IRIS Gate: a research foundation for multi-LLM scientific convergence

Five AI models reasoning in parallel, then converging on scientific hypotheses, is no longer theoretical — a rich ecosystem of frameworks, statistical methods, and implementation patterns now exists to build it. This report synthesizes 60+ papers and frameworks from 2024–2025 to provide the technical and methodological foundations for IRIS Gate: a staged, multi-model convergence system for scientific hypothesis generation using Claude, GPT, Grok, Gemini, and DeepSeek. The research reveals that architectural diversity across models is the single most important factor for ensemble quality, that structured JSON prompts can boost reasoning accuracy from 31% to 97%, and that Bayesian-optimal aggregation provably outperforms naive majority voting. What follows is organized into seven sections mapping directly to IRIS Gate's design requirements, with implementation-ready patterns throughout.

---

1. Multi-LLM ensemble architectures have matured rapidly

The field has converged on three dominant paradigms for combining multiple LLMs, each with distinct trade-offs relevant to IRIS Gate.

Mixture-of-Agents (MoA) is the most directly applicable pattern. Introduced by Wang et al. (June 2024, Together AI/Stanford), MoA uses a layered architecture where each layer contains multiple LLM agents that receive all outputs from the previous layer as auxiliary context. The key empirical finding: even incorporating outputs from *weaker* models improves a strong model's responses — a phenomenon the authors call "collaborativeness of LLMs." Clioapp On AlpacaEval 2.0, MoA using only open-source models achieved **65.1% versus GPT-4o's 57.5%**. GitHub Performance improves monotonically with additional layers, with 2–3 layers representing the cost-quality sweet spot. The reference implementation at `github.com/togethercomputer/MoA` is under 50 lines of Python.

Debate frameworks have received rigorous theoretical grounding. The NeurIPS 2024 paper by Estornell et al. provides the first mathematical analysis of multi-LLM debate dynamics, proving that **homogeneity in model capabilities leads to static debate dynamics** — essentially an echo chamber effect. This directly validates IRIS Gate's design choice of five *different* model families. Du et al. (MIT, 2023) showed that iterative multi-round debate significantly reduces hallucinations and improves mathematical reasoning, while Zhou &

Chen's Adaptive Heterogeneous Multi-Agent Debate (A-HMAD, 2025) demonstrated **4–6% higher accuracy and >30% fewer factual errors** by assigning distinct expert roles to different agents. A critical practical finding from October 2025 research: anonymizing model identity during cross-evaluation reduces sycophancy and self-bias.

Archon (Stanford Scaling Intelligence Lab, September 2024) is the most implementation-ready framework. It provides a modular architecture combining generators, fusers, critics, rankers, and verifiers in configurable layers, with automated architecture search via Bayesian optimization. It already supports Anthropic, OpenAI, Google, Groq, xAI, and Bedrock APIs through a JSON config file. Archon **outperformed GPT-4o and Claude 3.5 Sonnet by an average of 15.1%** across six benchmarks. For IRIS Gate, Archon's `github.com/ScalingIntelligence/Archon` codebase provides a battle-tested starting point.

The recommended hybrid architecture for IRIS Gate combines all three: Layer 1 generates hypotheses in parallel across all five models, Layer 2 runs anonymized cross-model debate (2–3 rounds), and Layer 3 performs confidence-weighted aggregation through a designated synthesis model. Google's AI Co-Scientist (February 2025) validated this "generate, debate, evolve" pattern by reducing hypothesis generation from weeks to days R&D World and correctly reproducing findings that took human researchers over a decade. R&D World

---

2. Structured prompts and meta-prompting as the compilation layer

The research on prompt structure for scientific contexts is unambiguous: **JSON-structured prompts with hierarchical domain knowledge dramatically outperform natural language**. The TMK (Task-Method-Knowledge) framework showed that converting domain knowledge into JSON-structured format jumped GPT-o1's planning accuracy from **31.5% to 97.3%** on PlanBench. arXiv A 2025 Frontiers in AI study comparing four prompting strategies for statistical reasoning found that hybrid prompting — combining explicit instructions, reasoning scaffolds, and format constraints — was the only approach that consistently produced accurate results for inferential statistics, while zero-shot prompting failed entirely.

For IRIS Gate's "compilation" step where one LLM enriches a raw question before distributing it, Suzgun & Kalai's Meta-Prompting paper (January 2024, Stanford) provides the canonical architecture. Their "conductor/expert" pattern transforms a single LLM into a multi-faceted orchestrator that decomposes complex tasks into subtasks, assigns each to a fresh "expert" instance, and synthesizes results. Prompt Hub The critical design principle is the **"fresh eyes" constraint**: each downstream model receives independent context with no access to prior conversation, preventing error compounding. The implementation is

available at `github.com/suzgunmirac/meta-prompting`.

Three mature prompt optimization frameworks can tune IRIS Gate's compilation layer:

- **DSPy** (Stanford NLP, ICLR 2024) separates the *interface* ("what should the LM do?") from the *implementation* ("how to prompt it"), then optimizes prompts automatically via MIPROv2 or COPRO. DSPy It uses a counterintuitive 20% training / 80% validation split. DSPy Available at `github.com/stanfordnlp/dspy`.
- **TextGrad** (Stanford, published in Nature 2024) implements PyTorch-like "textual backpropagation" — forward pass generates predictions, backward pass produces textual feedback that updates prompts. Semantic Scholar It pushed GPT-3.5 to near-GPT-4 performance Textgrad and designed druglike molecules with desirable binding affinities. arXiv
- **OPRO** (Google DeepMind, ICLR 2024) uses LLMs as black-box optimizers, Semantic Scholar Prompt Hub achieving up to **50% improvement on Big-Bench Hard** through iterative prompt refinement. OpenReview

A key finding from the DSPy+HELM paper (2025): without per-model prompt optimization, benchmarks underestimate LLM performance by 4% on average, and **leaderboard rankings flip on 3 of 7 benchmarks**. arXiv IRIS Gate should maintain model-specific prompt templates, not a universal prompt.

The practical compilation pipeline for IRIS Gate should work as follows: the compiler LLM receives a raw scientific question, enriches it with JSON-structured domain parameters (binding constants, statistical thresholds, relevant constraints), generates a TMK-style hierarchical knowledge scaffold, then adapts the compiled prompt to each downstream model's strengths. Each downstream model operates with fresh context, and the compiler aggregates and verifies all outputs.

---

3. Epistemic calibration reveals when model agreement is meaningful

Understanding what cross-model agreement and disagreement *means* statistically is central to IRIS Gate's convergence scoring. The research paints a consistent picture: **all frontier LLMs are overconfident**, arXiv but the pattern of their errors differs, making ensemble methods valuable.

The KalshiBench study (December 2024) evaluated frontier models on prediction market questions and found Expected Calibration Error (ECE) ranging from **0.12 to 0.40** across models. arXiv A surprising finding: GPT-5.2 with extended reasoning showed the *worst*

calibration (ECE = 0.395) despite using more compute arXiv — what the authors term the "reasoning paradox." The Mind the Confidence Gap paper (February 2025) compared 9 LLMs spanning dense and MoE architectures, finding that dense models with RLHF displayed inherent calibration strengths but paradoxically suffered increased miscalibration on easier queries, arXiv while **MoE models benefited disproportionately from certain prompting interventions**.

For computing convergence across five models, the MUSE framework (EMNLP 2025) provides the most actionable approach. It uses **Jensen-Shannon Divergence (JSD)** — symmetric and bounded [0,1] — to measure disagreement among multiple LLMs, then selects well-calibrated subsets for aggregation. PubMed Central nih The formula for M models: $JSD(P_1...P_m) = H(\Sigma\ w_i P_i) - \Sigma\ w_i H(P_i)$. For categorical/classification outputs, Fleiss' kappa extends naturally to five raters, Wikipedia though the LLM-as-a-Judge literature recommends **Gwet's AC2 over Krippendorff's alpha** for the skewed distributions typical of LLM outputs. Earl-workshop

The most powerful finding for IRIS Gate comes from the Bayesian optimal aggregation literature. The "Beyond Majority Voting" paper (October 2024) proves that majority voting is optimal *only* when sampling repeatedly from the *same* model. For diverse models, **Optimal Weight (OW) aggregation provably dominates**: $w_i = \log(p_i / (1 - p_i))$ where p_i is each model's expected accuracy. This connects directly to the Bradley-Terry model used in RLHF. The paper also introduces Inverse Surprisingly Popular (ISP) — leveraging second-order information (what each model predicts *other* models will say) — which works without ground-truth labels.

The "Wisdom of the Silicon Crowd" study (Science Advances, 2024) validated crowd wisdom for LLMs empirically: aggregating 12 diverse LLMs on 31 forecasting questions achieved a Brier score statistically indistinguishable from human crowds. The "Wisdom of the Machines" study further showed that **model diversity beats temperature-induced diversity** — median aggregation of deterministic outputs from diverse models outperformed 67% of individual guesses. OpenReview Semantic Scholar

The practical convergence scoring system for IRIS Gate should combine: raw agreement proportion, pairwise kappa matrix (10 pairs from 5 models), JSD for probabilistic outputs, coefficient of variation for numerical estimates, and Bayesian OW weighting when model accuracy estimates are available. The `github.com/LARK-NLP-Lab/MUSE` and `github.com/uqlm` repositories provide ready-made Python implementations.

---

4. Staged scientific pipelines define the chamber architecture

Eight major frameworks implementing AI-powered staged scientific discovery were

identified, providing direct architectural blueprints for IRIS Gate's chambers.

Sakana AI's "AI Scientist v2" (April 2025) represents the most sophisticated staged pipeline. arXiv +2 It uses a Progressive Agentic Tree Search Sakana with four experimentation stages: S1 (Preliminary Investigation) establishes feasibility via minimal prototypes, S2 (Hyperparameter Tuning) optimizes critical parameters, S3 (Research Agenda Execution) implements the core research systematically, and S4 (Ablation Studies) assesses component importance. Sakana Each stage has explicit stopping criteria, and **best-first search selects the optimal experimental node to seed the next stage**. GitHub A Vision-Language Model critiques generated figures as a quality gate. Sakana The framework eliminated dependency on human-authored code templates arXiv that limited v1. sakana Sakana

Google's AI Co-Scientist (February 2025) implements a "generate → debate → evolve" cycle arXiv using seven specialized agents: a Supervisor managing asynchronous execution, Generation and Reflection agents for hypothesis creation and critique, a Ranking agent running Elo-style tournaments, a Proximity agent for deduplication, an Evolution agent for iterative refinement, PYMNTS and a Meta-review agent for synthesizing feedback. Medium The most striking empirical result: the system **correctly reproduced unpublished experimental findings on antimicrobial resistance in 2 days** that required over a decade of human research. R&D World Quality improves monotonically with test-time compute. Google Research

MOOSE-Chem (ICLR 2025) takes a mathematically rigorous approach to hypothesis discovery, formally decomposing it as `hypothesis = f(background, inspirations)`. Its three-stage pipeline — Inspiration Retrieval, Hypothesis Composition, Hypothesis Ranking arXiv — successfully rediscovered hypotheses from 51 post-2024 Nature/Science papers. MOOSE-Chem3 (2025) adds an in-context reinforcement learning loop where simulated experimental feedback guides hypothesis re-ranking. arXiv

SciAgents (MIT, published in Advanced Materials 2025) uniquely integrates large-scale ontological knowledge graphs with multi-agent reasoning. nih +2 Its agent chain — Ontologist → Scientist 1 → Scientist 2 → Critic Cloudwalk — produces structured JSON outputs covering hypothesis, outcomes, mechanisms, design principles, unexpected properties, and novelty assessment, Wiley Online Library with each hypothesis generating ~8,100 words of research documentation. GitHub PubMed

The cross-cutting patterns for IRIS Gate's chamber design are clear:

- **Observation Chamber**: Knowledge graph construction (SciAgents) + literature RAG (MOOSE-Chem) + background survey assembly
- **Hypothesis Generation Chamber**: Multi-model parallel brainstorming with novelty

checking (AI Scientist v2) + inspiration-based composition (MOOSE-Chem)

- **Experimental Design Chamber**: Progressive tree search with sub-stages and explicit stopping criteria (AI Scientist v2) + tool-augmented planning (ChemCrow/Coscientist) OAE Publishing
- **Convergence Validation Chamber**: Elo tournament ranking (AI Co-Scientist) + Critic agent review (SciAgents) + replication with statistical aggregation (AI Scientist v2's mean ± std)

Every successful framework implements quality gates between stages — typically LLM-based evaluation with structured scoring rubrics — and feedback loops where downstream validation results propagate back to upstream hypothesis refinement.

---

5. The technical stack for parallel multi-API orchestration

The backend architecture for calling five LLM APIs simultaneously, streaming responses, and computing real-time convergence has a well-established pattern.

LiteLLM (`github.com/BerriAI/litellm` , 18k+ GitHub stars) is the clear choice for the unified API layer. It provides an OpenAI-compatible interface for 100+ providers GitHub TrueFoundry with **~12ms median proxy overhead**, Createaiagent supporting async streaming via `acompletion(model=..., stream=True)` , litellm standardized exception mapping across providers, LiteLLM built-in rate limiting, LiteLLM router fallbacks, LiteLLM and per-request cost tracking. Statsig A single config YAML file maps model aliases to provider-specific endpoints DeepWiki for Claude, GPT, Grok, Gemini, and DeepSeek.

The core streaming pattern uses `asyncio.gather()` to fire all five API calls simultaneously, DEV Community with each task yielding chunks through an async iterator. Medium **FastAPI with WebSocket support** serves as the orchestration layer Leapcell (ASGI-based, FastAPI supports 45k+ concurrent WebSocket connections). Medium The WebSocket protocol uses typed JSON messages: `"chunk"` messages stream per-model token output, `"metrics"` messages push periodic convergence updates, and `"complete"` messages deliver final aggregate scores.

The real-time metrics engine computes convergence incrementally as responses stream. The recommended stack includes Jaccard similarity for fast lexical overlap Newscatcherapi Studymachinelearning (computable on every chunk), `sentence-transformers` (all-MiniLM-L6-v2) for semantic similarity Hugging Face (computed every 2–5 seconds as text accumulates), and the full convergence scorer (JSD, kappa, Bayesian OW) on completion. A

`StreamingMetrics` class tracks per-model full text and triggers metric computation at configurable intervals.

For the WebGL frontend, **Three.js** renders NetBurner five model nodes as colored spheres positioned in a circle, with connection lines between them whose opacity reflects pairwise similarity. As convergence increases, nodes animate toward the center via `lerp()` interpolation. Particle systems (via `three-nebula` GitHub or `three.quarks`) GitHub can visualize token flow from each model. D3.js provides supplementary 2D metric overlays. D3

The complete stack: FastAPI + uvicorn (ASGI server) Medium → LiteLLM (unified LLM gateway) Createaiagent → asyncio (concurrent task management) DEV Community → sentence-transformers + numpy + scikit-learn (metrics) → WebSocket (real-time transport) → Three.js + D3.js (visualization). Error handling uses exponential backoff per provider with LiteLLM's built-in retry policy, Statsig Statsig and a semaphore-based rate limiter prevents exceeding per-provider RPM/TPM quotas.

---

6. Scientific prompting that maximizes signal density

The research on reducing unnecessary hedging while maintaining accuracy reveals counterintuitive findings that directly inform IRIS Gate's system prompts.

Simple persona labels ("You are a scientist") have minimal effect on accuracy. Zheng et al. (2024) tested 162 personas across multiple LLM families on 2,410 factual questions and found no improvement over control. Mollick et al. (2025) confirmed that in-domain expert personas had "no significant impact on performance." SSRN However, **auto-generated detailed expert identities** using the ExpertPrompting framework (Xu et al., 2023) significantly improved answer quality — preferred **48.5% versus 23%** over vanilla answers. arXiv The difference is granularity: a paragraph-length identity with specific credentials and expertise outperforms a one-line role label.

The most impactful technique is **expert-to-expert audience specification**. Google's AI Co-Scientist templates consistently include "This description is intended for an audience of domain experts" learnprompting — which increases technical vocabulary, eliminates explanatory padding, reduces hedging qualifiers, and increases willingness to make specific claims. An empirical study by Vasquez (2025) measuring "conversational pressure" found that co-facilitative framing reduced hedging from 4.31/5 to **1.39/5** across 120 trials (κ = 0.84 inter-rater reliability). Scientific Document

For scientific reasoning specifically, **self-consistency outperforms chain-of-thought**. Rueda et al. (2025) tested seven prompting techniques on graduate-level scientific

questions (GPQA): self-consistency achieved 52.99% accuracy while standard CoT scored only 43.75% — the *worst* performer. For probabilistic reasoning, Nature Communications (2025) showed that Bayesian teaching examples dramatically improve LLM inference while standard CoT does not.

The practical anti-hedging system prompt for IRIS Gate should combine several elements. It should explicitly ban low-information phrases ("it's important to note," "it's worth mentioning"), require epistemic status tags ([ESTABLISHED], [PROBABLE], [SPECULATIVE]) on each claim, request point estimates with stated confidence intervals rather than vague ranges, and specify expert audience throughout. A critical model-specific finding: prompt format sensitivity varies up to **40% across model families**, arXiv so IRIS Gate must maintain separate optimized templates per model rather than a universal prompt.

---

7. Putting it together: the IRIS Gate implementation blueprint

Synthesizing across all seven research domains, the IRIS Gate architecture emerges as a four-chamber pipeline with a meta-prompting compilation layer and real-time convergence scoring.

Chamber 0 — Prompt Compilation (single compiler LLM): Receives raw scientific question. Enriches with JSON-structured domain parameters (TMK hierarchy), quantitative constraints, and literature context. Generates five model-specific prompts adapted to each downstream model's strengths and format preferences. Uses DSPy's MIPROv2 for offline prompt optimization per model. DSPy

Chamber 1 — Parallel Hypothesis Generation: All five models receive their compiled prompts simultaneously via LiteLLM async streaming. litellm Each operates with fresh context (no cross-contamination). Outputs stream to frontend via WebSocket, with real-time Jaccard and embedding similarity computed incrementally. System prompts use ExpertPrompting with auto-generated identities, arXiv expert audience specification, epistemic status tagging, and anti-hedging directives.

Chamber 2 — Anonymized Cross-Model Debate: Each model receives all five anonymized hypotheses and provides structured critique (following A-HMAD's role-specialized pattern). Two to three debate rounds with convergence monitoring. Models assigned complementary roles: methodological rigor, creative alternatives, evidence synthesis, mathematical verification, literature grounding. Convergence metrics (JSD, Fleiss' kappa, cosine similarity of reasoning traces) computed after each round.

Chamber 3 — Convergence Validation: Bayesian OW aggregation combines model outputs with accuracy-weighted scoring. Elo-style tournament ranking (per Google Co-

Scientist) produces hypothesis quality ordering. Google Research Learn Prompting The four-category disagreement taxonomy — within-aligned, within-misaligned, between-aligned, between-misaligned — flags cases requiring human attention. arXiv Final convergence score combines raw agreement, kappa, JSD convergence, and CV, with task-specific weighting. Hypotheses above a threshold advance; below-threshold cases trigger additional debate rounds or human review.

The key statistical insight underpinning the entire system: **model disagreement is signal, not noise.** Cross-family disagreement is systematic and predictable, reflects genuinely ambiguous items, and can be modeled to improve downstream decisions. arXiv With five architecturally diverse models (dense RLHF, dense Constitutional AI, dense multimodal, MoE reasoning, and MoE code-augmented), IRIS Gate maximizes the Q-statistic measuring error diversity — the mathematical prerequisite for ensemble superiority over any individual model. Emergent Mind

Conclusion

IRIS Gate sits at the intersection of three rapidly maturing fields: multi-LLM orchestration, automated scientific discovery, and ensemble uncertainty quantification. The research reveals several non-obvious design principles. First, model diversity matters more than model quality — five different architectures provably outperform five instances of the strongest model. OpenReview Second, structured JSON prompts with hierarchical domain knowledge aren't a minor optimization but a transformative one, with accuracy gains exceeding 3x in procedural tasks. arXiv Third, Bayesian-optimal weighting should replace majority voting, as the two are only equivalent in the degenerate case of identical models. arXiv Fourth, self-consistency (multiple independent reasoning paths compared) outperforms chain-of-thought for scientific reasoning. arXiv And fifth, the "fresh eyes" constraint — each model receiving independent context in debate rounds — prevents the error compounding that degrades iterated self-reflection.

The most actionable starting points are Archon for the multi-model orchestration scaffold, LiteLLM for the unified API layer, Createaiagent TrueFoundry DSPy for prompt compilation and optimization, DSPy and the MUSE/UQLM packages for convergence scoring. Together, these provide perhaps 60–70% of the infrastructure needed, with IRIS Gate's novel contribution being the domain-specific chamber architecture, the compilation layer embedding scientific priors, and the real-time convergence visualization.