Test-Time Compute Games

Ander Artola Velasco
Max Planck Institute for Software Systems
Kaiserslautern, Germany
avelasco@mpi-sws.org

Dimitrios Rontogiannis
Max Planck Institute for Software Systems
Kaiserslautern, Germany
drontogi@mpi-sws.org

Stratis Tsirtsis
Hasso Plattner Institute
Potsdam, Germany
stratis.tsirtsis@hpi.de

Manuel Gomez-Rodriguez
Max Planck Institute for Software Systems
Kaiserslautern, Germany
manuel@mpi-sws.org

Abstract

Test-time compute has emerged as a promising strategy to enhance the reasoning abilities of large language models (LLMs). However, this strategy has in turn increased how much users pay cloud-based providers offering LLM-as-a-service, since providers charge users for the amount of test-time compute they use to generate an output. In our work, we show that the market of LLM-as-a-service is socially inefficient—providers have a financial incentive to increase the amount of test-time compute, even if this increase contributes little to the quality of the outputs. To address this inefficiency, we introduce a reverse second-price auction mechanism where providers bid their offered price and (expected) quality for the opportunity to serve a user, and users pay proportionally to the marginal value generated by the winning provider relative to the second-highest bidder. To illustrate and complement our theoretical results, we conduct experiments with multiple instruct models from the `Llama` and `Qwen` families, as well as reasoning models distilled from `DeepSeek-R1`, on benchmark datasets covering mathematics, science, and open-ended question answering.

1 Introduction

The massive computational resources required to run large language models (LLMs) have led many users to rely on a growing market of cloud-based service providers that offer LLM-as-a-service. A key driver of this computational demand is the use of test-time compute (TTC)—additional computations performed by an LLM at inference time to improve its performance using techniques such as chain-of-thought [1], tree-of-thought [2], and best-of-n sampling [3]. Importantly, in the LLM-as-a-service market, the cost of these additional computations ultimately falls on the users, who pay for all tokens generated by a model during inference, including intermediate tokens that are not visible in the final outputs observed by the users.

In this context, providers have the flexibility to (dynamically) adjust the amount of test-time compute an LLM uses to respond to a user's query.[1] However, in a competitive market, this flexibility creates a new strategic dimension beyond how providers price their services. In particular, providers can strategically increase the amount of test-time compute allocated to a user's query to maximize profit, even if the additional test-time compute contributes little to the quality of the response. Consider a simple illustrative example where, for a given set of queries with verifiable ground truth (*e.g.*,

[1] For example, GPT-5 uses real-time routing to adjust reasoning depth, and thus test-time compute. See `https://openai.com/index/introducing-gpt-5/`.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

diagnosing patients based on their medical records), two different providers can run their LLMs in either a low-TTC mode (*e.g.*, standard generation) or a high-TTC mode (*e.g.*, chain-of-thought), with average accuracies and generation costs for the providers given by:

Provider 1			**Provider 2**		
	Avg. accuracy	Avg. gen. cost		Avg. accuracy	Avg. gen. cost
Low TTC	70%	\$0.25	Low TTC	50%	\$0.5
High TTC	90%	\$1	High TTC	95%	\$10

If both providers price their models with a 25% profit margin over their generation costs, a user who values each percentage point of accuracy as $\$0.02$ would always select the first provider, who offers them higher value (*i.e.*, $\$0.02 \times \text{accuracy} - \text{price}$) independently of the TTC mode chosen by the second provider. However, to increase their profit, the first provider is financially incentivized to choose the high-TTC mode, even though the low-TTC mode would maximize the sum of the user value and provider profit, and would therefore be socially optimal.[2]

In our work, we formally characterize the above gap and show that, when providers act to maximize profit in a competitive market, they cannot, in general, be expected to use the socially optimal amount of test-time compute. Further, we conceptualize and analyze a forward-looking market where users and providers participate in a reverse second-price auction, and show that, in contrast to the current pay-for-compute market, this auction mechanism is guaranteed to be socially optimal.

Our contributions. We start by modeling the current LLM-as-a-service market as a normal-form game [4] between N providers who choose their test-time compute allocations to maximize profit, and users pay according to the price per compute set by their selected provider. Building on this characterization, we make the following contributions:

1. We show that the LLM-as-a-service market admits a pure Nash equilibrium and that, under natural assumptions, the market providers are guaranteed to reach this equilibrium in finite time.
2. We characterize the test-time compute used by providers in an LLM-as-a-service market at a Nash equilibrium and show that, in general, it is not socially optimal.
3. We introduce a reverse second-price auction mechanism where providers bid with their offered price and (expected) quality for the opportunity to serve a user, and the user's price is determined by the marginal value generated by the winning provider relative to the second-highest bidder.
4. We show that the above auction mechanism is socially optimal, and it guarantees that users always obtain at least their second-best achievable value for the task and providers secure non-negative profits.

To complement our theoretical results, we conduct experiments with multiple instruct models from the `Llama` and `Qwen` families, as well as reasoning models distilled from `DeepSeek-R1`, on math, science, and open-ended question answering benchmark datasets. Our results show that the existing pay-for-compute market is up to 19% (socially) inefficient, as measured by the Price of Anarchy. The code and data for our experiments are available at `https://github.com/Human-Centric-Machine-Learning/strategic-ttc`.

Further related work. Our work connects to the rapidly growing literature on the economic and strategic aspects of machine learning systems [5, 6, 7, 8, 9, 10], and more specifically, LLMs-as-a-service [11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22]. Therein, most closely related to ours is a strand of recent works that analyzes the incentives of providers of LLM-as-a-service when they act strategically. Specifically, Velasco et al. [18, 19] study how providers can overcharge users by misreporting the number of tokens used by the LLM to encode a given text, and introduce an auditing method to detect such behavior. In a similar vein, Sun et al. [22] and Cao et al. [21] consider a setting where a provider injects additional reasoning tokens to inflate the user's payment, while [16] and Cai

[2]We formally introduce user value and provider utilities in Section 2. For the exact derivations related to this example, refer to Appendix A.

et al. [15] study a scenario in which providers have a financial incentive to be unfaithful to users by deploying cheaper-to-run models to maximize profit.

Further, our work relates to a line of work that analyzes the capabilities enabled by test-time compute [23, 24, 25, 26, 27]. While most of the literature has focused primarily on performance aspects, there is increasing interest in analyzing the economic implications of test-time compute. In particular, recent works by Wang et al. [28], Zellinger and Thomson [29] and Erol et al. [30] argue for incorporating the substantial costs of test-time compute into LLM evaluation and ranking, while Wan et al. [31], Komiyama et al. [32] and Kalayci et al. [33] develop methods for selecting test-time compute resources once performance gains become marginal under majority voting and best-of-n. However, our work is the first to show that, in the LLM-as-a-service market, providers are incentivized to strategically decide about the amount of test-time compute used by the LLMs they serve, and to examine the resulting effects on the social welfare.

Finally, our work also draws on the classic literature in auctions and mechanism design [34, 35, 36, 37, 38], which has studied optimal auction design [39, 40, 41, 42, 43, 44, 45, 46, 47], mechanisms to incentivize truthful bidding [48, 49, 50, 51, 52, 53], and the (in)efficiency of various markets when players act in their own interest [54, 55, 56, 57]. Within this literature, the mechanism we propose for test-time compute markets is closely related to scoring auctions studied in the context of procurement contracts and supplier selection [37, 58, 59], where bidders submit offers specifying not only a price but also additional attributes of their offer (*e.g.*, quality). In many settings, such scoring auctions have been shown to outperform first-price auctions, achieving higher efficiency [60, 61]; yet, to the best of our knowledge, they have not been considered in the context of LLM-as-a-service.

2 A Game-Theoretic Model of Test-Time Compute

We model the LLM-as-a-service market as a normal-form (test-time compute) game $\mathcal{G}$ [4] between N providers who simultaneously deploy their own pre-trained model and compete for user demand by strategically selecting the level of test-time compute they use. In practice, each provider $i \in [N]$ can select different levels of test-time compute for different types of tasks (*e.g.*, low for fact retrieval and high for mathematical reasoning or coding), and their choice of compute for one task does not restrict their choices for other tasks. As a result, providers compete to serve each task independently, and each task defines an independent game between providers.

More formally, given a task $\mathcal{T}$ characterized by a set of queries $\mathcal{Q}$, each provider can select a test-time compute level $\theta \in \Theta$ for the model they serve from a finite set Θ of test-time compute levels.[3] For instance, in test-time compute methods such as majority voting [62] and best-of-n sampling [3], each level of test-time compute θ naturally corresponds to the number of generated samples and, in chain-of-thought [1], it may correspond to different "reasoning effort" levels. Then, given a test-time compute level θ, the model served by provider i produces an average output quality $q_i(\theta) \in \mathbb{R}_+$ for the task $\mathcal{T}$, which may scale differently across providers, *i.e.*, $q_i \neq q_j$ for $i \neq j$. In practice, the average output quality q_i may correspond to average accuracy across queries if the task $\mathcal{T}$ has a verifiable ground truth, or to an average user satisfaction score if the task $\mathcal{T}$ is open-ended. Moreover, there is extensive empirical evidence [23, 24, 25] that the quality q_i typically increases with the level of test-time compute, *i.e.*,

$$q_i(\theta) \leq q_i(\theta') \quad \text{for} \quad \theta \prec \theta', \tag{1}$$

where $\prec$ denotes a total order in the space Θ describing whether θ corresponds to a level of compute that is on average lower than θ' (*e.g.*, as measured by the average number of tokens used by the model per query).

Although provider i can increase the average quality $q_i(\theta)$ they offer to users by selecting a higher level of test-time compute θ, this comes at the expense of a higher (average) generation cost $c_i(\theta) \in \mathbb{R}_+$ for the provider. As a result, to compensate for higher generation costs, the price $p_i(\theta)$ charged by each provider i increases with the level of test-time compute θ, *i.e.*, $p_i(\theta) < p_i(\theta')$ for $\theta \prec \theta'$. Here, since each provider i is unlikely to set a price per unit of test-time compute lower than the cost,[4] we assume any increase in test-time compute leads to a positive marginal profit for the provider, *i.e.*,

$$p_i(\theta) - c_i(\theta) < p_i(\theta') - c_i(\theta') \quad \text{for} \quad \theta \prec \theta'. \tag{2}$$

[3] For ease of exposition, we consider that all providers share the same action space Θ.

[4] In practice, a provider's generation cost and price for a model's output is proportional to the number of tokens it used to generate the output.

Table 1: Market share function, $s_i(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i}))$

Perfect rationality	Bounded rationality
$\mathbb{1}\{V_i(\theta_i) > \max\{\boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})\}\}$	$\frac{e^{\beta V_i(\theta_i)}}{e^{\beta V_0} + \sum_{j=1}^{N} e^{\beta V_j(\theta_j)}}$

Further, given a test-time compute level θ, each provider i offers a certain average value $V_i(\theta)$ to users for the task $\mathcal{T}$, which is given by the difference between the average output quality $q_i(\theta)$ produced by their model and the average price $p_i(\theta)$ charged by the provider for task $\mathcal{T}$. That is,

$$V_i(\theta) = q_i(\theta) - p_i(\theta), \tag{3}$$

where we assume no ties in the values offered by different providers, *i.e.*, $V_i(\theta) \neq V_j(\theta')$ for all $i \neq j$ and all $\theta, \theta' \in \Theta$, since in practice both q_i and p_i refer to average quantities and thus take real values.[5]

Given the test-time compute levels selected by all providers, $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_N) \in \Theta^N$, users allocate their demand across providers based on the value $V_i(\theta_i)$ offered by each provider, but they also have the option to abstain from using any provider if none of them offers a value higher than an abstention threshold value $V_0 \in \mathbb{R}$. More formally, we characterize the allocation of user demand across providers through a market share function $s : \mathbb{R}^{N+1} \to [0,1]^{N+1}$ that maps the values offered by all providers and the abstention value to the fraction of queries served by each provider and the fraction of queries where users abstain. Moreover, as is typical in game theory, we use the notation $s_i(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i}))$ to denote the market share of provider i as a function of their offered value $V_i(\theta_i)$ while keeping fixed all other values $\boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})$ (*i.e.*, the abstention value and the values offered by providers other than i).

In the remainder of the paper, we will focus on the two canonical market share functions shown in Table 1. The first function corresponds to a Bertrand-style market [63] where users are perfectly rational and always select the provider offering them the highest value. The second function corresponds to a market where users are boundedly rational [64] and select providers probabilistically based on their offered values.[6] Here, the parameter $\beta > 0$ controls the degree of rationality; as β increases, users become more rational and, in the limit $\beta \to \infty$, users become perfectly rational.

Given the choices of test-time compute levels made by all providers and the resulting market share they attain, each provider i obtains a utility $U_i(\theta_i; \boldsymbol{\theta}_{-i})$ given by

$$U_i(\theta_i; \boldsymbol{\theta}_{-i}) = \underbrace{s_i(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i}))}_{\text{Provider's market share}} \cdot \underbrace{(p_i(\theta_i) - c_i(\theta_i))}_{\text{Provider's profit}}. \tag{4}$$

In the next section, we will analyze the (social) welfare $\mathrm{W}(\boldsymbol{\theta})$ generated by an LLM-as-a-service market, which naturally corresponds to the sum of the total value obtained by users and the total utility obtained by all providers. More formally:

$$\begin{aligned} \mathrm{W}(\boldsymbol{\theta}) &\coloneqq \sum_{i=1}^{N} U_i(\boldsymbol{\theta}_i; \boldsymbol{\theta}_{-i}) + \sum_{i=1}^{N} V_i(\boldsymbol{\theta}_i) \cdot s_i(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) + V_0 \cdot s_0(V_0; \boldsymbol{V}(\boldsymbol{\theta})) \\ &= \sum_{i=1}^{N} s_i(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) \cdot \underbrace{(q_i(\theta_i) - c_i(\theta_i))}_{\coloneqq \mathrm{W}_i(\theta_i)} + V_0 \cdot s_0(V_0; \boldsymbol{V}(\boldsymbol{\theta})), \end{aligned} \tag{5}$$

where prices $p_i(\theta)$ cancel out, and $\mathrm{W}_i(\theta_i)$ denotes each provider's contribution to social welfare, reflecting the quality they offer relative to the generation cost incurred by their chosen level of test-time compute.

[5] All results extend straightforwardly to settings with ties without altering the conclusions.

[6] In the context of random utility models (RUMs), boundedly rational users are those who only have access to noisy estimations of the value offered by each provider [65].

3 Social Welfare of Test-Time Compute Games

In this section, we will show that, in the LLM-as-a-service market, we can expect the test-time compute levels selected by the market providers to be suboptimal in terms of social welfare. To this end, as is typical in game theory, we will analyze the (pure) Nash equilibria of the test-time compute game $\mathcal{G}$ introduced in Section 2.

A choice of test-time compute levels $\boldsymbol{\theta}^\dagger \in \Theta^N$ is a pure Nash equilibrium of $\mathcal{G}$ if no provider can improve their utility through a unilateral change in test-time compute, *i.e.*,

$$U_i(\theta_i^\dagger; \boldsymbol{\theta}_{-i}^\dagger) \geq U_i(\theta_i'; \boldsymbol{\theta}_{-i}^\dagger) \text{ for all } i \in [N], \theta_i' \in \Theta. \tag{6}$$

To determine the existence of pure Nash equilibria, we resort to the theory of potential games [66]. Specifically, we start by showing that every test-time compute game is a *generalized ordinal potential game*, that is, there exists a function $\Phi : \Theta^N \to \mathbb{R}$, called a potential, such that any unilateral deviation of some provider i that strictly increases their utility also strictly increases the potential. More specifically, let

$$\Phi(\boldsymbol{\theta}) = \begin{cases} \log\left(p_{\pi(1)}(\theta_{\pi(1)}) - c_{\pi(1)}(\theta_{\pi(1)})\right) + C \cdot V_{\pi(2)}(\theta_{\pi(2)}) & (\beta = \infty) \\ \sum_{i=1}^{N} \log\left(p_i(\theta_i) - c_i(\theta_i)\right) + \beta \cdot \sum_{i=1}^{N} V_i(\theta_i) - \log\left(e^{\beta V_0} + \sum_{i=1}^{N} e^{\beta V_i(\theta_i)}\right) & (\beta < \infty), \end{cases} \tag{7}$$

where $C > 0$ is a constant whose value is explicitly given in Appendix C.1, and π is the ordering of providers in decreasing order of the value they offer under $\boldsymbol{\theta}$. Then, we have the following theorem:

Theorem 1. *The function Φ defined in Eq. 7 is a potential for the test-time compute game $\mathcal{G}$,* i.e.*, for all $\boldsymbol{\theta} \in \Theta^N$, $i \in [N]$, and $\theta_i' \in \Theta$, it holds that $U_i(\theta_i'; \boldsymbol{\theta}_{-i}) > U_i(\theta_i; \boldsymbol{\theta}_{-i}) \Rightarrow \Phi(\theta_i'; \boldsymbol{\theta}_{-i}) > \Phi(\theta_i; \boldsymbol{\theta}_{-i})$.*

Intuitively, Theorem 1 reveals that, although each provider acts selfishly to increase their own utility, the structure of the market is such that their actions jointly optimize the potential $\Phi(\boldsymbol{\theta})$, which captures the overall state of the market. In this context, it is worth noting that competition between providers drives up both the value offered to the users and (some of) the providers' profits.

Following well-known results in game theory, the characterization of the test-time compute game $\mathcal{G}$ as an ordinal potential game readily implies that the game admits a (not necessarily unique) pure Nash equilibrium [66].To understand why this holds, consider that the levels of test-time compute selected by the providers start at some arbitrary point $\boldsymbol{\theta}^1 \in \Theta^N$ and follow *better-response dynamics* over time. Formally, this means that, at each time step t, the levels of test-time compute $\boldsymbol{\theta}^t$ are such that either (i) there exists a single provider $i \in [N]$ who changes their level of test-time compute to $\theta_i^{t+1} \neq \theta_i^t$ with $U_i(\theta_i^{t+1}; \boldsymbol{\theta}_{-i}^t) > U_i(\theta_i^t; \boldsymbol{\theta}_{-i}^t)$ or (ii) $\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t$ if no such change is possible by any provider. Then, since the domain Θ^N is finite, the sequence $\boldsymbol{\theta}^t$ eventually becomes constant, at which point no provider can further increase their utility and the market reaches a pure Nash equilibrium. In this context, note that better-response dynamics are particularly natural in practice. One can think of the better-response of a provider as the (minor) release of a model that has not undergone additional pre-training or post-training, but whose performance has changed due to optimizations in test-time compute methods.

Further, we proceed by identifying properties that hold in the equilibrium of a test-time compute game $\mathcal{G}$, starting from an explicit characterization of the levels of test-time compute selected by providers under perfect rationality. When users are perfectly rational, they always opt for the provider that offers them the highest value (see Table 1). As a consequence, each provider's incentive is to select a level of test-time compute that offers higher value than what their competitors are offering. This observation allows us to determine the equilibrium reached by the providers using the maximum value $V_i^* := \max_{\theta \in \Theta} \{q_i(\theta) - p_i(\theta)\}$ that each provider can offer, where, without loss of generality, we assume that providers are indexed such that $V_1^* \geq V_2^* \geq \cdots \geq V_N^*$ and we will refer to the provider who can offer the highest value V_1^* as the *dominant* provider. In particular, we have the following theorem:

Theorem 2. *In any pure Nash equilibrium $\boldsymbol{\theta}^\dagger$ of a test-time compute game $\mathcal{G}$ with perfectly rational users, (i) the dominant provider serves all queries with $\theta_1^\dagger = \max_{\theta_1 \in \Theta} \{\theta_1 \mid V_1(\theta_1) > V_2^*\}$, and (ii)*

there exists at least one provider $i \neq 1$ such that

$$V_i(\theta_i^\dagger) > \max_{\theta_1 \in \Theta} \left\{ V_1(\theta_1) \;\middle|\; V_1(\theta_1) < V_2^* \quad \text{and} \quad p_1(\theta_1) - c_1(\theta_1) > p_1(\theta_1^\dagger) - c_1(\theta_1^\dagger) \right\},$$

where the right-hand side of the inequality is $-\infty$ if the set is empty.

The above result states that, in equilibrium, the dominant provider captures the market by selecting a level of test-time compute that offers higher value to the users than what their competitors can possibly offer and, once they achieve that, they increase their test-time compute as much as possible to maximize their utility (see also Eq. 2). At the same time, at least one competitor offers a sufficiently high-value alternative while gaining zero utility—otherwise, the dominant provider could also lower the value they offer, and the situation would not be an equilibrium.

For games $\mathcal{G}$ with boundedly rational users ($\beta < \infty$), the levels of test-time compute in equilibrium cannot be characterized explicitly, since all providers serve a positive fraction of user queries and their chosen levels of test-time compute depend on the costs, prices, and qualities $\{c_i, p_i, q_i\}_{i=1}^N$ of all providers. Nonetheless, we next show that, as long as users are sufficiently rational, any Nash equilibrium of the game $\mathcal{G}$ with $\beta < \infty$ is also an equilibrium of the game $\mathcal{G}$ with perfectly rational users and, hence, also satisfies the properties of Theorem 2. Formally, we have the following theorem:

Theorem 3. *Consider a set of providers specified by $\{c_i, p_i, q_i\}_{i=1}^N$. Then, there exists a (finite) level of user rationality $\beta_0 > 0$ such that, for any $\beta > \beta_0$, any pure Nash equilibrium $\boldsymbol{\theta}^\dagger$ of the game $\mathcal{G}$ with boundedly rational users is also a Nash equilibrium of the game $\mathcal{G}$ with perfectly rational users.*

Taken together, the above results provide some intuition regarding the provider utilities and user value in equilibrium. However, they do not elucidate to what extent we can expect the actions of providers in equilibrium to be optimal in terms of social welfare. To shed light on this question, in what follows, we will analyze the *price of anarchy* (PoA), a standard game-theoretic measure [54, 38] that compares the social welfare achieved by the test-time compute levels $\boldsymbol{\theta}^\dagger$ selected by the providers in equilibrium against the test-time compute levels that maximize the social welfare of the market. More formally, the price of anarchy of a test-time compute game $\mathcal{G}$ is given by

$$\text{PoA}\left(\mathcal{G}\right) \coloneqq \frac{\max_{\theta \in \Theta^N} \mathrm{W}(\boldsymbol{\theta})}{\mathrm{W}(\boldsymbol{\theta}^\dagger)}. \tag{8}$$

By definition, the price of anarchy is always at least 1, and it equals 1 only when the compute levels $\boldsymbol{\theta}^\dagger$ selected by providers at equilibrium also optimize the social welfare. However, this is unlikely to happen, because rational providers act to maximize their individual utilities rather than coordinating to maximize the social welfare of the market. To better understand what conditions can lead to $\text{PoA}(\mathcal{G}) > 1$, and since explicitly characterizing the test-time compute level $\operatorname{argmax}_{\theta \in \Theta^N} \mathrm{W}(\boldsymbol{\theta})$ in an arbitrary game $\mathcal{G}$ is generally intractable, we compare the test-time compute $\boldsymbol{\theta}^\dagger$ selected by the providers at equilibrium against the test-time compute $\boldsymbol{\theta}^*$ selected by the providers in an idealized situation in which they maximize their individual contribution $\mathrm{W}_i(\theta_i^*)$ to the welfare. By doing so, we obtain the following lower-bound on the price of anarchy:

Theorem 4. *The price of anarchy of the test-time compute game $\mathcal{G}$ is lower-bounded, up to higher order terms, as*

$$\text{PoA}\left(\mathcal{G}\right) \geq 1 + \frac{\mathrm{W}_{\pi^*(1)}(\theta^*_{\pi^*(1)}) - \mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)})}{\mathrm{W}_{\pi^*(1)}(\theta^*_{\pi^*(1)})} + f(\beta),$$

where $f(\beta) = \mathcal{O}\left(e^{-\beta \Delta V}\right)$, $\Delta V > 0$ is a constant depending on the game instance, and π, π^* are permutations of providers in decreasing order of the value they offer at $\boldsymbol{\theta}^\dagger$ and $\boldsymbol{\theta}^*$, respectively. Refer to the proof in Appendix C.4 for the exact expression of f, ΔV, and the higher order corrections to the above lower bound.

The first term in the lower-bound of the price of anarchy given by Theorem 4 quantifies the difference between the contributions to the social welfare by the providers who offer the highest value in an idealized situation in which they select a test-time compute level that maximizes their individual contributions to social welfare (*i.e.*, $\boldsymbol{\theta}^*$) and in equilibrium (*i.e.*, $\boldsymbol{\theta}^\dagger$), while the term $f(\beta)$ quantifies the effect of users' bounded rationality, which vanishes exponentially fast as users become fully rational ($\beta \to \infty$).

Although Theorem 4 shows that competition between providers can lead to levels of test-time compute that are inefficient in terms of social welfare, it is worth highlighting that, even in the absence of such inefficiencies, there is still room for the social welfare to improve further. Specifically, as can be seen in Eq.5, the social welfare depends not only on the providers' compute choices, but also on the market-share allocation (see Table1). In an ideal scenario, each provider would choose θ_i^* maximizing their contribution $W_i(\theta_i^*)$, and users would select the provider with the highest such contribution. Yet, even under perfect rationality, the provider with the highest $W_i(\theta_i^*)$ may not deliver the highest value to users. In the next section, we introduce a forward-looking market that aims to realize this ideal.

4 An Auction Mechanism for Test-Time Compute

To design a market in which social welfare is maximized, we draw inspiration from mechanism design [67, 34] and propose a reverse second-price auction that, by design, incentivizes each provider to choose the level of test-time compute θ_i^* that maximizes their individual contribution $W_i(\theta_i^*)$ to social welfare. Within this auction, the user acts as a seller providing a task, and providers act as bidders declaring the quality and price they offer, while a third-party platform allocates the task to the provider with the best quality–price trade-off and sets payments based on all stated qualities and prices.

Formally, we denote the induced game as $\widetilde{\mathcal{G}}$ and, for simplicity, overload notation from Sections 2 and 3, noting that definitions may differ. The process begins with a user submitting a (small) set of queries $\mathcal{Q}$ representative of a task $\mathcal{T}$ they wish to solve to the third-party platform conducting the auction. All N providers then bid simultaneously for the opportunity to serve the task $\mathcal{T}$. Specifically, each provider i selects a level of test-time compute θ_i and uses the set $\mathcal{Q}$ of queries to estimate the average quality $q_i(\theta_i) \in \mathbb{R}_+$ of their model on the task $\mathcal{T}$.[7] Then, each provider selects a price $p_i \in \mathbb{R}_+$ at which they are willing to serve the task, and submits a bid $(q_i(\theta_i), p_i)$ to the platform, as is common in scoring auctions [60, 37].[8] Finally, the platform assigns the task $\mathcal{T}$ to the provider who offers the highest value $V_i(\theta_i, p_i) \coloneqq q_i(\theta_i) - p_i$ where, similarly to Section 3, we assume there are no ties between providers.

To align the incentives of the providers with social welfare, the third-party platform sets the payment made by the user to the winning provider using a second-price payment rule—the user pays an amount that does not depend on the price $p_{\pi(1)}$ bid by the winning provider, but instead equals the difference between the quality delivered by the winning provider and the value offered by the second-best provider. Formally, the payment is given by

$$P(\boldsymbol{\theta}, \mathbf{p}) = q_{\pi(1)}(\theta_{\pi(1)}) - V_{\pi(2)}(\theta_{\pi(2)}, p_{\pi(2)}), \tag{9}$$

where π is the ordering of providers in decreasing order of the value they offer, and $\boldsymbol{\theta}$ and $\mathbf{p}$ denote the vectors of test-time compute levels and prices selected by all providers, respectively. Based on the payment in Eq. 9, the utility obtained by each provider is then given by

$$U_i(\theta_i, p_i; \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}) = (P(\boldsymbol{\theta}, \mathbf{p}) - c_i(\theta_i)) \cdot \mathbb{1}\left\{ V_i(\theta_i, p_i) > \max_{j \neq i} V_j(\theta_j, p_j) \right\}, \tag{10}$$

where $c_i(\theta) \in \mathbb{R}_+$ is the generation cost, and the utility of all providers except for the winning one equals zero.

Similar to the competitive market setting in Section 2, we define social welfare as the sum of the total utility obtained by all providers and the net value obtained by the user, which is the difference between the quality offered by the winning provider and the payment set by the third-party platform. More concretely, since the only provider with non-zero utility is the winning one, social welfare is given by

$$\begin{aligned} \mathrm{W}(\boldsymbol{\theta}, \mathbf{p}) &= \underbrace{U_{\pi(1)}(\theta_{\pi(1)}, p_{\pi(1)}; \boldsymbol{\theta}_{-\pi(1)}, \mathbf{p}_{-\pi(1)})}_{\text{Winning provider's utility}} + \underbrace{q_{\pi(1)}(\theta_{\pi(1)}) - P(\boldsymbol{\theta}, \mathbf{p})}_{\text{Net user value}} \\ &= q_{\pi(1)}(\theta_{\pi(1)}) - c_{\pi(1)}(\theta_{\pi(1)}) = W_{\pi(1)}\left(\theta_{\pi(1)}\right), \end{aligned}$$

[7] In many domains, such as coding or information retrieval, estimating the quality $q_i(\theta_i)$ can be done automatically with limited overhead. We further discuss this in Section 6.

[8] Given a selected test-time compute level θ_i, providers have no incentive to misreport their model's quality $q_i(\theta_i)$, since the platform can verify the quality of their responses, as we formally show in Appendix B.

where the second equality follows from Eqs. 9 and 10 and highlights that, due to the structure of the payment rule, the social welfare depends entirely on the quality, generation cost, and level of test-time compute of the winning provider $\pi(1)$.

Next, we proceed to analyze the equilibrium properties of the game $\widetilde{G}$ induced by the auction. Our starting point is the observation that, under the payment rule given by Eq. 9, each provider i has a dominant strategy, that is, a combination of test-time compute $\theta_i^\dagger$ and price $p_i^\dagger$ that maximizes their utility regardless of how the other providers act. Formally, a dominant strategy for provider i satisfies for all $(\theta_i, \boldsymbol{\theta}_{-i}) \in \Theta^N$ and $(p_i, \mathbf{p}_{-i}) \in \mathbb{R}_+^N$:

$$U_i(\theta_i^\dagger, p_i^\dagger; \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}) \geq U_i(\theta_i, p_i; \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}), \tag{11}$$

and, consequently, the choice of test-time compute levels $\boldsymbol{\theta}^\dagger$ and prices $\mathbf{p}^\dagger$ is a pure Nash equilibrium of $\widetilde{\mathcal{G}}$. The following theorem shows that such strategies (and hence, equilibria) always exist and characterizes their associated level of test-time compute and price:

Theorem 5. *In the game $\widetilde{\mathcal{G}}$, the choice of test-time compute level $\theta_i^\dagger = \theta_i^* = \arg\max_{\theta \in \Theta} W_i(\theta)$ and price $p_i^\dagger = c_i\left(\theta_i^*\right)$ is a dominant strategy for provider i.*

The above result follows from the fact that, conditioned on winning the auction, the payment defined in Eq. 9 does not depend on the price $p_{\pi(1)}$. Thus, a provider can simultaneously increase their utility and the value they offer to the user by selecting the compute level θ_i^* to maximize $V_i(\theta_i, c_i(\theta_i)) = \mathrm{W}_i(\theta_i)$, and bidding the corresponding quality $q_i(\theta_i^*)$ and the lowest possible price—their generation cost c_i.

As an immediate consequence, in the Nash equilibrium $(\boldsymbol{\theta}^\dagger, \mathbf{p}^\dagger)$ of the game $\widetilde{\mathcal{G}}$, the value $V_i(\theta_i, p_i)$ offered by each provider i is equal to the maximum individual contribution to social welfare $\mathrm{W}_i(\theta_i^*)$ they can make. Since the platform selects the provider who offers the highest value to serve the task $\mathcal{T}$, in equilibrium, the winning provider is the one who can make the highest individual contribution to social welfare, *i.e.*, $\pi_{(1)} = \arg\max_{i \in [N]} \mathrm{W}_i(\theta_i^*)$. Then, the price of anarchy of the game $\widetilde{\mathcal{G}}$ satisfies

$$\mathrm{PoA}\left(\widetilde{\mathcal{G}}\right) \coloneqq \frac{\max_{\boldsymbol{\theta}, \mathbf{p}} \mathrm{W}(\boldsymbol{\theta}, \mathbf{p})}{\mathrm{W}(\boldsymbol{\theta}^\dagger, \mathbf{p}^\dagger)} \overset{(*)}{=} \frac{\max_i \max_{\theta_i \in \Theta} \mathrm{W}_i(\theta_i)}{\mathrm{W}_{\pi(1)}(\theta^*_{\pi(1)})} = 1, \tag{12}$$

where $(*)$ holds because it is possible to set the prices $\mathbf{p}$ such that any provider wins the auction. Together, the above results show that, at equilibrium, the game $\widetilde{\mathcal{G}}$ achieves the maximum possible value of social welfare.

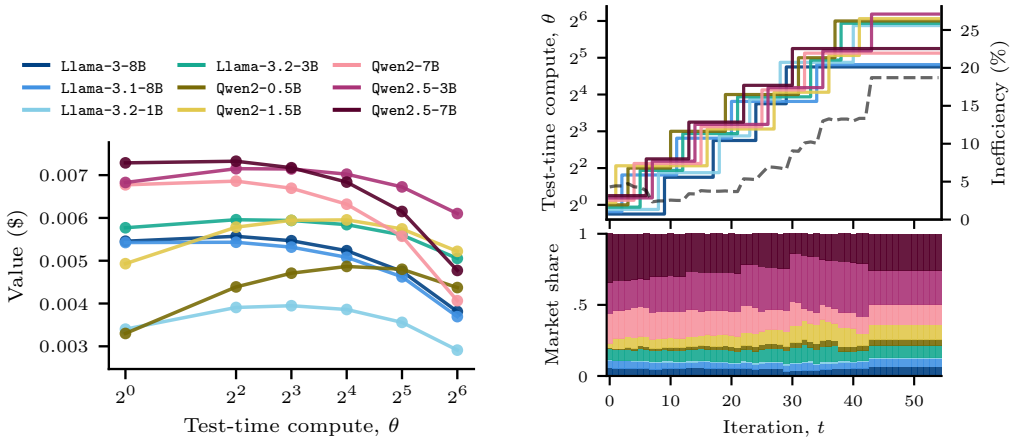
In this context, it is worth noting that, due to the choice of payment rule in Eq. 9, the game $\widetilde{\mathcal{G}}$ enjoys several additional desirable properties. First, one can easily verify that the winning provider receives a payment strictly higher than the price they bid (*i.e.*, $P(\boldsymbol{\theta}, \mathbf{p}) > p_{\pi(1)}$), and therefore always obtains positive utility. Second, the value received by the user matches the value offered by the second-best provider, *i.e.*, $q_{\pi(1)}(\theta_{\pi(1)}) - P(\boldsymbol{\theta}, \mathbf{p}) = V_{\pi(2)}(\theta_{\pi(2)}, p_{\pi(2)})$. We further explore these properties, as well as their implications for users and providers, in our experiments in Section 5.

5 Experiments

In this section, we analyze the outcomes of test-time compute games with two goals: to validate the theoretical results of Section 3 on game dynamics and equilibria, and to empirically show that current test-time compute markets generally do not maximize social welfare. We first outline our experimental setup and refer to Appendix D for details.

Experimental setup. For concreteness, here we focus on a particular game instance. Refer to Appendix E.2 for similar results for games with different levels of user rationality, and across other test-time compute methods (majority voting and chain-of-thought) and tasks.

We consider a test-time compute game, as defined in Section 2, in which each provider serves a different LLM, which we use to identify providers. More concretely, we study a game $\mathcal{G}$ with $N = 9$ providers, each serving a non-reasoning model selected from one of two families: (i) four models from the `Llama` family, `Llama-{3-8B,3.1-8B,3.2-1B,3.2-3B}-Instruct`, and (ii) five models from the `Qwen` family, `Qwen-{2-0.5B,2-1.5B,2-7B,2.5-3B,2.5-7B}-Instruct`.



(a) Value offered by all providers

(b) Dynamics

Figure 1: **Outcome of a test-time compute game.** The figure shows the outcome of a test-time compute game with $N = 9$ providers using best-of-n to serve queries from the GSM8K dataset. Panel (a) shows the value $V_i(\theta)$ that each provider offers to users as a function of their test-time compute. Panel (b) shows the better-response dynamics of the game; the upper and lower sub-panels correspond, respectively, to the test-time compute $\boldsymbol{\theta}^t$ of each provider (with a small vertical jitter for visibility) together with the inefficiency of the game, $\max_{\boldsymbol{\theta}} \mathrm{W}(\boldsymbol{\theta})/\mathrm{W}(\boldsymbol{\theta}^t) - 1$ (dashed black curve), and to the market share of each provider. We set $\beta = 1000$.

In the game $\mathcal{G}$, the task $\mathcal{T}$ that providers compete to serve consists of queries drawn from `GSM8K` [68], which contains mathematical questions with verifiable ground-truth answers. To serve this task, all providers use the same test-time compute method, namely best-of-n [3], in which providers can generate θ independent model outputs for a given query, and as the final response, they select the highest-scored one according to a reward model.

Lastly, we need to specify how the providers' compute choices determine both the value $V_i(\theta)$ they offer to users, and their own utilities $U_i(\boldsymbol{\theta})$. To this end, as we report in Appendix E.1, we first measure the average accuracy and the average number of generated tokens of all models for all test-time compute levels. Based on these measurements, we compute, for each provider and compute level θ, the prices $p_i(\theta)$ by multiplying the average number of generated tokens per query by the average per-token price. These token prices are obtained from the Hugging Face list of providers (see Appendix D). Then, we infer the provider costs $c_i(\theta)$ by assuming a fixed profit margin per token of 25% for all providers. Finally, we linearly map model accuracies into user values by considering that each percentage point of accuracy (*i.e.*, an accuracy of 0.01) is worth $\$0.008$, and set $\beta = 1000$.

Results. Figure 1 shows the outcome of the game $\mathcal{G}$ described above. In Panel (a), we show that the values $V_i(\theta)$ offered by providers are approximately concave with respect to their compute: users initially benefit from a slight increase in compute; however, as shown in Figure 3 in Appendix E.1.1, the number of tokens—and thus the price paid by the user—increases rapidly with compute, eventually diminishing their value.

To visualize how these differences in offered value translate into specific market outcomes, we show in Panel (b) of Figure 1 the dynamics of a test-time compute game where providers sequentially adjust their compute levels to increase their utility. In accordance with Theorem 1, since $\mathcal{G}$ is a potential game, the better-response dynamics converge to a pure Nash equilibrium. Importantly, we find that, once providers have reached an equilibrium, the market is socially inefficient: the price of anarchy is 1.19, representing a 16% loss in social welfare. Altogether, the above results suggest that competition in test-time compute markets does not naturally lead to the socially optimal outcome.

Lastly, in Table 3 (Appendix E.3), we show that compared to the test-time compute game $\mathcal{G}$, the auction mechanism $\widetilde{\mathcal{G}}$ increases social welfare by 25% and user value by 30%, while reducing the provider's utility by 25%. For other game instances, however, both users and providers can benefit simultaneously from the auction mechanism (see Table 4), while still guaranteeing that the mechanism is socially efficient (Eq. 12).

6 Discussion and Limitations

In this section, we highlight several limitations of our work and discuss avenues for future research.

Model. Our definition of test-time compute games considers only factors related to the quality of LLM outputs and financial aspects, such as the pricing of LLMs by providers and their generation costs. In practice, however, other factors (*e.g.*, a provider's popularity or the reliability of their API) may influence users' choices, leading to more complex behavior than captured by the canonical market share functions in Table 1. Consequently, it would be interesting to extend our definition of test-time compute games to account for these additional factors and develop auction mechanisms for such (complex) games. Moreover, in our work, we assume that all users derive the same (average) value from a given LLM and test-time compute level; however, in practice, different users may value output quality differently, even for the same query and output. Extending test-time compute games to heterogeneous users is an interesting direction for future work.

Evaluation. We have studied the equilibria of test-time compute games on simulated markets using several state-of-the-art LLMs, including both reasoning and non-reasoning models, and test-time compute methods. In each game, a set of simulated providers competes to serve queries drawn from a benchmarking dataset, either with verifiable ground truth or with reference correct and incorrect answers. However, it would be interesting to study the equilibria of test-time compute games in settings where providers compete to serve open-ended queries and quality is measured automatically without relying on predefined correct outputs. Further, as discussed in Appendix D, our evaluation of reasoning models does not explicitly vary their reasoning effort, since the models we consider do not support this; instead, we have used the number of reasoning tokens produced as a proxy for the test-time compute (reasoning) level. Lastly, it would be important to analyze the dynamics of the LLM-as-a-service market using real-world data, and deploy and evaluate our proposed auction mechanism in a real-world deployment.

Practical considerations. Our second-price auction requires that providers accurately estimate the (average) quality that the model they serve can deliver for a given task. In many domains where LLMs are currently used, such as coding or information retrieval, automated evaluation procedures make this feasible with limited overhead. At the same time, the platform implementing the auction can leverage these procedures to verify and evaluate the quality of a provider's responses. As we show in Appendix B, our auction can be naturally extended with such evaluations to ensure that providers maximize their utility by truthfully reporting their quality estimates. Looking ahead, the feasibility of an auction in settings with less well-established quality metrics and benchmarks will depend on continued progress in the LLM evaluation literature. Lastly, we note that, while second-price auctions possess many desirable properties, they also make a provider's payment dependent on the bids of other providers, leaving the mechanism potentially vulnerable to manipulation or collusion [69, 70, 71].

Adoption incentives. Our empirical results in Appendix E.3 suggest that, while the auction mechanism introduced in Section 4 is guaranteed to increase social welfare, in some settings this may come at the expense of provider utility. However, the same results also show that users typically obtain higher value when participating in the auction, especially compared to settings in which they have limited information about the value offered by providers (*i.e.*, when the parameter β is low). As a result, users may be more likely to participate in a platform implementing the auction, which could, in turn, incentivize provider participation, as it offers access to a larger share of the user base.

7 Conclusions

In this work, we introduced *test-time compute games*, a game-theoretic model in which LLM providers in a market strategically select the level of test-time compute used by their model to compete for user queries and maximize their profit. Based on this model, we have demonstrated that current pricing approaches in markets of LLMs-as-a-service incentivize providers to set their level of test-time compute in a way that is not aligned with social welfare. To address this, we proposed a forward-looking market based on a reverse second-price auction that provably aligns provider incentives with social welfare. We hope that our work inspires users, LLM providers, and online platforms to explore alternative market structures for generative AI that place social welfare at the center of their design choices.

Acknowledgements

Gomez-Rodriguez acknowledges support from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 945719 and 101169607). Tsirtsis acknowledges supports from the Alexander von Humboldt Foundation in the framework of the Alexander von Humboldt Professorship (Humboldt Professor of Technology and Regulation awarded to Sandra Wachter) endowed by the Federal Ministry of Education and Research via the Hasso Plattner Institute.

References

[1] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[2] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.

[3] Yinlam Chow, Guy Tennenholtz, Izzeddin Gur, Vincent Zhuang, Bo Dai, Aviral Kumar, Rishabh Agarwal, Sridhar Thiagarajan, Craig Boutilier, and Aleksandra Faust. Inference-aware fine-tuning for best-of-n sampling in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=77gQUdQhE7.

[4] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, USA, 2007. ISBN 0521872820.

[5] Jon Kleinberg and Manish Raghavan. Algorithmic monoculture and social welfare. *Proceedings of the National Academy of Sciences*, 118(22):e2018340118, 2021. doi: 10.1073/pnas.2018340118. URL https://www.pnas.org/doi/abs/10.1073/pnas.2018340118.

[6] Stratis Tsirtsis, Behzad Tabibian, Moein Khajehnejad, Adish Singla, Bernhard Schölkopf, and Manuel Gomez-Rodriguez. Optimal decision making under strategic behavior. *Management Science*, 70(12):8506–8519, 2024. doi: 10.1287/mnsc.2021.02567. URL https://doi.org/10.1287/mnsc.2021.02567.

[7] Ohad Einav and Nir Rosenfeld. A market for accuracy: Classification under competition. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=RPPBhhRddB.

[8] Eden Saig and Nir Rosenfeld. Evolutionary prediction games. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=qsYbytjmQK.

[9] Nir Rosenfeld and Haifeng Xu. Machine learning should maximize welfare, but not by (only) maximizing accuracy, 2025. URL https://arxiv.org/abs/2502.11981.

[10] Etienne Gauthier, Francis Bach, and Michael I Jordan. Statistical collusion by collectives on learning platforms. In *Forty-second International Conference on Machine Learning*, 2025.

[11] Paul Duetting, Vahab Mirrokni, Renato Paes Leme, Haifeng Xu, and Song Zuo. Mechanism design for large language models. In *Proceedings of the ACM Web Conference 2024*, pages 144–155, 2024.

[12] Emanuele La Malfa, Aleksandar Petrov, Simon Frieder, Christoph Weinhuber, Ryan Burnell, Raza Nazar, Anthony Cohn, Nigel Shadbolt, and Michael Wooldridge. Language-models-as-a-service: Overview of a new paradigm and its challenges. *Journal of Artificial Intelligence Research*, 80:1497–1523, 2024.

[13] Rafid Mahmood. Pricing and competition for generative AI. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=8LbJfEjIrT.

[14] Benjamin Laufer, Jon Kleinberg, and Hoda Heidari. Fine-tuning games: Bargaining and adaptation for general-purpose models. In *Proceedings of the ACM Web Conference 2024*, pages 66–76, 2024.

[15] Will Cai, Tianneng Shi, Xuandong Zhao, and Dawn Song. Are you getting what you pay for? auditing model substitution in LLM APIs. In *NeurIPS 2025 Workshop on Regulatable ML*, 2025. URL https://openreview.net/forum?id=thhrtv9P0s.

[16] Eden Saig, Ohad Einav, and Inbal Talgam-Cohen. Incentivizing quality text generation via statistical contracts. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=wZgw4CrxwK.

[17] Dirk Bergemann, Alessandro Bonatti, and Alex Smolin. The economics of large language models: Token allocation, fine-tuning, and optimal pricing. EC '25, page 786, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400719431. doi: 10.1145/3736252.3742625. URL https://doi.org/10.1145/3736252.3742625.

[18] Ander Artola Velasco, Stratis Tsirtsis, Nastaran Okati, and Manuel Gomez Rodriguez. Is your LLM overcharging you? tokenization, transparency, and incentives. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=aGaBFE9pta.

[19] Ander Artola Velasco, Stratis Tsirtsis, and Manuel Gomez-Rodriguez. Auditing pay-per-token in large language models. In *The 29th International Conference on Artificial Intelligence and Statistics*, 2026.

[20] Krishnamurthy Iyer, Alec Sun, Haifeng Xu, and You Zu. How to sell a service with uncertain outcomes. In *Proceedings of the 26th ACM Conference on Economics and Computation*, EC '25, page 837, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400719431. doi: 10.1145/3736252.3742633. URL https://doi.org/10.1145/3736252.3742633.

[21] Yuhan Cao, Yu Wang, Sitong Liu, Miao Li, Yixin Tao, and Tianxing He. Pay for the second-best service: A game-theoretic approach against dishonest llm providers. In *Proceedings of the ACM Web Conference 2026*, WWW '26, page 249–260, New York, NY, USA, 2026. Association for Computing Machinery. ISBN 9798400723070. doi: 10.1145/3774904.3792446. URL https://doi.org/10.1145/3774904.3792446.

[22] Guoheng Sun, Ziyao Wang, Bowei Tian, Meng Liu, Zheyu Shen, Shwai He, Yexiao He, Wanghao Ye, Yiting Wang, and Ang Li. Coin: Counting the invisible reasoning tokens in commercial opaque llm apis, 2025. URL https://arxiv.org/abs/2505.13778.

[23] Hao Wen, Yifan Su, Feifei Zhang, Yunxin Liu, Yunhao Liu, Ya-Qin Zhang, and Yuanchun Li. Parathinker: Native parallel thinking as a new paradigm to scale llm test-time compute, 2025. URL https://arxiv.org/abs/2509.04475.

[24] Yuyang Wu, Yifei Wang, Ziyu Ye, Tianqi Du, Stefanie Jegelka, and Yisen Wang. When more is less: Understanding chain-of-thought length in LLMs. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=6QDFsYxtI1.

[25] Zhenni Bi, Kai Han, Chuanjian Liu, Yehui Tang, and Yunhe Wang. Forest-of-thought: Scaling test-time compute for enhancing LLM reasoning. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 4253–4267. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/bi25a.html.

[26] Runze Liu, Junqi Gao, Jian Zhao, Kaiyan Zhang, Xiu Li, Biqing Qi, Wanli Ouyang, and Bowen Zhou. Can 1b LLM surpass 405b LLM? rethinking compute-optimal test-time scaling. In *Workshop on Reasoning and Planning for Large Language Models*, 2025. URL https://openreview.net/forum?id=CvjX9Lhpze.

[27] Charlie Victor Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=4FWAwZtd2n.

[28] Junlin Wang, Siddhartha Jain, Dejiao Zhang, Baishakhi Ray, Varun Kumar, and Ben Athiwaratkun. Reasoning in token economies: Budget-aware evaluation of LLM reasoning strategies. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 19916–19939, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1112. URL https://aclanthology.org/2024.emnlp-main.1112/.

[29] Michael J. Zellinger and Matt Thomson. Economic evaluation of llms, 2025. URL https://arxiv.org/abs/2507.03834.

[30] Mehmet Hamza Erol, Batu El, Mirac Suzgun, Mert Yuksekgonul, and James Zou. Cost-of-pass: An economic framework for evaluating language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=vC9S20zsgN.

[31] Guangya Wan, Zixin Stephen Xu, Sasa Zorc, Manel Baucells, Mengxuan Hu, Hao Wang, and Sheng Li. Beacon: Bayesian optimal stopping for efficient llm sampling. *arXiv preprint arXiv:2510.15945*, 2025.

[32] Junpei Komiyama, Daisuke Oba, and Masafumi Oyamada. Best-of-infinity: Asymptotic performance of test-time LLM ensembling. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=3qiCnLf3jf.

[33] Yusuf Kalayci, Vinod Raman, and Shaddin Dughmi. Optimal stopping vs best-of-n for inference time optimization. *arXiv preprint arXiv:2510.01394*, 2025.

[34] V. Krishna. *Auction Theory*. Academic Press, 2009. ISBN 9780080922935. URL https://books.google.de/books?id=qW1128ktG1gC.

[35] P.R. Milgrom. *Putting Auction Theory to Work*. Churchill Lectures in Economics. Cambridge University Press, 2004. ISBN 9780521536721. URL https://books.google.de/books?id=AkeHTU7XW4kC.

[36] Paul R. Milgrom and Robert J. Weber. A theory of auctions and competitive bidding. *Econometrica*, 50(5): 1089–1122, 1982. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/1911865.

[37] Yeon-Koo Che. Design competition through multidimensional auctions. *The RAND Journal of Economics*, 24(4):668–680, 1993. ISSN 07416261. URL http://www.jstor.org/stable/2555752.

[38] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V Vazirani, editors. *Algorithmic game theory*. Cambridge University Press, Cambridge, England, September 2007.

[39] Gregory E Kersten. Multiattribute procurement auctions: Efficiency and social welfare in theory and practice. *Decision Analysis*, 11(4):215–232, 2014.

[40] Motty Perry and Philip J Reny. An efficient auction. *Econometrica*, 70(3):1199–1212, 2002.

[41] Seiji Takanashi, Takehiro Kawasaki, Taiki Todo, and Makoto Yokoo. Efficiency in truthful auctions via a social network. *arXiv preprint arXiv:1904.12422*, 2019.

[42] Roger B Myerson. Optimal auction design. *Mathematics of operations research*, 6(1):58–73, 1981.

[43] John G Riley and William F Samuelson. Optimal auctions. *The American Economic Review*, 71(3): 381–392, 1981.

[44] Eric Maskin and John Riley. Optimal auctions with risk averse buyers. *Econometrica: Journal of the Econometric Society*, pages 1473–1518, 1984.

[45] Kshipra Bhawalkar and Tim Roughgarden. Welfare guarantees for combinatorial auctions with item bidding. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, pages 700–709. SIAM, 2011.

[46] Vasilis Syrgkanis and Eva Tardos. Composable and efficient mechanisms. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 211–220, 2013.

[47] Dirk Bergemann, Marek Bojko, Paul Duetting, Renato Paes Leme, Haifeng Xu, and Song Zuo. Data-driven mechanism design: Jointly eliciting preferences and information. In *Proceedings of the 26th ACM Conference on Economics and Computation*, EC '25, page 507, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400719431. doi: 10.1145/3736252.3742578. URL https://doi.org/10.1145/3736252.3742578.

[48] Ziv Bar-Yossef, Kirsten Hildrum, and Felix Wu. Incentive-compatible online auctions for digital goods. In *SODA*, volume 2, pages 964–970, 2002.

[49] John O Ledyard. Incentive compatibility. In *The New Palgrave dictionary of economics*, pages 1–9. Springer, 1987.

[50] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, 16 (1):8–37, 1961.

[51] Edward H Clarke. Multipart pricing of public goods. *Public choice*, pages 17–33, 1971.

[52] Theodore Groves. Incentives in teams. *Econometrica: Journal of the Econometric Society*, pages 617–631, 1973.

[53] Mohammad Akbarpour and Shengwu Li. Credible auctions: A trilemma. *Econometrica*, 88(2):425–467, 2020.

[54] Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. In *Annual symposium on theoretical aspects of computer science*, pages 404–413. Springer, 1999.

[55] Tim Roughgarden and Éva Tardos. How bad is selfish routing? *Journal of the ACM (JACM)*, 49(2): 236–259, 2002.

[56] Tim Roughgarden. Intrinsic robustness of the price of anarchy. *Journal of the ACM (JACM)*, 62(5):1–42, 2015.

[57] Jason Hartline, Darrell Hoy, and Samuel Taggart. Robust analysis of auction equilibria. *arXiv preprint arXiv:2310.03702*, 2023.

[58] Tim Roughgarden, Vasilis Syrgkanis, and Eva Tardos. The price of anarchy in auctions. *Journal of Artificial Intelligence Research*, 59:59–101, 2017.

[59] Yuhang Cai and Huchang Liao. An overview of multi-attribute auctions: bibliometrics, methodologies, applications and future directions. *Expert Systems with Applications*, 299:130083, 2026. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2025.130083. URL https://www.sciencedirect.com/science/article/pii/S0957417425036991.

[60] John Asker and Estelle Cantillon. Properties of scoring auctions. *The RAND Journal of Economics*, 39(1): 69–85, 2008. ISSN 07416261. URL http://www.jstor.org/stable/25046364.

[61] Yu Awaya, Naoki Fujiwara, and Marton Szabo. Quality and price in scoring auctions. *Journal of Mathematical Economics*, 116:103083, 2025.

[62] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=1PL1NIMMrw.

[63] Jean Tirole. *The theory of industrial organization*. The MIT Press. MIT Press, London, England, January 1988.

[64] Richard D. McKelvey and Thomas R. Palfrey. Quantal response equilibria for normal form games. *Games and Economic Behavior*, 10(1):6–38, 1995. ISSN 0899-8256. doi: https://doi.org/10.1006/game.1995.1023. URL https://www.sciencedirect.com/science/article/pii/S0899825685710238.

[65] Fan Yao, Chuanhao Li, Karthik Abinav Sankararaman, Yiming Liao, Yan Zhu, Qifan Wang, Hongning Wang, and Haifeng Xu. Rethinking incentives in recommender systems: are monotone rewards always beneficial? *Advances in Neural Information Processing Systems*, 36:74582–74601, 2023.

[66] Dov Monderer and Lloyd S. Shapley. Potential games. *Games and Economic Behavior*, 14(1):124–143, 1996. ISSN 0899-8256. doi: https://doi.org/10.1006/game.1996.0044. URL https://www.sciencedirect.com/science/article/pii/S0899825696900445.

[67] Noam Nisan and Amir Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35 (1):166–196, 2001. ISSN 0899-8256. doi: https://doi.org/10.1006/game.1999.0790. URL https://www.sciencedirect.com/science/article/pii/S089982569990790X.

[68] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[69] Kenneth Hendricks and Robert H. Porter. Collusion in auctions. *Annales d'Économie et de Statistique*, (15/16):217–230, 1989. ISSN 0769489X, 22726497. URL http://www.jstor.org/stable/20075758.

[70] George J Mailath and Peter Zemsky. Collusion in second price auctions with heterogeneous bidders. *Games and Economic Behavior*, 3(4):467–486, 1991. ISSN 0899-8256. doi: https://doi.org/10.1016/0899-8256(91)90016-8. URL https://www.sciencedirect.com/science/article/pii/0899825691900168.

[71] Yeon-Koo Che, Daniele Condorelli, and Jinwoo Kim. Weak cartels and collusion-proof auctions. *Journal of Economic Theory*, 178:398–435, 2018. ISSN 0022-0531. doi: https://doi.org/10.1016/j.jet.2018.09.005. URL https://www.sciencedirect.com/science/article/pii/S0022053118305842.

[72] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

[73] Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2025, 2025.

[74] Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL https://aclanthology.org/2022.acl-long.229/.

[75] Thibault Sellam, Dipanjan Das, and Ankur Parikh. BLEURT: Learning robust metrics for text generation. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.704. URL https://aclanthology.org/2020.acl-main.704/.

[76] Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: boosting language models with scaled ai feedback. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

[77] B O'Neill. Nonmetric test of the minimax theory of two-person zerosum games. *Proc. Natl. Acad. Sci. U. S. A.*, 84(7):2106–2109, April 1987.

[78] Bernhardt Lieberman. Human behavior in a strictly determined 3×3 matrix game. *Syst. Res.*, 5(4): 317–322, January 2007.

[79] James Pita, Manish Jain, Milind Tambe, Fernando Ordóñez, and Sarit Kraus. Robust solutions to stackelberg games: Addressing bounded rationality and limited observations in human cognition. *Artificial Intelligence*, 174(15):1142–1171, 2010. ISSN 0004-3702. doi: https://doi.org/10.1016/j.artint.2010.07.002. URL https://www.sciencedirect.com/science/article/pii/S0004370210000986.

A Additional Details for the Introductory Example

Here, we provide additional details regarding the introductory example in Section 1. Therein, two LLM providers compete to serve a user on a fixed task and may each choose between a low- and a high-TTC mode for their respective models. The resulting average output accuracy and generation cost associated with each mode are given by:

Provider 1		
	Avg. accuracy	Avg. gen. cost
Low TTC	70%	\$0.25
High TTC	90%	\$1

Provider 2		
	Avg. accuracy	Avg. gen. cost
Low TTC	50%	\$0.5
High TTC	95%	\$10

Suppose that providers have fixed their prices to obtain a margin of 25% over their generation costs for both low and high TTC modes, and that the user values each percentage of accuracy as $\$0.02$. Then, using the notation in Section 2, for each compute mode $\theta \in \{\text{Low}, \text{High}\}$ used by each provider $i = 1, 2$, the value they offer to the user is

$$V_i(\theta) = \underbrace{\$0.02 \times a_i(\theta)}_{q_i(\theta)} - \underbrace{1.25 \times c_i(\theta)}_{p_i(\theta)}, \tag{13}$$

where $a_i(\theta) \in [0, 100]$ are the average accuracy when selecting compute θ, and c_i are the generation costs. Using the specific values in the above table, we obtain:

$$\begin{cases} V_1(\text{Low}) = \$1.0875 \\ V_1(\text{High}) = \$0.55 \end{cases} \quad \text{and} \quad \begin{cases} V_2(\text{Low}) = \$0.375 \\ V_2(\text{High}) = -\$10.6. \end{cases} \tag{14}$$

Consequently, a perfectly rational user would select the first provider to serve the queries, no matter which compute level they use, and the market (for this particular task) is monopolistically dominated by the first provider. The total user value and provider utility of this market (*i.e.*, the social welfare W in Section 3) is, depending on the compute level selected by the first provider:

$$\begin{cases} \text{W}(\theta_1 = \text{Low}, \theta_2) = V_1(\text{Low}) + p_1(\text{Low}) - c_1(\text{Low}) = \$1.15 \\ \text{W}(\theta_1 = \text{High}, \theta_2) = V_1(\text{High}) + p_1(\text{High}) - c_1(\text{High}) = \$0.8, \end{cases} \tag{15}$$

regardless of the compute level θ_2 of the second provider. That is, the low compute mode is socially optimal. However, the utility obtained by the first provider is

$$\begin{cases} U_1(\text{Low}; \theta_2) = 0.25 \times \$0.25 = \$0.0625 \\ U_1(\text{High}; \theta_2) = 0.25 \times \$1 = \$0.25, \end{cases} \tag{16}$$

which strictly incentivizes the provider to select the high-TTC mode despite its lower social welfare. In this stylized example, the price of anarchy is:

$$\text{PoA} = \frac{\$1.15}{\$0.8} = 1.4375. \tag{17}$$

B Auction with Quality Verification

In this section, we show that the auction mechanism introduced in Section 4 can be naturally extended to allow the platform to verify the average quality of the models deployed by providers. As we will now demonstrate, this simple extension incentivizes providers to truthfully report their quality estimates.

Consider an extension of the auction $\widetilde{\mathcal{G}}$ that proceeds in multiple stages as follows (we overload notation for simplicity and also refer to it as $\widetilde{\mathcal{G}}$). A total of N providers bid simultaneously for the opportunity to serve a task $\mathcal{T}$ for a user. The auction begins with the user submitting to the platform a small set $\mathcal{Q}$ of queries representative of the task $\mathcal{T}$, which are then shared with all providers. Providers can use this information to estimate the quality of their models and determine the price they wish to offer (*e.g.*, by directly evaluating their model on $\mathcal{Q}$, using private historical data, *etc.*).[9]

In the first stage of the auction, each provider i submits a bid consisting of a pair $(\tilde{q}_i, p_i)$ to the platform, where $\tilde{q}_i$ is a reported estimate of their model quality on $\mathcal{T}$ and p_i is their bid price. The platform assigns the task $\mathcal{T}$ to the provider offering the highest value $V_i = \tilde{q}_i - p_i$. In the second stage, the winning provider, denoted by $\pi(1)$, selects a test-time compute level $\theta_{\pi(1)}$ (which need not be known to the platform or the user) and generates responses to the queries in $\mathcal{T}$. These responses are shared with both the user and the platform. Based on the observed responses, the platform computes an unbiased estimator $\hat{q}_{\pi(1)}$ of the true model quality on task $\mathcal{T}$ at compute level $\theta_{\pi(1)}$, *i.e.*,

$$\mathbb{E}[\hat{q}_{\pi(1)} \,|\, \theta_{\pi(1)}] = q_{\pi(1)}(\theta_{\pi(1)}). \tag{18}$$

For example, the platform may randomly sample a subset of the responses and evaluate their quality.[10] The platform then sets the payment to be made by the user to the provider for serving task $\mathcal{T}$ as

$$P(\tilde{\mathbf{q}}, \mathbf{p}) = \hat{q}_{\pi(1)} - V_{\pi(2)}, \tag{19}$$

where $\tilde{\mathbf{q}}$ and $\mathbf{p}$ denote the vectors of reported qualities and prices, respectively. Consequently, the (average) utility obtained by each provider is simply:

$$U_i(\tilde{q}_i, \theta_i, p_i; \tilde{\mathbf{q}}_{-i}, \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}) = \mathbb{E}\left[(P(\tilde{\mathbf{q}}, \mathbf{p}) - c_i(\theta_i)) \cdot \mathbf{1}\left\{ V_i > \max_{j \neq i} V_j \right\} \middle| \theta_{\pi(1)} \right], \tag{20}$$

where the expectation is taken over the randomness in the platform's estimator $\hat{q}_{\pi(1)}$, and note that only the provider serving the task obtains non-zero utility. Thus, the auction $\widetilde{\mathcal{G}}$ can be viewed as an extensive-form game in which each provider's action is a triplet $(\tilde{q}_i, \theta_i, p_i)$ and, analogously to Section 4, the social welfare is

$$\mathrm{W}(\tilde{\mathbf{q}}, \boldsymbol{\theta}, \mathbf{p}) = q_{\pi(1)}(\theta_{\pi(1)}) - c_{\pi(1)}(\theta_{\pi(1)}) = W_{\pi(1)}(\theta_{\pi(1)}). \tag{21}$$

The next result formalizes the fact that, in this extended auction $\widetilde{\mathcal{G}}$, truthfully reporting quality constitutes a (weakly) dominant strategy:

Theorem 6. *In the extended auction $\widetilde{\mathcal{G}}$, the strategy*

$$(\tilde{q}_i, \theta_i, p_i) = (q_i(\theta_i^*), \theta_i^*, c_i(\theta_i^*)) \tag{22}$$

is weakly dominant for each provider i, where $\theta_i^* \in \arg\max_{\theta \in \Theta}\{q_i(\theta) - c_i(\theta)\}$.

Proof. Fix an extended game $\widetilde{\mathcal{G}}$, a provider i, and any compute levels $\boldsymbol{\theta}_{-i}$, bid prices $\mathbf{p}_{-i}$, and bid qualities $\tilde{\mathbf{q}}_{-i}$. We will prove that, for provider i, selecting $\theta_i = \theta_i^*$, $p_i = c_i(\theta_i^*)$, and $\tilde{q}_i = q_i(\theta_i^*)$ is a weakly dominant strategy, that is, any deviation does not strictly increase their utility. Let $V_{-i}^{(1)} \geq V_{-i}^{(2)} \geq \cdots$ denote the values offered by the other providers in decreasing order, which are fixed since $(\tilde{\mathbf{q}}_{-i}, \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i})$ are fixed. Consider any possible deviation $(\tilde{q}_i, \theta_i, p_i)$ for provider i, and distinguish the following two cases.

[9] As in Section 4, quality refers to any agreed-upon metric between the user, the platform, and the providers, such as pass@k for coding tasks.

[10] There already exist online platforms that continuously monitor the quality of proprietary models on selected tasks, such as `https://marginlab.ai/trackers/claude-code-historical-performance/`.

Case 1: $V_{-i}^{(1)} > q_i(\theta_i^*) - c_i(\theta_i^*)$**.** Under the candidate strategy $(q_i(\theta_i^*), \theta_i^*, c_i(\theta_i^*))$, provider i loses the auction and receives utility zero. If the deviation also loses, the utility is likewise zero. If the deviation wins, the second-best value is $V_{-i}^{(1)}$ and the utility is

$$\begin{aligned} U_i(\tilde{q}_i, \theta_i, p_i;\, \tilde{\mathbf{q}}_{-i}, \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}) &= \mathbb{E}[\hat{q}_i - c_i(\theta_i) - V_{-i}^{(1)} \mid \theta_i] \\ &= q_i(\theta_i) - c_i(\theta_i) - V_{-i}^{(1)} \\ &= W_i(\theta_i) - V_{-i}^{(1)} \\ &\leq W_i(\theta_i^*) - V_{-i}^{(1)} \leq 0, \end{aligned}$$

where the first inequality uses $W_i(\theta_i) \leq W_i(\theta_i^*)$ by definition of θ_i^*, and the second uses $V_{-i}^{(1)} \geq q_i(\theta_i^*) - c_i(\theta_i^*) = W_i(\theta_i^*)$. Hence, no deviation strictly increases the utility of provider i above zero.

Case 2: $V_{-i}^{(1)} < q_i(\theta_i^*) - c_i(\theta_i^*)$**.** Under the candidate strategy, provider i wins and the second-best value is $V_{-i}^{(1)}$, yielding utility $U_i(q_i(\theta_i^*), \theta_i^*, c_i(\theta_i^*);\, \tilde{\mathbf{q}}_{-i}, \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}) = W_i(\theta_i^*) - V_{-i}^{(1)} > 0$. If the deviation loses, the resulting utility is $0 < W_i(\theta_i^*) - V_{-i}^{(1)}$, which is not profitable. If the deviation wins, the second-best value is again $V_{-i}^{(1)}$ (since the other providers' strategies are unchanged), and the utility is

$$\begin{aligned} U_i(\tilde{q}_i, \theta_i, p_i;\, \tilde{\mathbf{q}}_{-i}, \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}) &= \mathbb{E}[\hat{q}_i - c_i(\theta_i) - V_{-i}^{(1)} \mid \theta_i] \\ &= q_i(\theta_i) - c_i(\theta_i) - V_{-i}^{(1)} \\ &= W_i(\theta_i) - V_{-i}^{(1)} \\ &\leq W_i(\theta_i^*) - V_{-i}^{(1)} \\ &= U_i(q_i(\theta_i^*), \theta_i^*, c_i(\theta_i^*);\, \tilde{\mathbf{q}}_{-i}, \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}). \end{aligned}$$

Hence, no deviation strictly increases the utility of provider i above $W_i(\theta_i^*) - V_{-i}^{(1)}$.

In both cases, no deviation from $(q_i(\theta_i^*), \theta_i^*, c_i(\theta_i^*))$ strictly increases provider i's utility, so this strategy is weakly dominant. □

C Deferred Proofs

C.1 Proof of Theorem 1

We prove Theorem 1 for the case of perfectly rational ($\beta = \infty$) and boundedly rational ($\beta < \infty$) separately.

C.1.1 Potential game (perfect rationality)

We will show that, in the case where users are perfectly rational, the game $\mathcal{G}$ is a generalized ordinal potential game, *i.e.*, there exists a function $\Phi : \Theta^N \to \mathbb{R}$ such that for any provider i:

$$U_i\left(\theta_i'; \boldsymbol{\theta}_{-i}\right) - U_i\left(\theta_i; \boldsymbol{\theta}_{-i}\right) > 0 \Rightarrow \Phi\left(\theta_i'; \boldsymbol{\theta}_{-i}\right) - \Phi\left(\theta_i; \boldsymbol{\theta}_{-i}\right) > 0 \quad \text{for all } \boldsymbol{\theta} \in \Theta^N, \theta_i' \in \Theta.$$

Given the test-time compute levels of all providers $\boldsymbol{\theta}$, let π be the permutation ordering providers by the value they offer to users, *i.e.*,

$$V_{\pi(1)}(\theta_{\pi(1)}) > V_{\pi(2)}(\theta_{\pi(2)}) > \cdots > V_{\pi(N)}(\theta_{\pi(N)}).$$

and denote, for conciseness, $V_{\pi(1)} = V_{\pi(1)}(\theta_{\pi(1)})$ and $V_{\pi(2)} = V_{\pi(2)}(\theta_{\pi(2)})$. We define the potential function as a weighted sum of the second largest value and the utility gained by the provider $\pi(1)$, *i.e.*,

$$\begin{aligned} \Phi(\boldsymbol{\theta}) &= C \cdot V_{\pi(2)} + \log U_{\pi(1)}(\theta_{\pi(1)}; \boldsymbol{\theta}_{-\pi(1)}) \\ &= C \cdot V_{\pi(2)} + \log\left(p_{\pi(1)}(\theta_{\pi(1)}) - c_{\pi(1)}(\theta_{\pi(1)})\right), \end{aligned} \tag{23}$$

where $C > 0$ is a constant whose value we will specify later on.

To prove that the function given by Eq. 23 is a generalized ordinal potential function for the game $\mathcal{G}$, we distinguish two cases, depending on whether the provider i who unilaterally changes their test-time compute level to increase their utility (i) is the winning provider under $\boldsymbol{\theta}$, that is, $i = \pi(1)$, or (ii) is not the winning provider under $\boldsymbol{\theta}$.

In the first case, based on Eqs. 2, 4 and the fact that the provider i strictly increased their utility, it has to hold that $s_i(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) = s_i(V_i(\theta_i'); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) = 1$ and $\theta_i \prec \theta_i'$. This implies that the ordering of providers in terms of the value they offer to the users remains the same before and after the change of test-time compute level by provider i, and the second highest value under $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ coincide. Thus,

$$\begin{aligned} \Phi\left(\theta_i'; \boldsymbol{\theta}_{-i}\right) - \Phi\left(\theta_i; \boldsymbol{\theta}_{-i}\right) &= \log U_{\pi(1)}(\theta'_{\pi(1)}; \boldsymbol{\theta}_{-\pi(1)}) - \log U_{\pi(1)}(\theta_{\pi(1)}; \boldsymbol{\theta}_{-\pi(1)}) \\ &= \log U_i(\theta_i'; \boldsymbol{\theta}_{-i}) - \log U_i(\theta_i; \boldsymbol{\theta}_{-i}) > 0. \end{aligned}$$

In the second case, it holds that $s_i(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) = 0$, therefore, the only way for provider i to strictly increase their utility is by offering a higher value to the users than the previous winning provider and becoming the winning provider themselves. Therefore, under $\boldsymbol{\theta}'$, the winning provider is i and the provider who offers the second highest value to the users is the previous winning provider, which implies that $V'_{\pi'(2)} = V_{\pi(1)}$, where π' orders providers according to the values they offer under $\boldsymbol{\theta}'$. As a result, the difference in the function Φ is

$$\Phi\left(\theta_i'; \boldsymbol{\theta}_{-i}\right) - \Phi\left(\theta_i; \boldsymbol{\theta}_{-i}\right) = C \cdot (V_{\pi(1)} - V_{\pi(2)}) + \log U_i(\theta_i'; \boldsymbol{\theta}_{-i}) - \log U_{\pi(1)}(\theta_{\pi(1)}; \boldsymbol{\theta}_{-\pi(1)}), \tag{24}$$

where $V_{\pi(1)} - V_{\pi(2)} > 0$ because we have assumed no ties in the values providers can offer to the users. Moreover, since the action space Θ is finite, there has to exist a value $U_{max} > 0$ such that, for all j and $\boldsymbol{\theta}$, it holds that $U_j(\theta_j; \boldsymbol{\theta}_j) \leq U_{max}$. Similarly, there exists a value $U_{min} > 0$ such that for all j and $\boldsymbol{\theta}$, it holds that if $U_j(\theta_j; \boldsymbol{\theta}_j) > 0$, then $U_j(\theta_j; \boldsymbol{\theta}_j) > U_{min}$. Combining this with the fact that provider utilities are non-negative, Eq. 24 implies that

$$\Phi\left(\theta_i'; \boldsymbol{\theta}_{-i}\right) - \Phi\left(\theta_i; \boldsymbol{\theta}_{-i}\right) \geq C \cdot (V_{\pi(1)} - V_{\pi(2)}) + \log U_{min} - \log U_{max},$$

and it suffices to find a value for the constant C such that $C > \frac{\log U_{max} - \log U_{min}}{V_{\pi(1)} - V_{\pi(2)}}$. Since we have assumed no ties in the values that providers can offer to the users, there has to exist a δ_{min} such that $V_i(\theta_i) - V_j(\theta_j) > \delta_{min}$, uniformly across providers i, j and compute levels θ_i, θ_j. Setting $C = \frac{\log U_{max} - \log U_{min}}{\delta_{min}}$ concludes the proof.

C.1.2 Potential game (bounded rationality)

We will show that, in the case of boundedly rational users, the game $\mathcal{G}$ with $\beta < \infty$ is also a generalized ordinal potential game. To this end, we extend the potential in Eq. 23 under perfect rationality to a potential Φ under bounded rationality by observing that: (i) for $\beta < \infty$, all providers can have a non-zero market share (and hence non-zero utilities), (ii) each provider partially aims to improve their profit $p_i(\theta_i) - c_i(\theta_i)$ while still offering a sufficiently high value $V_i(\theta_i)$ relative to others, and (iii) in the log domain, utilities decompose as:

$$\log U_i(\theta_i; \boldsymbol{\theta}_{-i}) = \log\left(s_i(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) \cdot (p_i(\theta_i) - c_i(\theta_i))\right) \tag{25}$$

$$= \log\left(p_i(\theta_i) - c_i(\theta_i)\right) + \beta V_i(\theta_i) - \log\left(\sum_{j=0}^{N} \exp(\beta \cdot V_j(\theta_j))\right), \tag{26}$$

for $\boldsymbol{\theta} \in \Theta^N$ and any provider i, and we denote $V_0(\theta_0) \coloneqq V_0$ for simplicity. Based on the three observations above, let us define:

$$\Phi(\boldsymbol{\theta}) = \sum_{j=1}^{N} \log(p_j(\theta_j) - c_j(\theta_j)) + \beta \cdot \sum_{j=1}^{N} V_j(\theta_j) - \log\left(\sum_{j=0}^{N} \exp(\beta \cdot V_j(\theta_j))\right), \quad \forall \boldsymbol{\theta} \in \Theta^N, \tag{27}$$

where note that the last term is not summed over all providers, since it is a shared normalization constant related to the softmax allocation s under bounded rationality. To verify that Φ is indeed an ordinal potential function, consider a deviation θ_i' for provider i. Then:

$$\begin{aligned}
& U_i(\theta_i'; \boldsymbol{\theta}_{-i}) > U_i(\theta_i; \boldsymbol{\theta}_{-i}) \\
\iff & \log U_i(\theta_i'; \boldsymbol{\theta}_{-i}) > \log U_i(\theta_i; \boldsymbol{\theta}_{-i}) \\
\iff & \log(p_i(\theta_i') - c_i(\theta_i')) + \beta V_i(\theta_i') - \log\left(\sum_{j \neq i} e^{\beta V_j(\theta_j)} + e^{\beta V_i(\theta_i')}\right) \\
& \qquad > \log(p_i(\theta_i) - c_i(\theta_i)) + \beta V_i(\theta_i) - \log\left(\sum_{j=0}^{N} e^{\beta V_j(\theta_j)}\right) \\
\iff & \sum_{j \neq i, 0} [\log(p_j(\theta_j) - c_j(\theta_j)) + \beta V_j(\theta_j)] + \log(p_i(\theta_i') - c_i(\theta_i')) + \beta V_i(\theta_i') \\
& \qquad - \log\left(\sum_{j \neq i} e^{\beta V_j(\theta_j)} + e^{\beta V_i(\theta_i')}\right) \\
& \qquad > \sum_{j \neq i, 0} [\log(p_j(\theta_j) - c_j(\theta_j)) + \beta V_j(\theta_j)] + \log(p_i(\theta_i) - c_i(\theta_i)) + \beta V_i(\theta_i) \\
& \qquad - \log\left(\sum_{j=0}^{N} e^{\beta V_j(\theta_j)}\right).
\end{aligned}$$

and we can conclude that:

$$
\begin{aligned}
& U_i(\theta_i'; \boldsymbol{\theta}_{-i}) > U_i(\theta_i; \boldsymbol{\theta}_{-i}) \\
\iff & \sum_{j \neq i} [\log(p_j(\theta_j) - c_j(\theta_j)) + \beta V_j(\theta_j)] + \log(p_i(\theta_i') - c_i(\theta_i')) + \beta V_i(\theta_i') \\
& \quad - \log \left(\sum_{j \neq i} e^{\beta V_j(\theta_j)} + e^{\beta V_i(\theta_i')} \right) \\
& \quad > \sum_{j=1}^{N} [\log(p_j(\theta_j) - c_j(\theta_j)) + \beta V_j(\theta_j)] - \log \left(\sum_{j=0}^{N} e^{\beta V_j(\theta_j)} \right) \\
\iff & \Phi(\theta_i'; \boldsymbol{\theta}_{-i}) > \Phi(\boldsymbol{\theta}).
\end{aligned}
$$

C.2 Proof of Theorem 2

We fix a test-time compute game $\mathcal{G}$ under perfect rationality and any pure Nash equilibrium $\boldsymbol{\theta}^\dagger \in \Theta^N$. Firstly, observe that it must be

$$s_1\left(V_1(\theta_1^\dagger); \boldsymbol{V}_{-1}(\boldsymbol{\theta}_{-1}^\dagger)\right) = 1. \tag{28}$$

Indeed, if the above were false, by definition of s under perfect rationality, there would exist a provider $i \neq 1$ such that $V_i(\theta_i^\dagger) > V_1(\theta_1^\dagger)$. Then, given that providers are indexed in decreasing order of their maximum possible offered values, we have that $V_1^* > V_i^*$ and hence provider 1 could best respond and strictly increase their utility, meaning that $\boldsymbol{\theta}^\dagger$ would not be a Nash equilibrium. Hence, in $\boldsymbol{\theta}^\dagger$, provider 1 offers a strictly higher value than any other provider. Furthermore, if $V_1(\boldsymbol{\theta}_1^\dagger) < V_2^*$, then, at least provider 2 could best respond by increasing the value they offer to V_2^*, serving all queries and obtaining strictly positive utility. Thus, in $\boldsymbol{\theta}^\dagger$ it holds that $V_1(\theta_1) > V_2^*$, and since provider 1 is playing a best response conditional on serving all queries, it must be that:

$$\theta_1^\dagger = \operatorname*{argmax}_{\boldsymbol{\theta}_1 \in \Theta} \{p_1(\theta_1) - c_1(\theta_1) \mid V_1(\theta_1) > V_2^*\} = \max\{\theta_1 | V_1(\theta_1) > V_2^*\}, \tag{29}$$

where the second equality holds because providers have profits that are increasing with the compute level. Next, we derive conditions on the compute of providers $i \neq 1$ such that $\boldsymbol{\theta}^\dagger$ is a Nash equilibrium. To this end, observe that since $V_1(\theta_1^\dagger) > V_2^*$, any provider $i \neq 1$ is unable to serve any queries, and hence has utility 0 independently of their deviations in $\boldsymbol{\theta}^\dagger$. We then distinguish two cases:

- If for any $\theta_1 \in \Theta$ we have that $V_1(\theta_1) > V_2^*$, then any provider $i \neq 1$ cannot increase their utilities, and provider 1 selecting their compute as in Eq. 29 is not able to improve their utility. We conclude that in this case, providers $i \neq 1$ selecting their compute arbitrarily leads to a pure Nash equilibrium (although the equilibrium is not strict).
- Suppose there exists $\theta_1 \in \Theta$ such that $V_1(\theta_1) < V_2^*$ and that $p_1(\theta_1) - c_1(\theta_1) \geq p_1(\theta_1^\dagger) - c_1(\theta_1^\dagger)$ (note that we recover the previous case if $p_1(\theta_1) - c_1(\theta_1) < p_1(\theta_1^\dagger) - c_1(\theta_1^\dagger)$), with $\theta_1^\dagger$ as in Eq. 29. Then, provider 1 can strictly increase their utility by choosing θ_1 unless there exists another provider i with compute $\theta_i^\dagger$ such that $V_i(\theta_i^\dagger) > V_1(\theta_1)$.

We can summarize the above two cases by stating that at least one provider must offer a value higher than:

$$\max_{\theta_1 \in \Theta} \left\{ V_1(\theta_1) \middle| V_1(\theta_1) < V_2^* \quad \text{and} \quad p_1(\theta_1) - c_1(\theta_1) > p_1(\theta_1^\dagger) - c_1(\theta_1^\dagger) \right\} \tag{30}$$

with the constraint being lifted if the above set is empty. This concludes the proof.

C.3 Proof of Theorem 3

The intuition for the proof is that the softmax converges to an indicator function for the maximum offered value as $\beta \to \infty$, and hence, the equilibrium computes should also converge. We now make this intuition precise, and assume $V_0 = 0$ in what follows for simplicity. We will leverage the following well-known fact about the softmax function, which we prove here for completeness:

Lemma 1. *Let $\boldsymbol{X} = (x_1, \ldots, x_N) \in \mathbb{R}^N$ such that $x_1 > x_2 > \cdots > x_N$ and $\sigma^\beta(\boldsymbol{X})$ denote the softmax function with inverse temperature β. Then:*

$$|\sigma^\beta(\boldsymbol{X})_i - \mathbb{1}\{i = 1\}| \leq (N-1) \cdot \exp(-\beta \cdot (x_1 - x_2)), \quad \forall i = 1, \ldots, N. \tag{31}$$

Proof. From the softmax definition, we have:

$$\begin{cases} \sigma^\beta(\boldsymbol{X})_1 = \dfrac{1}{1 + \sum_{j=2}^N \exp(-\beta(x_1 - x_j))} \\ \sigma^\beta(\boldsymbol{X})_i = \dfrac{\exp(-\beta(x_1 - x_i))}{1 + \sum_{j=2}^N \exp(-\beta(x_1 - x_j))}, \quad i \geq 2. \end{cases} \tag{32}$$

Then, for $i \neq 1$ we have:

$$|\sigma^\beta(\boldsymbol{X})_i - \mathbb{1}\{i = 1\}| = \sigma^\beta(\boldsymbol{X})_i \leq \exp(-\beta(x_1 - x_i)) \leq (N-1) \cdot \exp(-\beta \cdot (x_1 - x_2)), \tag{33}$$

while for $i = 1$ we have:

$$|\sigma^\beta(\boldsymbol{X})_i - \mathbb{1}\{i = 1\}| = 1 - \sigma^\beta(\boldsymbol{X})_1 \leq \sum_{j=2}^N \exp(-\beta \cdot (x_1 - x_j)) \leq (N-1) \cdot \exp(-\beta \cdot (x_1 - x_2)). \tag{34}$$

□

To prove Theorem 3, we will use Lemma 1 with $\boldsymbol{X}$ corresponding to the vector of offered values $(\boldsymbol{V}(\boldsymbol{\theta}), 0)$ (which has $N+1$ components, including $V_0 = 0$). More precisely, let us denote by s^β and s^∞ the market share functions under bounded rationality and perfect rationality, respectively (see Table 1). Then, for any $\boldsymbol{\theta} \in \Theta^N$, and any provider i, we have:

$$\left| s_i^\beta(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) - s_i^\infty(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) \right| \leq N \cdot \exp\left(-\beta \cdot (V_{\pi(1)}(\theta_{\pi(1)}) - V_{\pi(2)}(\theta_{\pi(2)}))\right), \tag{35}$$

where π is the permutation ordering providers by their offered value at $\boldsymbol{\theta}$. Taking the maximum among providers i and the (finitely many) compute levels they can select in the above expression, and letting $\delta_{\min} > 0$ be the minimum gap in value offered by any two different providers (recall that we are assuming that providers always offer different values):

$$\max_{\boldsymbol{\theta}} \max_i \left| s_i^\beta(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) - s_i^\infty(V_i(\theta_i); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i})) \right| \leq N \cdot \exp\left(-\beta \cdot \delta_{\min}\right). \tag{36}$$

Thus, we have that $s^\beta \to s^\infty$ as $\beta \to \infty$, where the convergence is uniform over Θ^N and the provider index i. Then, since for each provider i, the profit $p_i(\theta_i) - c_i(\theta_i)$ is bounded on the finite domain Θ, we have that $\boldsymbol{U}^\beta \to \boldsymbol{U}^\infty$ as $\beta \to \infty$ uniformly over Θ^N and i, where $\boldsymbol{U}^\beta \colon \Theta^N \to \mathbb{R}^N$ denotes the function that maps a joint compute choice $\boldsymbol{\theta}$ to the provider's utilities as defined in Eq. 4 using the market allocation function s^β, and $\boldsymbol{U}^\infty$ is defined analogously. To use the above convergence result, we let

$$\varepsilon < \frac{1}{2} \min_{\boldsymbol{\theta}, \boldsymbol{\theta}'} \min_i \left\{ |U_i^\infty(\theta_i; \boldsymbol{\theta}_{-i}) - U_i^\infty(\theta_i'; \boldsymbol{\theta}'_{-i})| : \ |U_i^\infty(\theta_i; \boldsymbol{\theta}_{-i}) - U_i^\infty(\theta_i'; \boldsymbol{\theta}'_{-i})| > 0 \right\} \tag{37}$$

be at most half the minimum positive change in utility a provider can achieve in the game $\mathcal{G}$ under perfect rationality. Then, let β_0 be such that for any $\beta > \beta_0$, it holds that

$$\max_{\boldsymbol{\theta}} \max_i \left| U_i^\beta(\theta_i; \boldsymbol{\theta}_{-i}) - U_i^\infty(\theta_i; \boldsymbol{\theta}_{-i}) \right| \leq \varepsilon \tag{38}$$

and fix any pure Nash equilibrium $\boldsymbol{\theta}^\dagger$ of $\mathcal{G}$ for $\beta > \beta_0$. To show that $\boldsymbol{\theta}^\dagger$ is also an equilibrium of $\mathcal{G}$ under perfect rationality, we argue by contradiction, and suppose this premise is false. Then, there

exists a provider i that can change their compute in the game $\mathcal{G}$ under perfect rationality and strictly increase their utility, that is, there exists $\theta_i' \in \Theta$ such that $U_i^\infty(\theta_i'; \boldsymbol{\theta}_{-i}^\dagger) > U_i^\infty(\theta_i^\dagger; \boldsymbol{\theta}_{-i}^\dagger)$. Then:

$$
\begin{aligned}
2\varepsilon &\overset{(*)}{<} U_i^\infty(\theta_i'; \boldsymbol{\theta}_{-i}^\dagger) - U_i^\infty(\theta_i^\dagger; \boldsymbol{\theta}_{-i}^\dagger) && (39)\\
&= \Big[U_i^\infty(\theta_i'; \boldsymbol{\theta}_{-i}^\dagger) - U_i^\beta(\theta_i'; \boldsymbol{\theta}_{-i}^\dagger)\Big] + \Big[U_i^\beta(\theta_i'; \boldsymbol{\theta}_{-i}^\dagger) - U_i^\beta(\theta_i^\dagger; \boldsymbol{\theta}_{-i}^\dagger)\Big] \\
&\quad + \Big[U_i^\beta(\theta_i^\dagger; \boldsymbol{\theta}_{-i}^\dagger) - U_i^\infty(\theta_i^\dagger; \boldsymbol{\theta}_{-i}^\dagger)\Big] && (40)\\
&\overset{(**)}{\leq} \varepsilon + 0 + \varepsilon, && (41)
\end{aligned}
$$

where $(*)$ follows from Eq. 37 together with the contradiction hypothesis $U_i^\infty(\theta_i'; \boldsymbol{\theta}_{-i}^\dagger) - U_i^\infty(\theta_i^\dagger; \boldsymbol{\theta}_{-i}^\dagger) > 0$, and $(**)$ follows from bounding the first and third terms by ε via Eq. 38, and the second term by 0 since $\boldsymbol{\theta}^\dagger$ is a Nash equilibrium of $\mathcal{G}$ under bounded rationality, hence $U_i^\beta(\theta_i^\dagger; \boldsymbol{\theta}_{-i}^\dagger) \geq U_i^\beta(\theta_i'; \boldsymbol{\theta}_{-i}^\dagger)$. This is a contradiction, which proves that $\boldsymbol{\theta}^\dagger$ is also an equilibrium of $\mathcal{G}$ under perfect rationality.

C.4 Proof of Theorem 4

The following is an auxiliary result that will be used to prove Theorem 4:

Lemma 2. *Let $x_1 > x_2 > \cdots > x_N$ and $y_1, \ldots, y_N$ be real positive numbers, and denote both vectors by $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively. Denote by $\sigma^\beta(\boldsymbol{X})$ the softmax with inverse temperature β. Then,*

$$\begin{cases} \sum_{i=1}^N y_i \cdot \sigma^\beta(\boldsymbol{X})_i \leq y_1 + e^{-\beta(x_1 - x_2)} \cdot \sum_{i=2}^N (y_i + y_1) \\ \sum_{i=1}^N y_i \cdot \sigma^\beta(\boldsymbol{X})_i \geq y_1 - e^{-\beta(x_1 - x_2)} \cdot \sum_{i=2}^N (y_i + y_1) \end{cases} \tag{42}$$

Proof. From the softmax definition, we have:

$$\begin{cases} \sigma^\beta(\boldsymbol{X})_1 = \dfrac{1}{1 + \sum_{j=2}^N e^{-\beta(x_1 - x_j)}} \\ \sigma^\beta(\boldsymbol{X})_i = \dfrac{e^{-\beta(x_1 - x_i)}}{1 + \sum_{j=2}^N e^{-\beta(x_1 - x_j)}} \leq e^{-\beta(x_1 - x_i)}, \quad i \geq 2, \end{cases} \tag{43}$$

and,

$$\sum_{i=1}^N y_i \cdot \sigma^\beta(\boldsymbol{X})_i = y_1 + \sum_{i=2}^N (y_i - y_1) \cdot \sigma^\beta(\boldsymbol{X})_i. \tag{44}$$

We begin by showing that the first inequality in the lemma holds. Since $y_i, y_1 > 0$, we have $y_i - y_1 \leq |y_i - y_1| \leq y_i + y_1$, and since $\sigma^\beta(\boldsymbol{X})_i \geq 0$:

$$\sum_{i=2}^N (y_i - y_1)\, \sigma^\beta(\boldsymbol{X})_i \;\leq\; \sum_{i=2}^N (y_i + y_1)\, \sigma^\beta(\boldsymbol{X})_i. \tag{45}$$

Then, using that $\sigma^\beta(\boldsymbol{X})_i \leq e^{-\beta(x_1 - x_i)} \leq e^{-\beta(x_1 - x_2)}$ for $i \geq 2$, and that $y_i + y_1 > 0$:

$$\sum_{i=2}^N (y_i + y_1)\, \sigma^\beta(\boldsymbol{X})_i \;\leq\; e^{-\beta(x_1 - x_2)} \sum_{i=2}^N (y_i + y_1), \tag{46}$$

which combined with Eq. (44) yields the first inequality.

Similarly, we can show that the second inequality in 42 also holds. Indeed, from Eq. 44, we obtain:

$$\begin{aligned} \sum_{i=1}^N y_i \sigma^\beta(\boldsymbol{X})_i &= y_1 - \sum_{i=2}^N (y_1 - y_i) \cdot \sigma^\beta(\boldsymbol{X})_i && (47) \\ &\geq y_1 - \sum_{i=2}^N (y_1 + y_i) \cdot \sigma^\beta(\boldsymbol{X})_i && (48) \\ &\geq y_1 - \sum_{i=2}^N (y_1 + y_i) \cdot e^{-\beta(x_1 - x_i)} && (49) \\ &\geq y_1 - \sum_{i=2}^N (y_1 + y_i) \cdot e^{-\beta(x_1 - x_2)} && (50) \end{aligned}$$

□

We now prove Theorem 4. To this end, fix a test-time compute game $\mathcal{G}$ with $0 < \beta \leq \infty$ and a Nash equilibrium $\boldsymbol{\theta}^\dagger$. Further, for each provider i, denote by $\theta_i^* = \operatorname{argmax}_\theta \{q_i(\theta) - c_i(\theta)\}$ their socially

optimal compute, and let $\boldsymbol{\theta}^* = (\theta_i^*)_i$. Recall the definition of the price of anarchy in Eq. 8:[11]

$$\mathrm{PoA}(\mathcal{G}) = \frac{\max_{\theta\in\Theta^N} \mathrm{W}(\boldsymbol{\theta})}{\mathrm{W}(\boldsymbol{\theta}^\dagger)} \geq \frac{\mathrm{W}(\boldsymbol{\theta}^*)}{\mathrm{W}(\boldsymbol{\theta}^\dagger)} = \frac{\overbrace{\sum_{i=1}^{N} s_i(V_i(\theta_i^*); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i}^*)) \cdot \mathrm{W}_i(\theta_i^*)}^{(\diamond)}}{\underbrace{\sum_{i=1}^{N} s_i(V_i(\theta_i^\dagger); \boldsymbol{V}_{-i}(\boldsymbol{\theta}_{-i}^\dagger)) \cdot \mathrm{W}_i(\theta_i^\dagger)}_{(\bullet)}}. \tag{51}$$

Denote by π and π^* the permutations ordering providers by their value at the equilibrium $\boldsymbol{\theta}^\dagger$ and at $\boldsymbol{\theta}^*$, respectively. Then, we can use Lemma 2 to lower bound the numerator and upper bound the denominator in the above.

Starting with the numerator $(\diamond)$, we take in Lemma 2 $\boldsymbol{X}$ to be the vector $(\boldsymbol{V}(\boldsymbol{\theta}^*), 0)$ with $N+1$ components ordered by π^* (assuming without loss of generality that all providers can offer a value higher than the abstention value) and $\boldsymbol{Y}$ to be the vector $(\mathrm{W}_i(\theta_i^*), 0)$ with $N+1$ components ordered by π^*. Then, Lemma 2 implies:

$$(\diamond) \geq \mathrm{W}_{\pi^*(1)}(\theta^*_{\pi^*(1)}) \cdot \left(1 - e^{-\beta\Delta V^*}\right) - e^{-\beta\Delta V^*} \cdot \sum_{i\geq 2}^{N} \left(\mathrm{W}_{\pi^*(i)}(\theta^*_{\pi^*(i)}) + \mathrm{W}_{\pi^*(1)}(\theta^*_{\pi^*(1)})\right), \tag{52}$$

where $\Delta V^* = V_{\pi^*(1)}(\theta^*_{\pi^*(1)}) - V_{\pi^*(2)}(\theta^*_{\pi^*(2)})$ is the difference in values at $\boldsymbol{\theta}^*$ between the first and second providers.

Similarly, the term $(\bullet)$ can be upper-bounded by taking in Lemma 2 $\boldsymbol{X}$ to be the vector $(\boldsymbol{V}(\boldsymbol{\theta}^\dagger), 0)$ ordered by π (assuming without loss of generality that all providers can offer a value higher than the abstention value) and $\boldsymbol{Y}$ to be the vector $(\mathrm{W}_i(\theta_i^\dagger), 0)$ ordered by π. Then, we obtain:

$$(\bullet) \leq \mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)}) \cdot \left(1 + e^{-\beta\Delta V^\dagger}\right) + e^{-\beta\Delta V^\dagger} \cdot \sum_{i\geq 2}^{N} \left(\mathrm{W}_{\pi(i)}(\theta^\dagger_{\pi(i)}) + \mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)})\right), \tag{53}$$

where $\Delta V^\dagger = V_{\pi(1)}(\theta^\dagger_{\pi(1)}) - V_{\pi(2)}(\theta^\dagger_{\pi(2)})$.

Using such bounds on $(\diamond)$ and on $(\bullet)$, the price of anarchy satisfies:

$$\begin{aligned}
\mathrm{PoA}(\mathcal{G}) &\geq \frac{\mathrm{W}_{\pi^*(1)}(\theta^*_{\pi^*(1)}) \cdot \left(1 - e^{-\beta\Delta V^*}\right) - e^{-\beta\Delta V^*} \cdot \sum_{i\geq 2}^{N} \left(\mathrm{W}_{\pi^*(i)}(\theta^*_{\pi^*(i)}) + \mathrm{W}_{\pi^*(1)}(\theta^*_{\pi^*(1)})\right)}{\mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)}) \cdot \left(1 + e^{-\beta\Delta V^\dagger}\right) + e^{-\beta\Delta V^\dagger} \cdot \sum_{i\geq 2}^{N} \left(\mathrm{W}_{\pi(i)}(\theta^\dagger_{\pi(i)}) + \mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)})\right)} \\
&= \frac{\left(1 - e^{-\beta\Delta V^*}\right) - e^{-\beta\Delta V^*} \cdot \sum_{i\geq 2}^{N} \left(1 + \frac{\mathrm{W}_{\pi^*(i)}(\theta^*_{\pi^*(i)})}{\mathrm{W}^*}\right)}{\frac{\mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)})}{\mathrm{W}^*} \cdot \left(1 + e^{-\beta\Delta V^\dagger}\right) + e^{-\beta\Delta V^\dagger} \cdot \sum_{i\geq 2}^{N} \left(\frac{\mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)}) + \mathrm{W}_{\pi(i)}(\theta^\dagger_{\pi(i)})}{\mathrm{W}^*}\right)},
\end{aligned} \tag{54}$$

where we have defined $\mathrm{W}^* \coloneqq \mathrm{W}_{\pi^*(1)}(\theta^*_{\pi^*(1)})$. To simplify, we define:

$$\Delta\mathrm{W} \coloneqq \mathrm{W}_{\pi^*(1)}(\theta^*_{\pi^*(1)}) - \mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)}),$$

which represents the difference in the contribution to the social welfare by the provider offering the highest value at equilibrium, and at compute $\boldsymbol{\theta}^*$. Using this in the bound in Eq. 54 and Taylor-

[11] For ease of exposition, we take $V_0 = 0$, with the case $V_0 > 0$ following using an analogous argument.

expanding up to second order the function $t \mapsto 1/(1-t)$ around $t = 0$, we obtain:

$$\begin{aligned}
\text{PoA}(\mathcal{G}) &\geq \frac{\left(1 - e^{-\beta\Delta V^*}\right) - e^{-\beta\Delta V^*} \cdot \sum_{i\geq 2}^{N} \left(1 + \frac{\mathrm{W}_{\pi^*(i)}(\theta^*_{\pi^*(i)})}{\mathrm{W}^*}\right)}{1 - \Delta\mathrm{W}/\mathrm{W}^* + e^{-\beta\Delta V^\dagger}\left(1 + \sum_{i\geq 2}^{N} \frac{\mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)}) + \mathrm{W}_{\pi(i)}(\theta^\dagger_{\pi(i)})}{\mathrm{W}^*}\right) + \mathcal{O}\left(\|e^{-\beta\Delta V^\dagger}, \Delta\mathrm{W}\|^2\right)} \\
&= \left(\left(1 - e^{-\beta\Delta V^*}\right) - e^{-\beta\Delta V^*} \cdot \sum_{i\geq 2}^{N} \left(1 + \frac{\mathrm{W}_{\pi^*(i)}(\theta^*_{\pi^*(i)})}{\mathrm{W}^*}\right)\right) \\
&\quad \cdot \left(1 + \Delta\mathrm{W}/\mathrm{W}^* - e^{-\beta\Delta V^\dagger} - e^{-\beta\Delta V^\dagger} \sum_{i\geq 2}^{N} \frac{\mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)}) + \mathrm{W}_{\pi(i)}(\theta^\dagger_{\pi(i)})}{\mathrm{W}^*}\right) \\
&\quad + \mathcal{O}\left(\|e^{-\beta\Delta V^\dagger}, e^{-\beta\Delta V^*}, \Delta\mathrm{W}\|^2\right) \\
&= \left(1 - e^{-\beta\Delta V^*}\left(1 + \sum_{i\geq 2}^{N} \left(1 + \frac{\mathrm{W}_{\pi^*(i)}(\theta^*_{\pi^*(i)})}{\mathrm{W}^*}\right)\right)\right) \\
&\quad \cdot \left(1 + \Delta\mathrm{W}/\mathrm{W}^* - e^{-\beta\Delta V^\dagger}\left(1 + \sum_{i\geq 2}^{N} \frac{\mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)}) + \mathrm{W}_{\pi(i)}(\theta^\dagger_{\pi(i)})}{\mathrm{W}^*}\right)\right) \\
&\quad + \mathcal{O}\left(\|e^{-\beta\Delta V^\dagger}, e^{-\beta\Delta V^*}, \Delta\mathrm{W}\|^2\right) \\
&= 1 + \Delta\mathrm{W}/\mathrm{W}^* - e^{-\beta\Delta V^\dagger} \cdot \left(1 + \sum_{i\geq 2}^{N} \frac{\mathrm{W}_{\pi(1)}(\theta^\dagger_{\pi(1)}) + \mathrm{W}_{\pi(i)}(\theta^\dagger_{\pi(i)})}{\mathrm{W}^*}\right) \\
&\quad - e^{-\beta\Delta V^*} \cdot \left(1 + \sum_{i\geq 2}^{N} \left(1 + \frac{\mathrm{W}_{\pi^*(i)}(\theta^*_{\pi^*(i)})}{\mathrm{W}^*}\right)\right) \\
&\quad + \mathcal{O}\left(\|e^{-\beta\Delta V^\dagger}, e^{-\beta\Delta V^*}, \Delta\mathrm{W}\|^2\right)
\end{aligned}$$

To obtain the same form as in the statement of Theorem 4, we can let $\Delta V = \min(\Delta V^*, \Delta V^\dagger)$ and define $f(\beta)$ such that:

$$\begin{aligned}
f(\beta) = -\frac{e^{-\beta\Delta V^\dagger}}{\mathrm{W}^*} \cdot \left(\mathrm{W}^* + \sum_{i\geq 2}^{N} \left(W_{\pi(1)}(\theta^\dagger_{\pi(1)}) + \mathrm{W}_{\pi(i)}(\theta^\dagger_{\pi(i)})\right)\right) \\
- \frac{e^{-\beta\Delta V^*}}{\mathrm{W}^*} \cdot \left(\mathrm{W}^* + \sum_{i\geq 2}^{N} \left(\mathrm{W}^* + \mathrm{W}_{\pi^*(i)}(\theta^*_{\pi^*(i)})\right)\right) = \mathcal{O}\left(e^{-\beta\Delta V}\right).
\end{aligned}$$

C.5 Proof of Theorem 5

Fix a game $\widetilde{\mathcal{G}}$, a provider i, and any compute levels $\boldsymbol{\theta}_{-i}$ and bid prices $\mathbf{p}_{-i}$. We will prove that, for provider i, selecting $\theta_i = \theta_i^*$ and $p_i = c_i(\theta_i^*)$ is a dominant strategy. To this end, we first show that given a fixed θ_i, the provider maximizes their utility by bidding $p_i = c_i(\theta_i)$, and then show that $\theta_i = \theta_i^*$ is their best choice of compute. Recall that the utility of provider i is (Eq. 10):

$$\begin{aligned} U_i(\theta_i, p_i; \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}) &= (P(\boldsymbol{\theta}, \mathbf{p}) - c_i(\theta_i)) \cdot \mathbb{1}\left\{V_i(\theta_i, p_i) > \max_{j \neq i} V_j(\theta_j, p_j)\right\} \\ &= \left(q_{\pi(1)}(\theta_{\pi(1)}) - c_i(\theta_i) - (q_{\pi(2)}(\theta_{\pi(2)}) - p_{\pi(2)})\right) \cdot \mathbb{1}\left\{i = \pi(1)\right\}. \end{aligned}$$

Bidding generation cost is optimal. Let us first prove that, for a fixed θ_i, bidding $p_i = c_i(\theta_i)$ is always optimal. To this end, we assume that the provider bids $p_i = c_i(\theta_i)$ and show that no deviation to a different price bid can increase their utility. Consider first that provider i wins the bid when bidding $(q_i(\theta_i), c_i(\theta_i))$. Thus, it must be that $i = \pi(1)$ and $q_i(\theta_i) - c_i(\theta_i) > q_{\pi(2)}(\theta_{\pi(2)}) - p_{\pi(2)}$. Then, consider a deviation where provider i bids a different price $p_i \neq c_i(\theta_i)$. With their new bid $(q_i(\theta_i), p_i)$, they either lose the auction, in which case they strictly decrease their utility to 0, or they are still winning the auction. However, in the latter case, since conditional on winning, their utility is independent of p_i, we conclude that this deviation maintains their utility, and hence the provider cannot profit from it.

Now, suppose that provider i initially loses the auction by bidding $(q_i(\theta_i), c_i(\theta_i))$, which implies that:

$$q_i(\theta_i) - c_i(\theta_i) < q_{\pi(1)}(\theta_{\pi(1)}) - p_{\pi(1)}. \tag{55}$$

Then, any deviation where they bid a higher $p_i > c_i(\theta_i)$ will further decrease their offered value, and hence they will still lose the auction and maintain their null utility. In case the provider deviates to a lower price $p_i < c_i(\theta_i)$ and wins the auction by doing so, denote by π' the permutation ordering providers after the deviation. Then, $\pi'(1) = i$ and $\pi'(2) = \pi(1)$. Consequently, the new utility of provider i after the deviation will be:

$$U_i(\theta_i, p_i; \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}) = q_i(\theta_i) - c_i(\theta_i) - \left(q_{\pi(1)}(\theta_{\pi(1)}) - p_{\pi(1)}\right) < 0, \tag{56}$$

meaning that the deviation is not profitable. Hence, we conclude that bidding $p_i = c_i(\theta_i)$ is always optimal.

Bidding the socially optimal compute is dominant. We now prove that bidding with $\theta_i = \theta_i^*$ is dominant for provider i. To this end, we assume that for any compute θ_i, provider i selects their optimal price $p_i = c_i(\theta_i)$. Then, the utility of the provider can be written as:

$$\begin{aligned} U_i(\theta_i, c_i(\theta_i); \boldsymbol{\theta}_{-i}, \mathbf{p}_{-i}) = &\left(q_{\pi(1)}(\theta_{\pi(1)}) - c_i(\theta_i)\right) \cdot \mathbb{1}\left\{i = \pi(1)\right\} \\ &- \left(q_{\pi(2)}(\theta_{\pi(2)}) - p_{\pi(2)}\right) \cdot \mathbb{1}\left\{i = \pi(1)\right\} \end{aligned}$$

Since, conditional on provider i winning the bid, the second term in the above expression does not depend on the action of provider i, their optimal compute level is to select:

$$\begin{aligned} &\operatorname*{argmax}_{\theta_i \in \Theta} \left\{q_i(\theta_i) - c_i(\theta_i) | i = \pi(1)\right\} \\ =&\operatorname*{argmax}_{\theta_i \in \Theta} \left\{q_i(\theta_i) - c_i(\theta_i) \middle| q_i(\theta_i) - c_i(\theta_i) > q_{\pi(2)}(\theta_{\pi(2)}) - p_{\pi(2)}\right\} \\ =&\operatorname*{argmax}_{\theta_i \in \Theta} \left\{q_i(\theta_i) - c_i(\theta_i)\right\} \\ =&\theta_i^*, \end{aligned}$$

which proves the claim.

D Additional Experimental Details

Here, we provide additional experimental details for our empirical evaluation in Section 5. The code and data for our experiments is available at `https://github.com/Human-Centric-Machine-Learning/strategic-ttc`.

Hardware setup. Our experiments are executed on a compute server equipped with $2 \times$ Intel Xeon Gold 5317 CPU, $1{,}024$ GB main memory, and $2 \times$ A100 Nvidia Tesla GPU (80 GB, Ampere Architecture). In each experiment, a single Nvidia A100 GPU is used.

Models. We use the following LLMs in our experiments. From the `Llama` family, we use `Llama-3-8B-Instruct`, `Llama-3.1-8B-Instruct`, `Llama-3.2-1B-Instruct`, and `Llama-3.2-3B-Instruct`. From the `Qwen` family, we use `Qwen-2-0.5B-Instruct`, `Qwen-2-1.5B-Instruct`, `Qwen-2-7B-Instruct`, `Qwen-2.5-3B-Instruct`, and `Qwen-2.5-7B-Instruct`. Finally, we include three reasoning models distilled from `DeepSeek-R1`: `DeepSeek-R1-Distill-Llama-8B`, `DeepSeek-R1-Distill-Qwen-1.5B`, and `DeepSeek-R1-Distill-Qwen-7B`. Additionally, when using best-of-n sampling, we use the `ArmoRM-Llama3-8B-v0.1` reward model to score the generated outputs by the above models. We obtain all the models from the publicly available Hugging Face repository.[12]

Datasets. We use four datasets to evaluate the performance of all LLMs and simulate test-time compute games: `GPQA` [72], a multiple-choice STEM question-answering benchmark; `GSM8K` [68], a mathematical benchmark containing grade-school level problems; `AIME` [73], a mathematical reasoning benchmark with problems from the American Invitational Mathematics Examination; and `TruthfulQA` [74], which evaluates models on their ability to generate truthful answers to questions reflecting common misconceptions. The first three datasets have, for each query, a verifiable ground-truth answer that we use to evaluate accuracy. To measure quality for `TruthfulQA`, we consider the set of true and false reference answers that are provided for each question, and we use a `BLEURT`-based score [75] defined as the difference between the maximum similarity to the true reference answers and the maximum similarity to the false reference answers. Furthermore, given that `TruthfulQA` consists of open-ended questions where majority voting is not directly applicable, and involves short responses that do not benefit from extended chain-of-thought reasoning, we restrict experiments on this dataset to non-reasoning models using best-of-n. All datasets are obtained from the publicly available Hugging Face repository.[13][14][15][16]

Generation details. When generating model outputs and evaluating them on the above datasets, we adhere to the temperature settings recommended in the official Hugging Face model cards. That is, temperature 0.6 for the `Llama` family and the corresponding distilled reasoning model, and temperature 0.7 for the `Qwen` family and the corresponding distilled reasoning models. We disable top-p and top-k sampling when generating model outputs. To ensure robust evaluation and enable bootstrap resampling to estimate uncertainties, we generated 128 candidate responses per query for non-reasoning models and 32 for reasoning models. For the simulations, we used subsets of these pools (up to $\theta = 64$ for non-reasoning and the full distribution for reasoning quantiles). We report 95% bootstrapped confidence intervals for the qualities of all models. Our prompts and answer verification pipelines are adapted from established evaluation frameworks, specifically OpenCompass[17], EvalScope[18], and the LM Evaluation Harness[19].

Quality computation. For the datasets with verifiable ground truths (`GPQA`, `GSM8K`, and `AIME`), we linearly map average model accuracy to user quality $q_i(\theta)$ using a proportionality constant α, set to $\$0.008$ in our main experiments. We vary this constant in the additional experimental results in Appendix E. Our main results are robust to this linear, yet natural, choice. For instance, under the setting of Section 5, if users derive zero quality $q_i(\theta) = 0$ whenever the model's average accuracy is

[12] `https://huggingface.co`
[13] `https://huggingface.co/datasets/Idavidrein/gpqa`
[14] `https://huggingface.co/datasets/openai/gsm8k`
[15] `https://huggingface.co/datasets/Maxwell-Jia/AIME_2024`
[16] `https://huggingface.co/datasets/domenicrosati/TruthfulQA`
[17] `https://github.com/open-compass/opencompass`
[18] `https://github.com/modelscope/evalscope`
[19] `https://github.com/EleutherAI/lm-evaluation-harness`

below 70% and quality proportional to average accuracy otherwise, using the same constant $\$0.008$, then the price of anarchy of the test-time compute game is 1.18.

Test-time compute methods. In our experiments, we consider the following three test-time compute methods, each used to simulate a different test-time compute game $\mathcal{G}$ in Section 5:

- **Majority voting.** For each query, each model generates $\theta = \{2^0, \ldots, 2^6\}$ independent outputs. The final response of the model is taken as the most frequent response across the θ generations. The prices $p(\theta)$ and costs $c(\theta)$ are computed based on the total number of generated tokens across all θ outputs, averaged across queries.
- **Best-of-n [3].** For each query, the model generates $\theta = \{2^0, \ldots, 2^6\}$ independent outputs. The final response of the model is taken as the highest-scored response according to the scoring model `ArmoRM-Llama3-8B-v0.1`. Specifically, for `GSM8K`, `GPQA` and `AIME`, we use the default score provided by the reward model, while for `TruthfulQA`, we follow Cui et al. [76] and use the truthfulness attribute of the reward model. The prices $p(\theta)$ and costs $c(\theta)$ are computed based on the total number of generated tokens across all θ outputs, averaged across queries. We do not account for the cost of the reward model scoring in this computation. Empirically, we find that the additional overhead from scoring is small, contributing between approximately 0.5% and 1.2% of total inference time across models (e.g., ~0.8% for Qwen2-0.5B, ~0.5% for Qwen2-7B, ~1.2% for Llama-3.2-1B, and ~0.5% for Llama-3.1-8B), and thus omit it from the cost model.
- **Chain-of-thought [1].** Since the Hugging Face API does not allow explicit control over the level of test-time compute or reasoning used for chain-of-thought for the models we consider, we adopt the following heuristic approach. For each query, we generate 32 independent model outputs and then group these outputs into five quantile-based bins according to the number of reasoning tokens generated. More concretely, the first bin contains outputs whose reasoning-token counts fall within the lowest 20-th percentile, and analogously for the remaining bins. In this process, we discard any output in which the model generates fewer than 5 reasoning tokens or exceeds the maximum generation length of 2048 tokens (with 58.5%, 3.1%, and 11.1% of outputs retained on `GSM8K`, `AIME`, and `GPQA`, respectively).[20] Then, we compute the accuracy for each bin and average it over queries. While we emphasize that this approach is not equivalent to deterministically controlling for the reasoning level during chain-of-thought, it provides a reasonable first approximation of the relationship between reasoning effort—as measured by the number of reasoning tokens—and accuracy.

Test-time compute prices and costs. In our experiments in Section 5, we instantiate the test-time compute games $\mathcal{G}$ with each provider serving a different LLM. To determine the per-output token price for each provider/model, we refer to the Hugging Face list of inference providers[21] and, for each LLM, compute the average token price across the listed providers that offer access to that model. The resulting prices per million output tokens are as follows: \$0.1455, \$0.1245, \$0.10, and \$0.08 for `Llama-3-8B-Instruct`, `Llama-3.1-8B-Instruct`, `Llama-3.2-1B-Instruct`, and `Llama-3.2-3B-Instruct`, respectively; \$0.10, \$0.10, \$0.20, \$0.065, and \$0.1465 for `Qwen-2-0.5B`, `Qwen-2-1.5B`, `Qwen-2-7B`, `Qwen-2.5-3B`, and `Qwen-2.5-7B`, respectively; and \$0.125, \$0.10, and \$0.175 for the models `R1-D-Llama-8B`, `R1-D-Qwen-1.5B`, and `R1-D-Qwen-7B`, respectively. Then, we determine the per-output token generation cost by assuming that the per-token prices are 25% higher than the per-token costs, which simply corresponds to the per-token margin of providers. Lastly, for each player (provider) i in the game $\mathcal{G}$, each test-time method with compute level θ, and each dataset, we separately compute the quantities $p_i(\theta)$ (and correspondingly $c_i(\theta_i)$ using the margin 25%) by considering the average number of output tokens across all model outputs in the dataset and multiplying it by the per-token price. We additionally report inefficiency results for margins of 5% and 15% in Appendix E.3 to assess the sensitivity of our findings to this choice.

Dynamics of test-time compute games. To simulate the dynamics of a test-time compute game, we proceed as follows. First, we consider that all providers start with their lowest test-time compute level, that is, $\theta_i^1 = \min_{\theta \in \Theta} \theta, \forall i \in [N]$, which corresponds to using a single sample for best-of-n and

[20] As a result, the reported accuracies (*e.g.*, in Figure 5) reflect only the subset of outputs that pass this filtering step, and may therefore overestimate true model accuracy.

[21] `https://huggingface.co/docs/inference-providers/index`, consulted on December 30, 2025

majority voting, and generating responses in the first 20-th percentile in terms of reasoning tokens for chain-of-thought, see Appendix D. Then, at each iteration t, we update the providers' compute levels from $\boldsymbol{\theta}^t$ to $\boldsymbol{\theta}^{t+1}$ by randomly selecting a provider $i_t \in [N]$ who does not have maximum utility when the others select $\boldsymbol{\theta}^t_{-i_t}$. We keep $\boldsymbol{\theta}^{t+1}_{-i_t} = \boldsymbol{\theta}^t_{-i_t}$ fixed, and take $\theta^{t+1}_{i_t}$ to be the smallest compute that is larger than $\theta^t_{i_t}$ if $\theta^t_{i_t} \prec \operatorname{argmax}_\theta U_{i_t}(\theta; \boldsymbol{\theta}^t_{-i_t})$, or to be the largest compute that is smaller than $\theta^t_{i_t}$ if $\theta^t_{i_t} \succ \operatorname{argmax}_\theta U_{i_t}(\theta; \boldsymbol{\theta}^t_{-i_t})$. If all providers are already maximizing their utilities, the better-response dynamics have reached a Nash equilibrium. At each time step, we use the values offered by providers to determine their market share and compute the price of anarchy according to Eq. 8, see Appendix E.2.

Rationality parameter β. In Section 5 we focus on a rationality parameter $\beta = 1000$, and experiment with different values in Appendix E. We choose the value 1000 based real behavioral measurements. Indeed, the parameter β is, as discussed in Section 2, the standard rationality constant of quantal response equilibria, whose empirical value has been estimated in the behavioral literature to lie in the range $4 - - - 16$ when utilities are normalized to $[0, 1]$ [77, 64, 78, 79]. In our setting, user values are typically centered around 0.005 USD (see, *e.g.*, Figure 9), so after normalizing to our value scale this empirical range corresponds to β in the range $4/0.005 = 800$ to $16/0.005 = 3200$.

Licenses. The `Llama-3` models and `ArmoRM-Llama3-8B-v0.1` are licensed under the LLAMA 3.2 COMMUNITY LICENSE AGREEMENT.[22] The `Llama-3.1` models are licensed under the LLAMA 3.2 COMMUNITY LICENSE AGREEMENT.[23] The `Llama-3.2` models are licensed under the LLAMA 3.2 COMMUNITY LICENSE AGREEMENT.[24] The `Qwen` models are licensed under the Tongyi Qianwen LICENSE AGREEMENT.[25] The `DeepSeek-R1-Distill-Llama-8B`, `DeepSeek-R1-Distill-Qwen-1.5B` and `DeepSeek-R1-Distill-Qwen-7B` models are licensed under the MIT License. The `GPQA` and `AIME` datasets are licensed under Creative Commons Attribution 4.0, `GSM8K` is licensed under MIT License, and `TruthfulQA` is licensed under Apache License 2.0.

[22] `https://www.llama.com/llama3/license/`

[23] `https://www.llama.com/llama3_1/license/`

[24] `https://www.llama.com/llama3_2/license/`

[25] `https://github.com/QwenLM/Qwen/blob/main/Tongyi%20Qianwen%20LICENSE%20AGREEMENT/`

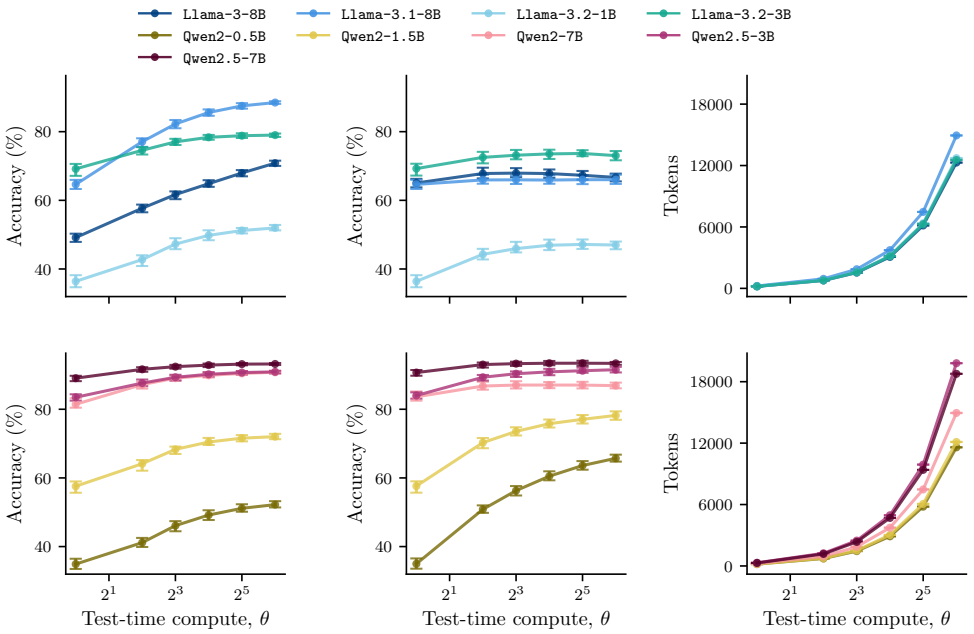
E Additional Experimental Results

This section contains additional experimental results that complement those discussed in Section 5. In Appendix E.1, we summarize the results (quality and number of generated tokens) of evaluating the LLMs served by each provider in the test-time compute games on `GSM8K`, `AIME`, `GPQA`, and `TruthfulQA`. Appendix E.2 shows additional outcomes of test-time compute games across all datasets and test-time compute methods. Further, Appendix E.3 compares the equilibrium outcomes of the test-time compute game $\mathcal{G}$ with the outcomes of the auction mechanism $\widetilde{\mathcal{G}}$ introduced in Section 4, across all datasets, test-time compute methods, profit margins (5%, 15%, and 25%), and accuracy-to-value conversion parameter values α.

E.1 Model evaluation

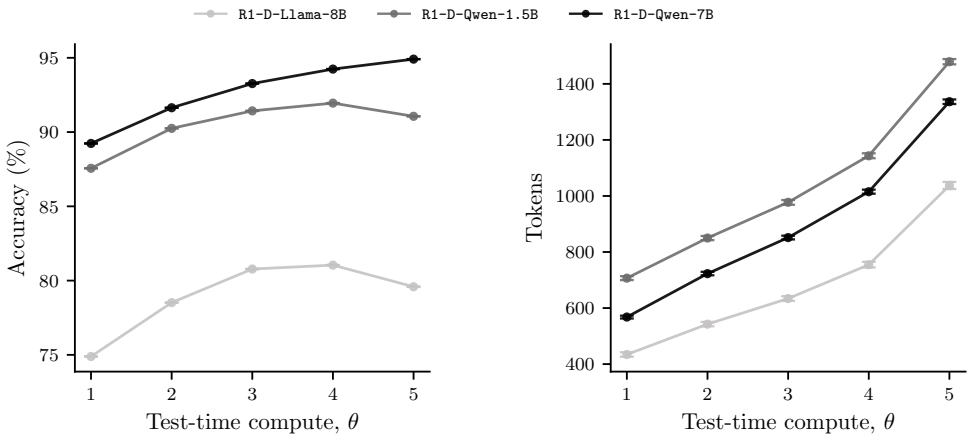
Here, we report, for each dataset (`GSM8K`, `AIME`, `GPQA`, and `TruthfulQA`) and each test-time compute method (majority voting and best-of-n for non-reasoning models, and chain-of-thought for reasoning models), the average quality of the models as a function of the test-time compute θ. We also report the average number of generated tokens to obtain the response to each query. See Appendix D for more details regarding the generation of model outputs.

E.1.1 Model Evaluation on GSM8K



(a) Majority Vote (b) Best-of-n (c) Number of Tokens

Figure 2: **Accuracy of `Llama` and `Qwen` models on GSM8K using majority voting and best-of-n.** Panels (a) and (b) show the average accuracy of various LLMs from the `Llama` and `Qwen` families over questions from the GSM8K dataset, where the responses of the models are obtained using majority voting or best-of-n, respectively. Panel (c) shows, as a function of the number of samples used to generate the response, the total number of tokens that the models generate to obtain the response to each question, averaged across questions. We show 95% confidence intervals obtained by bootstrapping 50 times.

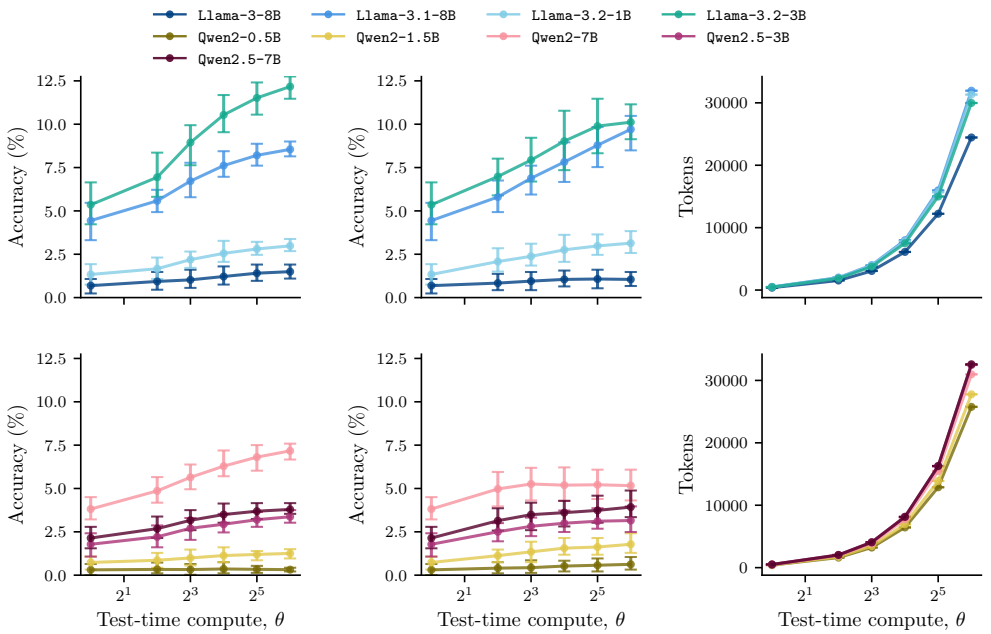


(a) Accuracy vs Reasoning Effort

(b) Tokens vs Reasoning Effort

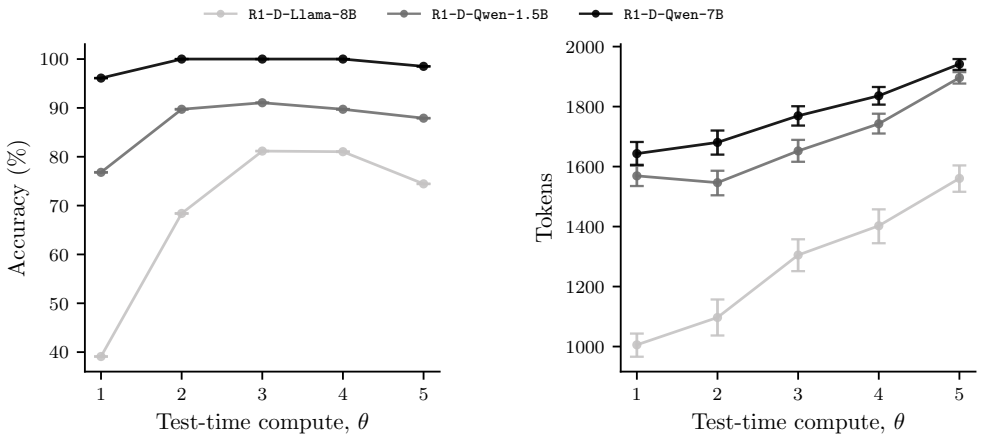
Figure 3: **Accuracy of reasoning models distilled from `DeepSeek-R1` on GSM8K using chain-of-thought.** Panel (a) shows the average accuracy of various reasoning models from the `Llama` and `Qwen` families distilled from `DeepSeek-R1` over questions from the GSM8K dataset. Here, the reasoning effort is defined by binning the model outputs into quantiles based on the number of reasoning tokens (see Appendix D). Panel (c) shows, as a function of the reasoning effort, the total number of tokens (including reasoning and non-reasoning tokens) that the models generate as a response to each question, averaged across questions. We show 95% confidence intervals obtained by bootstrapping 50 times. Refer to Appendix D for further details regarding the evaluation of the models. The reported accuracies are computed only over outputs in which the model generates at least 5 reasoning tokens and does not exceed the maximum generation length of 2048 tokens (see Appendix D for more details).

E.1.2 Model Evaluation on AIME



(a) Majority Vote (b) Best-of-n (c) Number of Tokens

Figure 4: **Accuracy of `Llama` and `Qwen` models on AIME using majority voting and best-of-n.** Panels (a) and (b) show the average accuracy of various LLMs from the `Llama` and `Qwen` families over questions from the AIME dataset, where the responses of the models are obtained using majority voting or best-of-n, respectively. Panel (c) shows, as a function of the number of samples used to generate the response, the total number of tokens that the models generate to obtain the response to each question, averaged across questions. Here, we compute the accuracies for majority voting, and best-of-n are computed across the same outputs, and hence both majority voting and best-of-n generate the exact same number of average tokens. We show 95% confidence intervals obtained by bootstrapping 50 times. Refer to Appendix D for further details regarding the evaluation of the models.

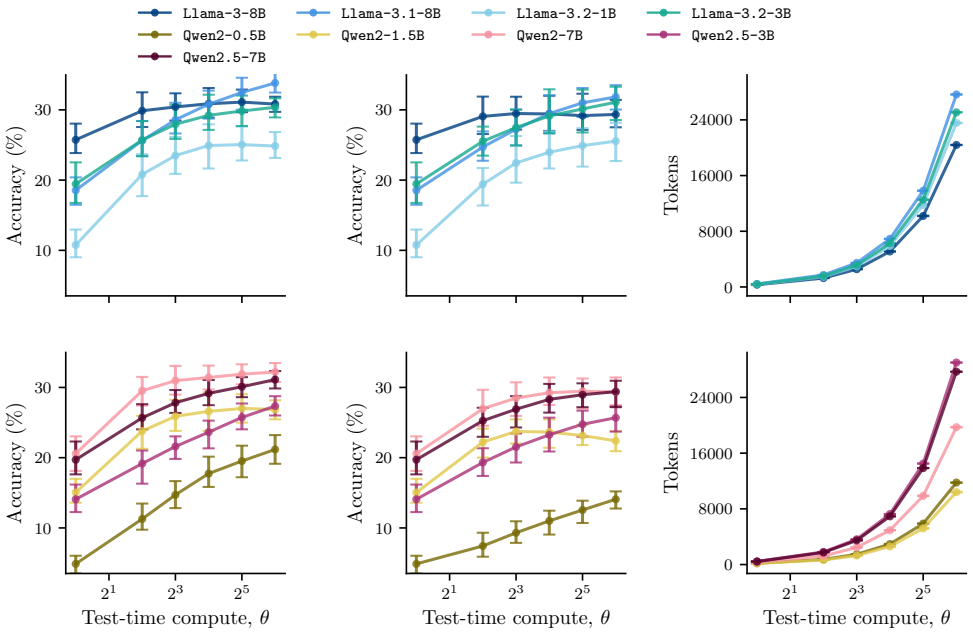


(a) Accuracy vs Reasoning Effort

(b) Tokens vs Reasoning Effort

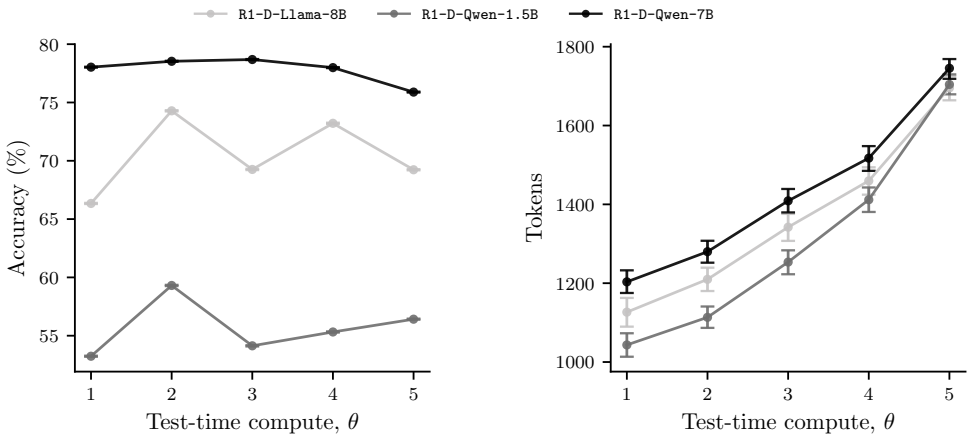
Figure 5: **Accuracy of reasoning models distilled from `DeepSeek-R1` on AIME using chain-of-thought.** Panel (a) shows the average accuracy of various reasoning models from the `Llama` and `Qwen` families distilled from `DeepSeek-R1` over questions from the AIME dataset. Here, the reasoning effort is defined by binning the model outputs into quantiles based on the number of reasoning tokens (see Appendix D). Panel (c) shows, as a function of the reasoning effort, the total number of tokens (including reasoning and non-reasoning tokens) that the models generate as a response to each question, averaged across questions. We show 95% confidence intervals obtained by bootstrapping 50 times. Refer to Appendix D for further details regarding the evaluation of the models. The reported accuracies are computed only over outputs in which the model generates at least 5 reasoning tokens and does not exceed the maximum generation length of 2048 tokens (see Appendix D for more details).

E.1.3 Model Evaluation on GPQA



(a) Majority Vote (b) Best-of-n (c) Number of Tokens

Figure 6: **Accuracy of `Llama` and `Qwen` models on GPQA using majority voting and best-of-n.** Panels (a) and (b) show the average accuracy of various LLMs from the `Llama` and `Qwen` families over questions from the GPQA dataset, where the responses of the models are obtained using majority voting or best-of-n, respectively. Panel (c) shows, as a function of the number of samples used to generate the response, the total number of tokens that the models generate to obtain the response to each question, averaged across questions. Here, we compute the accuracies for majority voting, and best-of-n are computed across the same outputs, and hence both majority voting and best-of-n generate the exact same number of average tokens. We show 95% confidence intervals obtained by bootstrapping 50 times. Refer to Appendix D for further details regarding the evaluation of the models.

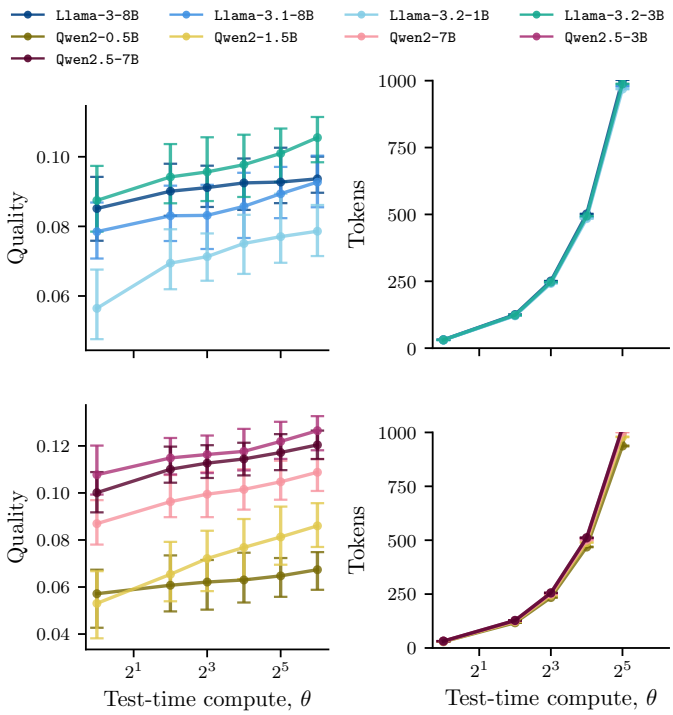


(a) Accuracy vs Reasoning Effort

(b) Tokens vs Reasoning Effort

Figure 7: **Accuracy of reasoning models distilled from `DeepSeek-R1` on GPQA using chain-of-thought.** Panel (a) shows the average accuracy of various reasoning models from the `Llama` and `Qwen` families distilled from `DeepSeek-R1` over questions from the GPQA dataset. Here, the reasoning effort is defined by binning the model outputs into quantiles based on the number of reasoning tokens (see Appendix D). Panel (c) shows, as a function of the reasoning effort, the total number of tokens (including reasoning and non-reasoning tokens) that the models generate as a response to each question, averaged across questions. We show 95% confidence intervals obtained by bootstrapping 50 times. Refer to Appendix D for further details regarding the evaluation of the models. The reported accuracies are computed only over outputs in which the model generates at least 5 reasoning tokens and does not exceed the maximum generation length of 2048 tokens (see Appendix D for more details).

E.1.4 Model Evaluation on TruthfulQA



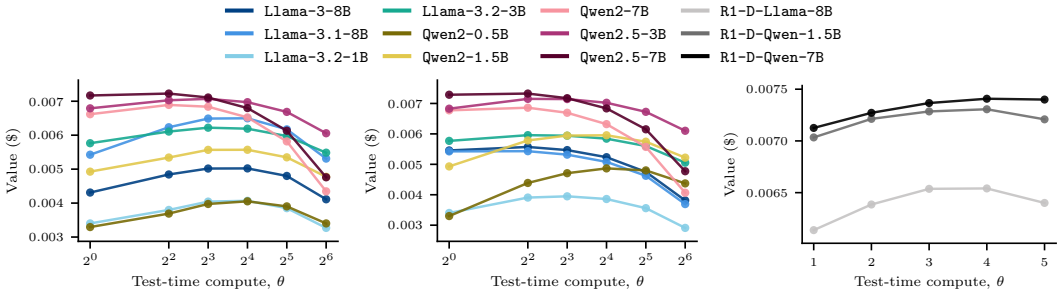
(a) Best-of-n (b) Number of Tokens

Figure 8: **Quality of `Llama` and `Qwen` models on TruthfulQA using best-of-n.** Panel (a) shows the average quality of various LLMs from the `Llama` and `Qwen` families over questions from the TruthfulQA dataset, where the responses of the models are obtained using best-of-n. To measure quality, we use a `BLEURT`-based score defined as the difference between the maximum similarity to a true reference answer and the maximum similarity to a false reference answer. Panel (b) shows, as a function of the number of samples used to generate the response, the total number of tokens that the models generate to obtain the response to each question, averaged across questions. We show 95% confidence intervals obtained by bootstrapping 50 times. Refer to Appendix D for further details regarding the evaluation of the models.

E.2 Dynamics and Equilibria of Test-Time Compute Games

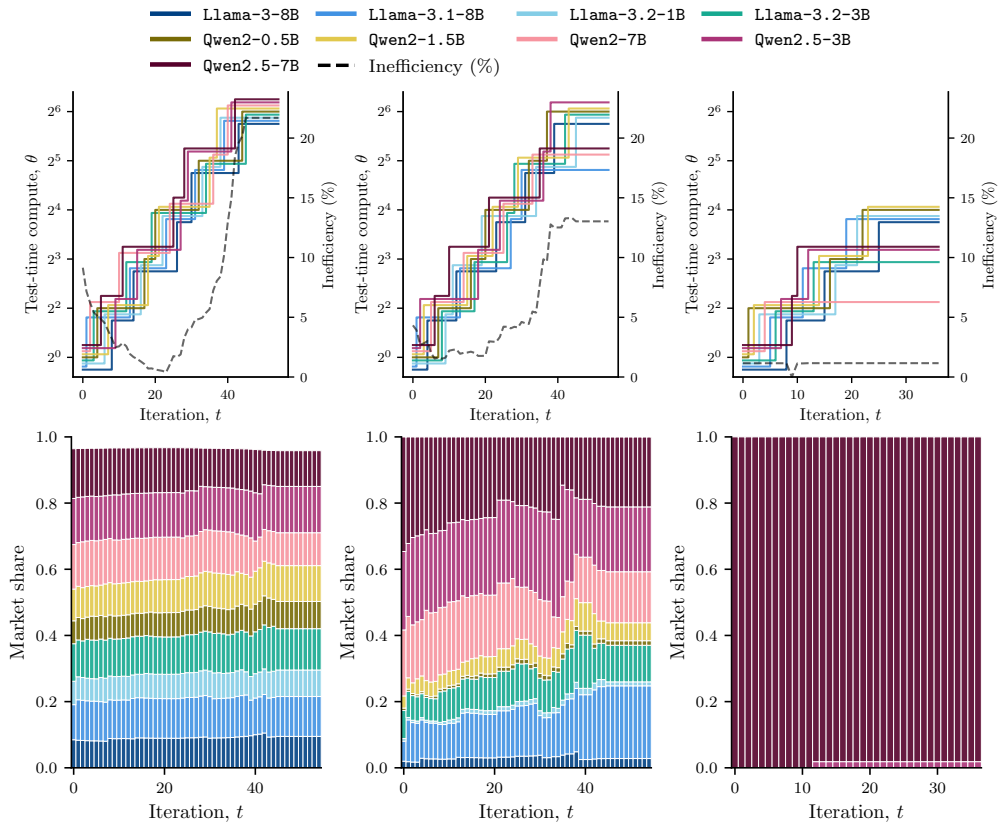
Here, for each dataset (`GSM8K`, `AIME`, `GPQA`, `TruthfulQA`) and each test-time compute method (majority voting and best-of-n for non-reasoning models, and chain-of-thought for reasoning models), we report: (i) the value offered by providers as a function of their test time compute, (ii) the compute levels and market shares when providers better-respond to each other, as a function of the iteration t, (iii) the evolution of the potential Φ (Eq. 7) when providers better-respond, and (iv), the inefficiency (PoA($\mathcal{G}$)-1) at equilibrium for each game as a function of user's rationality β. We also report the inefficiency at each time step, which corresponds to $\max_{\boldsymbol{\theta}} \mathrm{W}(\boldsymbol{\theta})/\mathrm{W}(\boldsymbol{\theta}^t)$ whenever $\boldsymbol{\theta}^t$ is not an equilibrium.

E.2.1 Test-Time Compute Equilibria on GSM8K



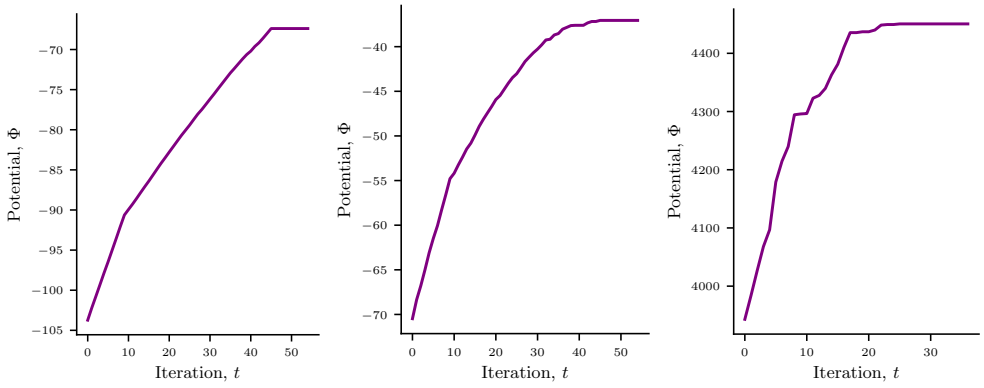
(a) Majority voting (b) Best-of-n (c) Chain-of-thought

Figure 9: **User values offered by providers in a test-time compute game on GSM8K.** The figure shows the user values $V_i(\theta)$ offered by providers in a test-time compute game $\mathcal{G}$, as a function of their test-time compute θ. Panel (a) and (b) correspond to games with $N = 9$ providers serving non-reasoning models from the `Llama` and `Qwen` families, where providers use, respectively, majority voting and best-of-n across θ samples. Panel (c) corresponds to a game with $N = 3$ providers serving reasoning models distilled from `DeepSeek-R1`, where θ represents reasoning effort, defined by binning the model outputs into quantiles based on the number of reasoning tokens (see Appendix D). In both games, providers serve queries Q from the `GSM8K` dataset, we set $\beta = 1000$ and consider that each (average) point of accuracy offers a value of $\$0.008$ to the users.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 10: **Dynamics of a test-time compute game using majority-voting.** The figure shows, for different levels of user rationality β, the better-response dynamics of a test-time compute game $\mathcal{G}$ where $N = 9$ providers sequentially select a test-time compute level that increases their utility. The upper panels show the compute levels θ selected by each provider and the resulting market inefficiency $(\text{PoA}(\mathcal{G}) - 1)$, and the lower panels show the market share of each provider. Here, all providers use majority-voting across θ samples as their test-time compute method to serve queries Q from the `GSM8K` dataset. We consider that providers operate with a margin of 25% between per-token price and per-token cost, and that each (average) point of accuracy offers a value of $\$0.008$ to the users.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 11: **Potential of a test-time compute game using majority-voting.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 10 where $N = 9$ providers sequentially select a test-time compute level that increases their utility.

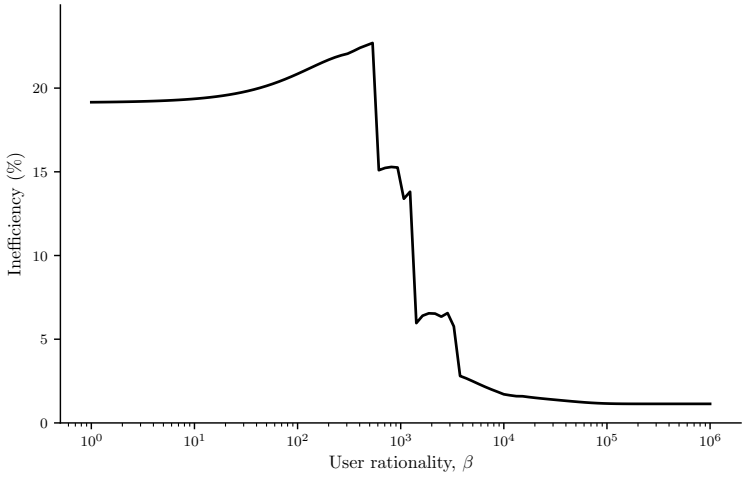
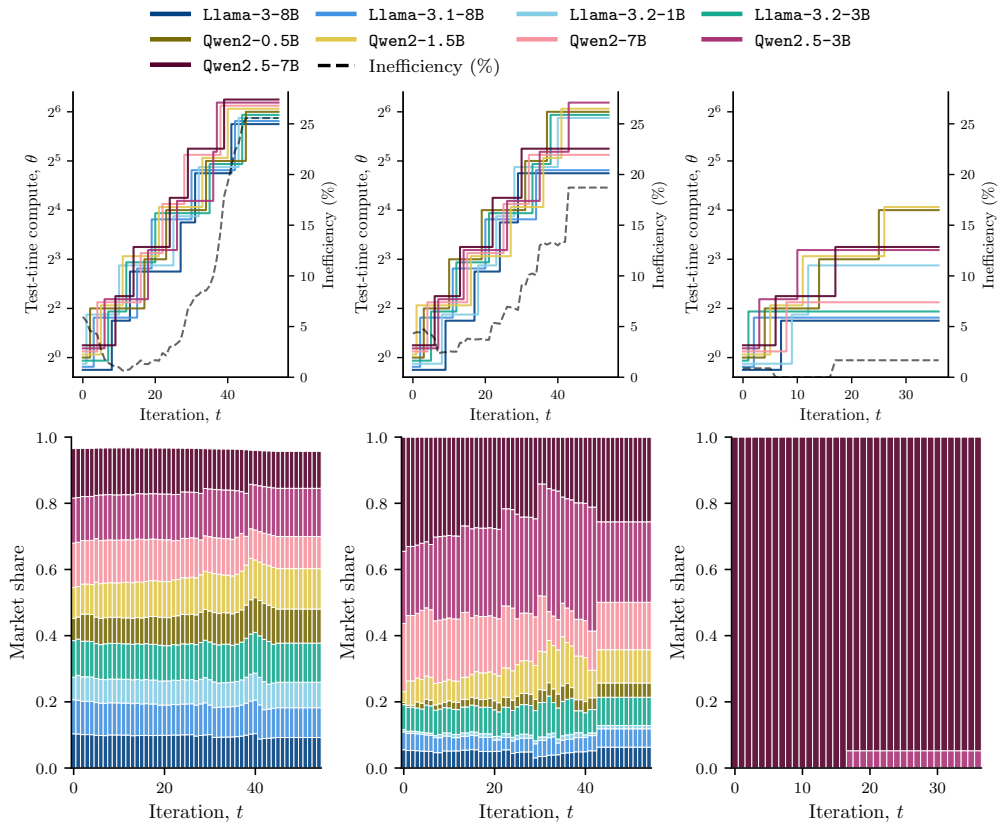
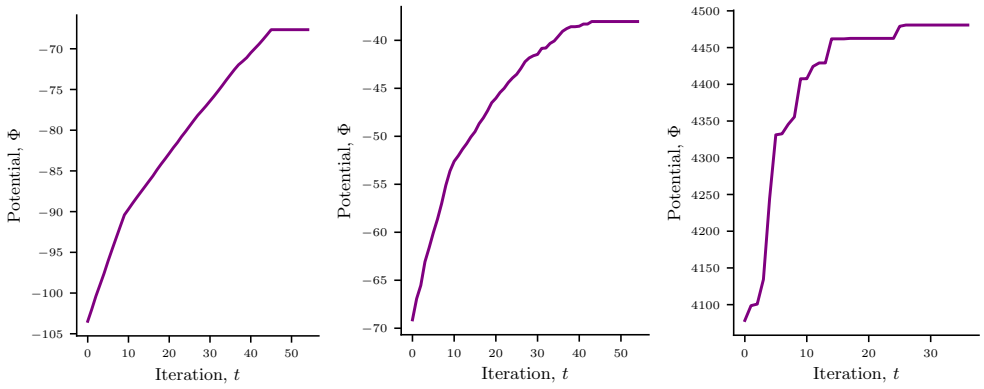


Figure 12: **Inefficiency of a test-time compute game using majority-voting.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 10, where $N = 9$ providers sequentially select a test-time compute level that increases their utility.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 13: **Better-response dynamics of a test-time compute game using best-of-n.** The upper panels show, for varying levels of user rationality β, better-response dynamics when providers serving models from the `Llama` and `Qwen` families use best-of-n to serve queries from the GSM8K dataset. The solid colored lines (left y-axis) represent the test-time compute θ selected by each provider at each iteration, corresponding to the number of samples used for best-of-n. The dashed black line (right y-axis) tracks the Market Inefficiency, defined as $(\text{PoA} - 1) \times 100$ (see Eq. 8). The lower panels show the evolution of the market share at each time step of the better-response dynamics. The initial compute level $\boldsymbol{\theta}^1$ is taken as the lowest possible compute. We apply a small vertical jitter to the strategy lines to distinguish overlapping providers and take a fixed profit margin of 25%.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 14: **Potential of a test-time compute game using best-of-n.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 13 where $N = 9$ providers sequentially select a test-time compute level that increases their utility.

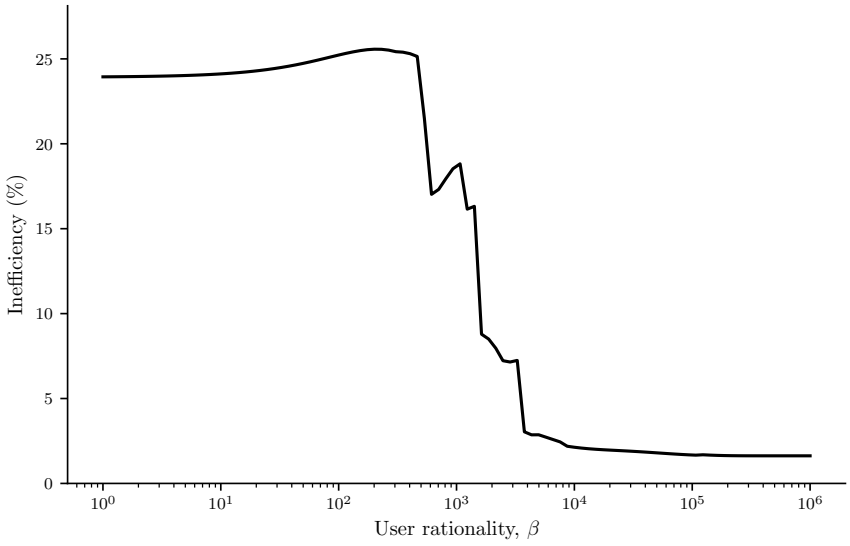
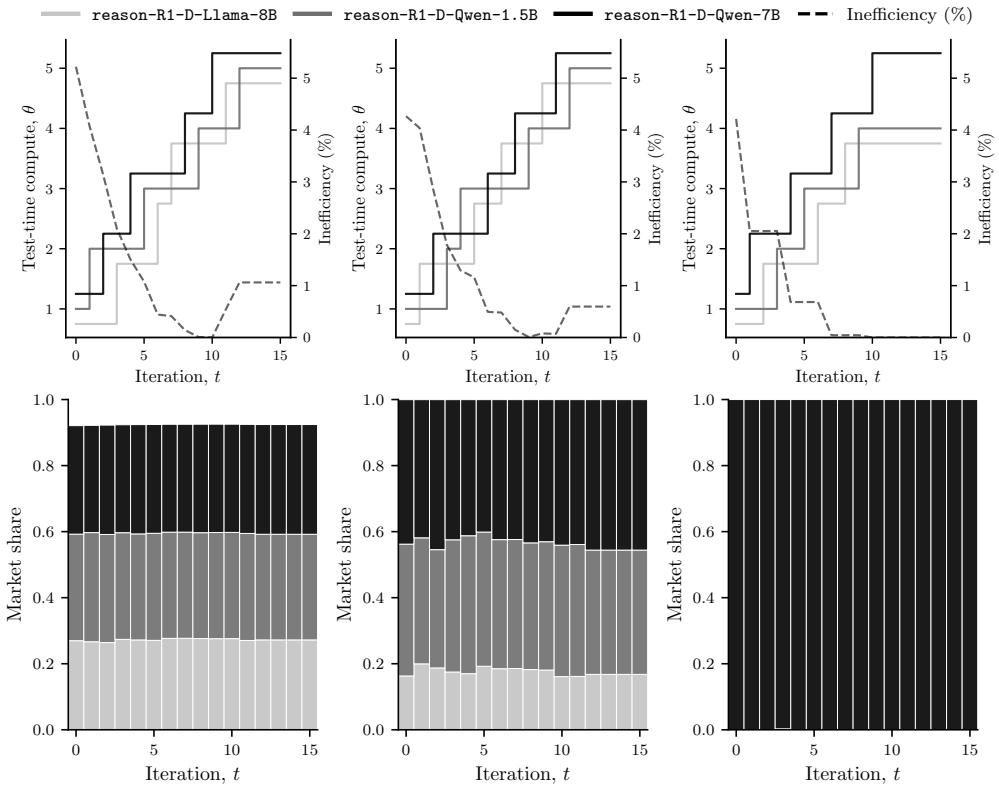
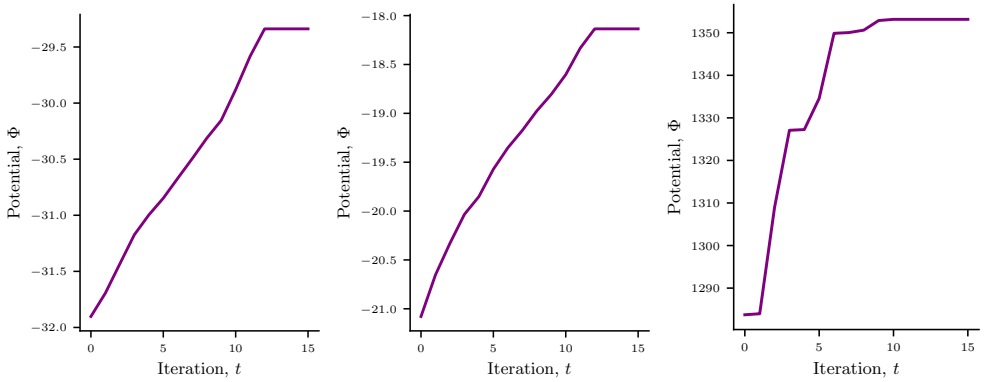


Figure 15: **Inefficiency of a test-time compute game using best-of-n.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 13, where $N = 9$ providers sequentially select a test-time compute level that increases their utility.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 16: **Better-response dynamics of a test-time compute game using CoT.** The upper panels show, for varying levels of user rationality β, better-response dynamics when providers serving models from the `Llama` and `Qwen` families distilled from `DeepSeek-R1` use chain-of-thought to serve queries from the GSM8K dataset. The solid colored lines (left y-axis) represent the test-time compute θ selected by each provider at each iteration, corresponding to the reasoning effort used. The dashed black line (right y-axis) tracks the Market Inefficiency, defined as $(\text{PoA} - 1) \times 100$ (see Eq. 8). The lower panels show the evolution of the market share at each time step of the better-response dynamics. The initial compute level $\boldsymbol{\theta}^1$ is taken as the lowest possible compute. We apply a small vertical jitter to the strategy lines to distinguish overlapping providers and take a fixed profit margin of 25%.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 17: **Potential of a test-time compute game using CoT.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 16 where $N = 3$ providers sequentially select a test-time compute level that increases their utility.

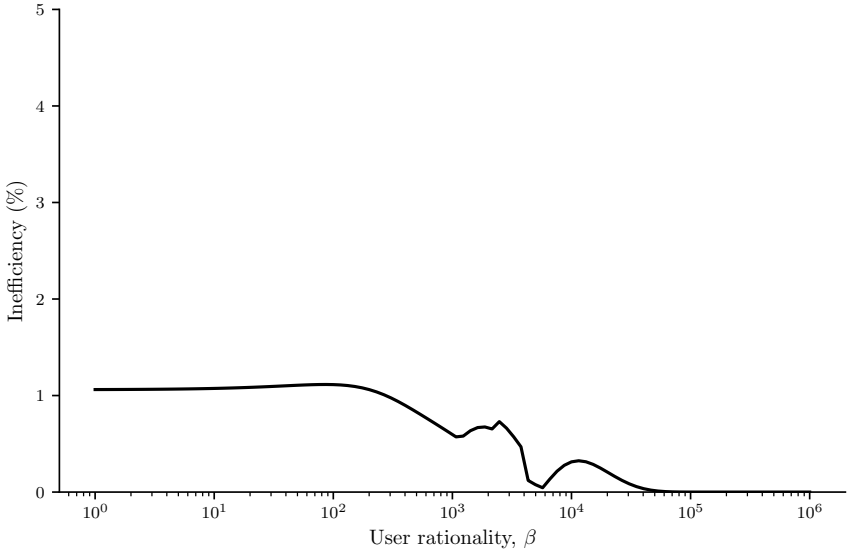
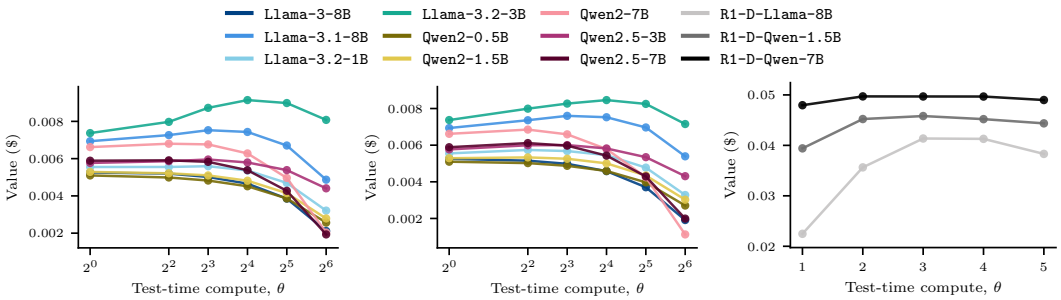


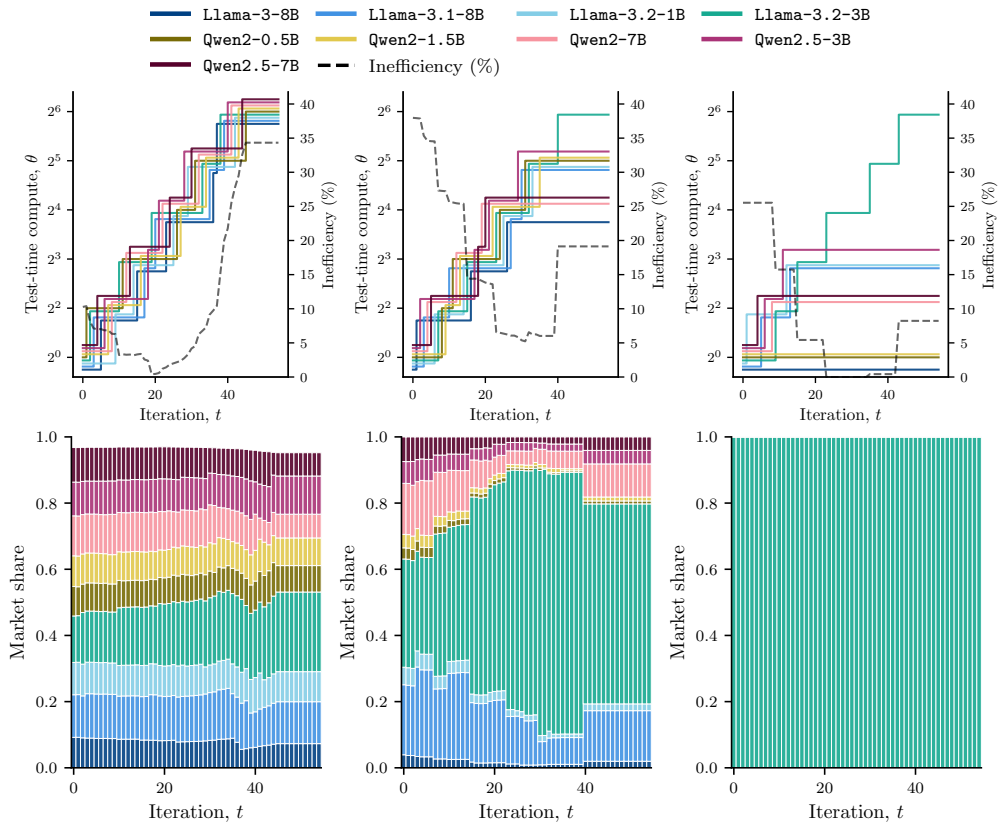
Figure 18: **Inefficiency of a test-time compute game using CoT.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 16, where $N = 3$ providers sequentially select a test-time compute level that increases their utility.

E.2.2 Test-Time Compute Equilibria on AIME



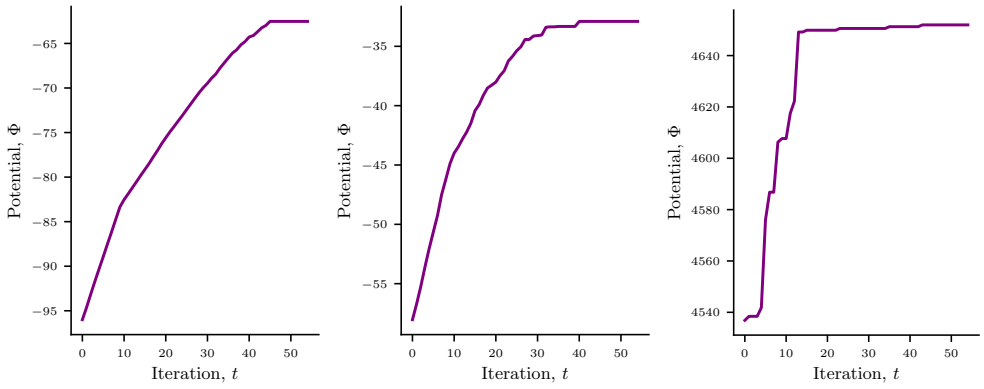
(a) Majority voting (b) Best-of-n (c) Chain-of-thought

Figure 19: **User values offered by providers in a test-time compute game on AIME.** The figure shows the user values $V_i(\theta)$ offered by providers in a test-time compute game $\mathcal{G}$, as a function of their test-time compute θ. Panel (a) and (b) correspond to games with $N = 9$ providers serving non-reasoning models from the `Llama` and `Qwen` families, where providers use, respectively, majority voting and best-of-n across θ samples. Panel (c) corresponds to a game with $N = 3$ providers serving reasoning models distilled from `DeepSeek-R1`, where θ represents reasoning effort, defined by binning the model outputs into quantiles based on the number of reasoning tokens (see Appendix D). In both games, providers serve queries Q from the `AIME` dataset, we set $\beta = 1000$ and consider that each (average) point of accuracy offers a value of $\$0.05$ to the users.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 20: **Dynamics of a test-time compute game using majority-voting.** The figure shows, for different levels of user rationality β, the better-response dynamics of a test-time compute game $\mathcal{G}$ where $N = 9$ providers sequentially select a test-time compute level that increases their utility. The upper panels show the compute levels θ selected by each provider and the resulting market inefficiency ($\text{PoA}(\mathcal{G}) - 1$), and the lower panels show the market share of each provider. Here, all providers use majority-voting across θ samples as their test-time compute method to serve queries Q from the `AIME` dataset. We consider that providers operate with a margin of 25% between per-token price and per-token cost, and that each (average) point of accuracy offers a value of $\$0.008$ to the users.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 21: **Potential of a test-time compute game using majority-voting.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 20 where $N = 9$ providers sequentially select a test-time compute level that increases their utility.

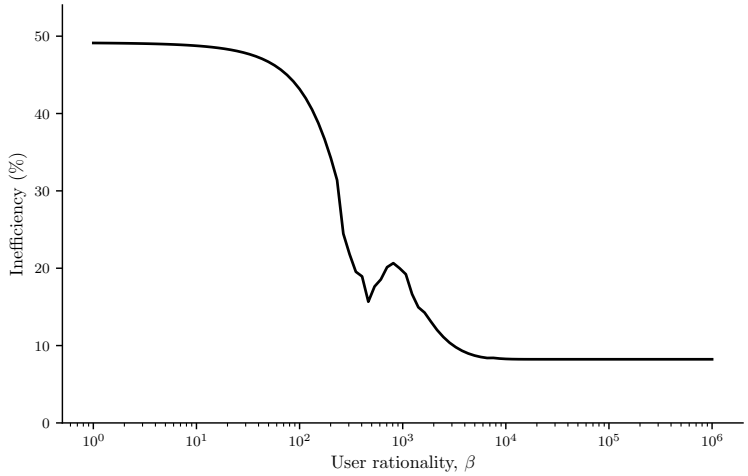
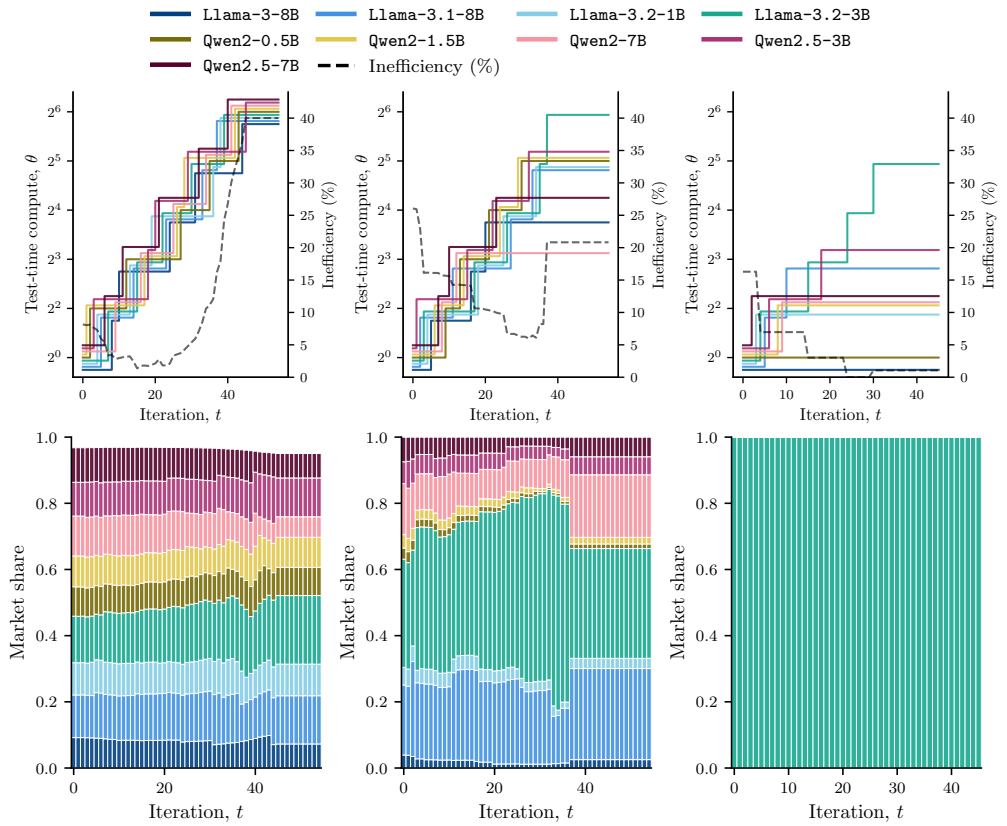
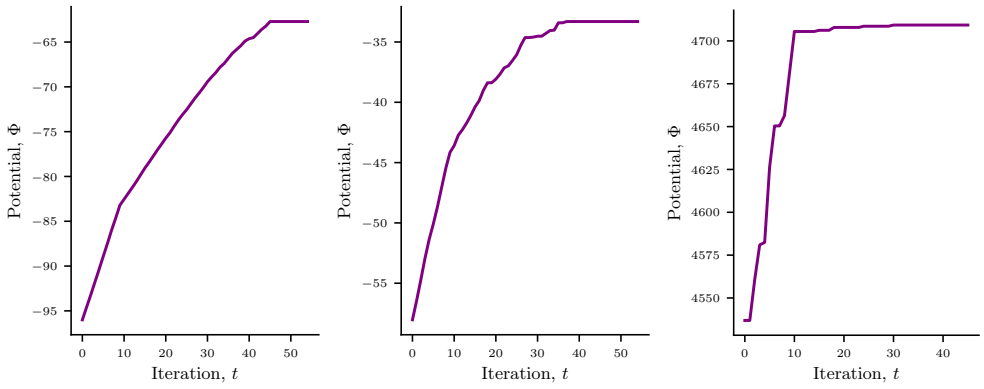


Figure 22: **Inefficiency of a test-time compute game using majority-voting.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 20, where $N = 9$ providers sequentially select a test-time compute level that increases their utility.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 23: **Better-response dynamics of a test-time compute game using best-of-n.** The upper panels show, for varying levels of user rationality β, better-response dynamics when providers serving models from the `Llama` and `Qwen` families use best-of-n to serve queries from the AIME dataset. The solid colored lines (left y-axis) represent the test-time compute θ selected by each provider at each iteration, corresponding to the number of samples used for best-of-n. The dashed black line (right y-axis) tracks the Market Inefficiency, defined as $(\text{PoA} - 1) \times 100$ (see Eq. 8). The lower panels show the evolution of the market share at each time step of the better-response dynamics. The initial compute level $\boldsymbol{\theta}^1$ is taken as the lowest possible compute. We apply a small vertical jitter to the strategy lines to distinguish overlapping providers and take a fixed profit margin of 25%.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 24: **Potential of a test-time compute game using best-of-n.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 23 where $N = 9$ providers sequentially select a test-time compute level that increases their utility.

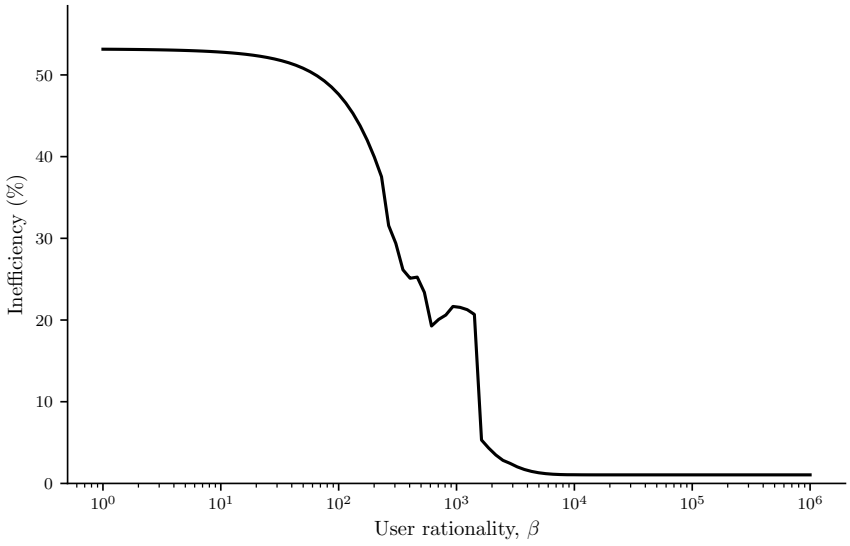
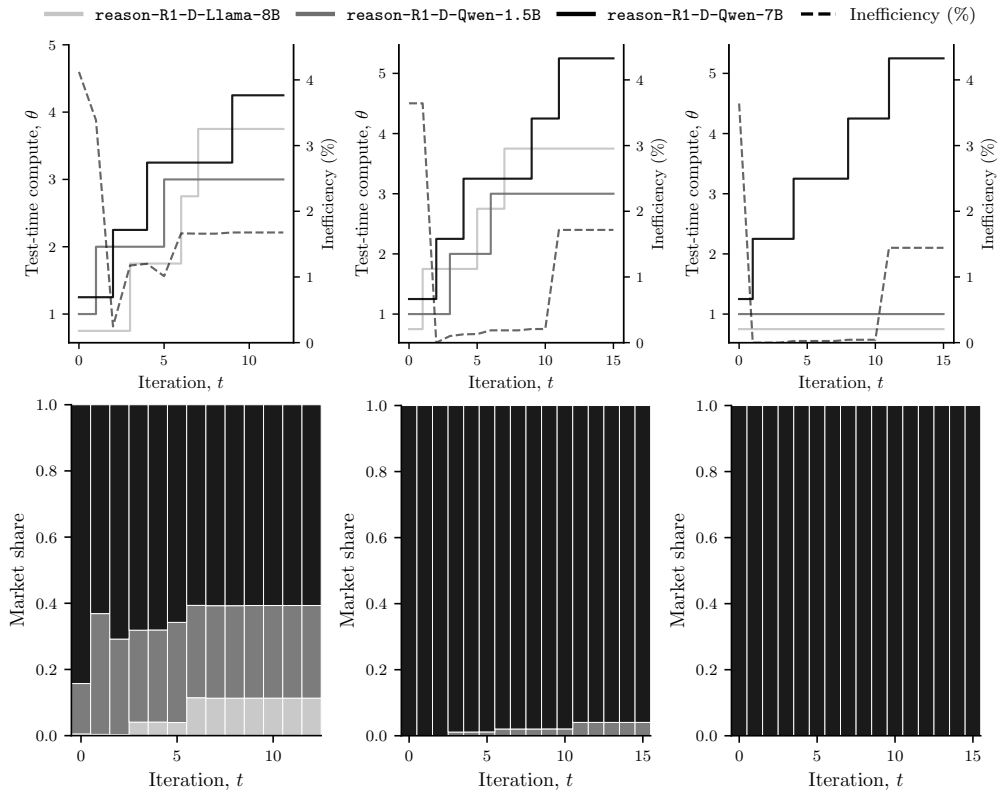
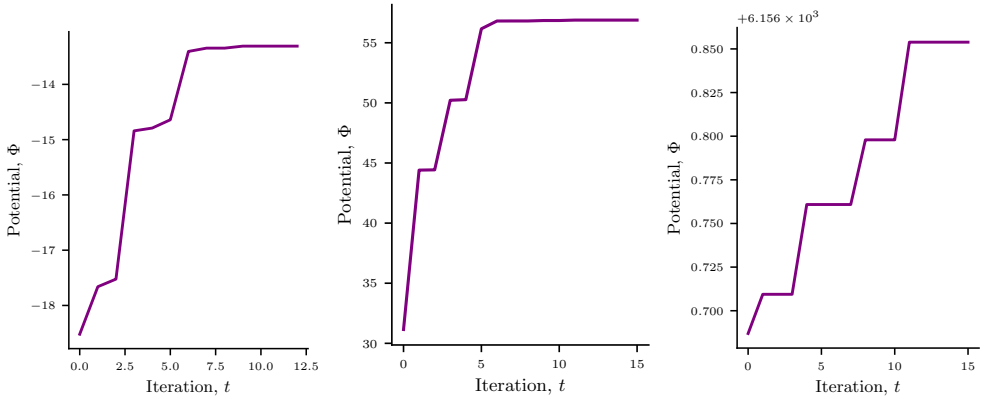


Figure 25: **Inefficiency of a test-time compute game using best-of-n.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 23, where $N = 9$ providers sequentially select a test-time compute level that increases their utility.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 26: **Better-response dynamics of a test-time compute game using CoT.** The upper panels show, for varying levels of user rationality β, better-response dynamics when providers serving models from the `Llama` and `Qwen` families distilled from `DeepSeek-R1` use chain-of-thought to serve queries from the AIME dataset. The solid colored lines (left y-axis) represent the test-time compute θ selected by each provider at each iteration, corresponding to the reasoning effort used. The dashed black line (right y-axis) tracks the Market Inefficiency, defined as $(\text{PoA} - 1) \times 100$ (see Eq. 8). The lower panels show the evolution of the market share at each time step of the better-response dynamics. The initial compute level $\boldsymbol{\theta}^1$ is taken as the lowest possible compute. We apply a small vertical jitter to the strategy lines to distinguish overlapping providers and take a fixed profit margin of 25%.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 27: **Potential of a test-time compute game using CoT.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 26 where $N = 3$ providers sequentially select a test-time compute level that increases their utility.

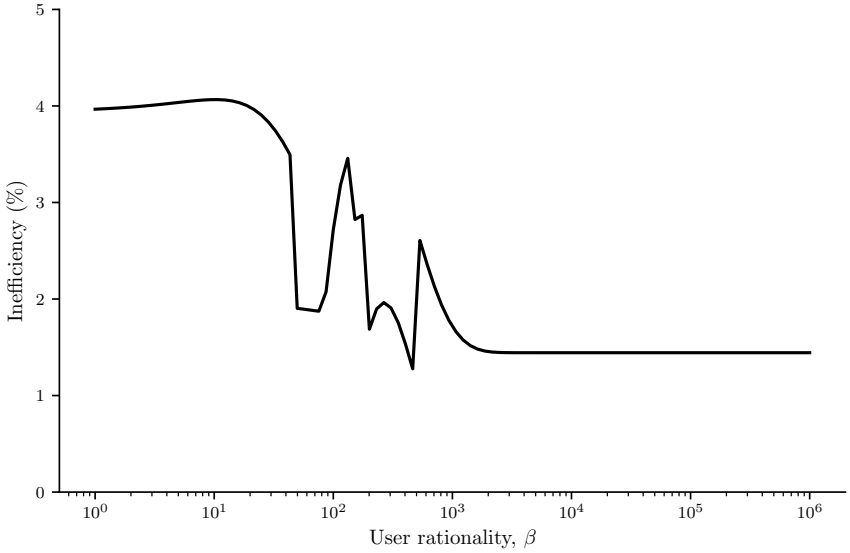
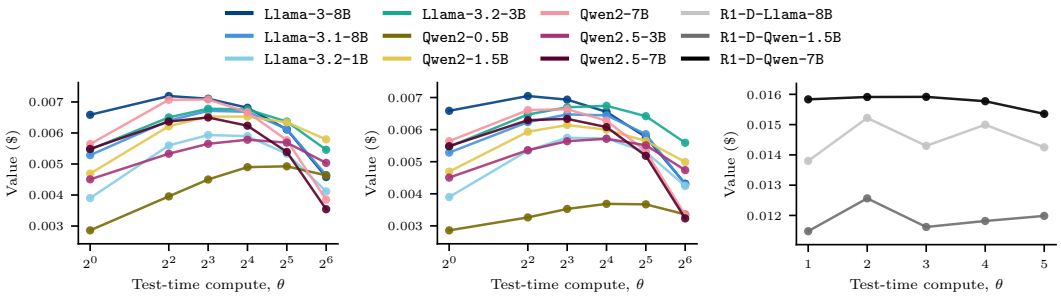


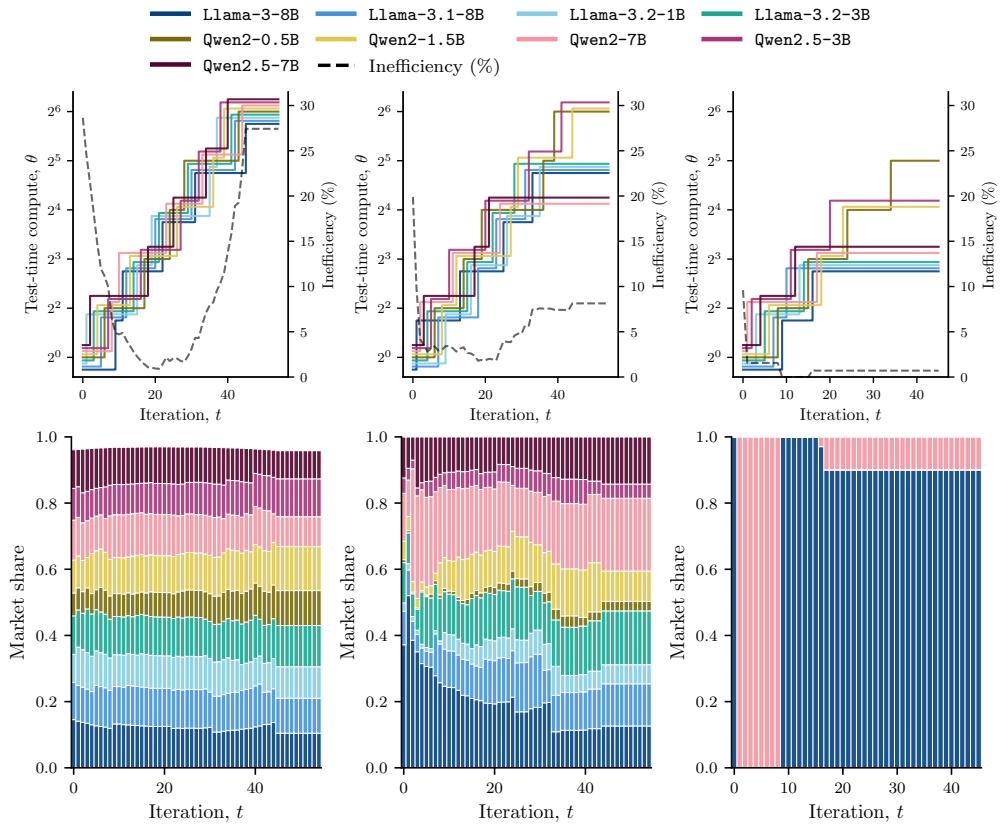
Figure 28: **Inefficiency of a test-time compute game using CoT.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 26, where $N = 3$ providers sequentially select a test-time compute level that increases their utility.

E.2.3 Test-Time Compute Equilibria on GPQA



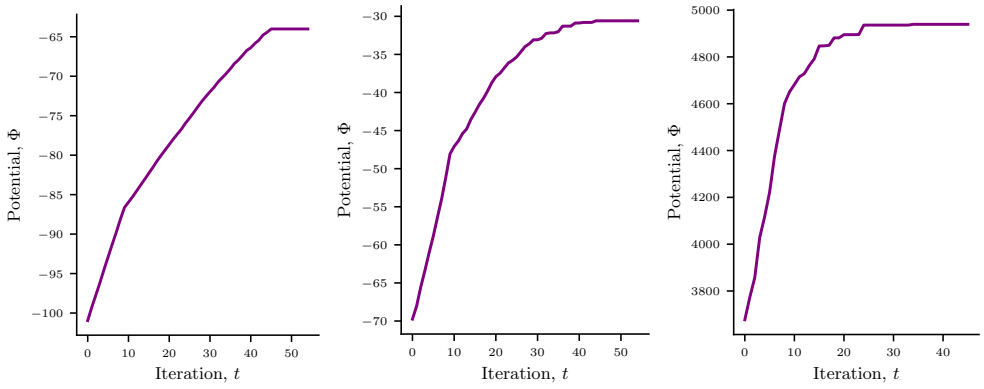
(a) Majority voting (b) Best-of-n (c) Chain-of-thought

Figure 29: **User values offered by providers in a test-time compute game on GPQA.** The figure shows the user values $V_i(\theta)$ offered by providers in a test-time compute game $\mathcal{G}$, as a function of their test-time compute θ. Panel (a) and (b) correspond to games with $N = 9$ providers serving non-reasoning models from the `Llama` and `Qwen` families, where providers use, respectively, majority voting and best-of-n across θ samples. Panel (c) corresponds to a game with $N = 3$ providers serving reasoning models distilled from `DeepSeek-R1`, where θ represents reasoning effort, defined by binning the model outputs into quantiles based on the number of reasoning tokens (see Appendix D). In both games, providers serve queries Q from the `GPQA` dataset, we set $\beta = 1000$ and consider that each (average) point of accuracy offers a value of $\$0.02$ to the users.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 30: **Dynamics of a test-time compute game using majority-voting.** The figure shows, for different levels of user rationality β, the better-response dynamics of a test-time compute game $\mathcal{G}$ where $N = 9$ providers sequentially select a test-time compute level that increases their utility. The upper panels show the compute levels θ selected by each provider and the resulting market inefficiency $(\text{PoA}(\mathcal{G}) - 1)$, and the lower panels show the market share of each provider. Here, all providers use majority-voting across θ samples as their test-time compute method to serve queries Q from the `GPQA` dataset. We consider that providers operate with a margin of 25% between per-token price and per-token cost, and that each (average) point of accuracy offers a value of $\$0.008$ to the users.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 31: **Potential of a test-time compute game using majority-voting.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 30 where $N = 9$ providers sequentially select a test-time compute level that increases their utility.

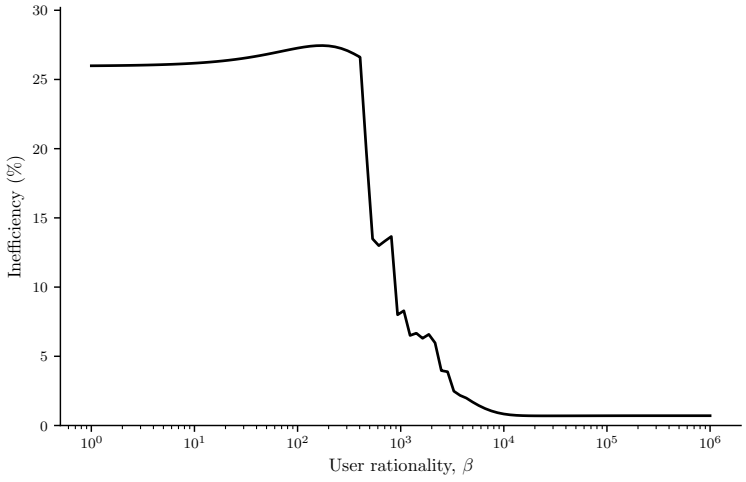
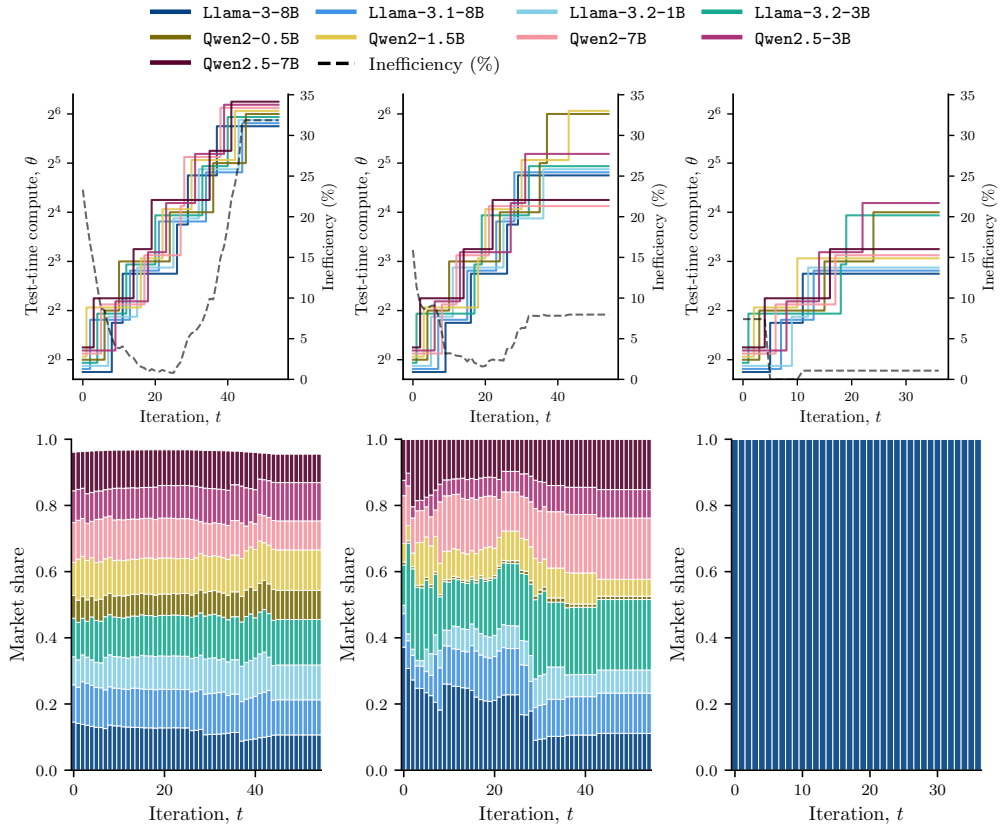
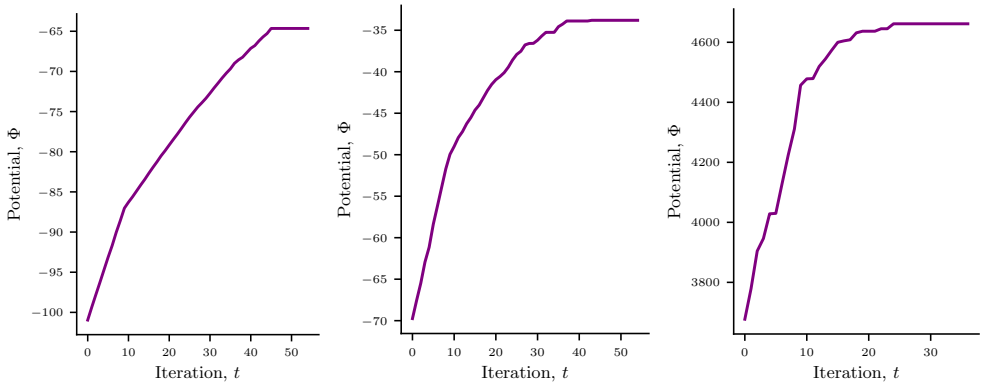


Figure 32: **Inefficiency of a test-time compute game using majority-voting.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 30, where $N = 9$ providers sequentially select a test-time compute level that increases their utility.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 33: **Better-response dynamics of a test-time compute game using best-of-n.** The upper panels show, for varying levels of user rationality β, better-response dynamics when providers serving models from the `Llama` and `Qwen` families use best-of-n to serve queries from the GPQA dataset. The solid colored lines (left y-axis) represent the test-time compute θ selected by each provider at each iteration, corresponding to the number of samples used for best-of-n. The dashed black line (right y-axis) tracks the Market Inefficiency, defined as $(\text{PoA} - 1) \times 100$ (see Eq. 8). The lower panels show the evolution of the market share at each time step of the better-response dynamics. The initial compute level $\boldsymbol{\theta}^1$ is taken as the lowest possible compute. We apply a small vertical jitter to the strategy lines to distinguish overlapping providers and take a fixed profit margin of 25%.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 34: **Potential of a test-time compute game using best-of-n.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 33 where $N = 9$ providers sequentially select a test-time compute level that increases their utility.

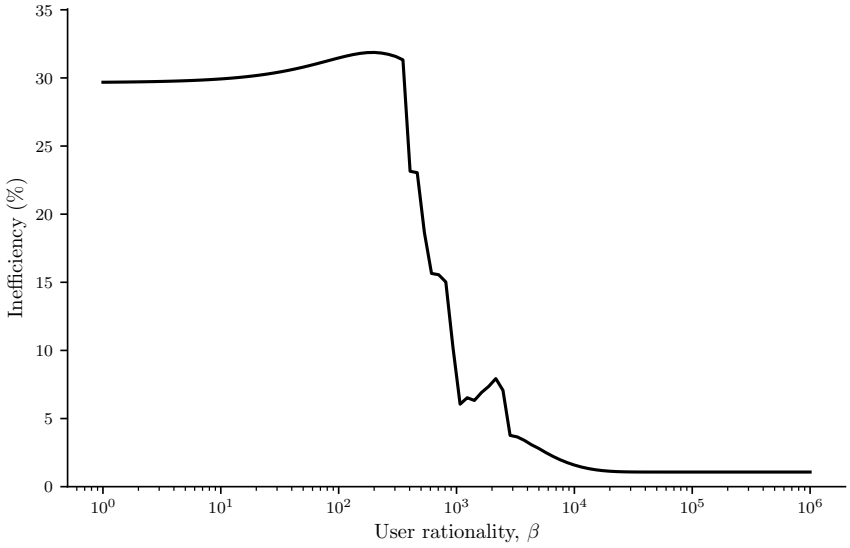
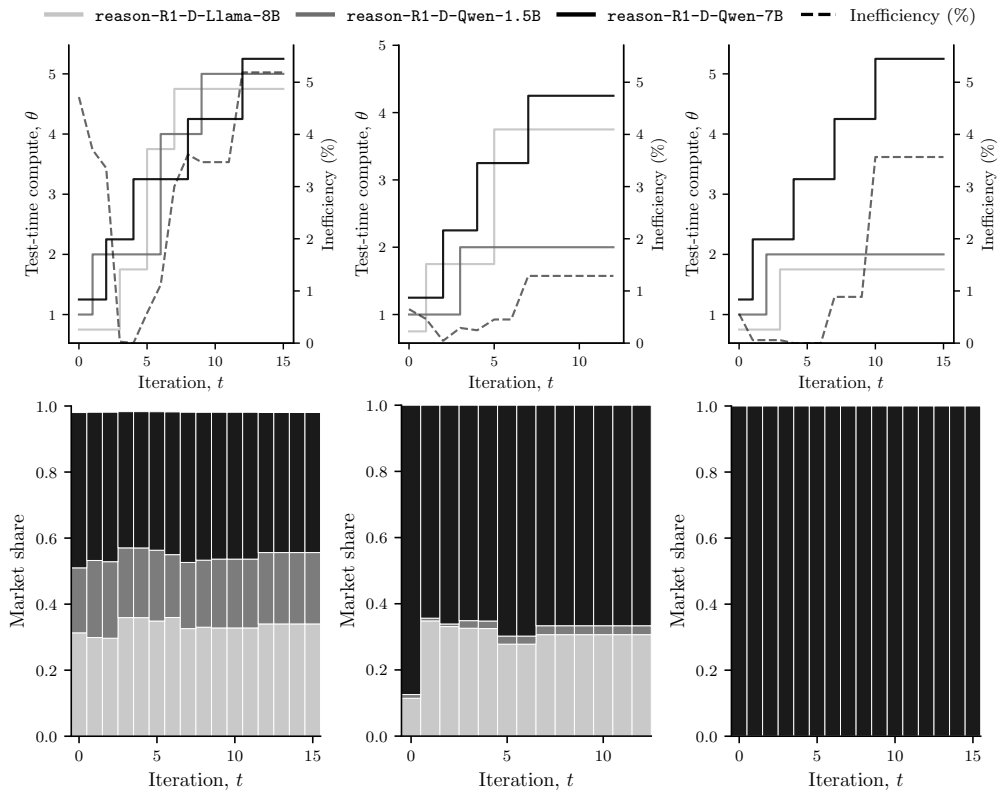
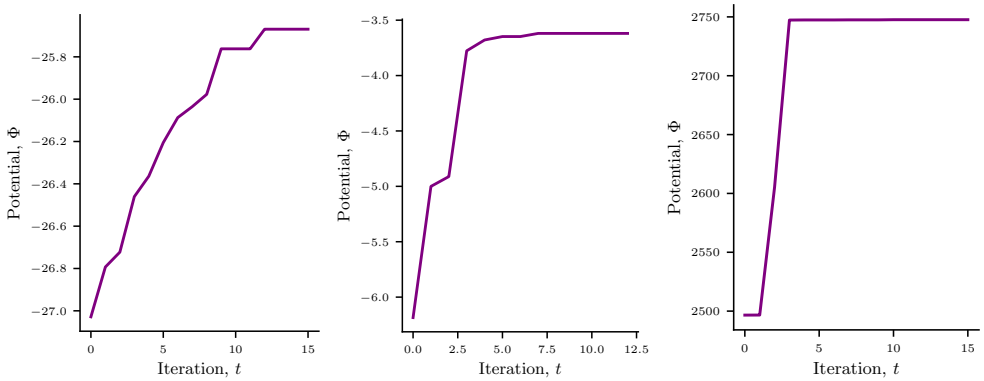


Figure 35: **Inefficiency of a test-time compute game using best-of-n.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 33, where $N = 9$ providers sequentially select a test-time compute level that increases their utility.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 36: **Better-response dynamics of a test-time compute game using CoT.** The upper panels show, for varying levels of user rationality β, better-response dynamics when providers serving models from the `Llama` and `Qwen` families distilled from `DeepSeek-R1` use chain-of-thought to serve queries from the GPQA dataset. The solid colored lines (left y-axis) represent the test-time compute θ selected by each provider at each iteration, corresponding to the reasoning effort used. The dashed black line (right y-axis) tracks the Market Inefficiency, defined as $(\text{PoA} - 1) \times 100$ (see Eq. 8). The lower panels show the evolution of the market share at each time step of the better-response dynamics. The initial compute level $\boldsymbol{\theta}^1$ is taken as the lowest possible compute. We apply a small vertical jitter to the strategy lines to distinguish overlapping providers and take a fixed profit margin of 25%.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 37: **Potential of a test-time compute game using CoT.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 36 where $N = 3$ providers sequentially select a test-time compute level that increases their utility.

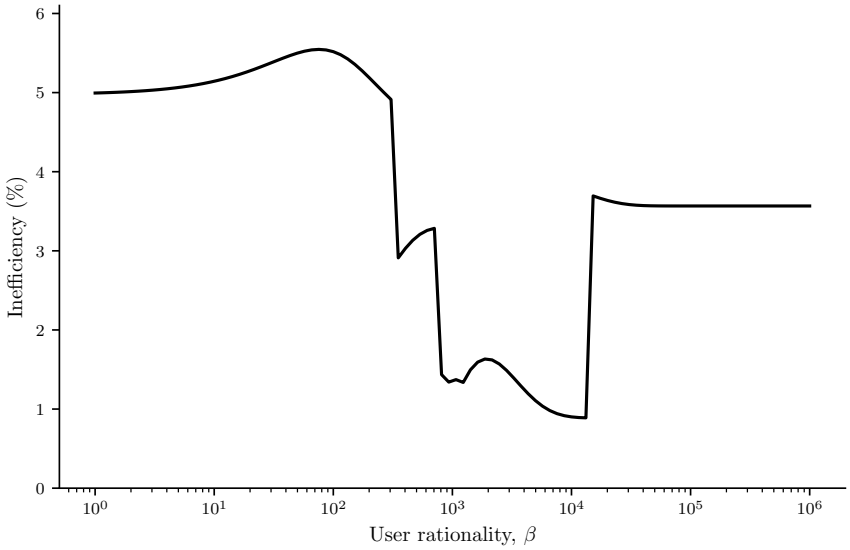
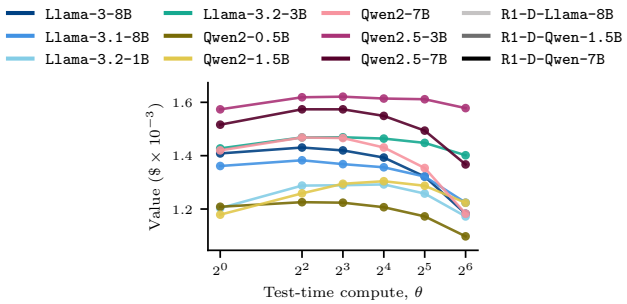


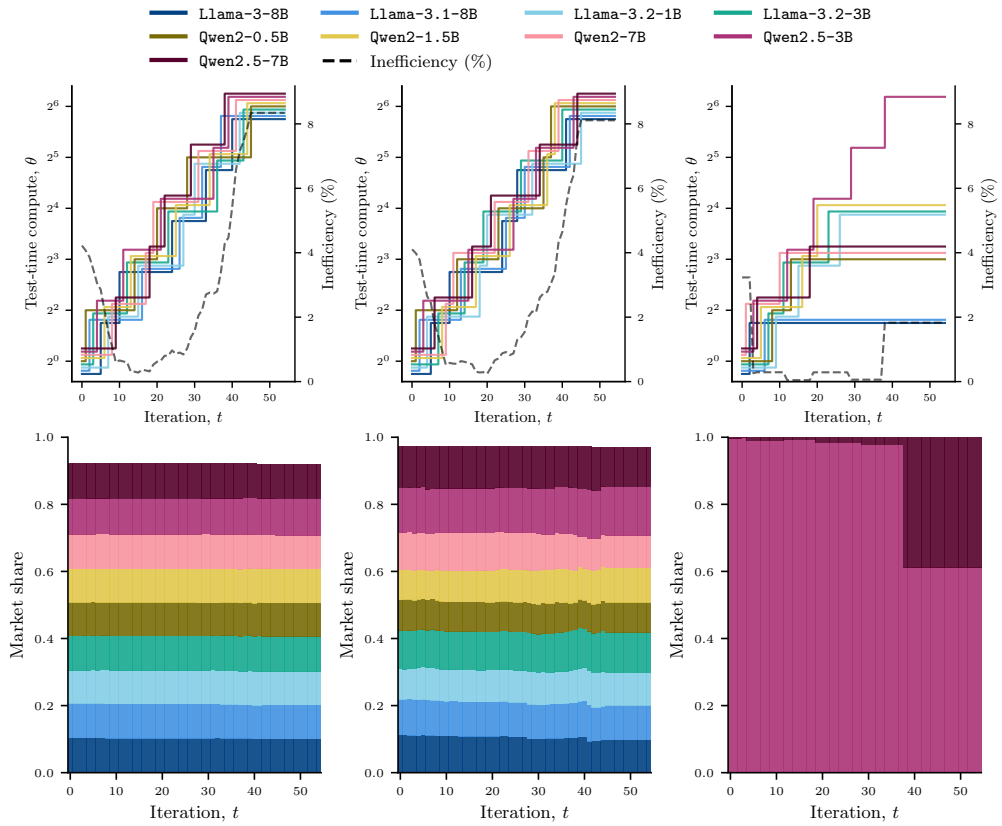
Figure 38: **Inefficiency of a test-time compute game using CoT.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 36, where $N = 3$ providers sequentially select a test-time compute level that increases their utility.

E.2.4 Test-Time Compute Equilibria on TruthfulQA



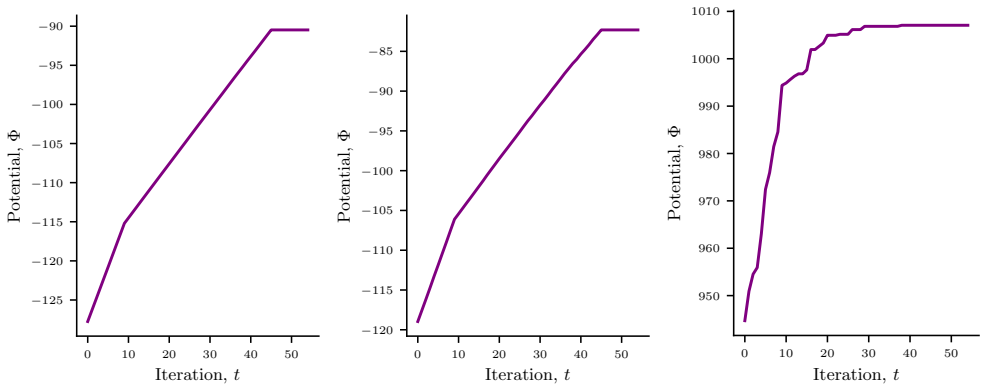
(a) Best-of-n

Figure 39: **User values offered by providers in a test-time compute game on TruthfulQA.** The figure shows the user values $V_i(\theta)$ offered by providers in a test-time compute game $\mathcal{G}$, as a function of their test-time compute θ. The panel corresponds to games with $N = 9$ providers serving non-reasoning models from the `Llama` and `Qwen` families, where providers use best-of-n across θ samples. Providers serve queries Q from the `TruthfulQA` dataset, we set $\beta = 1000$ and consider that each (average) point of quality (as reported in Figure 8) offers a value of $\$0.008$ to the users.



(a) $\beta = 2 \cdot 10^2$ (b) $\beta = 10^3$ (c) $\beta = 10^5$

Figure 40: **Better-response dynamics of a test-time compute game using best-of-n.** The upper panels show, for varying levels of user rationality β, better-response dynamics when providers serving models from the `Llama` and `Qwen` families use best-of-n to serve queries from the TruthfulQA dataset. The solid colored lines (left y-axis) represent the test-time compute θ selected by each provider at each iteration, corresponding to the number of samples used for best-of-n. The dashed black line (right y-axis) tracks the Market Inefficiency, defined as $(\text{PoA} - 1) \times 100$ (see Eq. 8). The lower panels show the evolution of the market share at each time step of the better-response dynamics. The initial compute level $\boldsymbol{\theta}^1$ is taken as the lowest possible compute. We apply a small vertical jitter to the strategy lines to distinguish overlapping providers and take a fixed profit margin of 25%.



(a) Potential ($\beta = 2 \cdot 10^2$) (b) Potential ($\beta = 10^3$) (c) Potential ($\beta = 10^5$)

Figure 41: **Potential of a test-time compute game using best-of-n.** The figure shows, for different levels of user rationality β, the evolution of the potential Φ (see Eq. 7) in the test-time compute games in Figure 40 where $N = 9$ providers sequentially select a test-time compute level that increases their utility.

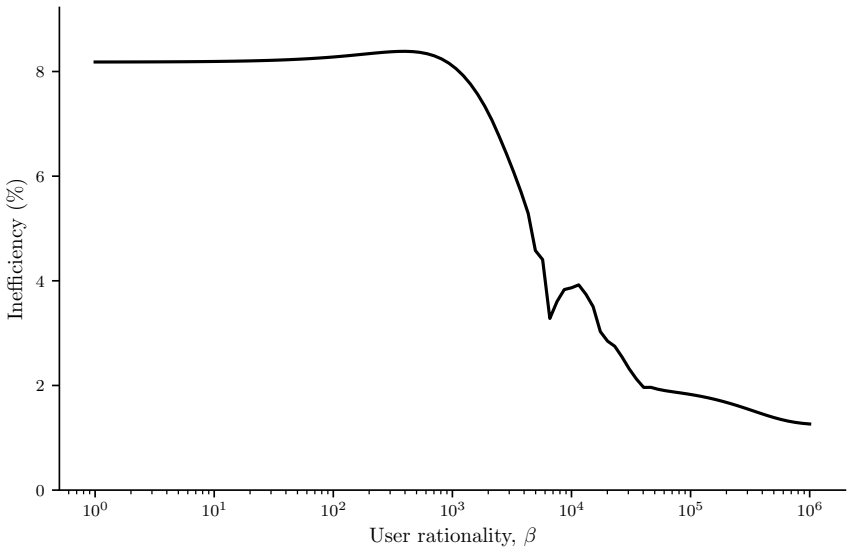


Figure 42: **Inefficiency of a test-time compute game using best-of-n.** The figure shows, as a function of users' rationality β, the inefficiency ($\text{PoA}(\mathcal{G}) - 1$) of the test-time compute game in Figure 40, where $N = 9$ providers sequentially select a test-time compute level that increases their utility.

E.3 Results of the auction mechanism

E.3.1 Results of the auction mechanism on GSM8K

Table 2: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on GSM8K using majority voting.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	16	4	2
User value	4.5	5.8	7.1	7.1
Price	1.6	1.2	0.34	0.30
Provider(s') utility	0.32	0.25	0.07	0.186
Social welfare	4.8	6.0	7.2	7.3

margin $= 0.15$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	16	4	2
User value	4.57	5.80	7.11	7.09
Price	1.62	1.24	0.34	0.31
Provider(s') utility	0.21	0.16	0.04	0.16
Social welfare	4.78	5.96	7.15	7.25

margin $= 0.05$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	16	4	2
User value	4.57	5.80	7.11	7.08
Price	1.62	1.24	0.34	0.32
Provider(s') utility	0.08	0.06	0.02	0.16
Social welfare	4.64	5.86	7.13	7.23

margin $= 0.25$ $\alpha = \$0.016$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	16	4	2
User value	11.41	13.15	14.56	14.33
Price	1.75	1.40	0.34	0.46
Provider(s') utility	0.35	0.28	0.07	0.32
Social welfare	11.75	13.43	14.63	14.66

margin $= 0.25$ $\alpha = \$0.004$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	16	2	1
User value	1.40	2.16	3.53	3.49
Price	1.53	1.15	0.17	0.12
Provider(s') utility	0.31	0.23	0.03	0.08
Social welfare	1.71	2.39	3.56	3.57

Table 3: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on GSM8K using best-of-n.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	16	4	2
User value	4.4	5.5	7.1	7.2
Price	1.6	1.2	0.33	0.32
Provider(s') utility	0.32	0.25	0.07	0.18
Social welfare	4.7	5.8	7.2	7.3

margin $= 0.15$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	16	4	2
User value	4.41	5.58	7.17	7.17
Price	1.60	1.26	0.33	0.33
Provider(s') utility	0.21	0.16	0.04	0.18
Social welfare	4.62	5.75	7.22	7.35

margin $= 0.05$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	16	4	2
User value	4.41	5.58	7.17	7.15
Price	1.60	1.26	0.33	0.34
Provider(s') utility	0.08	0.06	0.02	0.18
Social welfare	4.49	5.64	7.19	7.33

margin $= 0.25$ $\alpha = \$0.016$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	16	4	2
User value	11.05	13.15	14.69	14.48
Price	1.71	1.33	0.34	0.51
Provider(s') utility	0.34	0.27	0.07	0.38
Social welfare	11.40	13.41	14.76	14.86

margin $= 0.25$ $\alpha = \$0.004$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-7B`
Compute (θ)	32	32	2	1
User value	1.34	2.02	3.58	3.55
Price	1.52	1.18	0.17	0.11
Provider(s') utility	0.30	0.24	0.03	0.08
Social welfare	1.64	2.25	3.61	3.63

Table 4: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on GSM8K using chain-of-thought.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	4	4	4
User value	6.5	7.1	7.4	7.3
Price	0.16	0.17	0.23	0.30
Provider(s') utility	0.03	0.04	0.05	0.12
Social welfare	6.5	7.1	7.4	7.4

margin $= 0.15$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	4	4	4
User value	6.51	7.16	7.40	7.33
Price	0.16	0.18	0.23	0.30
Provider(s') utility	0.03	0.04	0.05	0.12
Social welfare	6.54	7.19	7.45	7.45

margin $= 0.05$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	4	4	4
User value	6.51	7.16	7.40	7.33
Price	0.16	0.18	0.23	0.30
Provider(s') utility	0.03	0.04	0.05	0.12
Social welfare	6.54	7.19	7.45	7.45

margin $= 0.25$ $\alpha = \$0.016$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	4	4	4
User value	14.06	14.72	15.03	14.75
Price	0.17	0.20	0.23	0.52
Provider(s') utility	0.03	0.04	0.05	0.33
Social welfare	14.09	14.76	15.08	15.08

margin $= 0.25$ $\alpha = \$0.004$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	4	3	3
User value	2.93	3.42	3.61	3.62
Price	0.15	0.17	0.17	0.17
Provider(s') utility	0.03	0.03	0.03	0.03
Social welfare	2.96	3.45	3.65	3.65

E.3.2 Results of the auction mechanism on AIME

Table 5: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on AIME using majority voting.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.05$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	32	32	8
User value	4.24	7.25	8.08	7.63
Price	3.09	2.08	2.40	2.11
Provider(s') utility	0.62	0.42	0.48	1.63
Social welfare	4.85	7.66	8.56	9.26

margin $= 0.15$ $\alpha = \$0.05$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	32	32	8
User value	4.24	7.25	8.08	7.59
Price	3.09	2.08	2.40	2.15
Provider(s') utility	0.40	0.27	0.31	1.63
Social welfare	4.64	7.52	8.39	9.22

margin $= 0.05$ $\alpha = \$0.05$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	32	32	8
User value	4.24	7.25	8.08	7.55
Price	3.09	2.08	2.40	2.19
Provider(s') utility	0.15	0.10	0.11	1.62
Social welfare	4.38	7.35	8.19	9.17

margin $= 0.25$ $\alpha = \$0.1$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	32	32	16
User value	13.63	18.36	18.56	16.06
Price	3.15	2.37	2.40	4.31
Provider(s') utility	0.63	0.47	0.48	3.35
Social welfare	14.26	18.83	19.04	19.41

margin $= 0.25$ $\alpha = \$0.025$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	16	16	8
User value	0.38	2.98	3.89	3.61
Price	2.93	1.17	1.20	1.26
Provider(s') utility	0.59	0.23	0.24	0.78
Social welfare	0.96	3.22	4.13	4.39

Table 6: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on AIME using best-of-n.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.05$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	32	16	8
User value	3.95	6.55	8.25	7.72
Price	3.09	1.73	1.20	1.34
Provider(s') utility	0.62	0.35	0.24	0.86
Social welfare	4.57	6.90	8.49	8.58

margin $= 0.15$ $\alpha = \$0.05$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	32	16	8
User value	3.95	6.55	8.25	7.67
Price	3.09	1.73	1.20	1.40
Provider(s') utility	0.40	0.23	0.16	0.87
Social welfare	4.36	6.78	8.41	8.54

margin $= 0.05$ $\alpha = \$0.05$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	32	16	8
User value	3.95	6.55	8.25	7.62
Price	3.09	1.73	1.20	1.44
Provider(s') utility	0.15	0.08	0.06	0.87
Social welfare	4.10	6.64	8.31	8.49

margin $= 0.25$ $\alpha = \$0.1$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	32	32	16
User value	12.56	16.28	16.71	16.32
Price	3.21	2.19	2.40	2.58
Provider(s') utility	0.64	0.44	0.48	1.62
Social welfare	13.20	16.72	17.19	17.94

margin $= 0.25$ $\alpha = \$0.025$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3.2-3B`
Compute (θ)	32	16	8	8
User value	0.30	2.56	3.93	3.65
Price	2.93	1.32	0.60	0.88
Provider(s') utility	0.59	0.26	0.12	0.40
Social welfare	0.88	2.82	4.05	4.05

Table 7: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on AIME using chain-of-thought.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.05$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	3	4	4	1
User value	47.64	48.86	48.99	45.84
Price	0.26	0.33	0.34	4.16
Provider(s') utility	0.05	0.07	0.07	3.92
Social welfare	47.70	48.92	49.06	49.76

margin $= 0.15$ $\alpha = \$0.05$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	3	4	4	1
User value	47.64	48.86	48.99	45.84
Price	0.26	0.33	0.34	4.16
Provider(s') utility	0.05	0.07	0.07	3.92
Social welfare	47.70	48.92	49.06	49.76

margin $= 0.05$ $\alpha = \$0.05$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	3	4	4	1
User value	47.64	48.86	48.99	45.84
Price	0.26	0.33	0.34	4.16
Provider(s') utility	0.05	0.07	0.07	3.92
Social welfare	47.70	48.92	49.06	49.76

margin $= 0.25$ $\alpha = \$0.1$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	3	4	4	1
User value	97.90	98.31	98.32	91.82
Price	0.29	0.34	0.34	8.18
Provider(s') utility	0.06	0.07	0.07	7.95
Social welfare	97.96	98.38	98.38	99.76

margin $= 0.25$ $\alpha = \$0.025$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	3	4	1
User value	22.72	24.37	24.32	22.86
Price	0.26	0.30	0.34	2.14
Provider(s') utility	0.05	0.06	0.07	1.91
Social welfare	22.77	24.43	24.39	24.76

E.3.3 Results of the auction mechanism on GPQA

Table 8: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on GPQA using majority voting.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.02$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3-8B`
Compute (θ)	32	8	4	2
User value	4.53	6.13	7.10	7.18
Price	2.33	1.21	0.38	0.20
Provider(s') utility	0.47	0.24	0.08	0.05
Social welfare	4.99	6.37	7.18	7.23

margin $= 0.15$ $\alpha = \$0.02$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3-8B`
Compute (θ)	32	8	4	2
User value	4.53	6.13	7.10	7.15
Price	2.33	1.21	0.38	0.23
Provider(s') utility	0.30	0.16	0.05	0.07
Social welfare	4.83	6.29	7.15	7.22

margin $= 0.05$ $\alpha = \$0.02$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3-8B`
Compute (θ)	32	8	4	2
User value	4.53	6.13	7.10	7.11
Price	2.33	1.21	0.38	0.27
Provider(s') utility	0.11	0.06	0.02	0.10
Social welfare	4.64	6.19	7.12	7.20

margin $= 0.25$ $\alpha = \$0.04$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2-7B`
Compute (θ)	32	16	4	4
User value	11.82	13.47	14.66	14.65
Price	2.51	1.54	0.49	0.50
Provider(s') utility	0.50	0.31	0.10	0.10
Social welfare	12.32	13.78	14.75	14.75

margin $= 0.25$ $\alpha = \$0.01$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3-8B`
Compute (θ)	32	8	2	2
User value	1.08	2.43	3.50	3.46
Price	2.18	1.13	0.19	0.23
Provider(s') utility	0.44	0.23	0.04	0.08
Social welfare	1.52	2.65	3.54	3.54

Table 9: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on GPQA using best-of-n.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.02$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3-8B`
Compute (θ)	32	16	4	2
User value	4.15	5.94	6.93	6.84
Price	2.35	1.15	0.37	0.39
Provider(s') utility	0.47	0.23	0.07	0.24
Social welfare	4.62	6.17	7.01	7.08

margin $= 0.15$ $\alpha = \$0.02$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3-8B`
Compute (θ)	32	16	4	2
User value	4.15	5.94	6.93	6.81
Price	2.35	1.15	0.37	0.43
Provider(s') utility	0.31	0.15	0.05	0.26
Social welfare	4.46	6.09	6.98	7.07

margin $= 0.05$ $\alpha = \$0.02$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3-8B`
Compute (θ)	32	16	4	2
User value	4.15	5.94	6.93	6.77
Price	2.35	1.15	0.37	0.47
Provider(s') utility	0.11	0.05	0.02	0.29
Social welfare	4.27	5.99	6.95	7.06

margin $= 0.25$ $\alpha = \$0.04$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3-8B`
Compute (θ)	32	8	4	2
User value	11.20	13.07	14.24	14.09
Price	2.54	1.50	0.37	0.38
Provider(s') utility	0.51	0.30	0.07	0.23
Social welfare	11.71	13.37	14.31	14.32

margin $= 0.25$ $\alpha = \$0.01$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Llama-3-8B`
Compute (θ)	32	16	4	2
User value	0.89	2.31	3.28	3.27
Price	2.19	1.12	0.37	0.34
Provider(s') utility	0.44	0.22	0.07	0.20
Social welfare	1.32	2.53	3.36	3.47

Table 10: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on GPQA using chain-of-thought.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.02$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	3	4	2
User value	13.95	15.45	15.35	15.25
Price	0.24	0.24	0.31	0.91
Provider(s') utility	0.05	0.05	0.06	0.72
Social welfare	14.00	15.50	15.42	15.97

margin $= 0.15$ $\alpha = \$0.02$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	3	4	2
User value	13.95	15.45	15.35	15.25
Price	0.24	0.24	0.31	0.91
Provider(s') utility	0.05	0.05	0.06	0.72
Social welfare	14.00	15.50	15.42	15.97

margin $= 0.05$ $\alpha = \$0.02$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	3	4	2
User value	13.95	15.45	15.35	15.25
Price	0.24	0.24	0.31	0.91
Provider(s') utility	0.05	0.05	0.06	0.72
Social welfare	14.00	15.50	15.42	15.97

margin $= 0.25$ $\alpha = \$0.04$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	3	4	2
User value	29.82	31.53	31.02	30.62
Price	0.24	0.24	0.31	1.70
Provider(s') utility	0.05	0.05	0.06	1.51
Social welfare	29.87	31.57	31.08	32.13

margin $= 0.25$ $\alpha = \$0.01$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`R1-D-Qwen-7B`
Compute (θ)	4	4	3	1
User value	6.37	7.32	7.75	7.56
Price	0.22	0.24	0.27	0.50
Provider(s') utility	0.04	0.05	0.05	0.32
Social welfare	6.41	7.37	7.81	7.89

E.3.4 Results of the auction mechanism on TruthfulQA

Table 11: **Comparison of the equilibrium between $\mathcal{G}$ and the auction $\widetilde{\mathcal{G}}$ on TruthfulQA using best-of-n.** The table shows, for $\widetilde{\mathcal{G}}$, the value received by the user (once the auction is conducted), the price they pay (according to Eq. 9), the provider's utility, the social welfare, and the provider (LLM) that wins the auction. For $\mathcal{G}$, the table shows the providers' average utility, the users' average value and price, all weighted by their equilibrium market shares, together with the social welfare. We vary the provider margin—used to derive the per-token cost from the per-token price—and the proportionality parameter α, which linearly converts a percentage point of accuracy into a monetary value (see Appendix D). All quantities (except the compute) have units ($\$ \times 10^{-3}$).

margin $= 0.25$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-3B`
Compute (θ)	32	32	32	16
User value	1.17	1.25	1.58	1.58
Price	0.21	0.22	0.10	0.10
Provider(s') utility	0.04	0.04	0.02	0.04
Social welfare	1.22	1.30	1.60	1.62

margin $= 0.15$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-3B`
Compute (θ)	32	32	32	4
User value	1.17	1.25	1.58	1.58
Price	0.21	0.22	0.10	0.06
Provider(s') utility	0.03	0.03	0.01	0.04
Social welfare	1.20	1.28	1.59	1.62

margin $= 0.05$ $\alpha = \$0.008$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-3B`
Compute (θ)	32	32	32	4
User value	1.17	1.25	1.58	1.58
Price	0.21	0.22	0.10	0.06
Provider(s') utility	0.01	0.01	0.00	0.05
Social welfare	1.18	1.26	1.58	1.62

margin $= 0.25$ $\alpha = \$0.016$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-3B`
Compute (θ)	32	32	32	32
User value	2.62	2.83	3.29	3.19
Price	0.22	0.23	0.13	0.23
Provider(s') utility	0.04	0.05	0.03	0.12
Social welfare	2.66	2.87	3.32	3.32

margin $= 0.25$ $\alpha = \$0.004$	Game $\mathcal{G}$			Auction $\widetilde{\mathcal{G}}$
	$\beta = 2 \cdot 10^2$	$\beta = 10^3$	$\beta = 10^5$	`Qwen2.5-3B`
Compute (θ)	32	32	8	2
User value	0.47	0.49	0.79	0.78
Price	0.21	0.21	0.03	0.03
Provider(s') utility	0.04	0.04	0.01	0.03
Social welfare	0.51	0.54	0.79	0.81

E.3.5 Attribution of the welfare gain of the auction mechanism.

Compared to the test-time compute game $\mathcal{G}$, the auction $\widetilde{\mathcal{G}}$ improves social welfare through two mechanisms. First, it incentivizes each provider to select the compute θ_i^* that maximizes their individual contribution $W_i(\theta_i^*)$ to social welfare (Theorem 5). Second, it assigns the task to the provider with the highest such contribution. In $\mathcal{G}$, these two mechanisms are coupled: the user rationality β shapes both the equilibrium compute levels $\boldsymbol{\theta}^\dagger$ and the market shares. To isolate the effect of compute from that of allocation, we consider the best-of-n game on GSM8K from Section 5 at its equilibrium $\boldsymbol{\theta}^\dagger$ for $\beta = 10^3$. In this instance, the same provider, Qwen2.5-7B, offers the highest value $V_i(\theta_i^\dagger)$, makes the highest contribution $W_i(\theta_i^\dagger)$ to social welfare, and wins the auction (Table 3). We compute the social welfare that would be attained if Qwen2.5-7B served all users at its equilibrium compute $\theta_i^\dagger$. We then compare it to the social welfare of the auction, in which the same provider serves all users at θ_i^*. Since both outcomes share the same allocation, any difference in social welfare is due to compute alone. Table 12 shows that, at equilibrium, Qwen2.5-7B uses $\theta_i^\dagger = 16$ samples and yields a social welfare of \$0.00643. Under the auction, it uses $\theta_i^* = 2$ samples and attains \$0.00736. Hence, in this instance, the auction's welfare gain over the optimally allocated equilibrium comes entirely from correcting over-compute.

Table 12: **Effect of compute on social welfare in the best-of-n game on GSM8K at $\beta = 10^3$, with the allocation held fixed.**

Setting	Provider	Compute	Social welfare
$\mathcal{G}$, all users served at equilibrium compute	Qwen2.5-7B	$\theta_i^\dagger = 16$	\$0.00643
Auction $\widetilde{\mathcal{G}}$	Qwen2.5-7B	$\theta_i^* = 2$	\$0.00736

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide extensive empirical evidence supporting the inefficiency of current LLM-as-a-service markets in Appendix E. In Section 4, we introduce an auction mechanism that is efficient, and we prove this claim in Appendix C.5 .

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In Section 6 we discuss the assumptions and limitations of our work, both at a theoretical and at a practical level.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We detail the theoretical assumptions when stating each one of the theoretical results in Section 2, Section 3 and Section 4. The proofs are given in Appendix C.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide concrete details on the experiments we conduct in Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We provide the code and data required to reproduce our experiments, along with instructions for running them, at `https://github.com/Human-Centric-Machine-Learning/strategic-ttc`.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: We do not train any model in our paper, but provide in Appendix D the details of how we have evaluated the LLMs in our simulated markets.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We provide 95% confidence bars in the relevant experiments by bootstrapping.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix D provides the details about the hardware setup we used in our experiments. We do not report execution time because it is not relevant for our experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have reviewed the NeurIPS Code of Ethics, and our paper conforms with it.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss in Section 6 the broader impact of our work.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We discuss licenses and credits in Appendix D.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide the full code and data used in our experiments at`https://github.com/Human-Centric-Machine-Learning/strategic-ttc`.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Our work analyzes the market of LLMs-as-a-service, but we do not consider LLMs as a non-standard component of our work.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.