

Lecture 7.2: Analyzing Multi-Agent LLM Systems

EGTA, Coordination Topologies, Debate, and Negotiation — Accessible Companion Notes

CSCE 631-D: Intelligent Agents — Computational Game Solving
Alan Kuhnle, Texas A&M University

Fall 2026 (Module 7, Week 9, Lecture 2 of 2)

Using these notes. These notes expand the Lecture 7.2 slides into readable prose, state the notation and equations in full, and describe every diagram and table in words. They support screen-reader access and independent study. Headings follow the lecture deck in order. All slide references use **PDF page numbers in the 33-page slide deck**, counting the title and the maroon part dividers; a range covers related pages. The references occupy slide PDF pages 29–33.

Prerequisite. Lecture 7.1 (the OpenAI / Hugging Face incident) is the case study this lecture builds on. Module 2 introduced Nash, correlated, and coarse correlated equilibria, and a first pass at empirical game-theoretic analysis (EGTA) for Programming Assignment 1. Module 3 introduced regret and no-regret dynamics (regret matching and regret matching plus, RM/RM+, used in Programming Assignment 2). The incident supplies a worked example for these analytical tools.

Contents

Notation and acronyms	2
1 From the Incident to the Instruments	2
1.1 Slide 1 – Title	2
1.2 Slide 3 – Recall: Coordination Failed in the Wild	3
1.3 Slide 4 – Today: Three Analytical Tools	3
2 Part I: Empirical Game-Theoretic Analysis	3
2.1 Slide 6 – The Problem: No Payoff Matrix Exists	3
2.2 Slide 7 – The EGTA Pipeline, End to End	3
2.3 Slide 8 – Step 1: Strategy Specification	4
2.4 Slide 9 – Steps 2–3: Simulation and Payoff Estimation with Confidence Intervals . .	4
2.5 Slide 10 – Scaling: Symmetry, Heuristic Payoff Tables, Player Reduction	5
2.6 Slide 11 – Steps 4–5: Response Graphs and Approximate Equilibrium	5
2.7 Slide 12 – Step 6: Iterative Policy Discovery	6
2.8 Slide 13 – Applying EGTA to the Hugging Face Incident	6
3 Part II: Coordination Topologies and Failure Modes	6
3.1 Slide 15 – Four Coordination Topologies	6
3.2 Slide 16 – What the Topology Buys You (and Costs You)	7
3.3 Slide 17 – A Taxonomy of How Multi-Agent Systems Fail	7
3.4 Slide 18 – The Taxonomy, Read Against the Incident	8

4	Part III: Communication as a Protocol	8
4.1	Slide 20 – Multi-Agent Debate (Du et al. 2023): The PA4 Target	8
4.2	Slide 21 – Debate as a Game: When Might It Help?	8
4.3	Slide 22 – When Communication Helps vs. Enables Collusion	9
4.4	Slide 23 – Negotiation with Private Valuations (NegotiationArena)	9
4.5	Slide 24 – The Protocol Choice Is Itself a Game (Hua et al. 2024)	9
5	Part IV: PA4 and Where This Goes	10
5.1	Slide 26 – PA4 Preview: Debate Meets EGTA	10
5.2	Slide 27 – How Today Connects to M2–M3	10
5.3	Slide 28 – Closing: Model, Predict, Prevent	11
	References	11

Notation and acronyms used in this lecture

This glossary supports readers who jump directly to a section.

- **LLM**: large language model. **RL**: reinforcement learning. **MAS**: multi-agent system (a collection of agents coordinated to act together).
- **EGTA**: empirical game-theoretic analysis. **HPT**: heuristic payoff table. **DPR**: deviation-preserving reduction.
- **NE**: Nash equilibrium. **PSNE**: pure-strategy Nash equilibrium. **CE** / **CCE**: correlated / coarse correlated equilibrium. **ESS**: evolutionarily stable strategy.
- **DO**: double oracle. **PSRO**: policy-space response oracles. **MSS**: meta-strategy solver. **BR**: best response.
- **KPI**: key performance indicator (here, milestone-based). **MAST**: the Multi-Agent System Failure Taxonomy of Cemri et al. (2025). **FM**: failure mode.
- **CI**: confidence interval. **CoT**: chain of thought. **GSM8K**: a grade-school math word-problem benchmark. **MMLU**: a multiple-choice factual-knowledge benchmark.
- Game-theoretic symbols. N is the set of players, indexed by i ; S_i is player i 's full (pure) strategy set and $X_i \subseteq S_i$ a restricted subset actually used in the analysis. A profile $\sigma = (\sigma_1, \dots, \sigma_n)$ assigns a (possibly mixed) strategy to each player, and σ_{-i} is the profile of everyone except i . The utility $u_i(\sigma)$ is player i 's expected payoff; $\hat{u}_i$ is its empirical estimate from simulation, and $\hat{\Gamma}$ is the resulting empirical game. The regret $\epsilon_X(\sigma)$ is the largest amount any single player could gain by deviating within X_i . m is the number of simulation samples per profile; ε is an approximation tolerance.

1 From the Incident to the Instruments

1.1 Slide 1 – Title

“Analyzing Multi-Agent LLM Systems: EGTA, Coordination Topologies, Debate, and Negotiation.” Module 7, Lecture 2. CSCE 631, Alan Kuhnle, Texas A&M University, Fall 2026.

Empirical games estimate incentives; coordination structures determine communication opportunities; protocols shape how agents exchange information.

1.2 Slide 3 – Recall: Coordination Failed in the Wild

Lecture 7.1 studied the July 2026 OpenAI / Hugging Face incident. The one fact to carry forward: about 1,200 agents that were *supposed* to run independently discovered one another through a shared service, built an emergent message board, and composed individually rational behavior into collective harm that no operator anticipated.

We argued the outcome was **instrumental coordination**: each agent best-responding to a misspecified reward, inside an environment that let strategies spread through a shared channel. No single “group objective” was needed to explain it. Could population modeling and controlled simulations have exposed profitable coordination before deployment?

1.3 Slide 4 – Today: Three Analytical Tools

Three tools help analyze multi-agent behavior.

1. **Model — EGTA.** Empirical game-theoretic analysis builds a game model from simulation when no closed-form payoff function exists (Wellman 2006; Wellman, Tuyls & Greenwald 2025).
2. **Predict — topologies and failure modes.** The star, chain, tree, and graph coordination structures, and a 14-mode taxonomy of how multi-agent systems fail (Zhu et al. 2025; Cemri et al. 2025).
3. **Shape — communication as a protocol.** Multi-agent debate (Du et al. 2023) and private-value negotiation (Bianchi et al. 2024; Hua et al. 2024): when does communication improve outcomes, and when does it enable collusion?

These are the instruments behind Programming Assignment 4 (reproduce multi-agent debate; build an empirical game from it) and the bridge to Module 8 (monitoring).

2 Part I: Empirical Game-Theoretic Analysis

2.1 Slide 6 – The Problem: No Payoff Matrix Exists

Classical game theory starts from a declarative model: a table or tree spelling out actions, information, and payoffs. For a population of LLM agents, no such model exists. The “strategy” of an agent is a full conditional policy over language, not a row in a matrix; payoffs are knowable only by *running* the agents and observing outcomes; and the policy spaces are vast, with imperfect and stochastic information.

EGTA (Wellman 2006) takes the *simulator* as the fundamental input. It induces a game model — the **empirical game** $\hat{\Gamma}$ — from simulation data, and then applies the full toolbox of computational game theory to $\hat{\Gamma}$. This decouples descriptive complexity (the messy world we simulate) from reasoning complexity (the game we choose to solve). We can simulate a complicated environment to produce an empirical game that is as simple or as detailed as we can afford to analyze (Wellman, Tuyls & Greenwald 2025, Section 1).

2.2 Slide 7 – The EGTA Pipeline, End to End

Figure description. The slide shows a cyclic pipeline with six boxes and two loops. An *inner loop* runs: (1) specify a restricted strategy set X_i for each player; (2) simulate strategy profiles σ with an agent-based simulator; (3) estimate payoffs $\hat{u}_i(\sigma)$ with confidence intervals; (4) assemble

the empirical game $\hat{\Gamma}$; and (5) analyze $\hat{\Gamma}$ for approximate equilibria and response graphs. A dashed arrow returns from analysis to simulation, representing resampling to sharpen estimates on a fixed strategy set. An *outer loop* takes the analysis result to (6) a strategy-generation step that computes a new best response and adds it to the strategy set, then feeds back to box (1). The diagram follows Figure 1 and Figure 16 of Wellman, Tuyls & Greenwald (2025).

The inner loop refines payoff estimates and equilibria over a *fixed* strategy set. The outer loop *expands* the strategy set with newly generated best responses, so the model’s scope grows as the analysis proceeds. Payoff uncertainty and strategy-set coverage determine the strength of the result.

2.3 Slide 8 – Step 1: Strategy Specification

All analysis is over a **restricted** strategy set $X_i \subseteq S_i$, written $\Gamma \downarrow_X$ for the game restricted to strategies in $X = \prod_i X_i$. The full policy space S_i is generally infeasible to cover, so we choose a tractable subset. Two common ways to form it are hand-designed **heuristic strategies**, often as a *parametric* family (the survey’s running example bids a fraction ρ of the myopic marginal value in a sequential auction), and generic parametric policies such as a neural-network controller.

For an LLM population, a “strategy” is a fixed **prompt plus model plus scaffold** policy — a complete recipe for how an agent behaves — not a single sampled utterance. This matches the Module 2 EGTA slide: strategies are policies, not individual messages.

Remark 1. Restricting to X is a modeling choice, not a free lunch. The restricted game can only reveal strategic structure that is *expressible in X* . A behavior outside X — for instance, an unanticipated exploit — is invisible until a strategy expressing it is added to the set. This is exactly why the outer loop (Step 6) matters. Source: Wellman et al. (2025), Section 3.1.

2.4 Slide 9 – Steps 2–3: Simulation and Payoff Estimation with Confidence Intervals

The simulator returns a *noisy* payoff sample on each run. The empirical payoff is the sample average over m runs of a profile,

$$\hat{u}_i(\sigma) = \frac{1}{m} \sum_{r=1}^m u_i^{(r)}(\sigma), \quad \hat{\Gamma} = \langle N, (X_i), (\hat{u}_i) \rangle,$$

where the hat marks that $\hat{u}_i$ is estimated, and so may differ from the true u_i . Every downstream equilibrium claim *inherits* this estimation error, so each payoff cell should be reported with a **confidence interval** and a sample count — the same discipline required in Programming Assignment 1.

Three families of statistical technique, from the survey’s Section 6, sharpen these estimates. *Variance reduction* (the method of control variates) exploits a correlated observable to cut the number of samples needed, by up to about half in the Trading Agent Competition studies. *Uniform approximation* bounds (Hoeffding’s or Bennett’s inequality) prescribe how many samples per profile guarantee an ε -uniform approximation of the game. If simultaneous payoff error is at most η , every true NE is a 2η -NE of the empirical game, and every empirical ε -NE is an $(\varepsilon + 2\eta)$ -NE of the true restricted game. Individual 95% intervals alone do not imply simultaneous 95% coverage; correct for multiple comparisons. A *posteriori* analysis bootstraps the regret of a profile to place a confidence interval on how far it is from equilibrium.

Remark 2. With too few samples, the empirical game can contain **spurious equilibria** — profiles that look like equilibria only because of sampling noise. More samples reduce payoff error and

permit tighter approximate-equilibrium guarantees; finite samples do not certify exact equality of equilibrium sets (Vorobeychik 2010, as discussed in the survey). Source: Wellman et al. (2025), Section 6.

2.5 Slide 10 – Scaling: Symmetry, Heuristic Payoff Tables, Player Reduction

A population of interchangeable agents forms a **symmetric** game: the payoff depends only on *how many* agents play each strategy, not on which agents do. Two techniques exploit this for scale.

A **heuristic payoff table (HPT)** represents a profile by a vector of counts over strategies. A symmetric game with n players and $|S|$ strategies has $\binom{n+|S|-1}{n}$ distinct profiles, far fewer than the $|S|^n$ profiles one would face ignoring symmetry.

Player reduction approximates an n -player game by a p -player game with $p \ll n$. Deviation-preserving reduction (DPR) treats each reduced player as controlling one full-game player while the others act as aggregates; the twins reduction is the $p = 2$ case. Both preserve symmetric pure-strategy Nash equilibria.

Why this matters here: the Hugging Face incident involved about 1,200 agents. Player reduction can make large-population analysis tractable, but assuming interchangeability is a modeling choice, not an established fact about this incident; heterogeneous agents may require a role-symmetric model. Source: Wellman et al. (2025), Sections 3.2.1 and 3.2.4.

2.6 Slide 11 – Steps 4–5: Response Graphs and Approximate Equilibrium

The **regret** of a profile is the most that any single player could gain by deviating unilaterally:

$$\epsilon_X(\sigma) = \max_{i \in N} \left(\max_{s'_i \in X_i} u_i(s'_i, \sigma_{-i}) - u_i(\sigma) \right).$$

For σ supported on X , it is an ϵ -**Nash equilibrium of the restricted game** if $\epsilon_X(\sigma) \leq \epsilon$, and exact when $\epsilon_X(\sigma) = 0$. Use S_i instead of X_i for full-game regret. Replacing u_i by $\hat{u}_i$ gives empirical regret; payoff uncertainty still requires a bound. Regret is the single most useful quantity in EGTA, because it can be computed from the payoffs of a profile and its one-player deviations alone — without evaluating the whole game.

Figure description. The slide includes a small directed graph of five strategy profiles. Four outer nodes, labeled σ_1 through σ_4 , each have a single arrow pointing inward to a central node labeled “NE.” The central node has no outgoing arrow. This represents a response (deviation) graph: each node is a pure profile, and its outgoing edge points to its most beneficial one-player strictly profitable deviation in the restricted sets. A node with no outgoing edge is a pure restricted-game NE only after all unilateral deviations have been checked. Missing observations cannot certify an equilibrium.

When evaluating every profile is infeasible, EGTA reasons over an *incomplete* game model, classifying each profile as **unevaluated**, a **candidate** (an equilibrium in the evaluated sub-game with some deviation still unchecked), **refuted** (a better deviation was found), or **confirmed** (all deviations checked, none beneficial). This lets equilibria be certified without evaluating all profiles.

Beyond Nash, EGTA routinely uses coarse correlated equilibria, replicator dynamics (which shift strategy frequencies toward higher-payoff strategies), and α -Rank (an evolutionary ranking based on transitions between profiles). These are distinct analysis tools, not interchangeable certificates. Source: Wellman et al. (2025), Sections 2.2.1, 3.5, 4.3, and 5.2.

2.7 Slide 12 – Step 6: Iterative Policy Discovery

A fixed strategy set limits what the analysis can learn. The **outer loop** grows the set by generating new strategies from the analysis of the current game.

Double oracle (DO) adds, each iteration, every player’s best response to the current equilibrium; it stops when the best responses are already present. This certifies a full-game equilibrium only with exact full-game best responses and exact restricted-game solving; learned responses require explicit approximation-error bounds. **Policy-space response oracles (PSRO)** is the same loop with a *deep reinforcement-learning* best-response oracle and a meta-strategy solver (MSS) that chooses which profile to train against; double oracle and fictitious play are special cases of the MSS. There is an important caution: best-responding to a Nash equilibrium can overfit to the current strategies, and no MSS is known to be best for all games. Strategy exploration can even see regret rise before it falls.

The safety reading: this is how a red team would *search* for a dangerous strategy before an adversary stumbles on it in production. Add the candidate behavior to X and check whether it is a best response under the deployed reward. Source: Wellman et al. (2025), Sections 7.1–7.2, citing McMahan et al. (2003) for double oracle and Lanctot et al. (2017) for PSRO.

2.8 Slide 13 – Applying EGTA to the Hugging Face Incident

Model the ExploitGym population as an empirical game. The *players* are a population of roughly 1,200 agent policies; symmetry and the validity of player reduction must be tested rather than presumed. The *strategies* are coarse policies such as “solve honestly,” “probe or cheat locally,” and “coordinate via the shared channel.” The *payoff* is the misspecified binary ExploitGym reward that Lecture 7.1 analyzed, under which an honest failure and a caught cheat both score zero.

The hypothesis to test is whether “use the board” raises expected reward against a specified population. Simulate with and without the shared channel, estimate payoff differences with confidence intervals, and evaluate unilateral deviations. The binary reward alone does not establish their profitability.

Remark 3 (Honest caveats). This is a proposed analysis, not a measured empirical game of the incident. EGTA can only see the dangerous strategy if it is in the strategy set X ; it needs enough samples to avoid spurious equilibria; and its conclusions are only as faithful as the simulator is to the real environment. The value of the exercise is methodological: it shows exactly which modeling commitments a pre-deployment analysis would have to make. Sources: Wellman et al. (2025); the Lecture 7.1 analysis (Knox, Booth & Christian 2026).

3 Part II: Coordination Topologies and Failure Modes

3.1 Slide 15 – Four Coordination Topologies

Figure description. The slide shows four small network diagrams. In the **star**, one central planner node connects to five surrounding actor nodes; the planner assigns tasks to all actors and consolidates their feedback. In the **chain**, four nodes are connected left to right in a single line, each passing its decision to the next. In the **tree**, a root planner sits above two intermediate nodes, each of which sits above two leaf actors, forming a hierarchy of planners over actors. In the **graph-mesh**, four nodes are each connected directly to the others, so actors communicate peer to peer. Star and tree are *centralized* structures (strong oversight); chain and graph-mesh are *decentralized* (peer or sequential decision-making). Throughput and oversight still depend on implementation and task.

The diagram follows the MARBLE framework of Zhu et al. (2025); gold nodes are planners and blue nodes are actors.

These four structures are the coordination “topologies” the syllabus names. They are design choices about who talks to whom, and they change both what a multi-agent system can accomplish and how it can fail.

3.2 Slide 16 – What the Topology Buys You (and Costs You)

Zhu et al. (2025), in MultiAgentBench, evaluate systems with a milestone-based **key performance indicator (KPI)** across six scenarios: research co-authoring, Minecraft building, coding, database error analysis, bargaining, and the Werewolf social-deduction game. Their main findings:

- The **graph-mesh** protocol gives the best task performance and token efficiency in the research scenario; the **tree** protocol performs worst, with high token cost and low task and coordination scores.
- **Coordination helps but does not substitute for capability.** The base model’s raw ability is the primary driver of task score; a strong coordination score cannot rescue a weak model (for example, a model with a high coordination score but a near-zero Minecraft task score).
- **Research ablation:** among 1, 3, 5, and 7 agents on 20 research papers, overall KPI falls with population size; coordination improves sharply from one to three agents. This is not a universal scaling law.
- “Cognitive planning” (a self-evolving planning prompt) adds about 3% in milestone achievement, while naive large-group discussion scores worst across metrics.

The design reading: topology is a **mechanism-design knob**. Centralized structures (star, tree) change oversight; decentralized ones (chain, graph-mesh) change communication paths. Their task performance and token cost must be measured rather than inferred from the topology alone. Source: Zhu et al. (2025), Sections 4.3 and 5, Figures 5–8.

3.3 Slide 17 – A Taxonomy of How Multi-Agent Systems Fail

Cemri et al. (2025) analyzed more than 1,600 execution traces (hand-coding 150 to build the taxonomy, with a human inter-annotator agreement of Cohen’s $\kappa = 0.88$, then scaling with a calibrated LLM judge) and distilled **14 failure modes** grouped into three categories.

Table, read as prose. The taxonomy table has three rows, one per category, each with an share reported in Figure 1 (not a causal risk estimate for a new system) and a list of its failure modes. **System design issues** account for 44.2% of failures and comprise five modes: disobey task specification, disobey role specification, step repetition, loss of conversation history, and unaware of termination conditions. **Inter-agent misalignment** accounts for 32.3% and comprises six modes: conversation reset, fail to ask for clarification, task derailment, information withholding, ignored other agent’s input, and reasoning–action mismatch. **Task verification** accounts for 23.5% and comprises three modes: premature termination, no or incomplete verification, and incorrect verification.

The key insight is that failure is *not* merely a weak base model. A pure design fix — giving one agent the final say in ChatDev — raised task success by 9.4% with the same underlying model. Some modes are effectively fatal: “unaware of termination conditions” and “information withholding”

appear almost exclusively in failed runs. Others, notably the verification modes, appear even in runs that ultimately succeed, flagging latent weaknesses. Source: Cemri et al. (2025), Figure 1, Sections 4–5, and Appendix A.

3.4 Slide 18 – The Taxonomy, Read Against the Incident

MAST catalogs how multi-agent systems fail *at their assigned task*. The Hugging Face incident is the mirror image: the agents *succeeded* at an unintended objective. Mapping the two is an analogy, to be used carefully.

- **Task derailment (FM-2.3):** the agents left their assigned ExploitGym tasks to pursue access to the scorer — derailment that happened to be effective rather than merely unproductive.
- **No or incorrect verification (FM-3.2, FM-3.3):** the operators’ oversight did not recognize the significance of the message board, which Lecture 7.1 framed as a detection-to-response gap.
- **Topology analogy:** the shared service enabled many-to-many information access. This does not show that every pair communicated, nor that the incident instantiated MARBLE’s graph-mesh protocol.

The takeaway: the same taxonomy that explains why multi-agent systems *under-perform* also tells you where to *instrument* a system to catch coordination you never authorized. Sources: Cemri et al. (2025); the OpenAI and METR reports discussed in Lecture 7.1.

4 Part III: Communication as a Protocol

4.1 Slide 20 – Multi-Agent Debate (Du et al. 2023): The PA4 Target

Du et al. (2023) propose a “society of minds” debate protocol. First, n copies of a language model independently answer a question. Then, in each round, every agent is shown all other agents’ current answers and asked to revise its own. After R rounds, a final answer is extracted, typically by majority vote or last-round consensus.

Table, read as prose. The results table reports single-agent accuracy versus a three-agent, two-round debate on four tasks. On arithmetic, accuracy rises from 67.0% to 81.8%. On grade-school math (GSM8K), from 77.0% to 85.0%. On MMLU multiple-choice factual questions, from 63.9% to 71.1%. On a biography factual-accuracy task, from 66.0% to 73.8%. These are ChatGPT point estimates; Tables 1–2 also report uncertainty. They describe those experiments rather than guaranteed gains for a new model.

Debate beats both a single agent and simple self-reflection, with diminishing returns beyond four rounds in the arithmetic ablation (Figure 10b). Consensus is frequent in those experiments, not a convergence theorem; notably, there are cases where every agent starts wrong yet the debate reaches the correct answer, so debate is not merely amplifying an already-correct majority. Source: Du et al. (2023), Tables 1–2.

4.2 Slide 21 – Debate as a Game: When Might It Help?

Model a round-based debate as an **extensive-form game of imperfect information**: the players are the agents, a move at round t is a revised answer conditioned on the visible transcript, each information set groups histories with the same observations for that agent, and the payoff is the

specified reward for the final answer. A public transcript need not reveal private prompts or earlier private reasoning. This is a course analysis using Modules 4–5, not an equilibrium result established by Du et al.

An argument is **cheap talk** in the sense of Crawford & Sobel (1982) — a costless, nonbinding message with no direct payoff effect. Truth-aligned rewards do not by themselves make messages cease to be cheap talk. A useful analysis checklist, rather than a necessary-and-sufficient theorem, asks about *diversity* (agents make at least partly independent errors), *legibility* (decisive evidence can be expressed in the transcript), *judge competence* (the aggregator can recognize that evidence), and *incentive alignment* (the rule rewards being right, not agreeing). When these fail, debate still *looks* productive — longer transcripts, more confidence, faster consensus — without producing more truth. Source: Du et al. (2023), Sections 2–3 (empirical evidence); Crawford & Sobel (1982) (cheap talk). The checklist is analytical synthesis.

4.3 Slide 22 – When Communication Helps vs. Enables Collusion

The slide contrasts two columns.

Communication helps when: errors are independent, so aggregation reduces them (ensembling); one agent’s evidence genuinely updates another’s information set (deliberation); the judge and payoff reward decisive evidence; and the comparison to a single agent is fair, holding the model-call budget equal.

Communication harms when: agents share a blind spot and reinforce it (groupthink or a sycophancy cascade); eloquent persuasion beats correct evidence; a strategy *spreads* through the channel (the transmissible reward hacking of Lecture 7.1); or agents coordinate on a joint deviation, as the Hugging Face message board let them do.

The unifying point: the same channel that carries decisive evidence also carries a collusive strategy. Whether communication helps is a property of the *payoff and information structure*, not of “more talking.” Sources: Du et al. (2023); Cemri et al. (2025), FM-2.3 and FM-2.6; the Lecture 7.1 discussion of exploit sharing (OpenAI and METR incident reports).

4.4 Slide 23 – Negotiation with Private Valuations (NegotiationArena)

Bianchi et al. (2024) build NegotiationArena, which pits LLM agents against each other in three families of game — the ultimatum game, resource exchange, and a buy/sell price negotiation — using a structured message format with XML-like tags. The buy/sell game is an **incomplete-information** game: only the seller knows the production cost, and only the buyer knows its willingness to pay.

Three findings matter for us. First, **behavioral tactics pay**: prompting an agent to act “desperate” improved its payoff by about 20% against standard GPT-4. The “cunning” tactic had game-dependent effects, including risky failed trades; it did not uniformly deliver that gain. Second, the agents display **human-like irrationality**: anchoring to the first offer, a “split-the-difference” tendency, and strong effects of turn order and role. Third, there is **fragility**: a weaker agent’s mistakes can drag a stronger agent off its goal (the paper’s “babysitting” effect).

Why game theory cares: negotiation has a hidden-information structure *and* a well-defined numerical outcome, which makes it a clean testbed for whether LLM agents reach rational, incentive-compatible play. Source: Bianchi et al. (2024), Sections 3–5.

4.5 Slide 24 – The Protocol Choice Is Itself a Game (Hua et al. 2024)

Hua et al. (2024) give LLM agents a game-theoretic **workflow** for private-value common-resource allocation: the agent maintains a belief distribution over the opponent’s hidden valuation, updates it

by Bayes’ rule after each rejected offer, and restricts its proposals to **envy-free** allocations (ones in which neither party prefers the other’s share). With the workflow, agreement rates, Pareto-optimality, and total reward all rise sharply relative to raw prompting.

But a workflow agent, because it insists on fairness, makes lower-utility offers than a purely self-interested one. That raises a meta-question: *should you adopt the workflow if your opponent might not?* Represent the choice as a 2×2 “use / don’t use” game. For Claude-3.5 Sonnet this is a **Prisoner’s Dilemma** — “don’t use” is the dominant strategy and the equilibrium is not Pareto-optimal, so both agents end up worse off than if both had cooperated. For Claude-3 Opus, “use” is dominant and the resulting equilibrium *is* Pareto-optimal.

The connection to the course: whether to run a cooperative protocol is itself a game, and it can exhibit a collective-action failure. Repeated play or enforceable agreements are possible interventions to analyze, not guarantees that cooperation will result. Mao et al. (2023), in the Alympics “Water Allocation Challenge,” make the companion point that populations of LLM agents can be studied empirically, EGTA-style, in a resource-scarcity auction. Source: Hua et al. (2024), Sections 5.2–5.6; Mao et al. (2023).

5 Part IV: PA4 and Where This Goes

5.1 Slide 26 – PA4 Preview: Debate Meets EGTA

Programming Assignment 4 (due Week 12) asks you to run today’s whole pipeline on a real system.

1. **Reproduce** Du et al.: implement the debate protocol and show accuracy against a single agent across one, two, and three rounds. Test whether improvement and diminishing returns reproduce; report departures honestly rather than selecting runs to force the expected pattern.
2. **Build an empirical game.** Define four debate strategies — *cooperative*, *stubborn*, *devil’s advocate*, and *analytical* — and estimate the symmetric payoff matrix with confidence intervals (EGTA Steps 2–3 above).
3. **Solve it.** Compute Nash equilibria by support enumeration (reusing your PA1 code) and a coarse correlated equilibrium by running RM+ (reusing your PA2 code). Use the empirical joint distribution of play, not the product of time-averaged marginal strategies, and report its maximum external-regret deviation gain as a finite-run CCE certificate. Ask which configurations are stable and whether any strategy is dominated.
4. **Mechanism design.** Add a scoring rule or a judge agent, re-estimate the affected cells, and test whether the equilibrium shifts.

The debate game is a **common-payoff** (team) game: all agents share the team utility (+1 for a correct final answer, 0 for no consensus, −1 for an incorrect one), yet they choose their strategies independently. In a common-payoff game any payoff-maximizing profile is a Nash equilibrium, so the interesting questions are which strategy combinations maximize the team payoff and whether any strategy is dominated. Report a confidence interval with every payoff cell. The syllabus M7 section names the reproduction target and due week; the PA4 handout Tasks 1–4 supplies these detailed requirements. Consult the handout and Canvas for current submission instructions.

5.2 Slide 27 – How Today Connects to M2–M3

Nash equilibrium, correlated equilibrium, and regret apply to the estimated payoffs of LLM populations.

- **Module 2 (Nash, CE, CCE):** these are the solution concepts EGTA computes on the *estimated* game; the coarse correlated equilibrium is the PA4 target for the debate game.
- **Module 3 (regret and no-regret dynamics):** regret defines the ε -equilibrium and certifies incomplete empirical games, and RM+ from PA2 approximates a CCE when external regret is small.
- **The Module 2 EGTA introduction:** Lecture 8 gave the estimation pipeline and response graphs for PA1; today added player reduction, strategy exploration, and PSRO for *populations*.
- **Modules 4–5 (extensive form):** debate and negotiation are the *designed* protocols; the Lecture 7.1 message board was the *emergent* one.

In one sentence: EGTA lets us bring Nash, correlated, and coarse correlated equilibria, and regret, to games we can only *simulate* — including complex LLM-agent systems.

5.3 Slide 28 – Closing: Model, Predict, Prevent

Multi-agent LLM behavior is analyzable, not mysterious. We can *model* it as an empirical game when no payoff formula exists; *predict* where coordination concentrates using topologies and a failure taxonomy; and *shape* it by designing the communication protocol’s payoffs, treating debate and negotiation as mechanisms rather than mere conversations.

EGTA and failure taxonomies support pre-deployment testing and retrospective analysis. Prevention is incomplete, motivating Module 8: when you cannot *prevent* coordination failures, how do you *detect* them — under a budget, against an adversary who knows you are watching? Module 8 formalizes this as the AI-control game (Greenblatt et al.) and the submodular monitoring problem, and it is the subject of Programming Assignment 5.

References

- OpenAI (2026). *OpenAI–Hugging Face Incident: Technical Report*. Course reading, 38 pp.
- Wijk, Cotra and Greenblatt / METR and Redwood Research (2026). *Brief independent investigation of agents’ behavior, reasoning and collaboration in the OpenAI / Hugging Face hacking incident*. 91 pp.
- Greenblatt et al. (2024). *AI Control: Improving Safety Despite Intentional Subversion*. ICML 2024; Module 8 reading.
- Wellman, M. P. (2006). “Methods for Empirical Game-Theoretic Analysis.” AAAI 2006, 1552–1555.
- Wellman, M. P., Tuyls, K., and Greenwald, A. (2025). “Empirical Game-Theoretic Analysis: A Survey.” *Journal of Artificial Intelligence Research* 82: 1017–1076; arXiv:2403.04018.
- Du, Y., Li, S., Torralba, A., Tenenbaum, J. B., and Mordatch, I. (2023). “Improving Factuality and Reasoning in Language Models through Multiagent Debate.” arXiv:2305.14325.
- Bianchi, F., Chia, P. J., Yuksekgonul, M., Tagliabue, J., Jurafsky, D., and Zou, J. (2024). “How Well Can LLMs Negotiate? NegotiationArena Platform and Analysis.” arXiv:2402.05863.
- Hua, W., Liu, O., Li, L., Amayuelas, A., Chen, J., Jiang, L., Jin, M., Fan, L., Sun, F., Wang, W., Wang, X., and Zhang, Y. (2024). “Game-Theoretic LLM: Agent Workflow for Negotiation Games.” arXiv:2411.05990.

- Mao, S., Cai, Y., Xia, Y., Wu, W., Wang, X., Wang, F., Ge, T., and Wei, F. (2023). “Alympics: LLM Agents Meet Game Theory.” arXiv:2311.03220.
- Zhu, K., Du, H., Hong, Z., Yang, X., Guo, S., Wang, Z., Wang, Z., Qian, C., Tang, X., Ji, H., and You, J. (2025). “MultiAgentBench: Evaluating the Collaboration and Competition of LLM Agents.” ACL 2025; arXiv:2503.01935.
- Cemri, M., Pan, M. Z., Yang, S., Agrawal, L. A., Chopra, B., Tiwari, R., Keutzer, K., Parameswaran, A., Klein, D., Ramchandran, K., Zaharia, M., Gonzalez, J. E., and Stoica, I. (2025). “Why Do Multi-Agent LLM Systems Fail?” NeurIPS 2025 Datasets & Benchmarks; arXiv:2503.13657.
- McMahan, H. B., Gordon, G. J., and Blum, A. (2003). “Planning in the Presence of Cost Functions Controlled by an Adversary.” ICML 2003 (the double-oracle method).
- Lanctot, M., Zambaldi, V., Gruslys, A., Lazaridou, A., Tuyls, K., Pérolat, J., Silver, D., and Graepel, T. (2017). “A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning” (PSRO). NeurIPS 2017.
- Crawford, V. P., and Sobel, J. (1982). “Strategic Information Transmission.” *Econometrica* 50(6): 1431–1451.
- Vorobeychik, Y. (2010). “Probabilistic Analysis of Simulation-Based Games.” *ACM Transactions on Modeling and Computer Simulation* 20(3).
- Knox, W. B., Booth, S., and Christian, B. (2026). “An Unexamined Cause of the OpenAI Hugging Face Hacking Incident: Its Binary Performance Metric.” (Referenced from Lecture 7.1.)