# BeerGameBench: An API-Mediated Protocol for Studying Agent Coordination Under Delayed Feedback

Innar Liiv[1] and Uku Sildoja[1]

Tallinn University of Technology, Tallinn, Estonia

**Abstract.** AI agents are starting to participate in organizational workflows, but most benchmarks still test individual task success rather than coordination under delayed feedback, partial information, and local incentives. This paper introduces BeerGameBench, a Beer Game-based benchmark protocol for studying delayed-feedback coordination in AI-agent, human, and mixed-team settings. The protocol specifies reusable scenarios, API observations and actions, visibility and communication regimes, logging requirements, and metrics for cost, bullwhip amplification, robustness, reliability, communication, explanation, and team behavior.

We also report one API-mediated classroom deployment that tested the protocol in practice. Students submitted independently deployed bot endpoints, and the evaluator ran them on public and private simulation seeds. The anonymized export contains 49 registered bots, 877 simulation runs, 844 completed runs, 806 student rerun request records, and public 36-week state-action histories for all completed public runs. This deployment is not a controlled AI-agent experiment; it is evidence that the benchmark can capture integration failures, nondeterminism, public/private seed robustness, repeated strategy iteration, and role-level order amplification.

Among completed runs with valid private scores, public and private costs were strongly rank-correlated (Spearman $\rho = 0.985$). Repeated completed submissions showed median first-to-latest public-score improvements in both Blackbox (local-role information only) and Glassbox (all-role data visible) regimes, while role-level traces revealed substantial heterogeneity in upstream order amplification. BeerGameBench's contribution is a reproducible protocol for comparing agent policies, visibility rules, communication structures, and API-mediated development loops in a simple delayed-feedback coordination task.

**Keywords:** Human-AI teaming · LLM agents · Multi-agent systems · Beer Game · Bullwhip effect · Collective intelligence

## 1 Introduction

Large language models and other AI agents are moving from single-turn assistance toward repeated participation in organizational decisions. They summarize

meetings, propose actions, negotiate with other software agents, and increasingly appear as teammates in workflows that require coordination rather than mere prediction. This shift creates an evaluation gap. A system may perform well on static question answering or isolated task completion while failing in the social conditions that make organizational work difficult: partial information, role asymmetry, delayed consequences, communication overhead, trust calibration, and local incentives that do not automatically align with collective outcomes.

This gap is fundamentally collective: the question is not only whether an AI system can choose a plausible action, but how many actors make decisions when information, incentives, and consequences are distributed. For autonomous agents, these are not peripheral concerns. They are central to deployment in domains where decisions are distributed across organizational roles and where the consequences of one actor's behavior propagate through a wider system.

We propose BeerGameBench, a benchmark protocol that adapts the MIT Beer Game [5, 17] for studying delayed-feedback coordination in AI-agent, human, and mixed-team settings. The Beer Game, developed in the system dynamics tradition, models a serial supply chain with retailer, wholesaler, distributor, and factory roles. Each role makes local ordering decisions based on limited information, while material and information delays produce the well-known bullwhip effect: small changes in customer demand can lead to large oscillations in upstream orders and inventories. The game has been used for decades to demonstrate misperceptions of feedback, bounded rationality, and coordination failure in dynamic decision making [10, 2].

The Beer Game is attractive as a coordination benchmark because it is simple enough to be standardized and complex enough to reveal non-trivial collective behavior. Each actor faces a locally meaningful task, but the quality of the whole system depends on how actors infer hidden states, anticipate delayed effects, coordinate across roles, and respond to shocks. This makes the environment especially relevant for evaluating AI agents as organizational participants rather than as isolated solvers.

Recent work has already begun to study LLMs and other AI agents in Beer Game or supply-chain settings. Deep reinforcement learning has been applied to Beer Game inventory decisions [14]. LLM-based supply-chain agents have been studied in consensus-seeking, inventory decision, and reliability settings [9, 22, 12]. Related work includes live GenAI Beer Game demonstrations [11], ICIS work on Socratic iterative reasoning in the Beer Game [1], and emerging supply-chain benchmarks such as SCM-Arena [8]. These studies make a simple novelty claim impossible and unnecessary. BeerGameBench does not claim to be the first use of LLM agents in a Beer Game-like supply-chain simulation. Its contribution is to formalize the Beer Game as a benchmark protocol for studying communication regimes, institutional visibility, human-AI team composition, and emergent social behavior alongside conventional cost and bullwhip metrics.

The paper makes four contributions:

1. We define a benchmark protocol for studying delayed-feedback coordination across agent, human, and mixed-team settings, including interfaces, scenarios, communication regimes, and logging requirements.
2. We propose a multidimensional evaluation framework that combines operational metrics with behavioral metrics for communication, coordination, adaptation, explanation, and human-AI teaming.
3. We report anonymized evidence from a real API-mediated classroom deployment, including public/private seed scoring, determinism checks, integration failures, and repeated student reruns.
4. We position the benchmark against recent LLM-based supply-chain and multi-agent benchmark work, clarifying the remaining gap for coordination research.

## 2 Background and Related Work

### 2.1 The Beer Game and the Bullwhip Effect

The Beer Game is rooted in system dynamics and industrial dynamics [5]. It captures a recurring organizational pattern: actors operating with local information and delayed feedback may produce collectively unstable outcomes even when no individual actor intends harm. Sterman's classic account of misperceptions of feedback showed that human decision makers often underestimate pipeline inventory and fail to account for accumulations, producing oscillatory behavior in dynamic systems [17]. Lee, Padmanabhan, and Whang formalized information distortion in supply chains as the bullwhip effect, showing how demand signal processing, order batching, price fluctuations, and rationing games can amplify variability upstream [10].

Experimental and behavioral operations research has used the Beer Game to study bounded rationality, coordination risk, information visibility, and inventory dynamics. Croson and Donohue found that access to inventory information can reduce but not eliminate behavioral causes of the bullwhip effect [2]. Later work examined order stability, coordination stock, and behavioral causes through control-theoretic perspectives [3, 20]. System dynamics scholars have also emphasized that the Beer Game remains valuable because it exposes feedback delays and local decision rules that are difficult to perceive without interactive experience [18, 13].

BeerGameBench inherits this tradition but redirects it toward AI-agent evaluation. In human experiments, the Beer Game reveals how people reason under delayed feedback. In AI-agent experiments, it can reveal whether agents reproduce human-like biases, avoid them through explicit calculation, or develop new failure modes linked to stochastic generation, prompt framing, memory limits, or coordination protocols.

### 2.2 AI Agents in Supply-Chain and Beer Game Settings

AI methods have been applied to Beer Game-like inventory decisions before the recent LLM wave. Oroojlooyjadid et al. trained a deep Q-network for inventory

optimization in the Beer Game, demonstrating that reinforcement learning can outperform simple policies in specific settings [14]. This work is important because it shows that the Beer Game can function as an algorithmic testbed, but its primary emphasis is operational optimization rather than communication, teaming, or social-process evaluation.

Recent LLM work is closer to BeerGameBench. Jannelli et al. propose supply-chain-specific consensus-seeking frameworks for LLM agents and evaluate them in an inventory management case, with emphasis on autonomous coordination and negotiation [9]. Zhao et al. introduce AIM-Bench for evaluating decision-making biases of agentic LLMs as inventory managers, including pull-to-center and bullwhip-related behaviors [22]. Long et al. study autonomous generative AI agents in the MIT Beer Game and identify model selection, guardrails, centralized data sharing, and prompt engineering as performance levers; they also introduce the notion of agent bullwhip, including decision instability generated by stochastic agent behavior [12]. Boussioux et al. use the Beer Game to compare Socratic iterative reasoning with baseline LLMs, RAG, and chain-of-thought across multi-week experiments [1]. SCM-Arena describes a reproducible behavioral evaluation protocol for LLM decision policies in Beer Game-derived supply-chain systems [8]. Public demonstrations such as the GenAI Beer Game also show growing interest in LLM-powered Beer Game simulations [11].

These works are not obstacles to BeerGameBench; they are evidence that the problem is timely. They also sharpen the necessary contribution. A new paper cannot rely on the mere fact that LLMs play the Beer Game. BeerGameBench must instead specify what is missing from the current landscape: a protocol that makes communication structure, institutional information design, human-AI team composition, and emergent collective behavior first-class experimental variables.

### 2.3 LLM Agents, Social Simulation, and Human-AI Teaming

LLM agents are increasingly studied as autonomous systems with perception, memory, planning, tool use, and interaction loops [21]. Multi-agent LLM research extends this problem from individual capability to collective behavior: agents exchange messages, negotiate, critique one another, specialize, or fail through correlated errors and communication overload. Generative-agent simulations show that LLM-based agents can produce rich social patterns in sandboxed environments [15]. Strategic-game benchmarks evaluate LLM agents in interactive decision settings [19], while modular benchmark frameworks such as AgentQuest emphasize reusable evaluation infrastructure [6].

Human-AI teaming research adds another layer. Gonzalez et al. argue that human-AI complementarity depends on task structure, role partitioning, attention, memory, reasoning, trust calibration, and shared mental models [7]. Duan et al. show that trust in AI teammates evolves over time and responds to communication behavior during collaboration [4]. Shi and DiFranzo frame AI disclosure as epistemic coordination, emphasizing that teams must manage not only decisions but also the visibility and attribution of AI assistance [16]. These ideas are

directly relevant to BeerGameBench because the Beer Game can instantiate repeated mixed-team interactions where trust, disclosure, and information sharing have measurable operational consequences.

### 2.4 Competitive Positioning and Research Gap

Table 1 summarizes the main difference between BeerGameBench and close prior work. The key distinction is not domain novelty but evaluation orientation.

**Table 1.** Close prior work and remaining gap.

| Work | Most relevant overlap and remaining gap |
|---|---|
| Oroojlooyjadid et al. [14] | Uses the Beer Game as an algorithmic environment for deep RL inventory optimization. The gap is communication, mixed teams, and social-process metrics. |
| Jannelli et al. [9] | Studies LLM consensus-seeking in supply-chain coordination. The gap is a general Beer Game protocol with comparable scenarios and logs. |
| Zhao et al. [22] | Evaluates decision biases and bullwhip-related behavior in agentic LLM inventory decisions. The gap is systematic communication-regime and human-AI team evaluation. |
| Boussioux et al. [1] | Runs direct Beer Game LLM experiments around Socratic iterative reasoning. The gap is benchmark infrastructure rather than a single reasoning intervention. |
| Long et al. [12] | Evaluates autonomous AI agents in the MIT Beer Game and introduces agent bullwhip. The gap is broader instrumentation around communication, trust, and institutional visibility. |
| SCM-Arena [8] | Provides a Beer Game-derived behavioral protocol for LLM supply-chain decisions. The gap is explicit human-AI teaming, communication coding, and collective-behavior framing. |

This positioning leads to a conservative claim: BeerGameBench is a standardized protocol for studying agent coordination in a Beer Game environment. Its distinctive role is to operationalize human-AI teaming, communication structure, and institutional information design as first-class benchmark variables in a delayed-feedback coordination task, rather than to claim first use of LLMs or AI agents in Beer Game-like supply-chain simulations.

## 3 Benchmark Design

BeerGameBench is designed as a protocol rather than a single fixed simulator. The classroom system reported here is one reference implementation; the protocol is meant to make later studies comparable while allowing researchers to

vary agents, communication structures, and institutional rules. A conforming implementation should define the following components.

### 3.1 Core Environment

The core environment follows the classical four-echelon Beer Game. Roles are retailer, wholesaler, distributor, and factory. At each time step, customer demand is observed by the retailer. Orders flow upstream and goods flow downstream with fixed or scenario-specific lead times. Each role observes its local inventory, backlog, incoming orders, deliveries, and optionally other information exposed by the scenario. Each role chooses one order quantity per period. Costs are computed from inventory holding and backlog penalties, with optional order smoothing or capacity penalties in extended scenarios.

The benchmark separates the simulator state from the agent interface. Agents must not access hidden global state unless the scenario explicitly grants visibility. This separation is essential: the experimental value of the Beer Game comes from bounded information and delayed consequences, not from solving a fully observed optimization problem.

### 3.2 Agent Interface

Each agent receives a structured observation at each period:

- role identity and local objective;
- current period number and remaining horizon;
- on-hand inventory, backlog, incoming shipment, and outstanding pipeline inventory if visible;
- downstream order received in the current period;
- optional history window of local observations and own actions;
- optional shared messages or institutional dashboard information.

The agent returns an order quantity and, in communication-enabled settings, one or more messages. For LLM agents, the benchmark records model name, prompt template, temperature, system instructions, response text, parsed action, parsing failures, and retry behavior. For deterministic agents, it records policy parameters. For human participants, it records interface state, action timestamps, messages, and consented survey measures.

### 3.3 Communication Regimes

Communication is a central independent variable. BeerGameBench specifies six regimes:

1. **No communication**: agents act only on local observations.
2. **Structured reporting**: agents may send fixed fields such as forecast, inventory concern, and intended order.

3. **Free-form natural language**: agents may send unconstrained messages to adjacent or all roles.
4. **Shared dashboard**: agents see selected global indicators, such as end-customer demand history or inventory positions.
5. **Mediator**: a non-ordering AI or human moderator summarizes state and recommends coordination.
6. **Negotiation**: agents may propose and revise shared ordering plans before final action.

These regimes make communication and visibility experimental variables rather than background assumptions. They allow researchers to ask whether natural-language communication reduces instability, whether structured communication is more reliable than open chat, whether mediators improve or distort shared understanding, and whether agents converge on useful conventions or harmful overconfidence.

### 3.4 Scenario Suite

The benchmark defines a minimal scenario suite for comparability:

- **Classic step demand**: demand changes from a low stable level to a higher stable level, reproducing canonical bullwhip conditions.
- **Seasonal demand**: demand follows a predictable cycle, testing memory and forecasting.
- **Shock and recovery**: a temporary demand spike tests adaptation and stabilization.
- **Lead-time change**: delivery delay increases or decreases mid-game, testing model updating.
- **Noisy observation**: observed orders or shipments include noise or missing data.
- **Institutional visibility**: scenarios vary customer-demand visibility, pipeline visibility, or downstream inventory visibility.

Each scenario is specified in a machine-readable configuration file, including horizon length, demand process, lead times, cost parameters, visibility rules, message topology, and random seed. Scenario configurations should be published with results so that comparisons do not depend on undocumented choices.

## 4 Reference Implementation and Classroom Deployment

The reference implementation instantiates the protocol as an API-mediated classroom benchmark. Participants registered independently deployed bot endpoints. For each simulation week, the evaluator called the endpoint with a structured JSON observation containing the mode label, current week, total horizon, seed, and week-by-week history visible to the bot. The bot returned a JSON object with order quantities for the four roles. The implementation stored the

parsed orders, role states, costs, completion status, and error messages in a relational database.

The deployed simulator used a 36-week horizon, four roles, a two-week shipping delay, initial inventory of 15 units per role, and inventory-holding and backlog costs of 0.5 and 1.0 per unit per week. Customer demand was 10 units before week 5 and 25 units from week 5 onward, with bounded integer noise in $[-2, 2]$. The classroom API used two local terms: Blackbox mode instructed bots to base decisions only on each role's local weekly information, while Glassbox mode allowed bots to use all roles' data for coordinated decision-making. Each queued run was first evaluated on a public seed. If the public run completed, the evaluator repeated the same public-seed run as a determinism check by comparing the final sequence of orders. Only deterministic successful bots were then evaluated on a private seed. The public run produced the leaderboard score and stored the full weekly trace; the private score was stored separately for robustness analysis.

The classroom setting also made integration reliability observable. API endpoints could fail through HTTP errors, timeouts, non-JSON responses, invalid order payloads, or nondeterministic behavior. These failures are not incidental implementation noise: in API-mediated social systems, reliable participation in the interface is part of the coordination problem. A benchmark that records only successful final scores would miss this layer of social and technical behavior.

Table 2 summarizes the anonymized export. The raw export includes registration metadata needed for classroom administration, but the manuscript reports only aggregate statistics and anonymized run-level evidence. No student names, email addresses, API URLs, or identifiable bot labels are included.

**Table 2.** Anonymized classroom deployment dataset.

| **Quantity** | **Value** |
|---|---:|
| Registered bots | 49 |
| Simulation runs | 877 |
| Completed runs | 844 |
| Student rerun requests | 806 |
| Blackbox runs / completed scored public runs | 579 / 561 |
| Glassbox runs / completed scored public runs | 298 / 283 |
| Completed public runs with 36-week traces / failed determinism runs with traces | 844 / 13 |
| Deployment window | 2025-10-14–2026-01-05 |

## 5 Preliminary Observational Evidence

The deployment was a classroom benchmark rather than a randomized experiment. Students chose whether and when to submit reruns, and mode participation differed between Blackbox and Glassbox. The results should therefore be

read as preliminary observational evidence about what the protocol can capture, not as causal evidence that one information regime is superior to another.

First, the public/private scoring design suggests that the public leaderboard was not merely measuring overfitting to a single visible seed. Among completed runs with valid private scores ($n = 843$), public and private total costs were strongly associated (Pearson $r = 0.952$, Spearman $\rho = 0.985$); 59.0% of private scores were within 10% of the public score and 87.7% were within 25%, while bot-mode best-public records remained strongly rank-preserving ($n = 66$, Spearman $\rho = 0.975$). Figure 1 shows the public/private relationship on log scales. The visible outliers are analytically useful: they identify policies that pass the interface and determinism gates but still produce unstable coordination under some demand realizations.

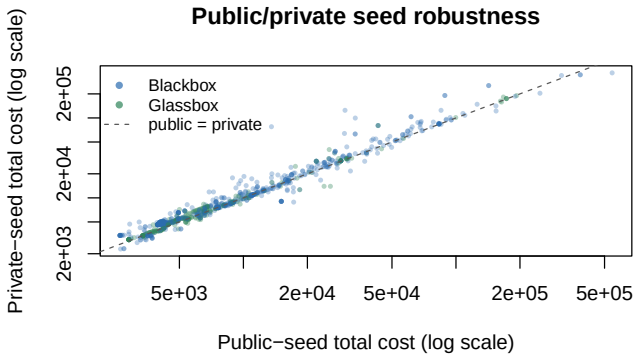


**Fig. 1.** Public/private seed robustness among completed runs with valid private scores. Points are anonymized run records; axes use logarithmic scales because coordination failures can produce heavy-tailed costs.

Second, the benchmark captured iterative adaptation. The 806 rerun requests show that the task was not only a static evaluator but also an API-mediated development loop. Using bot-mode pairs rather than runs as the unit, repeated completed submissions improved the latest public score for 23/35 Blackbox bot-mode pairs and 10/13 Glassbox bot-mode pairs; median first-to-latest improvements were 34.3% (IQR 0.0–89.1) and 66.7% (IQR 14.8–84.9), respectively. These figures are consistent with iterative policy refinement in the leaderboard and result-feedback loop. They do not show that the Glassbox mode causally improves performance, because mode support, bot selection, and rerun behavior were not controlled.

Third, best-by-bot scores show useful between-policy variation without requiring identity disclosure. The median best completed public cost by bot was 6,955 in Blackbox and 5,169 in Glassbox. This difference is directionally

consistent with the idea that institutional visibility and trace access may help coordination, but the correct interpretation is cautious: the observed modes differ in participation and in the population of bots that chose to support them.

Fourth, failure modes are themselves benchmark outcomes. The failed runs included determinism failures, HTTP/API errors, timeouts, and invalid or non-JSON responses. For autonomous agents deployed as services, parseability, latency, reproducibility, and endpoint availability are part of behavioral reliability. BeerGameBench therefore treats failures as evidence about API-mediated participation reliability rather than excluding them as mere noise.

Finally, the 36-week traces allow role-level analysis of coordination dynamics. Each completed trace records inventories, backlogs, shipments, incoming orders, and orders for retailer, wholesaler, distributor, and factory roles. Using each bot-mode pair's best completed public run, the median factory-vs-retailer order-variance ratio was 5.12 (IQR 1.00–13.31, $n = 46$) in Blackbox and 0.75 (IQR 0.20–6.85, $n = 20$) in Glassbox; because mode participation was self-selected and paired bot comparisons were mixed, this is evidence of trace-level heterogeneity rather than causal mode superiority.

## 6 Evaluation Framework

BeerGameBench evaluates both task performance and social behavior. Table 3 summarizes the proposed metric families.

The operational metrics are necessary but insufficient. Total cost can improve for reasons that are socially uninteresting, such as a single strong forecasting policy. Conversely, a team may communicate richly but perform poorly. The benchmark therefore treats performance and social process as complementary outcomes. For example, a structured communication protocol may slightly increase total cost while reducing variance and improving explanation reliability. Whether that trade-off is desirable depends on the research question.

### 6.1 Communication Coding

For communication-enabled runs, messages are logged and coded along several dimensions. A lightweight automatic coding scheme can classify messages into demand forecasts, inventory status, backlog status, intended action, uncertainty markers, requests, explanations, blame, and coordination proposals. For LLM messages, coding should be parser-first: messages are classified by a reproducible schema, with one retry or fallback to human coding if the classifier fails. Researchers should report classifier prompts, schemas, and inter-coder agreement when human coding is used.

### 6.2 Human-AI Measures

Mixed-team experiments should record both behavior and perception. Minimal measures include pre- and post-game trust, perceived competence of each role,

**Table 3.** Metric families for BeerGameBench.

| Metric family | Examples |
|---|---|
| Operational performance | Total cost, inventory cost, backlog cost, service level, average order quantity. |
| Bullwhip dynamics | Order variance amplification by role, demand distortion, downstream-to-upstream instability ratios. |
| Coordination quality | Recovery time after shock, convergence to stable ordering, synchronization across roles, frequency of stockouts. |
| Communication behavior | Message volume, message timing, use of forecasts, acknowledgement of uncertainty, contradiction rate, action-message consistency. |
| Social reasoning | References to other roles' constraints, inferred downstream demand, shared mental model indicators, system-level reasoning. |
| Human-AI teaming | Trust change, perceived reliability, disclosure condition, role adaptation, human override behavior. |
| Robustness | Variance across seeds, performance under demand shifts, parser failure rate, sensitivity to prompt and temperature. |
| Explanation reliability | Whether stated rationale matches observed action, stability of rationales across equivalent states. |

perceived communication usefulness, and willingness to delegate future decisions. Where disclosure is manipulated, participants should be told whether roles are human or AI according to the approved experimental condition and ethics protocol. BeerGameBench does not require deception, but it can support disclosure studies if they are ethically reviewed.

## 7 From Deployment Evidence to Controlled Studies

The classroom deployment strengthens the benchmark proposal, but it does not replace controlled evaluation. A complete comparative study should include heuristic base-stock policies, moving-average forecasting policies, prompt-only LLM agents, communication-enabled LLM agents, and mixed human-AI teams. Each condition should be evaluated on classic step demand, shock-and-recovery demand, and institutional visibility scenarios, with multiple seeds and preregistered parsing, timeout, and retry rules.

The deployment evidence helps specify what such studies must report: total and component costs, public/private robustness, role-specific order amplification, recovery time after shocks, rerun or adaptation history, and failed-run categories. It also shows why benchmark papers should publish trace schemas rather than only leaderboards. Without weekly state-action traces, claims about coordination, trust, visibility, and collective behavior cannot be checked.

A second development path is to make the protocol machine-discoverable rather than only described in prose: scenario schemas, API contracts, trace formats, and metric definitions should be readable by external agents as well as by researchers. This would support full-agent evaluations in which coding agents or command-line/Model Context Protocol (MCP)-enabled systems implement, test, and revise policies without prewritten bot code. It also raises a design choice for future work: nondeterministic agents can be rejected by determinism gates, evaluated through repeated sampling, or treated as a distinct behavioral property. The same protocol could then compare deterministic service bots, full-agent modes, and self-improving agent loops under shared seeds and logging rules.

## 8 Discussion

### 8.1 Why the Beer Game Still Matters

A likely criticism is that the Beer Game is too simple. This criticism is partly correct and partly the reason the benchmark is useful. Real supply chains include multiple products, nonlinear networks, strategic suppliers, capacity constraints, contracts, and external disruptions. BeerGameBench does not attempt to model all of this. Instead, it isolates a minimal but consequential coordination problem: actors with local information must coordinate under delays, and their locally reasonable actions can create global instability. Controlled simplicity is valuable when the research question concerns mechanisms of coordination rather than prediction in a single realistic domain.

### 8.2 Risks of LLM-Specific Benchmarks

LLM behavior is sensitive to model version, sampling parameters, prompts, system instructions, and tool wrappers. BeerGameBench therefore treats agent metadata as part of the experimental record. A published result should report exact model versions, prompt templates, temperature, retry logic, and parsing rules. Where proprietary APIs are used, researchers should state the date of access and preserve generated transcripts when allowed. This is especially important for agent-behavior claims because small prompt changes can alter not only performance but also communication tone, willingness to coordinate, and apparent trustworthiness.

### 8.3 Ethical and Social Considerations

BeerGameBench studies how deployed agents and bots participate in coordinated decision settings, but it should not anthropomorphize them. Agents may produce persuasive explanations that do not reflect stable internal reasoning. In mixed human-AI teams, participants may over-trust confident messages or misattribute responsibility for decisions. The benchmark should therefore log explanation-action consistency and support disclosure conditions that make AI

participation clear unless an ethics-approved study intentionally manipulates disclosure. If human participants are involved, the system should collect only necessary data, store transcripts securely, and separate personal identifiers from gameplay logs.

### 8.4 Limitations

The current protocol is strongest for studying decentralized coordination in a stylized linear supply chain. It does not replace real-world supply-chain optimization benchmarks, and it does not claim that performance in the Beer Game generalizes to all organizational settings. The classroom evidence is observational: mode participation, model choice, implementation effort, and rerun behavior were not randomized. Communication metrics also require careful validation: automatic classifiers can misread sarcasm, indirect requests, or role-specific jargon. Finally, mixed human-AI studies require human-subject review and cannot be fully replaced by simulated humans or LLM personas.

## 9 Conclusion

BeerGameBench reframes the MIT Beer Game as a benchmark for AI-agent coordination under delayed feedback. The benchmark is motivated by a practical and theoretical gap: autonomous agents are being evaluated as individual problem solvers, while real organizational deployment requires coordination among humans, AI systems, APIs, and institutional rules. The classroom deployment shows that the protocol can produce more than a leaderboard: it captures coordination traces, integration reliability, determinism, public/private robustness, and iterative adaptation across independently deployed bots. Its value lies not in claiming novelty for the Beer Game itself, but in using a standard delayed-feedback environment to ask concrete questions about communication, trust, shared mental models, institutional visibility, and emergent instability in autonomous agent systems.

**Acknowledgments.** The research idea, conceptual framing, classroom setup, data-collection process, and final interpretation are the authors' work. During preparation, the authors used ChatGPT and Codex (OpenAI), Claude (Anthropic), and Google Gemini for drafting support, language editing, structural refinement, consistency checks, and data-analysis assistance. The authors reviewed, edited, and validated the content and take full responsibility for the publication.

This work forms part of research at Tallinn University of Technology and uses an anonymized classroom deployment export plus a local Beer Game simulation prototype. It was supported by the European Union and the Estonian Research Council through project TEM-TA141.

**Disclosure of Interests.** The authors have no competing interests to declare that are relevant to the content of this article.

## References


1. Boussioux, L., Chen, A., Fan, M., Jain, A.: Socratic iterative reasoning: Enhancing LLM decision-making in the beer game supply chain. In: Proceedings of the International Conference on Information Systems (2025), https://aisel.aisnet.org/icis2025/gen_ai/gen_ai/41/, paper ICIS2025-2779
2. Croson, R., Donohue, K.: Behavioral causes of the bullwhip effect and the observed value of inventory information. Management Science **52**(3), 323–336 (2006). https://doi.org/10.1287/mnsc.1050.0436
3. Croson, R., Donohue, K., Katok, E., Sterman, J.D.: Order stability in supply chains: Coordination risk and the role of coordination stock. Production and Operations Management **23**(2), 176–196 (2014). https://doi.org/10.1111/j.1937-5956.2012.01422.x
4. Duan, W., Zhou, S., Scalia, M.J., Yin, X., Weng, N., Zhang, R., Freeman, G., McNeese, N.J., Gorman, J.C., Tolston, M.T.: Understanding the evolvement of trust over time within human-AI teams. Proceedings of the ACM on Human-Computer Interaction **8**(CSCW2) (2024). https://doi.org/10.1145/3687060
5. Forrester, J.W.: Industrial Dynamics. MIT Press, Cambridge, MA (1961)
6. Gioacchini, L., Siracusano, G., Sanvito, D., Gashteovski, K., Friede, D., Bifulco, R., Lawrence, C.: AgentQuest: A modular benchmark framework to measure progress and improve LLM agents. arXiv preprint arXiv:2404.06411 (2024). https://doi.org/10.48550/arXiv.2404.06411
7. Gonzalez, C., Donahue, K., Goldstein, D.G., Heidari, H., Jalali, M.S., Schelble, B.G., Singh, A., Woolley, A.W.: Toward a science of human-AI teaming for decision making: A complementarity framework. PNAS Nexus **5**(3), pgag030 (2026). https://doi.org/10.1093/pnasnexus/pgag030
8. Hasija, A., Castillo, V.E.: SCM-Arena: A behavioral benchmark for LLM decision-making in supply chains. SSRN Working Paper (2026). https://doi.org/10.2139/ssrn.6514506, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=6514506
9. Jannelli, V., Schoepf, S., Bickel, M., Netland, T.H., Brintrup, A.: Agentic LLMs in the supply chain: Towards autonomous multi-agent consensus-seeking. International Journal of Production Research (2025). https://doi.org/10.1080/00207543.2025.2604311
10. Lee, H.L., Padmanabhan, V., Whang, S.: Information distortion in a supply chain: The bullwhip effect. Management Science **43**(4), 546–558 (1997). https://doi.org/10.1287/mnsc.43.4.546
11. Long, C.X., Simchi-Levi, D., Calmon, A.P., Calmon, F.P.: The GenAI beer game. https://infotheorylab.github.io/beer-game/ (2026), accessed 2026-05-29
12. Long, C.X., Simchi-Levi, D., Zhu, F., Su, H., Calmon, A.P., Calmon, F.P.: Reliability and effectiveness of autonomous AI agents in supply chain management. arXiv preprint arXiv:2605.17036 (2026). https://doi.org/10.48550/arXiv.2605.17036
13. Martinez-Moyano, I.J.: History of the beer game. System Dynamics Review (2024). https://doi.org/10.1002/sdr.1767
14. Oroojlooyjadid, A., Nazari, M., Snyder, L.V., Takác, M.: A deep q-network for the beer game: Deep reinforcement learning for inventory optimization. Manufacturing & Service Operations Management **24**(1), 285–304 (2022). https://doi.org/10.1287/msom.2020.0939
15. Park, J.S., O'Brien, J.C., Cai, C.J., Morris, M.R., Liang, P., Bernstein, M.S.: Generative agents: Interactive simulacra of human behavior. In: Proceedings of the

36th Annual ACM Symposium on User Interface Software and Technology (2023). https://doi.org/10.1145/3586183.3606763
16. Shi, H., DiFranzo, D.: Who gets credit? operationalizing AI disclosure as epistemic coordination in human-AI teams. In: Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (2026). https://doi.org/10.1145/3772363.3799006
17. Sterman, J.D.: Misperceptions of feedback in dynamic decision making. Organizational Behavior and Human Decision Processes **43**(3), 301–335 (1989). https://doi.org/10.1016/0749-5978(89)90041-1
18. Sterman, J.D., Oliva, R., Linderman, K., Bendoly, E.: System dynamics perspectives and modeling opportunities for research in operations management. Journal of Operations Management **39–40**, 1–5 (2015). https://doi.org/10.1016/j.jom.2015.07.001
19. Tang, W., Zhou, Y., Xu, E., Cheng, K., Li, M., Xiao, L.: DSGBench: A diverse strategic game benchmark for evaluating LLM-based agents in complex decision-making environments. arXiv preprint arXiv:2503.06047 (2025). https://doi.org/10.48550/arXiv.2503.06047
20. Udenio, M., Vatamidou, E., Fransoo, J.C., Dellaert, N.: Behavioral causes of the bullwhip effect: An analysis using linear control theory. IISE Transactions **49**(10), 980–1000 (2017). https://doi.org/10.1080/24725854.2017.1325026
21. Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., Zhao, W.X., Wei, Z., Wen, J.R.: A survey on large language model based autonomous agents. Frontiers of Computer Science (2024). https://doi.org/10.1007/s11704-024-40231-1
22. Zhao, X., Xie, Y., Chen, C., Sun, Y.: AIM-Bench: Evaluating decision-making biases of agentic LLM as inventory manager. arXiv preprint arXiv:2508.11416 (2025). https://doi.org/10.48550/arXiv.2508.11416