The City College of New York

CSC 59866-E: Senior Project I

AI Agents for Decision Making in the Real World

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

April 29, 2026 CSC 59866

Advanced Topics: Software Coding Agents, Agentic RAG and Multimodal Embodied Audio-Vision-Language Agents

Logistics & The State of the Class

Recall Lecture 23: We explored software coding agents, agentic RAG, and digital embodiment across the web and open-ended games.

Today's Focus: We will transition from macro-scale physical systems (Supply Chains) to pure digital optimization (Stock Portfolios), and finally to micro-scale physical embodiment (Robotics).

Today's Agenda

1. **Real-World Supply Chain Logistics:** Distributed AI and Industry 4.0 (Sharifmousavi et al., 2024).
2. **Stock Portfolio Orchestration:** Deep RL and Multi-Agent Risk Management (Yu et al., 2019; Li & Tam, 2024).
3. **Physical AI Agents:** Foundation Models in Robotics and the Sim-to-Real Gap (Firoozi et al., 2024).

Real-World Supply Chain Logistics

Supply Chain 4.0

Traditional supply chains are highly linear and can easily be disrupted (e.g. 2021 Evergreen blocking the Suez)

In “Supply Chain 4.0” we can take advantage of IoT, cyber-physical systems, and big data to avoid these vulnerabilities.

One issue remaining is that scaling a centralized AI Agent coordination over tens of thousands of trucks is doomed to end in failure. We need a way to reduce the dependencies on one particular node.

Distributed Artificial Intelligence

We can use **Distributed AI** and **Multi-Agent Systems** to avoid the typical troubles that come from relying on one giant AI in the center of everything.

We can make every entity in the supply chain its own agent:

- *Suppliers -> Supplier Agent*
- *Inventory Managers -> Inventory Agents*
- *Vehicles and Logistics -> Vehicle and Logistic Agents*

If one Vehicle Agent goes down, the entire system doesn't collapse; instead, the Logistics Agent can simply pivot to the nearest available operational truck!

Logistics Opmization

At its core, supply chain logistics is a variation of the Vehicle Routing Problem (VRP). The global objective is to minimize total cost Z :

$$\min Z = \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij} + \sum_{i \in V} h_i I_i$$

Where:

- c_{ij} is the transport cost from node i to node j .
- x_{ij} is a binary decision variable (1 if route is used, 0 otherwise).
- h_i is the holding cost of inventory at node i .
- I_i is the inventory level.

The Agentic Shift: MAS approximates this global minimum through local, peer-to-peer negotiations (e.g., Contract Net Protocols) to satisfy constraints heuristically and rapidly.

Application: Agri-Food Supply Chains

Why use MAS in Agri-food? **Strict perishability constraints.**

Food supply chains must balance routing costs against the degradation rate of the product.

An *Inventory Agent* monitoring temperature IoT sensors detects a cooling failure. It autonomously negotiates with a *Routing Agent* to instantly reroute the truck to a closer distribution center, preventing total crop spoilage (Sharifmousavi et al., 2024).

Robustness and Fault Tolerance

- **Dynamic Routing:** MAS allows for dynamic routing strategies based on real-time disruptions.
- **Information Architecture:** Agents share quality management data continuously.
- Unlike centralized systems which freeze during server outages, a MAS exhibits graceful degradation. If one regional warehouse agent goes offline, neighboring agents adjust their supply requests autonomously.

Wrap-Up of Supply Chain Multi-Agent Systems

Takeaway: Physical logistics are too complex for centralized deterministic equations.

By decentralizing the math into multi-agent systems, supply chains become responsive, resilient, and adaptive.

Next, we take this exact same decentralized multi-agent approach and apply it to something completely digital, infinitely faster, and highly volatile: the stock market.

Stock Portfolio Orchestration

The Dynamic Stock Portfolio Optimization Problem

The Goal: Sequentially allocate wealth across a collection of assets (stocks, crypto, commodities) over consecutive trading periods to maximize returns while managing risk.

The Challenge: Financial markets are highly turbulent, non-stationary, and extremely noisy.

Traditional portfolio theory (Markowitz Mean-Variance) relies on historical covariance matrices that fail during sudden market crashes.

Single-Agent Deep RL

Modern finance uses Deep RL. The agent's action space A_t is a continuous vector of portfolio weights: $\sum a_{i,t} = 1$.

Yu et al. (2019): Introduced a model-based RL architecture using an Infused Prediction Module (IPM) to forecast price trends, a Data Augmentation Module (DAM) via GANs to simulate rare market crashes, and a Behavior Cloning Module (BCM) to mimic expert human traders.

The reward function is often the logarithm of the return rate: $R_t = \ln(\frac{V_t}{V_{t-1}})$

Risk Metrics

A major issue with basic RL is that maximizing pure return often leads to catastrophic losses. We must mathematically ground the reward in risk.

Sharpe Ratio (S_t): Measures risk-adjusted return.

$$S_t = \frac{E[R_p - R_f]}{\sigma_p}$$

(Where R_p is portfolio return, R_f is risk-free rate, σ_p is standard deviation).

Maximum Drawdown (MDD): Measures the largest peak-to-trough drop in portfolio value.

$$MDD = \frac{Peak\_Value - Trough\_Value}{Peak\_Value}$$

A good RL agent must penalize high MDD in its reward function.

Failures Using One Agent

A single RL agent struggles to balance the opposing goals of high return and low risk simultaneously in shifting markets.

If trained during a bull market, the agent becomes over-leveraged and crashes during a bear market. If trained during a bear market, the agent is too timid and misses gains.

Multi-Agent Self-Adaptive Framework

The Solution: Break the problem into multiple specialized agents (Li & Tam, 2024).

Agent 1 (The Aggressor): Trained explicitly to maximize total portfolio returns.

Agent 2 (The Defender): Trained explicitly to minimize volatility and Maximum Drawdown.

How do they cooperate instead of fighting each other?

The Market Observer Agent

A third agent, the **Market Observer**, does not trade. It acts as a macroscopic classifier. It evaluates macro-economic indicators to determine market turbulence.

The Math: The Observer outputs a dynamic weight scalar $\alpha_t \in [0, 1]$

$$Final\_Action_t = \alpha_t \times Action_{Return} + (1 - \alpha_t) \times Action_{Risk}$$

In calm markets, $\alpha_t \to 1$. In turbulent markets, $\alpha_t \to 0$. This multi-agent framework significantly outperforms single-agent benchmarks in financial crises.

Physical AI Agents

Physical AI Agents & Robotics - Kinds of Models

LLMs (Large Language Models): Used for high-level task planning (e.g., "To make a sandwich, first get bread, then get cheese").

VLMs (Vision-Language Models): Used for open-vocabulary perception. The robot can look at a messy table and understand "the bread is next to the toaster."

VLAs (Vision-Language-Action Models): The holy grail. The model directly outputs physical motor torques based on camera and text inputs.

Zero-Shot Generalization & Emergence

The greatest benefit of Large Models in robotics is **Zero-Shot Generalization**.

Because the model read Wikipedia during pretraining, if you tell the robot to "pick up the extinct animal," it knows to pick up the plastic dinosaur, even if the robotics engineers never labeled a dinosaur in the robot's physical training data.

The Sim-to-Real Gap and Data Scarcity

Why is robotics lagging behind ChatGPT? **Data Scarcity.**

You can scrape billions of words from the internet. You cannot scrape physical robot trajectories.

The Sim-to-Real Gap: We train agents in physics simulators (like MuJoCo or Isaac Gym) millions of times. But when deployed on physical hardware, friction, lighting, and sensor noise cause the agent to fail. Closing this gap is the current frontier of physical AI research.

Summary

Logistics: Complex, macro-physical networks require distributed, multi-agent negotiations to avoid the computational bottlenecks of centralized routing.

Finance: Volatile, noisy digital environments benefit from adversarial/cooperative multi-agent architectures (like MASA) to mathematically separate risk from reward optimization.

Robotics: Micro-physical embodiment relies on translating internet-scale reasoning (Foundation Models) into mathematical policies for continuous physical control.

The Common Thread: The shift from rigid, single-objective algorithms to adaptive, agentic orchestration.

Questions?