Tribunal Session Summary

Session: conclave-20260604-021619-is-10x-compute-required-9d87

Depth: T6

Advocates: 6 | Judges: 5 | Cost: $8.3537 | Time: 37m 24s

THE PROMPT: Is "10x Compute" required for each doubling of model intelligence?

THE RULING: No. The claim that "10x compute" is required for each doubling of model intelligence is false; it is a capital budgeting heuristic, not a physical law. The correct compute-optimal scaling laws show that 10x compute yields only a ~30% reduction in loss, whereas halving loss actually requires ~60–90x compute. Furthermore, "intelligence" (capability) does not scale linearly with loss; it is decoupled by algorithmic efficiency, test-time reasoning, and data quality, making any fixed compute multiplier impossible.

The Prompt

Is "10x Compute" required for each doubling of model intelligence?

The Ruling

No. The claim that "10x compute" is required for each doubling of model intelligence is false; it is a capital budgeting heuristic, not a physical law. The correct compute-optimal scaling laws show that 10x compute yields only a ~30% reduction in loss, whereas halving loss actually requires ~60–90x compute. Furthermore, "intelligence" (capability) does not scale linearly with loss; it is decoupled by algorithmic efficiency, test-time reasoning, and data quality, making any fixed compute multiplier impossible.

The strongest evidence is the mathematical derivation of the effective compute exponent. While the "10x" rule attempts to derive a constant from scaling laws, it conflates per-resource exponents (how loss scales with model size or data individually) with the effective frontier exponent (how loss scales with total compute budget). The correct effective exponent is ~0.15–0.18, meaning the relationship is far steeper than the slogan suggests. Additionally, real-world capability doublings—such as the jump from GPT-3 to GPT-4—required ~60–70x compute, aligning with the corrected math, not the 10x heuristic. The rule survives only as a rough planning guide for capital expenditure, not as a requirement for capability gains.

Core Analysis

The Mathematical Error Behind the Slogan

The core of the debate was a mathematical correction that dismantled the "10x" rule. The confusion stems from misapplying the exponents found in the Chinchilla scaling laws (Hoffmann et al., 2022).

- **The Per-Resource Exponents:** Chinchilla established that reducible loss scales with parameters (N) and data (D) according to exponents $\alpha \approx 0.34$ and $\beta \approx 0.28$. If you look at these numbers in isolation, you might mistakenly assume the compute exponent is in this range.
- **The Effective Frontier Exponent:** When you optimize for total compute budget (C) where $C \propto N \cdot D$, the math changes. The effective exponent for how loss scales with compute is derived by combining the two: $\alpha_{eff} = \frac{\alpha\beta}{\alpha+\beta}$.
- **The Arithmetic:** Plugging in the Chinchilla values ($\alpha \approx 0.34, \beta \approx 0.28$) yields $\alpha_{eff} \approx 0.15$. Even using updated fits (Besiroglu et al., 2024), the value is only ~0.18.
- **The Consequence:** A power law with an exponent of 0.15 is very steep. A 10x increase in compute reduces reducible loss by only $10^{-0.15} \approx 30\%$. To actually *halve* the reducible loss (a proxy for a major capability step), you need $2^{1/0.15} \approx 90\times$ more compute.

This proves the "10x" rule is mathematically invalid for loss reduction. The slogan conflates the *per-resource* slopes with the *compute-frontier* slope, a category error that makes the required compute look 6–9 times

cheaper than it actually is.

Loss vs. Intelligence: The Non-Linear Mapping

Even if the math supported a 10x loss reduction, "loss" is not "intelligence."

- **The Irreducible Floor:** Loss scales toward an irreducible entropy floor (E). You cannot "double" a quantity that asymptotes to a minimum. The concept of "doubling intelligence" is operationally meaningless on a loss scale.
- **Log-Linear Capability:** The mapping from loss reduction to downstream benchmark capability (e.g., MMLU, GSM8K) is log-linear, not multiplicative. This means each order of magnitude of compute tends to produce a *constant additive* improvement in capability, not a constant multiplicative doubling.
- **Metric Dependence:** Capabilities often emerge discontinuously. A model might show zero performance on a task for orders of magnitude of compute, then suddenly jump to competence. This makes a fixed "per-doubling" coefficient impossible—the compute-to-capability ratio is effectively infinite right before the jump and very low right after it.

The Decoupling Levers: Why 10x Isn't Required

Capability gains are increasingly achieved without proportional training compute increases, further invalidating the "requirement" claim.

- **Algorithmic Efficiency:** Research (Ho et al., 2024; Hernandez & Brown, 2020) shows the compute required to reach a fixed performance level has been halving roughly every 8–16 months. This compounds over time, meaning the "effective compute" available to a lab grows faster than just the hardware count.
- **Test-Time Compute:** Techniques like Chain-of-Thought (CoT) prompting allow models to "think longer" at inference time. On GSM8K, enabling CoT nearly halved the error rate (from ~82% to ~43%) with *zero* additional training compute. This shifts the cognitive load from pre-training (expensive, lumpy) to inference (cheap, elastic).
- **Data Quality:** Models like Microsoft's Phi series demonstrated that highly curated "textbook" data allows small models (1.3B) to match the performance of models 10–50x their size on reasoning tasks. This bends the scaling curve by shifting the intercept rather than increasing the slope.

Empirical Validation: The GPT-3 to GPT-4 Transition

The historical record supports the corrected math, not the slogan.

- **The Compute Jump:** Public estimates place the GPT-3 to GPT-4 training compute increase at roughly 60–70x.
- **The Capability Gain:** This jump produced a significant but not "infinite" leap in capability (e.g., MMLU score rising from ~43% to ~86%).
- **The Alignment:** The 60–70x compute multiplier aligns perfectly with the *corrected* theoretical requirement for halving loss (~60–90x), and contradicts the "10x" heuristic. The industry practice of scaling by "orders of magnitude" reflects the actual math (60–100x), not the simplified slogan (10x).

How We Got Here

Scorecard

Judge	GPT-5	Claude Sonnet 4	Gemini 3.1 Pro	Grok 4	DeepSeek V4 Pro	Winner
GPT-OSS 120B	ADOPT	ADOPT	REJECT	REJECT	REJECT	GPT-5
MiniMax M2.7	ADOPT	ADOPT	REJECT	REJECT	REJECT	Claude Sonnet 4
Zhipu GLM-4.7	ADOPT	ADOPT	REJECT	REJECT	REJECT	Claude Sonnet 4
Zhipu GLM-5.1	ADOPT	ADOPT	REJECT	REJECT	REJECT	GPT-5

Convergence Assessment

The convergence was **epistemic** (evidence-driven). The pivotal moment was the mathematical derivation of the effective exponent. Grok 4 flipped from YES to NO specifically because the arithmetic ($\alpha_{eff} \approx 0.15$) proved their previous position mathematically impossible. There was no evidence of models changing positions to agree with the majority; they changed positions because the math was checked and found wanting.

Closing Statements

GPT-5: GPT-5 maintained that a 10x increase in compute is not required. They refined their argument to focus on the correct compute-optimal exponent ($\alpha_{eff} \approx 0.15–0.18$), proving that 10x compute yields only a ~30% loss reduction. They emphasized that capability doublings are routinely achieved via test-time reasoning (CoT) and algorithmic efficiency without 10x training compute. Their position remained stable, grounded in the portfolio of levers available beyond raw FLOPs.

DeepSeek V4 Pro: DeepSeek initially argued the required compute was "thousands to millions" of times larger than 10x, based on the Kaplan exponent. After conceding this was an error, they revised to argue that while the *loss* reduction might require 8–12x compute (using per-resource exponents), "intelligence" doubling is not equivalent to loss halving. They maintained that the 10x rule is false because capability is metric-dependent and can be achieved via inference-time methods like CoT.

Claude Sonnet 4 (Claude Opus 4.8): Claude provided the definitive mathematical correction, deriving $\alpha_{eff} \approx 0.15–0.18$ and showing that halving loss requires ~60–90x compute. They argued that "10x" is a folk slogan that compresses a real phenomenon (power-law scaling) into a wrong formula. Their final position highlighted that loss scales predictably, but intelligence does not—it is log-linear and

metric-dependent. They made a major revision to their own math (abandoning a 0.5 exponent) to align with the correct derivation.

Gemini 3.1 Pro: Gemini argued that 10x physical compute isn't required due to algorithmic progress and test-time scaling. They initially claimed the 10x rule derived from an exponent of 0.3, but conceded this was a category error. Their refined position was that the 10x rule is "dead on arrival" mathematically (since 10x only cuts loss by 30%) and that alternative levers (inference scaling, data density) provide capability gains without proportional training compute.

Grok 4: Grok was the lone "YES" advocate initially, citing industry consensus (Musk, Morgan Stanley). They argued the 10x rule was the practical translation of scaling laws. However, in Round 7, they conceded fully. They accepted the corrected exponent ($\alpha_{eff} \approx 0.15$), acknowledged that 10x compute does not halve loss, and admitted that the GPT-3 to GPT-4 jump (~60–70x) aligned with the *corrected* math, not the 10x slogan. They concluded the rule is a budgeting heuristic, not a requirement.

Key Moments

- **The Exponent Derivation (Rounds 5–6):** GPT-5 and GPT-5dvocate-C independently derived the correct effective compute exponent ($\alpha_{eff} \approx 0.15$) using the Lagrange-multiplier solution $\alpha_{eff} = \alpha\beta/(\alpha+\beta)$. This mathematical proof exposed that Gemini 3.1 Pro and GPT-5dvocate-E were using per-resource exponents (~0.3) as if they were the compute exponent, a fatal error that invalidated the "10x halves loss" claim.
- **The "Millions of Times" Collapse (Round 5):** DeepSeek V4 Pro was forced to abandon their opening claim that halving loss requires "millions of times" more compute. This claim relied on the outdated Kaplan exponent ($\alpha \approx 0.05$), which applies to non-optimal training regimes. The correction to the Chinchilla regime reduced the multiplier from ~1,000,000x to ~60–90x.
- **The Flip (Round 7):** Grok 4, the sole advocate defending "YES," conceded their position. After the arithmetic correction showed 10x compute only reduces loss by ~30%, and acknowledging that GPT-3 to GPT-4 required ~60–70x compute, E admitted the "10x" rule was a budgeting heuristic, not a law.
- **The CoT Counterexample (Round 3):** GPT-5 successfully defended the claim that single-sample Chain-of-Thought prompting nearly halves error on GSM8K with zero extra training compute. This provided a concrete, empirical falsification of the idea that 10x *training* compute is "required" for a capability doubling.

Reality Checks

Claim	Challenge	Result
DeepSeek V4 Pro: Halving loss requires ~1,000,000x compute (based on Kaplan $\alpha \approx 0.05$).	**GPT-5 / Claude Sonnet 4:** The Kaplan exponent applies to non-optimal scaling. The Chinchilla compute-optimal exponent is ~0.15, requiring ~60–90x compute.	**Full Concession.** DeepSeek abandoned the "millions of times" claim and accepted the corrected exponent.

Claim	Challenge	Result
Gemini 3.1 Pro / Grok 4: 10x compute halves reducible loss (based on $\alpha \approx 0.3$).	**Claude Sonnet 4 / GPT-5:** The 0.3 figure is a per-resource exponent. The effective frontier exponent is $\alpha_{eff} = \alpha\beta/(\alpha+\beta) \approx 0.15$. At 0.15, 10x compute only reduces loss by ~30%.	**Full Concession.** Both advocates admitted the arithmetic error. Grok 4 subsequently flipped their position from YES to NO.
Grok 4: "10x compute doubles intelligence" is an industry consensus validated by Morgan Stanley and Elon Musk.	**Claude Sonnet 4:** This is a circular citation cascade. Musk's statement is a single soundbite, not a peer-reviewed derivation. No paper shows a 10x compute increase systematically doubles benchmark performance.	**Concession.** Grok 4 admitted the claim relied on hearsay and investor slides rather than scientific evidence.

Next Steps

Next Step	Why It Matters	Confidence
Define "Intelligence Doubling"	The debate revealed "intelligence" lacks a cardinal metric. A standardized definition (e.g., error-halving on a fixed benchmark suite) is needed to make future scaling claims falsifiable.	High (GPT-OSS 120B, MiniMax M2.7, and Zhipu GLM-4.7)
Measure Inference Compute Scaling	As training compute hits diminishing returns, inference-time compute (CoT, search) is becoming the primary lever for reasoning gains. We need better accounting of total FLOPs (training + inference) per task.	High (GPT-OSS 120B, MiniMax M2.7, and Zhipu GLM-4.7)
Track Algorithmic Efficiency Rates	The "2-3x per year" efficiency gain is critical. If this slows, the physical compute burden rises. Continuous monitoring of this trend is essential for forecasting.	Medium (Zhipu GLM-4.7)

Dissenting Opinions

No dissenting opinions were filed in this session.

Glossary

Term	Definition
Compute-Optimal Scaling	A training regime where the model size (N) and dataset size (D) are scaled together to minimize loss for a fixed compute budget (C).
Effective Exponent (α_{eff})	The exponent describing how reducible loss scales with total compute budget ($L \propto C^{-\alpha_{eff}}$). Derived from per-resource exponents as $\alpha_{eff} = \alpha\beta/(\alpha+\beta)$.
Irreducible Loss	The theoretical minimum error a model can achieve, determined by the entropy (noise) inherent in the data distribution.
Log-Linear Scaling	A relationship where a logarithmic change in one variable (compute) produces a linear change in another (capability). This implies constant *additive* gains per order of magnitude, not multiplicative doublings.
Per-Resource Exponents	The exponents (α, β) describing how loss scales with model size or data size *individually*, holding the other constant.
Reducible Loss	The portion of model error that can be eliminated by scaling compute, data, or parameters.
Test-Time Compute	Compute expended during the inference phase (when the model is generating answers) to improve performance, e.g., via Chain-of-Thought reasoning or search algorithms.

How The Conclave Works

The Conclave is a structured multi-model deliberation system. Instead of asking one AI model a question and trusting its answer, the Conclave convenes a panel of independent models, forces them to argue, and subjects their conclusions to judicial review. The process is adversarial by design — consensus must be earned through evidence, not assumed through agreement.

Advocates are independent AI models (e.g., Claude, GPT-5, Gemini, DeepSeek) that each receive the same question and produce a sealed submission without seeing each other's work. Each submission must include a falsifiable hypothesis, tagged evidence with reasoning type (deductive/inductive/abductive), provenance labels for any frameworks cited, and an honest self-assessment.

Challenges follow submissions. Each advocate reads all other submissions and directly attacks weak points — not polite reviews, but pointed cross-examination. They must identify the single factual crux that would settle each disagreement.

Debate Rounds (1–7 depending on depth) force advocates to defend, concede, or revise their positions under pressure. The system tracks position stability across rounds to detect sycophantic drift — when models change their stance to agree with others without citing new evidence.

Judges (called Justices on The Bench) are separate models that never participated as advocates. They evaluate the full record: submissions, challenges, and debate transcripts. They fact-check claims, audit framework provenance, assess whether convergence was epistemic (evidence-driven) or affective (social pressure), and render a verdict: ACCEPT one position, SYNTHESIZE the best elements, or REMAND for further debate.

Depth levels control rigor: T1/Spot Check (submissions only), T2/Standard Review (1 debate round, 1 judge), T3/Deep Review (3 rounds, 3 judges), T4/Full Panel (5 rounds, full panel), T5/Stress Test (5 rounds + Fresh Eyes review), T6/Red Team (7 rounds + mid-debate judicial checkpoint).

The system is deterministic code — it cannot be sycophantic. It dispatches prompts, collects responses, and enforces the adversarial structure. The models argue; the code referees.