

Example: Multi-Robot Coordination

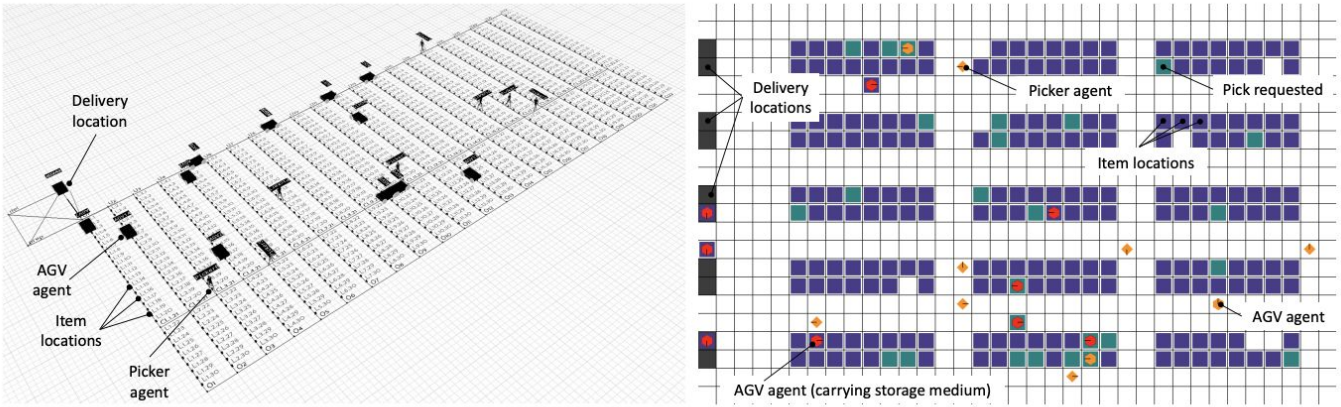


Fig. 1. Left: Dematic PTG simulator with human pickers and AGVs. Right: TA-RWARE GTP simulator with picking bots (diamond) and AGVs (hexagon).

Krnjaic et al., 2024 (*MARL for Warehouse Logistics*), Fig. 1

Structural Failure of Classical Industry Heuristics

The Follow-Me (FM) heuristic binds specific AGVs to follow assigned humans, which **forces transport vehicles to travel redundant loops and causes transport congestion**.

The Pick, Don't Move (PDM) heuristic confines workers to static zones, which **triggers chronic worker under-utilization when incoming orders skip their aisles**.

The Closest Task Assignment (CTA) heuristic greedily routes nearest available robots to open storage totes, which **causes dense robot clustering and severe physical intersection deadlocks**.

Fixed heuristic rules lack dynamic responsiveness because **they cannot adapt to shifting spatial demand skews or irregular customer order profiles**.

Saptarashmi Bandyopadhyay

Warehouse Orchestration as a POSG

Saptarashmi Bandyopadhyay

Structural Failure of Classical Industry Heuristics

- Warehouse fleet orchestration is formally structured as a **Partially Observable Stochastic Game (POSG) defined over N heterogeneous worker agents**.
- The agent population separates into two distinct roles: automated guided vehicles V and humans workers P, yielding W = V ∪ P.
- Partial observability restricts each agent to **local observation histories h□ⁱ containing nearby worker locations, active order items, and vehicle payloads**.
- The cooperative objective targets a shared return by **maximizing total discounted pick rate while subtracting per-step operational latency penalties**.

$$\mathcal{G} = \left(\mathcal{I},\, \mathcal{S},\, \{\mathcal{A}^i\}_{i\in\mathcal{I}},\, \{\mathcal{O}^i\}_{i\in\mathcal{I}},\, \mathcal{P},\, \Omega,\, \{\mathcal{R}^i\}_{i\in\mathcal{I}}\right), \quad i \in \mathcal{I} = \{1, \ldots, N\}$$

$$\max_{\pi} \mathbb{E}\left[\sum_{t=1}^{T} \gamma^{t-1} r_t^i\right], \quad r_t^i = r_{\text{pick}} + r_{\text{delivery}} - c_{\text{step}}$$

Saptarashmi Bandyopadhyay

Action Space Pruning via Invalid Action Masking

Unconstrained navigation choices scale with all warehouse storage locations, causing **the raw action space to span thousands of discrete rack destinations**.

Agents waste exploration budget without constraints because **unmasked policies repeatedly select irrelevant shelves containing unrequested items**.

Invalid action masking prunes the output distribution by **setting policy network pre-softmax logits of invalid destinations to negative infinity**.

Dynamic masks enforce coordination by **restricting AGVs to active order lines and limiting pickers to locations with pending, unserviced vehicles**.

$$\tilde{\ell}(a^i) = \begin{cases} \ell(a^i) & \text{if } a^i \in \mathcal{A}^i_{\text{valid}}(s_t) \\ -\infty & \text{if } a^i \notin \mathcal{A}^i_{\text{valid}}(s_t) \end{cases} \implies \pi^i(a^i \mid h^i_t) = \frac{\exp(\tilde{\ell}(a^i))}{\sum_{a' \in \mathcal{A}^i} \exp(\tilde{\ell}(a'))}$$

Sattarashmi Bandyopadhyay

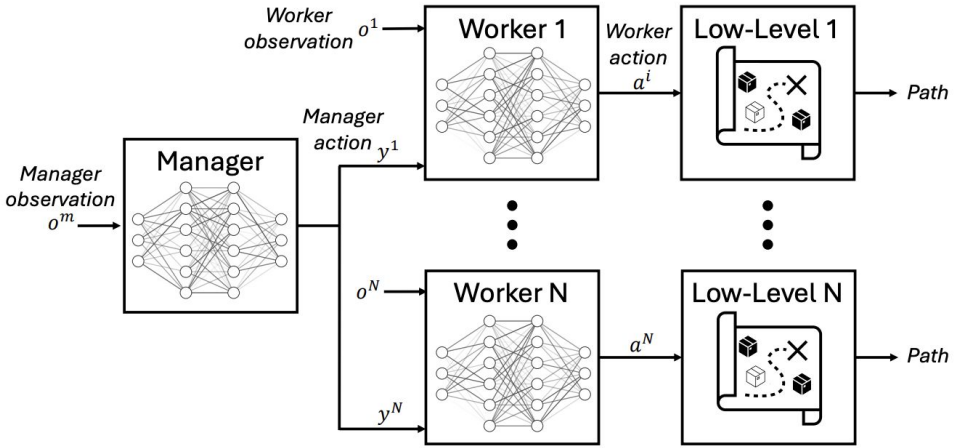
3-Layer Hierarchical MARL Architecture

Large physical environments require spatial abstraction because **routing robots directly across thousands of grid tiles creates intractable action spaces**.

The top-level Manager Agent observes global warehouse state and **partitions the facility into discrete functional zones assigned to workers**.

Worker agents receive assigned zones and **select specific storage rack locations within their designated partition**.

Low-level decentralized path controllers execute primitive movement steps by **calculating shortest collision-free grid paths using A* search**.



Krnjaic et al., 2024 (*MARL for Warehouse Logistics*), Fig. 2.

Saptarashmi Bandyopadhyay

Relevant Cooperative Training Paradigms

Independent Actor-Critic (IAC)

Trains separate policy and critic networks per agent, which **enables specialized behaviors but suffers from sample inefficiency**.

Shared Network Actor-Critic (SNAC)

Shares network weights across identical agent types, which **speeds up training but induces policy symmetry that causes robot path deadlocks**.

Shared Experience Actor-Critic (SEAC)

Maintains independent policies while **sharing off-policy experience trajectories across workers via auxiliary policy gradient updates**.

Hierarchical Wrapping

Creates HIAC, HSNAC, and HSEAC, which **breaks symmetric deadlocks because the manager assigns distinct spatial goals to identical workers**.

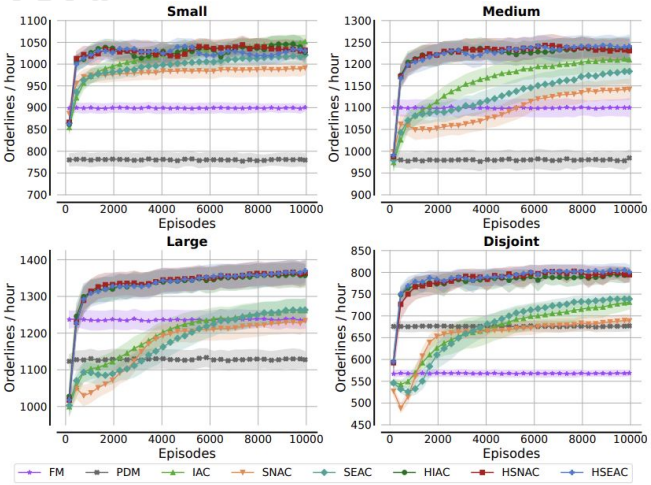
Results at Industrial Scale

Hierarchical MARL systems **outperform established industry heuristics across both PTG and GTP operating models**.

In large-scale Dematic PTG facilities, hierarchical models **exceed Follow-Me heuristics by up to 41.3% and Pick-Don't-Move by up to 26.5% in pick rate**.

In robotic TA-RWARE environments, hierarchical policies **surpass Closest Task Assignment by over 28% in pick rate while eliminating vehicle gridlocks**.

Sample efficiency accelerates under hierarchical decomposition because **partitioning action spaces enables policies to reach peak pick rates in fewer training episodes**.



Krnjaic et al., 2024 (*MARL for Warehouse Logistics*), Fig. 3

Saptarashmi Bandyopadhyay

Questions?

Saptarashmi Bandyopadhyay