ePACT: ENERGY-PERFORMANCE-AWARE COMMITMENT TRACKING FOR LLM SERVING

Anonymous authors
Paper under double-blind review

ABSTRACT

The power grid must continuously balance electricity supply and demand, making predictable demand important for electricity provisioning. Data center operators therefore often procure electricity under contracts that specify hourly energy commitments, based on which the expected demand can be planned in advance. Deviating from these commitments in either direction can lead to unfavorable settlements or penalties. For data centers serving increasingly dynamic LLM workloads, this creates a new systems challenge: serving decisions directly affect energy consumption, yet the realized consumption must track a precommitted hourly target while preserving request-level SLOs. We formulate this problem as energy-Performance-Aware Commitment Tracking and implement ePACT, an LLM serving system built around a two-level controller that periodically revisits the energy vs SLO-performance trade-off and adjusts the serving configuration. A global planner updates near-term energy targets from the remaining hourly commitment and observed consumption, while a local decision maker simulates candidate active-replica and GPU-clock configurations and selects the SLO-feasible action that best matches the target under asymmetric deviation costs. Adaptive action search limits planning overhead while serving proceeds asynchronously. We evaluate ePACT against representative latency-first and energy-first baselines, showing its ability to maintain service quality while better aligning realized energy consumption with varying hourly commitments.

1 INTRODUCTION

LLM serving increasingly contributes to large, variable data center electricity demand (Shehabi et al., 2024). The International Energy Agency estimates that data centers consumed 485 TWh in 2025 and projects approximately 950 TWh in 2030 (International Energy Agency, 2026). Operators are also incorporating computing flexibility into energy procurement: Google reports integrating 1 GW of data center demand response into long-term utility contracts (Terrell, 2026). Serving decisions therefore matter not only for computational efficiency, but also for how operators fulfill their electricity procurement commitments.

Advance electricity procurement can commit data centers to specified hourly consumption volumes. In hourly day-ahead markets such as PJM, buyers establish financially binding hourly energy positions before delivery (PJM Interconnection, n.d.b;n). These commitments help the grid schedule generation against expected demand and maintain real-time balance (EIA, n.d.). If actual consumption differs from the committed volume, the buyer can face imbalance settlements or contractual charges (Elexon, n.d.; WAPA, 2026). Importantly, under contracts with volume-deviation charges, unused committed energy can still impose an economic loss on the operator (EDF Energy, 2015). When requests are waiting, that energy could instead support additional serving capacity and lower latency. We therefore consider a preset energy commitment for each hour, with potentially different costs for overuse and underuse. For LLM serving, this creates a distinct objective: *tracking hourly energy commitments while maintaining request-level Service-Level Objectives (SLOs).*

Tracking these commitments online raises two challenges. Challenge 1: *Workload uncertainty.* The system must serve incoming requests while tracking the current hour's energy commitment without knowing future arrivals. Each serving decision changes both cumulative energy consumption and the backlog carried into subsequent time intervals, so later decisions must account for both earlier

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

consumption and the remaining work. Challenge 2: *SLO-energy trade-off.* Favoring low latency can consume energy too aggressively and overshoot the commitment, whereas aggressively reducing energy use can increase SLO violations and still incur costs by undershooting the committed volume. Serving decisions must therefore protect service quality while limiting both overuse and underuse.

The performance and energy objectives of existing LLM serving systems do not directly address this two-sided tracking problem. Systems such as vLLM, Sarathi, DistServe, and Splitwise improve serving efficiency through memory management, batch scheduling, and phase separation (Kwon et al., 2023; Agrawal et al., 2024b; Zhong et al., 2024; Patel et al., 2024), while energy-aware systems such as DynamoLLM and throttLL'eM reduce inference energy subject to performance constraints (Stojkovic et al., 2025; Kakolyris et al., 2025). These objectives favor lower latency or lower consumption, rather than alignment with a supplied hourly commitment. Neither objective alone resolves the joint risk of SLO violations and deviations on either side of that commitment.

We introduce ePACT, which divides each hourly settlement horizon into decision intervals, deciding *how much energy to consume* and *which serving configuration to use* for each interval. To tackle Challenge 1, a global planner updates interval targets from measured consumption and the remaining hourly commitment. To tackle Challenge 2, a local decision maker then uses the current serving state to evaluate candidate serving actions, admitting candidates predicted to meet the SLO and ranking them by asymmetric deviation cost relative to the interval target. Actual energy consumption closes the loop at the next boundary. Adaptive action search first samples the configuration space broadly, then refines promising neighborhoods while requests continue to execute.

In summary, we make the following contributions:

- We formulate online LLM serving under preset hourly energy commitments as a two-sided tracking problem: minimize asymmetric overuse and underuse costs subject to request-level deadlines, accounting for energy across active, draining, and standby replicas without knowing future arrivals.
- We design ePACT, a two-level controller that adjusts interval targets using measured consumption and jointly selects replica counts and GPU clocks using snapshot-based SLO predictions. Coarse-to-fine search reuses promising configurations while serving continues asynchronously.
- We evaluate ePACT on H20 and H200 serving setups through single-hour comparisons, component ablations, and 24-hour trace simulations. It achieves mean absolute hourly deviation of 2.16% and 2.31%, respectively, with near-vLLM SLO attainment; ablations isolate the roles of global feedback and local configuration adaptation.

2 Preliminaries & Related Work

We review how serving mechanisms and SLOs constrain configuration choices, then distinguish commitment tracking from existing energy objectives.

LLM serving and SLOs. An autoregressive request first processes its prompt in prefill and then generates output tokens through repeated decode steps. Iteration-level scheduling admits and retires requests as generation proceeds (Yu et al., 2022), while vLLM's PagedAttention improves the management of their key-value caches. Sarathi-Serve uses chunked prefill to reduce interference with decoding; DistServe separates the two phases to optimize goodput under time-to-first-token (TTFT) and time-per-output-token (TPOT) constraints. Serving quality is commonly expressed through latency SLOs, where a request is compliant when its latency remains within a prescribed threshold (Gujarati et al., 2020; Sun et al., 2024). To account for heterogeneous request lengths, prior inference-serving work also uses scaled SLOs, which set the latency threshold relative to a baseline or isolated execution latency through a multiplicative SLO scale (Li et al., 2023; Fan et al., 2023; Jiang et al., 2024; 2025; Li et al., 2026). We use request-level deadlines to express this service requirement while controlling serving capacity.

Energy commitments and load shaping. A power cap constrains instantaneous demand, whereas an energy commitment concerns cumulative consumption within a specified hour. Volume-tolerance provisions can charge for deviations in either direction, with contract-specific settlement periods and tolerances (EDF Energy, n.d.; 2025). Related carbon-aware systems shift flexible computation in time. Google's Carbon-Intelligent Compute Management (Radovanović et al., 2023) uses Virtual Capacity Curves to shift flexible workloads, while the probabilistic carbon-aware scheduling framework (Hall

et al., 2025) combines day-ahead planning with real-time tracking to reduce carbon cost and peak power. In contrast to shifting flexible jobs, our problem is to track a supplied hourly commitment while serving requests with latency deadlines.

Energy-aware inference. DynamoLLM coordinates instance count, parallelism, and frequency to reduce energy under SLOs. throttLL'eM predicts serving behavior to scale instances and GPU frequency, while GreenLLM applies SLO-aware frequency control to prefill and decode (Liu et al., 2025). For prefill-decode disaggregated serving, VoltanaLLM combines frequency control with routing (Yu et al., 2026), and DualScale couples phase-aware placement with fine-grained frequency control (Basit et al., 2026). These studies establish the value of adapting capacity and frequency. ePACT uses these controls to reduce deviation from an interval target, including cases where the preferred SLO-admitted configuration consumes more energy.

Online scheduling and simulation. Llumnix responds to heterogeneous requests through runtime rescheduling and live migration. Vidur uses profiling-calibrated simulation to explore serving configurations (Agrawal et al., 2024a). In ePACT, simulator predictions support local selection: they check predicted deadlines and estimate each configuration's contribution to commitment tracking.

3 System Overview

ePACT translates a supplied hourly energy commitment into online serving decisions through a two-level controller. The global planner updates interval energy targets; the local decision maker searches for, evaluates, and selects configurations for a co-located serving pool. We describe the component responsibilities and feedback loop below; Figure 1 shows the architecture.

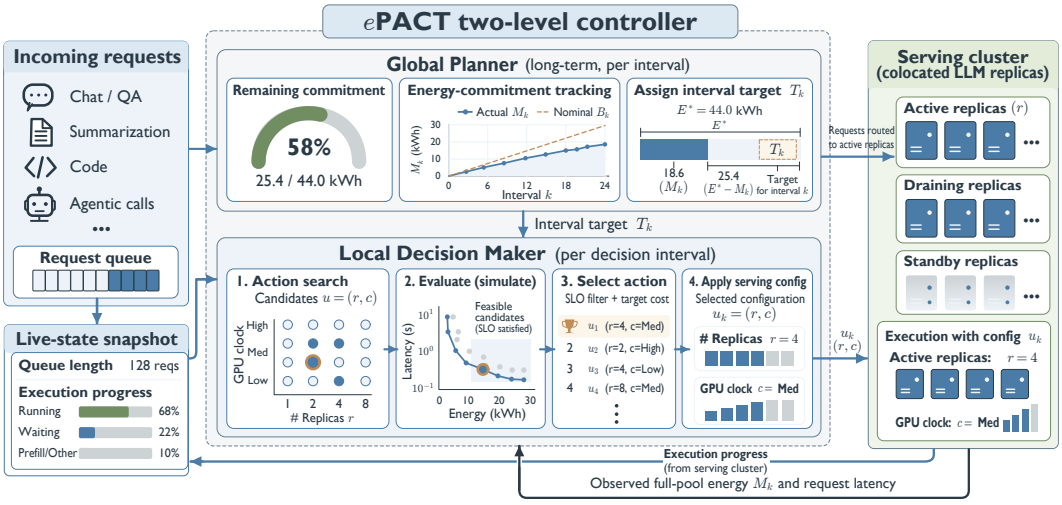

Figure 1: ePACT architecture. Remaining commitment, cumulative-energy tracking, and interval-target assignment belong to the global planner. Adaptive search, prediction, SLO checking, selection, and configuration application belong to the local decision maker. The serving pool co-locates prefill and decode. Mini-plots illustrate state and control, not measured results.

3.1 Control Components

Global planner. The global planner tracks cumulative energy consumption within the current hour to determine the remaining energy commitment. At each decision boundary, it allocates the remaining commitment across the remaining intervals and supplies the current interval target. Recomputing this target from measured consumption allows subsequent decisions to account for earlier overuse or underuse instead of following a fixed allocation.

Local decision maker. The local decision maker searches over serving configurations defined by two parameters: the *active-replica count*, which determines how many replicas accept new requests, and the *GPU clock*, which sets their GPU operating frequency. Using the target and a snapshot of current

requests and replica states, a profiling-calibrated simulator predicts each candidate's interval energy consumption and request completion times. Among candidates that pass the predicted SLO check, the local decision maker selects the configuration with the lowest asymmetric deviation cost relative to the interval energy target. Section 4 formalizes target updates, selection, and fallbacks.

3.2 Serving and Feedback

Arrivals and execution progress update the live state captured at each decision boundary. Serving continues under the current configuration during search; a selected configuration is applied only if it remains valid. A replica removed from active service finishes its running requests before entering standby, and standby replicas retain model weights for reactivation. Energy accounting covers active, draining, and standby replicas, so reducing active capacity does not remove idle consumption from the commitment ledger. These measurements feed the next global target update; Appendix A details routing and application.

4 Energy-Commitment Tracking

The controller must connect an hour-end energy objective to decisions made before future requests are known. We first formulate that objective, then derive interval targets and explain how local selection and adaptive search use them.

4.1 Problem Formulation

Input. Requests arrive online with varying prompt and response lengths. These lengths and arrival fluctuations determine the batches formed by the runtime and the capacity needed to meet deadlines. The controller observes arrived requests and their execution progress, but not future arrivals. For each hour, $\mathcal{R}$ denotes its arriving requests, whose deadlines may extend beyond that hour.

Energy commitment. Each hour has a preset energy commitment $E^* > 0$ and overuse and underuse rates λ_+ and λ_-, measured in cost per unit energy. These are fixed inputs, not procurement decisions made by ePACT. We omit the hour index below and track the controlled GPU pool; Appendix B.1 explains its relationship to facility consumption. Overuse and underuse incur their respective costs.

Constraints. The service requirement constrains which energy-saving configurations are acceptable. Request i, arriving at a_i, has reference latency L_i^{ref}, its standalone service time on an isolated replica. The SLO scale α defines its deadline $d_i = a_i + \alpha L_i^{\mathrm{ref}}$. The service policy permits a deadline-miss fraction of at most ϵ; $\epsilon = 0$ admits none.

Action space. ePACT adjusts active-replica count and GPU clock jointly. A configuration $u = (r, c)$ activates $r \in \{1, \dots, N\}$ replicas from a resident pool of size N and selects a shared GPU clock c from supported levels. More active replicas provide greater concurrent capacity but also keep more resources active; changing the clock adjusts execution speed and power within each replica. Both choices affect batch formation, request completions, and the backlog carried forward.

Optimization objective. We divide the hourly settlement horizon into K decision intervals of duration Δ, and let an online policy π choose configuration u_k at each decision boundary. If e_k denotes actual GPU-pool energy in interval k, then total consumption is $M_K = \sum_{k=0}^{K-1} e_k$. With f_i denoting the actual request completion time, the optimization objective is

$$\begin{aligned} \min_{\pi} \quad & C(M_K) = \lambda_+ \left[M_K - E^*\right]_+ + \lambda_- \left[E^* - M_K\right]_+ , \\ \text{subject to} \quad & \left|\{i \in \mathcal{R} : f_i > d_i\}\right| \le \epsilon |\mathcal{R}|, \end{aligned} \tag{1}$$

where $[x]_+ = \max(x, 0)$. This objective penalizes both directions of energy deviation under the prescribed service requirement. Each hour is assessed separately: overuse in one hour is not offset by underuse in another.

From an hourly commitment to interval targets. Equation (1) assesses energy at the end of each hour, but the controller must choose a configuration at every decision boundary. ePACT therefore assigns each interval an adjustable energy target T_k that guides current decisions toward the remaining commitment. The global planner computes this target from measured progress; the local decision maker uses it to rank predicted configurations.

4.2 Global Planner: Dynamic Interval Targets

The global planner converts cumulative consumption into the next interval's energy target. Keeping every target fixed would leave earlier overuse or underuse uncorrected; updating targets lets the remaining intervals compensate within the same hour.

Allocating the interval energy target. Suppose the operator has a prior nominal allocation $b_k \geq 0$ for each decision interval, with $\sum_{k=0}^{K-1} b_k = E^*$. At the start of interval k, the planner has observed cumulative consumption $M_k = \sum_{j<k} e_j$, so the actual energy still needed to reach the commitment is $E^* - M_k$. Meanwhile, if $B_k = \sum_{j<k} b_j$ denotes the nominal allocation assigned to elapsed intervals, then $E^* - B_k$ is the nominal allocation remaining over the rest of the hour.

Scaling the nominal allocation. The ratio $\frac{(E^*-M_k)}{(E^*-B_k)}$ compares the remaining commitment with the remaining nominal allocation. A value above one indicates less consumption than planned and raises the target; a value below one indicates more consumption and lowers it. The planner scales b_k accordingly: $T_k = b_k \frac{E^*-M_k}{E^*-B_k}$. Scaling by this ratio retains the relative nominal allocations while correcting for measured consumption. By default, the nominal allocations are equal, $b_k = \frac{E^*}{K}$, so the target update becomes $T_k = \frac{E^*-M_k}{K-k}$. The next interval therefore receives an equal share of the actual remaining commitment, not an unchanged share of the original commitment.

Illustrative example for determining next interval's energy target. Consider an hourly commitment of $E^* = 600\,\mathrm{kJ}$ divided into $K = 6$ intervals with nominal allocation $b_k = 100\,\mathrm{kJ}$ each. After the first interval, $B_1 = 100\,\mathrm{kJ}$ and five intervals remain. If $M_1 = 70\,\mathrm{kJ}$, underuse leaves $530\,\mathrm{kJ}$ to allocate, raising the next target to $T_1 = 100\frac{600-70}{600-100} = 106\,\mathrm{kJ}$. If $M_1 = 130\,\mathrm{kJ}$ instead, overuse leaves $470\,\mathrm{kJ}$, lowering the next target to $T_1 = 100\frac{600-130}{600-100} = 94\,\mathrm{kJ}$. Only the current target is issued; subsequent targets are recomputed from updated measurements rather than fixed in advance. This example illustrates how the global planner flexibly sets the next energy target based on the observed cumulative consumption.

Using targets without overriding SLOs. An interval target is a cost reference, not a hard cap. Bursts may require extra energy to meet SLOs, while little useful work can make a high target unreachable. Consumption above E^* can even yield a negative target, not negative physical energy. The local decision maker therefore checks predicted service quality before matching T_k.

4.3 Local Decision Maker: SLO Checks and Target Matching

Given the interval target, request SLOs, and live-state snapshot, the local decision maker checks predicted service quality before ranking configurations by energy-deviation cost. Appendix Figure 4 illustrates the choice; Section 4.4 explains how adaptive search proposes candidates.

Predicting completion times and energy. Every candidate starts from the same captured state. The simulator replays the serving scheduler, substituting profiling-calibrated step duration and power for GPU execution. It also replays previously observed arrivals, without accessing future trace arrivals. Energy prediction $\widehat{E}_k(u)$ covers one decision interval; completion prediction $\widehat{f}_i(u)$ continues the same configuration until the captured requests finish. These horizons differ because a request that extends beyond the interval must still be checked against its deadline. We calibrate and validate the predictor offline; Appendix C.5 reports prediction accuracy and control overhead.

Admitting candidates by SLO. Let $\mathcal{I}_k$ be the unfinished requests in the snapshot. The controller compares their predicted completion times with their original deadlines, including delay already incurred. A candidate passes when its predicted miss count satisfies

$$\left|\{i \in \mathcal{I}_k : \widehat{f}_i(u) > d_i\}\right| \leq \epsilon|\mathcal{I}_k|. \tag{2}$$

Let $\mathcal{F}_k$ denote the evaluated candidates with complete, valid predictions that pass this check. With $\epsilon = 0$, every captured request must be predicted to finish on time. This is a check on the predicted workload, not a guarantee of actual deadlines under unseen arrivals.

Matching the target among admitted candidates. Among admitted candidates, neither lower energy nor lower latency alone determines the choice. Instead, the local decision maker compares each candidate's predicted energy with T_k using the terminal objective's overuse and underuse rates:

$$J_k(u) = \lambda_+ \left[\widehat{E}_k(u) - T_k\right]_+ + \lambda_- \left[T_k - \widehat{E}_k(u)\right]_+, \; u_k \in \arg\min_{u \in \mathcal{F}_k} J_k(u). \tag{3}$$

The selected candidate is the closest *cost-weighted* match, which need not minimize absolute energy deviation. Thus, passing the SLO check establishes eligibility; the target and penalty rates determine the energy preference. This interval score guides decisions but does not replace the hour-end objective in Equation (1).

Service-first fallback. If no evaluated candidate passes the check, ePACT first minimizes predicted deadline misses. Ties favor the largest worst-request SLO margin - how far the most constrained request lies inside or outside its deadline - and then the lowest energy-deviation cost. Equal-cost admitted candidates use the same service-based tie-breaks. If no candidate has a valid prediction, ePACT retains the current configuration.

4.4 Adaptive Action Search

Local selection can rank only the configurations already evaluated; adaptive search determines which to evaluate next. ePACT starts broadly, then refines promising neighborhoods. The search retains evaluated configurations and their predicted request completion times and interval energy, using these records to choose unvisited candidates until the stopping condition below is met.

Constructing the catalog. ePACT selects up to eight replica counts and five supported GPU clocks at approximately uniform rank intervals, preserving both endpoints of each domain, yielding at most $A = 40$ configurations. The search assumes no monotonic relationship between replica count, clock frequency, energy, or SLO attainment.

Coarse initialization and warm starts. Each search begins with a coarse lattice formed from the lowest, middle, and highest ranks along both dimensions, augmented with the current configuration and historical warm-start candidates. Duplicate configurations are removed. Historical configurations seed the search, but their predictions and scores are recomputed for the current decision.

Adaptive refinement. Starting from the initial candidates, ePACT performs a coarse-to-fine search around the best configuration found so far, where "best" follows the selection rule in Section 4.3. At each resolution, it evaluates unvisited neighboring configurations by changing one or both configuration dimensions by the current search stride. The search starts with a relatively coarse stride, approximately one quarter of each dimension's rank span. Whenever a better configuration is found, ePACT recenters the search at the same stride. Once all neighbors at the current resolution have been explored, it halves the stride and continues at a finer resolution. The search stops after all one-step neighbors of the best configuration have been evaluated. If none of the evaluated configurations has a complete, valid prediction, ePACT evaluates the remaining configurations as a fallback.

Execution and overhead. Candidates proposed together are evaluated in parallel using persistent CPU workers. Search runs asynchronously with serving and terminates by completion of the unit-step neighborhood or catalog exhaustion, without a wall-time cutoff. If $n_{\text{eval}} \leq A$ configurations are evaluated, the simulation work is $O(n_{\text{eval}} T_{\text{sim}})$, where T_{sim} denotes the cost of one candidate simulation. Each configuration is evaluated at most once per decision.

5 Evaluation

We evaluate how well ePACT tracks hourly energy commitments while maintaining request-level SLOs, focusing on three questions:

- **Q1: Single-hour service-energy performance.** How does ePACT compare with baselines across SLO scales under fixed and decreasing commitments?
- **Q2: Controller ablation.** How do the global planner and local decision maker in ePACT contribute to SLO attainment and hourly commitment tracking?

- **Q3: Full-day commitment tracking.** At a fixed SLO scale, how does ePACT perform over a full day of changing workloads and hourly commitments?

The single-hour study tests sensitivity to SLO and commitment choices and isolates controller components. The full-day study tests continuous operation as arrivals and commitments vary.

5.1 Experimental Setup

Serving setups. We evaluate H20 and H200 GPU pools. Each pool contains 32 GPUs, with one vLLM-hosted single-GPU replica of Qwen3.8-27B-FP8 per device. Each hourly commitment spans 60 one-minute control intervals. Appendices C.1 and B.1 detail execution and GPU metering.

Workloads. Both studies use requests from the Azure LLM inference trace (Microsoft, 2024); the full-day study replays $969{,}781$ requests.

Preset hourly commitments. The single-hour study uses two settings. A *fixed commitment* tests sensitivity to the SLO requirement without changing the energy objective. A *decreasing commitment* tests whether looser SLOs allow tracking lower energy targets. In both settings, a run's hourly commitment is fixed; "decreasing" describes the targets assigned across SLO scales, not a change within a run. Full-day commitments instead vary by hour. All targets are preset and shared across methods; Appendix C.2 gives their values and construction.

Baselines. We compare ePACT against two representative baselines:

- **vLLM.** We use unmodified vLLM as the latency-first representative, keeping all 32 replicas resident and active throughout each run. GPUs can operate up to the experiment's hardware-specific clock ceiling; this baseline does not proactively reduce clocks or deactivate replicas to save energy.
- **DynamoLLM.** We implement DynamoLLM's energy-aware serving policy as the energy-minimizing representative. It adjusts active capacity and GPU clocks from observed demand and latency pressure, and restores capacity for urgent requests. It uses the same model, workload, and GPU pool as ePACT.

Ablations. We evaluate two variants that successively remove the controller's two levels:

- **Static-Target.** This variant removes the global planner and uses fixed nominal targets, $T_k = b_k$, without correcting accumulated energy deviation. The local decision maker still checks predicted SLOs and adjusts replica counts and GPU clocks to minimize deviation cost.
- **Fixed-Config.** This variant also removes the local decision maker and keeps one configuration throughout the hour: 28 active replicas at 1980 MHz on H20 and 1710 MHz on H200.

Metrics. In our evaluation, we use two primary metrics:

- **SLO attainment.** We report the fraction of requests meeting the deadlines in Section 4.1. Single-hour comparisons and ablations use $\alpha \in \{2.5, 3, 3.5\}$; the full-day comparison fixes $\alpha = 3$. Requests are grouped by arrival time and scored at completion, including completions after the evaluation horizon. Full-day attainment pools all arrivals rather than averaging hourly percentages.
- **Energy-commitment deviation.** We report signed hourly deviation $\frac{(M_K - E^*)}{E^*}$ as a percentage: positive values indicate overuse and negative values indicate underuse. Energy is assigned to the hour in which it is consumed. For full-day tracking, we also report mean absolute hourly deviation, hours within $\pm 5\%$, and the sum of nonmonetary, asymmetric hourly deviation costs. These costs weight overuse and underuse differently and settle each hour separately, so opposite deviations cannot cancel. Appendix C.3 specifies the weights and units.

5.2 Single-Hour Service-Energy Comparison

Figure 2 answers Q1 for both settings on H20 and H200. We discuss baselines first, then ablations in Section 5.3. Appendix Tables 2 and 3 list the values.

ePACT keeps high attainment while aligning to the preset commitment. Across both settings and GPU types, ePACT stays within 0.09 percentage points of vLLM's attainment and achieves the

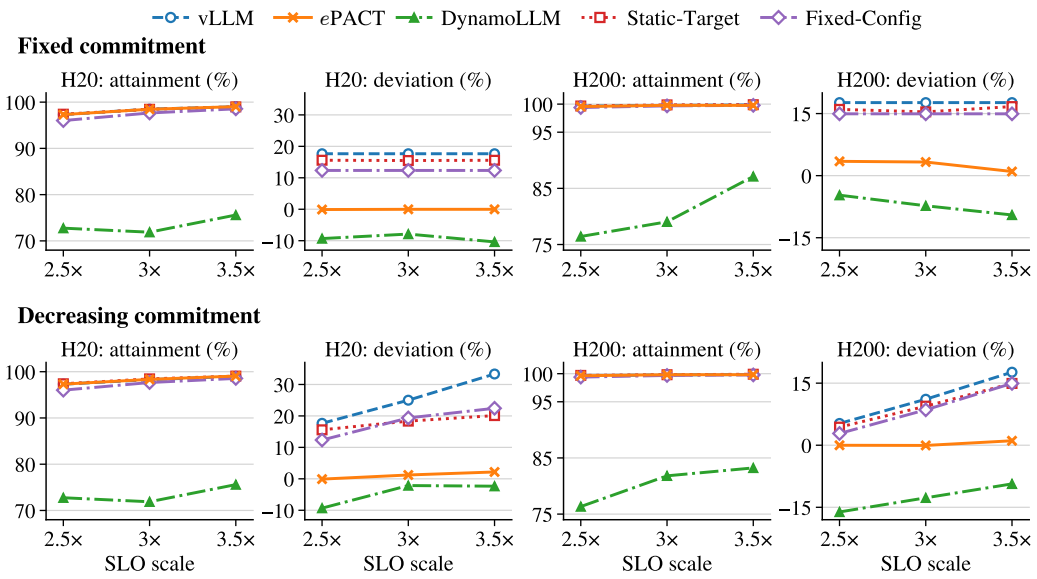

Figure 2: Single-hour comparisons with fixed (top) and decreasing (bottom) commitments. Columns show H20 and H200 attainment and signed deviation; positive deviation denotes overuse.

smallest absolute deviation among the five methods. With fixed commitments, absolute deviation is at most 0.09% on H20 and 3.46% on H200, preserving tracking as the SLO changes. With decreasing commitments, the corresponding maximum deviations are 2.18% and 1.08%: ePACT uses the flexibility of looser SLOs to track lower targets. H200 energy falls from 15.27 to 13.81 MJ while attainment remains above 99.63%.

Full-capacity serving does not track commitments. vLLM overuses the fixed commitments by 17.65% on both GPUs. Under decreasing commitments, its overuse rises from 17.65% to 33.33% on H20 and from 5.26% to 17.65% on H200. The increase reflects smaller target denominators, not increased serving capacity: the baseline has no feedback that adapts consumption to the commitment.

Energy-minimizing serving falls short on both objectives. Across both settings, DynamoLLM undershoots the commitments and has lower SLO attainment: 71.88%-75.59% on H20 and 76.34%-87.12% on H200. Energy-oriented downscaling may leave insufficient capacity during bursts, allowing queueing delay to accumulate before recovery. When extra capacity both reduces latency and better matches the interval target, ePACT does not forgo that capacity merely to save energy.

5.3 Controller Ablation

To answer Q2, we compare ePACT with Static-Target and Fixed-Config within each setting in Figure 2.

Global feedback corrects accumulated deviation. Removing the global planner increases absolute deviation in every comparison, while Static-Target's attainment remains within 0.13 percentage points of ePACT. Its overuse is 15.41%-16.69% with fixed commitments and 4.37%-20.11% with decreasing commitments. For H200 at $3\times$ under the latter setting, global feedback reduces deviation from $+9.43\%$ to -0.05%. Feedback corrects the cumulative error left by fixed targets.

Local selection preserves flexibility. With the global planner disabled, Static-Target still adapts configurations to minimize predicted deviation cost subject to the SLO check; Fixed-Config cannot adapt. Static-Target has higher SLO attainment throughout both settings. Fixed-Config often has lower energy deviation, but sacrifices attainment: at H20 $2.5\times$, the gap is 1.38 percentage points. With decreasing H20 commitments at $3\times$ and $3.5\times$, local selection improves both metrics, reducing overuse to 18.35% and 20.11% from 19.36% and 22.43%. Local selection therefore preserves service-energy flexibility, but does not uniformly improve tracking without global feedback. Together, the ablations separate adapting capacity to the current workload from correcting accumulated energy error: local selection provides the former, while measured-energy feedback supplies the latter.

5.4 24-HOUR COMMITMENT TRACKING

To answer Q3, Table 1 compares ePACT with the baselines over 24 hours at fixed $\alpha = 3$ on H20 and H200. Unlike the single-hour study, both arrivals and commitments change across hours. Requests and serving state persist across hour boundaries, as detailed in Appendix C.3.

Table 1: Full-day comparison at fixed $3\times$ SLO. Energy metrics use 24 separate hourly settlements; attainment pools all arriving requests. The hour count is out of 24.

GPU	Method	Attainment (%)	Mean abs. dev. (%)	Hours within $\pm 5\%$	Cost
H20	vLLM	99.4836	11.4304	7	23.5683
H20	DynamoLLM	80.2132	8.9597	5	14.3261
H20	ePACT	**99.4892**	**2.1630**	**22**	**6.1838**
H200	vLLM	99.9644	11.4304	7	18.6780
H200	DynamoLLM	83.8908	12.5358	4	15.0900
H200	ePACT	**99.9371**	**2.3092**	**21**	**4.5320**

Sustained service quality with low hourly deviation on both GPUs. On H20, ePACT keeps 99.4892% SLO attainment while reducing mean absolute hourly deviation from vLLM's 11.4304% to 2.1630%, with 22 of 24 hours within $\pm 5\%$. On H200, it attains 99.9371% while reducing mean absolute hourly deviation from 11.4304% to 2.3092%, with 21 of 24 hours within $\pm 5\%$. Across both pools, close hourly tracking is compatible with near-vLLM service quality rather than requiring the attainment loss observed for DynamoLLM.

Hourly results reveal the remaining tracking error. On H200, ePACT exceeds the commitment by 5.80% in hour 7 and 5.26% in hours 21 and 23, despite attainment above 99.9% in each. These residuals distinguish average tracking quality from exact agreement in every hour. A daily energy total could hide offsetting hourly errors; mean absolute deviation measures their magnitude, while the within-band count indicates how consistently the controller tracks separate hourly commitments. Appendix Figures 5 and 6 and Tables 4 and 5 show the hourly trajectories and outcomes.

5.5 PREDICTION ACCURACY AND PLANNING OVERHEAD

Held-out predictor validation. We validate the predictions used for local selection on held-out workloads, with detailed results in Appendix C.5. With frozen models and no refitting, five fixed-batch workload classes, each repeated twice on one replica at 1980 MHz, yield latency WAPE of 0.44%-2.66% and energy WAPE of 0.19%-2.89%. Here WAPE is $100\frac{\sum_j |\widehat{y}_j - y_j|}{\sum_j y_j}$ for measured values y_j and predictions $\widehat{y}_j$. Another nonoverlapping trace slice uses 97 requests with continuous arrivals and batching. Latency WAPE is 4.15%. The signed energy error over the 120-second window is $+0.76\%$. This window error is distinct from energy WAPE. These checks characterize single-replica settings.

Asynchronous planning cost. The reported run completes 60 asynchronous decisions with a mean search wall time of 23.35 seconds. Serving continues during search; this measurement does not by itself establish zero latency impact. The control interval therefore should not be confused with an instantaneous response time.

6 CONCLUSION

We presented ePACT, an LLM serving system that tracks preset hourly energy commitments while maintaining request-level SLOs. Its objective accounts for both overuse and underuse rather than optimizing latency or energy consumption alone. A global planner corrects interval targets using measured consumption, while a local decision maker uses simulator-guided adaptive action search to select replica counts and GPU clocks under predicted SLO constraints. On H20 and H200, ePACT maintains near-vLLM attainment with mean absolute hourly deviation of at most 2.31% over 24 hours. The results show the value of linking service-aware configuration choices to measured progress toward a fixed commitment.

AI USE STATEMENT

(This section is **required** and does not count toward the page limit.)

In this work, we used generative AI tools to aid and polish writing, including language refinement, grammar checking, typo correction, and improving clarity and readability. We did not use generative AI tools to generate research ideas, design the methodology, or conduct experiments. All AI-assisted text was reviewed and revised by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Amey Agrawal, Nitin Kedia, Jayashree Mohan, Ashish Panwar, Nipun Kwatra, Bhargav S. Gulavani, Ramachandran Ramjee, and Alexey Tumanov. Vidur: A large-scale simulation framework for LLM inference. In *Proceedings of Machine Learning and Systems*, volume 6, pp. 351–366, 2024a. URL https://proceedings.mlsys.org/paper_files/paper/2024/hash/b74a8de47d2b3c928360e0a011f48351-Abstract-Conference.html.

Amey Agrawal, Nitin Kedia, Ashish Panwar, Jayashree Mohan, Nipun Kwatra, Bhargav Gulavani, Alexey Tumanov, and Ramachandran Ramjee. Taming throughput-latency tradeoff in LLM inference with Sarathi-Serve. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pp. 117–134. USENIX Association, 2024b. URL https://www.usenix.org/conference/osdi24/presentation/agrawal.

Omar Basit, Yunzhao Liu, Z. Jonny Kong, and Y. Charlie Hu. DualScale: Energy-efficient disaggregated LLM serving via phase-aware placement and DVFS. *arXiv preprint arXiv:2602.18755*, 2026. URL https://arxiv.org/abs/2602.18755.

EDF Energy. No crying over spilt milk: How volume tolerance works in electricity contracts. EDF Energy, Talk Power, 2015. URL https://www.edfenergy.com/large-business/talk-power/blogs/no-crying-over-spilt-milk-how-volume-tolerance-works-electricity-contracts. 21 September 2015; accessed 12 September 2026.

EDF Energy. Flexible electricity contracts, September 2025. URL https://www.edfenergy.com/sites/default/files/flexible_elec_contracts_efactsheet_250920.pdf. September 2025 edition.

EDF Energy. TalkPower Energy Academy: Plan, lesson 1, n.d. URL https://www.edfenergy.com/large-business/energy-academy/plan/lesson-one. Accessed 12 September 2026.

EIA. Hourly electric grid monitor: About the EIA-930 data. U.S. Energy Information Administration, n.d. URL https://www.eia.gov/electricity/gridmonitor/about. Accessed 18 September 2026.

Elexon. The electricity trading arrangements: A beginner's guide, n.d. URL https://bscdocs.elexon.co.uk/guidance-notes/the-electricity-trading-arrangements-a-beginners-guide. Balancing and Settlement Code guidance; accessed 18 September 2026.

Hongxiang Fan, Stylianos I. Venieris, Alexandros Kouris, and Nicholas D. Lane. Sparse-dysta: Sparsity-aware dynamic and static scheduling for sparse multi-dnn workloads. In *Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture*, pp. 353–366, 2023. doi: 10.1145/3613424.3614263.

Arpan Gujarati, Reza Karimi, Safya Alzayat, Wei Hao, Antoine Kaufmann, Ymir Vigfusson, and Jonathan Mace. Serving DNNs like Clockwork: Performance predictability from the bottom up. In *14th USENIX Symposium on Operating Systems Design and Implementation (OSDI 20)*, pp. 443–462. USENIX Association, 2020. URL https://www.usenix.org/conference/osdi20/presentation/gujarati.

Sophie Hall, Francesco Micheli, Giuseppe Belgioioso, Ana Radovanović, and Florian Dörfler. Carbon-aware computing for data centers with probabilistic performance guarantees. *IEEE Transactions on Power Systems*, 2025. doi: 10.1109/TPWRS.2025.3630499. URL https://ieeexplore.ieee.org/document/11250739/. Early access, published 17 November 2025.

International Energy Agency. Key questions on energy and AI: Executive summary. IEA, 2026. URL https://www.iea.org/reports/key-questions-on-energy-and-ai/executive-summary. Accessed 12 September 2026.

Youhe Jiang, Ran Yan, Xiaozhe Yao, Yang Zhou, Beidi Chen, and Binhang Yuan. HexGen: Generative inference of large language model over heterogeneous environment. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 21946–21961. PMLR, 2024. URL https://proceedings.mlr.press/v235/jiang24f.html.

Youhe Jiang, Ran Yan, and Binhang Yuan. HexGen-2: Disaggregated generative inference of LLMs in heterogeneous environment. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=Cs6MrbFuMq.

Andreas Kosmas Kakolyris, Dimosthenis Masouros, Petros Vavaroutsos, Sotirios Xydis, and Dimitrios Soudris. throttLL'eM: Predictive GPU throttling for energy efficient LLM inference serving. In *IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2025. doi: 10.1109/HPCA61900.2025.00103. URL https://ieeexplore.ieee.org/document/10946751/.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles (SOSP)*, 2023. URL https://arxiv.org/abs/2309.06180.

Zhuohan Li, Lianmin Zheng, Yinmin Zhong, Vincent Liu, Ying Sheng, Xin Jin, Yanping Huang, Zhifeng Chen, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. AlpaServe: Statistical multiplexing with model parallelism for deep learning serving. In *17th USENIX Symposium on Operating Systems Design and Implementation (OSDI 23)*, pp. 663–679. USENIX Association, 2023. URL https://www.usenix.org/conference/osdi23/presentation/li-zhouhan.

Zikun Li, Zhuofu Chen, Remi Delacourt, Gabriele Oliaro, Zeyu Wang, Qinghan Chen, Shuhuai Lin, April Yang, Zhihao Zhang, Zhuoming Chen, Yi-Hsiang Lai, Xinhao Cheng, Xupeng Miao, and Zhihao Jia. Adaserve: Accelerating multi-slo llm serving with slo-customized speculative decoding. In *Proceedings of the 21st European Conference on Computer Systems*, 2026. doi: 10.1145/3767295.3769315.

Qunyou Liu, Darong Huang, Marina Zapater, and David Atienza. GreenLLM: SLO-aware dynamic frequency scaling for energy-efficient LLM serving. *arXiv preprint arXiv:2508.16449*, 2025. URL https://arxiv.org/abs/2508.16449.

Microsoft. Azure LLM inference trace 2024. Azure Public Dataset, 2024. URL https://github.com/Azure/AzurePublicDataset/blob/master/AzureLLMInferenceDataset2024.md. Timestamped inference traces; accessed 18 September 2026.

NVIDIA. *NVML API Reference Guide: Device Queries*, 2024. URL https://docs.nvidia.com/deploy/archive/R550/nvml-api/group__nvmlDeviceQueries.html. Release R550.

NVIDIA. *NVIDIA System Management Interface Documentation*, n.d. URL https://docs.nvidia.com/deploy/nvidia-smi/index.html. Accessed 23 September 2026.

Pratyush Patel, Esha Choukse, Chaojie Zhang, Aashaka Shah, Íñigo Goiri, Saeed Maleki, and Ricardo Bianchini. Splitwise: Efficient generative LLM inference using phase splitting. In *51st Annual International Symposium on Computer Architecture (ISCA)*, 2024. URL https://arxiv.org/abs/2311.18677.

PJM Interconnection. Energy storage offers efficiency, flexibility to power the grid, June 2026. URL https://www.pjm.com/-/media/DotCom/about-pjm/newsroom/fact-sheets/energy-storage-fact-sheet.pdf. 15 June 2026.

PJM Interconnection. Guide to billing: Spot market energy, n.d.a. URL https://www.pjm.com/markets-and-operations/billing-settlements-and-credit/guide-to-billing. Accessed 23 September 2026.

PJM Interconnection. Energy market, n.d.b. URL https://www.pjm.com/markets-and-operations/energy.aspx. Accessed 18 September 2026.

Ana Radovanović, Ross Koningstein, Ian Schneider, Bokan Chen, Alexandre Duarte, Binz Roy, Diyue Xiao, Maya Haridasan, Patrick Hung, Nick Care, Saurav Talukdar, Eric Mullen, Kendal Smith, MariEllen Cottman, and Walfredo Cirne. Carbon-aware computing for datacenters. *IEEE Transactions on Power Systems*, 38:1270–1280, 2023. doi: 10.1109/TPWRS.2022.3173250. URL https://arxiv.org/abs/2106.11750.

Arman Shehabi, Sarah J. Smith, Alex Hubbard, Alex Newkirk, Nuoa Lei, Md Abu Bakar Siddik, Billie Holecek, Jonathan Koomey, Eric Masanet, and Dale Sartor. 2024 United States data center energy usage report. Technical report, Lawrence Berkeley National Laboratory, December 2024. URL https://datacenters.lbl.gov/publications/2024-lbnl-data-center-energy-usage-report.

Jovan Stojkovic, Chaojie Zhang, Íñigo Goiri, Josep Torrellas, and Esha Choukse. DynamoLLM: Designing LLM inference clusters for performance and energy efficiency. In *IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pp. 1348–1362, 2025. doi: 10.1109/HPCA61900.2025.00102.

Biao Sun, Ziming Huang, Hanyu Zhao, Wencong Xiao, Xinyi Zhang, Yong Li, and Wei Lin. Llumnix: Dynamic scheduling for large language model serving. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pp. 173–191. USENIX Association, 2024. URL https://www.usenix.org/conference/osdi24/presentation/sun-biao.

Michael Terrell. A new milestone for smart, affordable electricity growth. Google, March 2026. URL https://blog.google/innovation-and-ai/infrastructure-and-cloud/global-network/demand-response-data-center-milestone/. 19 March 2026.

The Green Grid. PUE: A comprehensive examination of the metric. Technical Report 49, The Green Grid, 2012. URL https://www.thegreengrid.org/en/resources/library-and-tools/237-WP%2349---PUE%3A-A-Comprehensive-Examination-of-the-Metric. Edited by Victor Avelar, Dan Azevedo, and Alan French.

WAPA. Energy imbalance service: Rate schedule DSW-EI4, schedule 4 to OATT. Western Area Power Administration, U.S. Department of Energy, 2026. URL https://www.wapa.gov/wp-content/uploads/2026/08/DSW-EI4.pdf. Rate effective 1 October 2016; version noting extensions through 30 September 2031.

Gyeong-In Yu, Joo Seong Jeong, Geon-Woo Kim, Soojeong Kim, and Byung-Gon Chun. Orca: A distributed serving system for Transformer-Based generative models. In *16th USENIX Symposium on Operating Systems Design and Implementation (OSDI 22)*, pp. 521–538. USENIX Association, 2022. URL https://www.usenix.org/conference/osdi22/presentation/yu.

Jiahuan Yu, Aryan Taneja, Junfeng Lin, and Minjia Zhang. VoltanaLLM: Energy-efficient and SLO-aware disaggregated LLM serving via adaptive frequency control and state-space routing. In *ISC High Performance*, 2026. URL https://arxiv.org/abs/2509.04827v3.

Yinmin Zhong, Shengyu Liu, Junda Chen, Jianbo Hu, Yibo Zhu, Xuanzhe Liu, Xin Jin, and Hao Zhang. DistServe: Disaggregating prefill and decoding for goodput-optimized large language model serving. In *18th USENIX Symposium on Operating Systems Design and Implementation (OSDI 24)*, pp. 193–210. USENIX Association, 2024. URL https://www.usenix.org/conference/osdi24/presentation/zhong-yinmin.

A Additional Implementation Details

This appendix details replica management, prediction semantics, the measurement boundary, and supplementary evaluation results.

Component boundaries and feedback. Figures 3 and 4 detail the two control components shown in Figure 1. The hourly commitment is supplied externally. Measured GPU energy updates the global planner; arrivals and execution progress update the live state captured by the local decision maker.

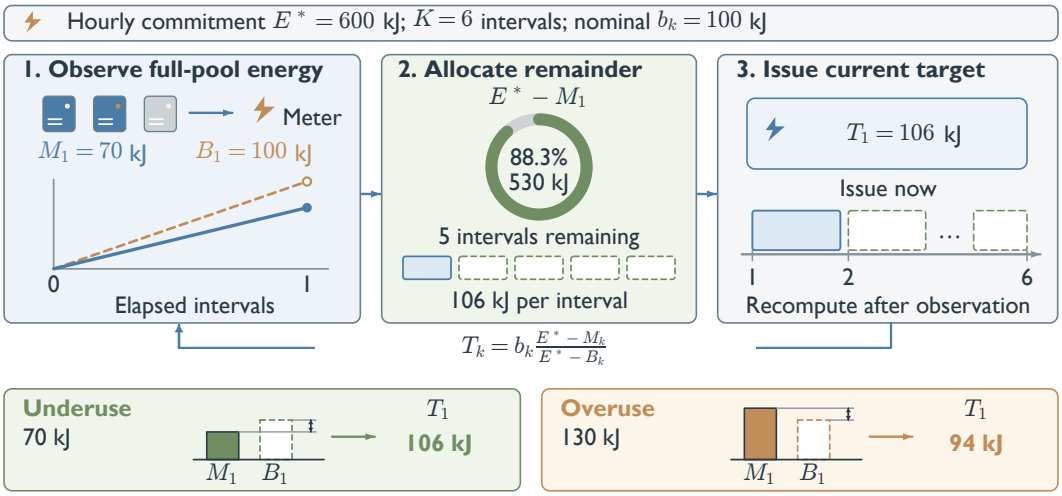

Figure 3: Global planning from measured progress to the current interval target. Dashed energy bars mark 100 kJ; future targets are recomputed after observation.

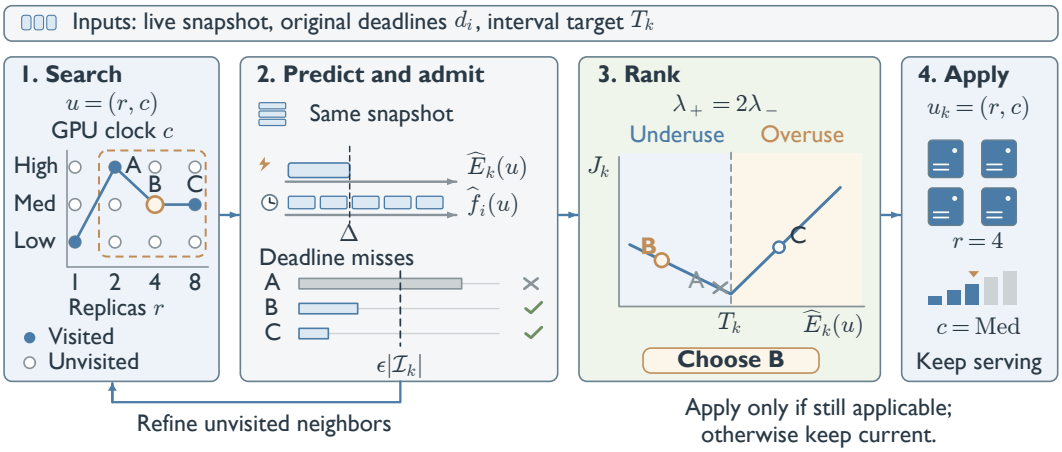

Figure 4: Local prediction and action selection. A fails the SLO check; B has lower weighted deviation cost than C despite being farther from the energy target.

Autoscaling and replica lifecycle. The lifecycle in Section 3.2 preserves in-flight execution during capacity changes. A replica selected for deactivation stops accepting new requests and drains its running requests; reactivation uses model weights retained in GPU memory rather than reloading them.

Request routing. The router assigns each incoming request to an active replica with the fewest unfinished requests. When the active-replica count changes, queued requests that have not begun execution may be redistributed among active replicas. Running requests remain on their current replicas, including those draining toward standby.

Asynchronous decisions. Each control decision uses a snapshot captured at the decision boundary. Because execution continues during search, the live state can change before a result is ready. The controller applies the result only if it remains valid for that state; otherwise, it retains the current configuration.

Native-step projection. Beyond the projection described in Section 4.3, the simulator preserves queue and cache semantics, progress feedback, and pending steps; a pending step completes before a clock transition. The current calibration combines phase-specific duration components, a decode plateau model, and empirical power corrections. A different GPU family requires new calibration of these numerical components; the controller and search alone do not establish prediction accuracy.

B Scope and Limitations

The two-sided deviation cost models contracts with unfavorable deviations, rather than every electricity tariff. A forward purchase is a financial commitment; using less does not automatically cancel it (PJM Interconnection, n.d.a). Elexon describes surplus settlement as a sale back to the system, while EDF's volume-tolerance example passes losses from lower-priced resale to customers. WAPA's DSW-EI4 schedule illustrates deficit premiums and discounted surplus credits, but is a transmission-level example, not a universal data center retail contract. Its deviation bands and monthly netting are not reproduced here. The fixed procurement payment is common to the compared policies and omitted from the objective; the reported cost models deviation exposure, not the complete electricity bill. The experimental weights in Appendix C.3 do not represent a universal market ratio.

The predicted SLO check does not guarantee actual deadlines or target reachability: standby power creates a consumption floor, insufficient work can prevent reaching a high commitment, and heavy load can require more energy than committed. ePACT accelerates useful requests rather than generating work to consume energy. Energy storage can complement serving control, but requires a separate dispatch policy and has capacity and conversion-efficiency limits (PJM Interconnection, 2026). Procurement, storage dispatch, and tariff-band optimization are outside the evaluated controller.

B.1 GPU Energy and Facility-Level Commitments

What the energy metric includes. We track the electrical energy of the controlled GPU pool, including associated GPU memory and active, draining, and standby consumption. This is device energy, not CPU, host-memory, networking, cooling, or total facility energy. NVIDIA's Management Library (NVML), also used by `nvidia-smi`, exposes device power and cumulative energy counters (NVIDIA, 2024; n.d.). The implemented measurement path integrates timestamped power observations over the accounting interval and sums energy across the pool; it retains cumulative counters for continuity checks. These readings include GPU-associated circuitry and are not per-request or per-process energy measurements. The 24-hour simulator estimates the same device-level quantity from calibrated power and execution duration rather than a facility meter.

Mapping a facility commitment to a GPU target. Power usage effectiveness (PUE) is total facility energy divided by total IT-equipment energy; its denominator includes servers, storage, and networking, not only GPUs (The Green Grid, 2012). For deployment, let F_h^* be the facility commitment in hour h, $\widehat{\rho}_h$ an expected PUE for that hour, and $\widehat{E}_h^{\text{other IT}}$ the expected IT energy outside the controlled GPU pool. An operator can allocate the residual GPU commitment as

$$E_h^{*,\text{GPU}} = \frac{F_h^*}{\widehat{\rho}_h} - \widehat{E}_h^{\text{other IT}}. \tag{4}$$

A positive residual becomes the controller's supplied E^*; a nonpositive residual requires revisiting the facility allocation. Dividing by PUE alone gives an IT budget and approximates a GPU budget only when other IT consumption is negligible or has already been budgeted separately. This mapping lets ePACT control the serving-dependent component of a facility commitment without treating GPU telemetry as a utility meter.

Operational scope of the mapping. Expected PUE and other IT demand should be estimated for the relevant load and hour, then checked against facility measurements; an annual average PUE is not an exact hourly conversion. GPU tracking translates to facility alignment only to the extent that these

estimates hold. The experiments report GPU-pool results without a PUE multiplier and do not claim measured facility-level tracking.

C Evaluation Protocol and Supplementary Results

This appendix specifies execution procedures, commitment calibration, and supplementary measurements for Section 5.

C.1 Execution and Configuration

Execution and validation scope. The single-hour comparisons use matched workloads, SLO definitions, and commitments across the five policies. The H20 decreasing-commitment matrix is the previously reported physical study. The 24-hour studies use CPU simulation with GPU-specific profiles. Policy comparisons do not independently establish prediction accuracy; Appendix C.5 reports the scope of the available single-replica validation.

Hardware and runtime. The model fits on one GPU, so all methods use tensor-parallel degree one. Beyond the shared setup in Section 5.1, vLLM 0.27.1 uses a 16,384-token context limit, a 2,048-token scheduling budget, and at most 64 sequences per replica. H20 clock levels are $\{345, 750, 1170, 1575, 1980\}$ MHz. On H200, we cap the maximum evaluated clock at 1710 MHz because preliminary runs at 1980 MHz triggered thermal throttling; Fixed-Config therefore uses 1710 MHz on H200. One Fixed-Config execution trajectory is scored under the three SLO settings on each GPU setup.

C.2 Commitment Settings and Calibration

Historical workload as expected demand. For the full-day H20 comparison, a separate historical day matching the evaluation day's weekday supplies expected hourly demand; the preceding Saturday is used before replaying May 11 (Saturday). The H200 record reuses the H20 historical target fractions but multiplies them by each evaluation-hour vLLM energy from an offline simulation. Targets are frozen before the compared controller runs. The H200 study therefore tests tracking supplied targets, not unseen-day procurement forecasting; its offline reference calculation is distinct from the information available to the online controller.

Energy required to preserve attainment. Separate calibration experiments on the historical workload identify the lowest GPU-pool energy among evaluated configurations that meet the prescribed SLO-attainment requirement. This empirical estimate guides the commitment for the corresponding hour and serving setup. The estimate is a minimum over evaluated configurations, not a globally optimal energy bound.

Targets follow workload and SLO difficulty. The full-day target fractions range from 75% to 100%, with reference energies constructed as described above. High-pressure hours use 95% or 100%, while less demanding hours admit lower commitments. Tables 4 and 5 list the fractions and absolute targets for each pool. Targets remain fixed even when replay-day demand makes exact matching difficult.

Single-hour target metadata. Both settings use SLO scales 2.5, 3, and 3.5. The fixed-commitment setting uses 85% of the corresponding full-capacity vLLM energy at every scale: 16.138982 MJ on H20 and 13.663689 MJ on H200. The decreasing-commitment setting uses 85%, 80%, and 75% on H20, and 95%, 90%, and 85% on H200. The latter H200 targets are 15.271181, 14.467435, and 13.663689 MJ, respectively. Each target is fixed throughout its run, not decreased online as requests arrive. The two settings compare policies under different prescribed objectives, rather than estimating a minimum feasible energy budget.

Runs for the two single-hour settings. The decreasing-commitment H20 results retain the previously reported runs and subsequent corrected entries; they are not obtained by re-settling the fixed-commitment controller runs. In the decreasing-commitment H200 setting, vLLM (32 active replicas at 1710 MHz) and Fixed-Config (28 at 1710 MHz) each use a single execution trajectory scored at the three SLO scales. ePACT, Static-Target, and DynamoLLM are executed separately at each scale. The H200 fixed- and decreasing-commitment controller runs are distinct even where their target values coincide.

C.3 Measurement and Reporting

Matched execution and metering. Comparisons use the same workload, resident capacity, clock catalog, deadlines, commitments, and settlement boundaries. The GPU measurement boundary is defined in Appendix B.1. For the single-hour study, SLO attainment includes all arrivals and their full completion tails, whereas energy settlement covers only the first 3,600 seconds. Post-hour energy is excluded from hourly deviation and accounted for separately. Single-run attainment differences do not establish statistical significance.

Continuous tracking across hourly boundaries. In the full-day replay, queued and running requests, original deadlines, and serving state persist across hourly boundaries; only the commitment and its ledger change. A request's SLO outcome belongs to its arrival hour, even if it completes later, while energy belongs to the hour in which it is consumed. Hours 0–23 are trace-relative indices. The replay is continuous, not 24 independently initialized runs.

Hourly deviation costs. We report a nonmonetary, energy-equivalent deviation cost. Let E_h and E_h^* be actual GPU-pool energy and its commitment in hour h, both in kWh. Dimensionless weights of 2 for overuse and 1 for underuse give

$$C_h = 2\max(E_h - E_h^*, 0) + \max(E_h^* - E_h, 0), \tag{5}$$

The daily score $\sum_{h=0}^{23} C_h$ has units of kWh-equivalent: it is weighted deviation, not physical energy consumption or a currency charge. The weights express an experimental asymmetry without assuming an electricity price. Equivalently, this score is the monetary objective in Equation (1) divided by a common reference price per kWh, with a fixed overuse-to-underuse rate ratio. Multiplying by that reference price would restore monetary units; actual tariffs may require different weights or a different settlement function. No fixed procurement payment is added, and deviations in different hours do not cancel.

Daily energy totals. For the H20 24-hour study, the supplied totals are 496.1070 MJ in commitments and 538.5299, 506.9002, and 453.9654 MJ consumed by vLLM, ePACT, and DynamoLLM, respectively. These totals provide context, not the settlement metric: Table 1 sums hourly costs rather than using a net daily difference. Hourly entries are rounded independently and need not sum exactly to the full-precision daily totals.

C.4 Supplementary Figures and Tables

Tables 2 and 3 report rounded single-hour values for Figure 2. Figures 5 and 6, together with Tables 4 and 5, provide the detailed 24-hour views for H20 and H200.

Table 2: Single-hour H20 results under both commitment settings. Percentages are rounded; deviation is signed. Bold highlights ePACT, not column maxima.

	2.5× SLO		3× SLO		3.5× SLO	
Method	Attain. (%)	Dev. (%)	Attain. (%)	Dev. (%)	Attain. (%)	Dev. (%)
Fixed commitment						
vLLM	97.30	+17.65	98.49	+17.65	99.00	+17.65
ePACT	**97.31**	−0.09	**98.46**	−0.02	**99.02**	−0.02
DynamoLLM	72.75	−9.32	71.88	−7.89	75.59	−10.38
Static-Target	97.39	+15.58	98.45	+15.49	98.97	+15.57
Fixed-Config	96.01	+12.34	97.66	+12.34	98.55	+12.34
Decreasing commitment						
vLLM	97.30	+17.65	98.28	+25.00	99.00	+33.33
ePACT	**97.31**	−0.09	**98.34**	+1.22	**99.06**	+2.18
DynamoLLM	72.75	−9.32	71.88	−2.14	75.59	−2.34
Static-Target	97.39	+15.58	98.45	+18.35	99.05	+20.11
Fixed-Config	96.01	+12.34	97.66	+19.36	98.55	+22.43

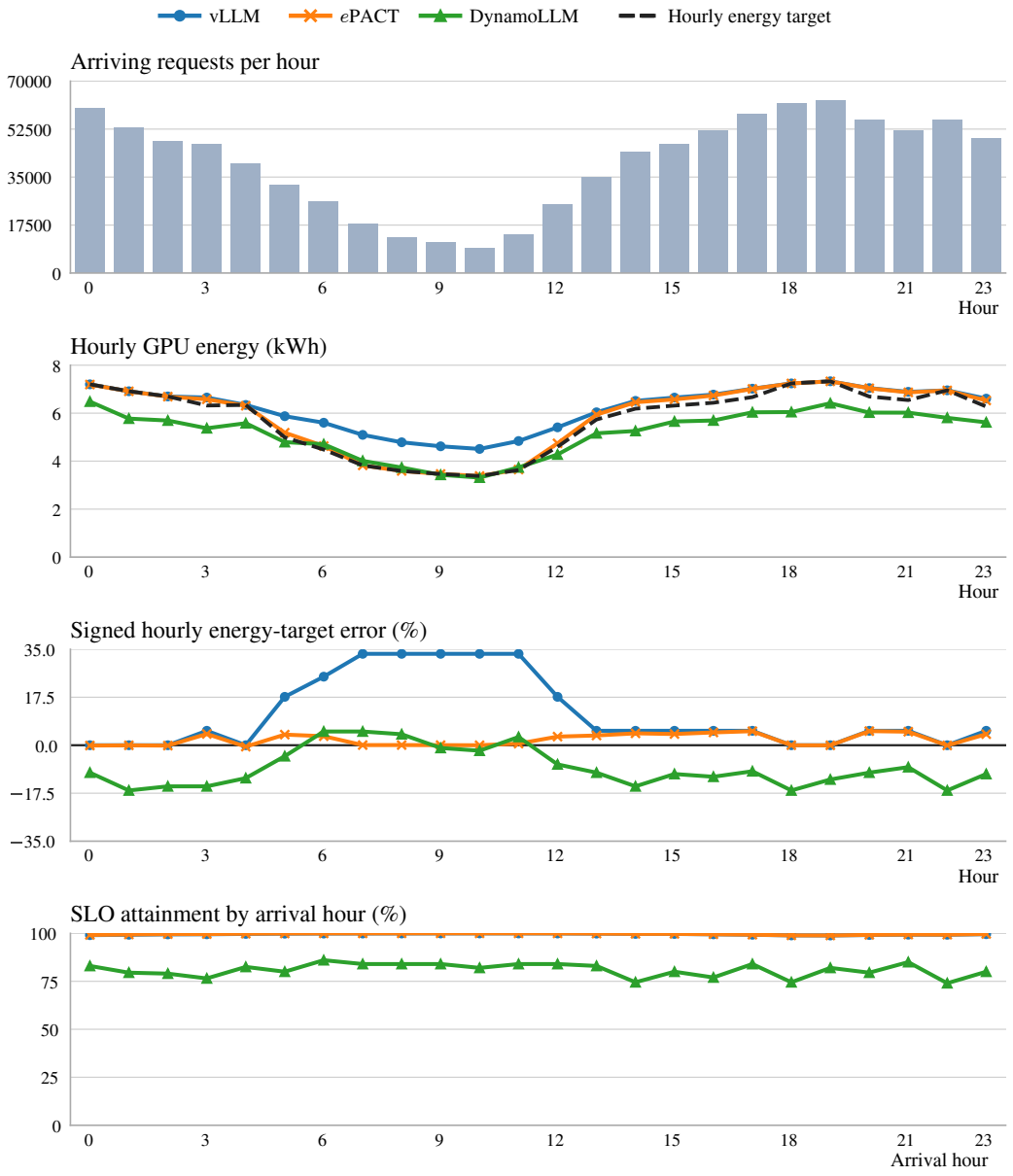

Figure 5: Hourly overview of the 24-hour H20 comparison at fixed 3× SLO. The panels show arrivals, GPU energy, signed commitment deviation, and SLO attainment by arrival hour.



Figure 6: Hourly overview of the 24-hour H200 comparison at fixed 3× SLO. The panels show arrivals, GPU energy, signed commitment deviation, and SLO attainment by arrival hour.

Table 3: Single-hour H200 results under both commitment settings. Percentages are rounded; deviation is signed. Bold highlights ePACT, not column maxima.

	2.5× SLO		3× SLO		3.5× SLO	
Method	Attain. (%)	Dev. (%)	Attain. (%)	Dev. (%)	Attain. (%)	Dev. (%)
Fixed commitment						
vLLM	99.64	+17.65	99.77	+17.65	99.86	+17.65
ePACT	**99.57**	+3.46	**99.79**	+3.29	**99.77**	+1.00
DynamoLLM	76.43	−4.72	79.05	−7.28	87.12	−9.50
Static-Target	99.67	+16.01	99.86	+15.41	99.89	+16.69
Fixed-Config	99.39	+14.94	99.68	+14.94	99.78	+14.94
Decreasing commitment						
vLLM	99.64	+5.26	99.77	+11.11	99.86	+17.65
ePACT	**99.64**	−0.01	**99.81**	−0.05	**99.82**	+1.08
DynamoLLM	76.34	−16.13	81.81	−12.71	83.22	−9.34
Static-Target	99.70	+4.37	99.82	+9.43	99.90	+14.93
Fixed-Config	99.39	+2.84	99.68	+8.55	99.78	+14.94

Table 4: ePACT's hourly H20 results at SLO scale 3 on the May 11, 2024 trace. Target fractions refer to the historical vLLM calibration. Attainment groups requests by arrival hour; positive deviation indicates overuse. Hourly values are rounded as reported.

Hour	Target fraction	Target (MJ)	Attainment (%)	Deviation (%)
00	100%	25.928	99.15	−0.12
01	100%	24.889	99.41	−0.04
02	100%	24.111	99.60	−0.14
03	95%	22.747	99.62	+4.10
04	100%	22.833	99.81	−0.52
05	85%	17.975	99.92	+3.87
06	80%	16.133	99.96	+3.29
07	75%	13.758	99.97	+0.05
08	75%	12.923	99.99	+0.06
09	75%	12.471	100.00	+0.03
10	75%	12.175	100.00	−0.01
11	75%	13.066	99.99	+0.58
12	85%	16.552	99.97	+3.14
13	95%	20.643	99.87	+3.55
14	95%	22.272	99.72	+4.33
15	95%	22.737	99.72	+4.08
16	95%	23.164	99.52	+4.66
17	95%	24.009	99.29	+5.12
18	100%	26.066	98.95	−0.01
19	100%	26.382	98.90	−0.04
20	95%	24.095	99.20	+5.14
21	95%	23.560	99.37	+4.93
22	100%	25.027	99.31	−0.08
23	95%	22.592	99.66	+4.03

Table 5: ePACT's hourly H200 results at SLO scale 3 on the May 11, 2024 trace. Target fractions scale the offline evaluation-day vLLM reference. Attainment groups requests by arrival hour; positive deviation indicates overuse. Hourly values are rounded as reported.

Hour	Target fraction	Target (MJ)	Attainment (%)	Deviation (%)
00	100%	21.176	99.92	−0.49
01	100%	19.742	99.95	−1.22
02	100%	19.137	99.97	−1.56
03	95%	17.833	99.94	−0.13
04	100%	17.833	99.99	−1.57
05	85%	14.091	99.92	+1.12
06	80%	12.664	99.91	+3.39
07	75%	10.911	99.93	+5.80
08	75%	10.296	99.91	+3.66
09	75%	9.978	99.89	+1.03
10	75%	9.777	99.86	+1.37
11	75%	10.382	99.93	+3.46
12	85%	13.028	99.92	+1.49
13	95%	16.184	99.98	+2.61
14	95%	17.324	99.97	+1.53
15	95%	17.785	99.97	+2.79
16	95%	18.328	99.98	+3.88
17	95%	19.265	99.92	+4.26
18	100%	21.374	99.89	−0.00
19	100%	21.867	99.89	−0.07
20	95%	19.162	99.92	+3.25
21	95%	18.764	99.97	+5.26
22	100%	20.379	99.91	−0.20
23	95%	17.896	99.97	+5.26

C.5 Prediction Accuracy and Control Overhead

We validate frozen prediction models on held-out workloads and report planning time.

Validation without refitting. The first validation uses five workload classes excluded from fitting, repeated twice, on a single replica at 1980 MHz with fixed batches. Class-level end-to-end latency WAPE ranges from 0.44% to 2.66%, and energy WAPE ranges from 0.19% to 2.89%. The largest request-level relative latency error among 18 requests is 3.12%; the worst evaluated energy cell has 3.28% error. The duration and power models are frozen before evaluation and are not refitted using the held-out measurements.

Continuous-arrival validation. A second validation uses 97 requests from a nonoverlapping slice of the same trace, excluded from the corresponding fit, with continuous arrivals and continuous batching on one replica. End-to-end latency WAPE is 4.15%, the 95th percentile of per-request relative prediction error is 14.74%, and 84 of 97 predictions fall within $\pm 5\%$ of measured latency. The signed energy error over the 120-second window is $+0.76\%$. These checks characterize the stated single-replica settings; they do not establish an error bound for every 32-replica configuration or full-day trajectory.

Interpreting prediction errors. For measured values y_j and predictions $\widehat{y}_j$, WAPE is $100\frac{\sum_j |\widehat{y}_j - y_j|}{\sum_j y_j}$, computed separately for latency and energy. It is an aggregate error, not the fraction of requests within 5% error. The continuous-arrival energy result is a signed window error, not an energy-WAPE measurement across multiple windows.

Observed control and batching behavior. The reported ePACT run completes 60 asynchronous decisions with a mean search wall time of 23.35 seconds. Serving continues while the search runs; this measurement does not by itself establish that planning has no latency impact. Batch sizes 1, 2, 3, and 4 occur in 265,295, 113,992, 38,503, and 1,199 steps, respectively, confirming that execution is not restricted to batch size one.