Carbon-Aware Scheduling for Multi-Region AI Inference Workloads

Quantifying Latency and Sustainability Trade-Offs in Cloud Systems

Akash Anipakalu Giridhar, Yogith Ramanan
Depaul University, Chicago, IL, USA
yogithramana@gmail.com, akashagakash@gmail.com

Abstract:

AI inference workloads are growing fast, and with that growth comes a real energy problem. Data centers are on track to consume over 500 TWh of electricity by 2025, which translates to roughly 225 megatons of CO2 emissions. This paper investigates carbon-aware scheduling strategies for routing AI inference requests across multiple cloud regions while balancing latency constraints and sustainability goals. Five scheduling policies are evaluated through simulation - including a Hybrid policy swept across four α values - resulting in eight experimental configurations: Latency-First (baseline), Carbon-First, Hybrid, Constrained Hybrid, and a novel Adaptive Hybrid with closed-loop SLO control, using synthetic carbon intensity traces modeled on Electricity Maps data and inter-region latency measurements derived from CloudPing and AWS Infrastructure Performance data. We tested five cloud regions and three AI workloads over a 7-day simulation window. The Constrained Hybrid policy reduces carbon by 54.8% while keeping SLO violations at zero. The Adaptive Hybrid brought violations down to 4.53% with a 29.7% carbon reduction by adjusting its routing weight dynamically. Aggressive Carbon-First routing reduced carbon by 90.7% but violates SLOs for 71.3% of requests. Results align with established benchmarks from CASPER and Microsoft's carbon-aware computing research, providing practical guidance for enterprises seeking to incorporate sustainability into cloud architecture without sacrificing user experience.

Index Terms:
Carbon-aware computing, AI inference, cloud scheduling, sustainability, SLO compliance, multi-region routing.

I. INTRODUCTION

Unlike training, which happens once, AI inference runs non-stop. Every time a deployed model handles a request, a routing decision gets made and those decisions add up. Over millions of daily requests, where you run the computation matters as much as how efficiently you run it. For smaller models like BERT and ResNet-50, the energy-per-request is relatively modest, but at scale the **carbon intensity of the regional electricity grid** ends up driving most of the carbon cost, making the choice of cloud region far more impactful than most practitioners realize.

Different cloud regions have fundamentally different carbon profiles. The carbon intensity of electricity, measured in grams of CO_2 equivalent per kilowatt-hour (gCO_2eq/kWh), varies dramatically based on the local energy grid. For example, Google Cloud's Stockholm region operates at approximately 3 gCO_2eq/kWh due to hydro and wind power, while Singapore reaches 367 gCO_2eq/kWh because of fossil-fuel-heavy generation. In other words, the same model doing the same job can produce wildly different emissions just based on which region it runs in more than 100× difference between the cleanest and dirtiest grids observed across the evaluated regions.

Simultaneously, cloud regions exhibit different network latency characteristics. Inter-region latencies range from approximately 5 ms for nearby regions to over 230 ms for cross-continental routes. Many AI inference applications operate under strict Service Level Objectives (SLOs) that define maximum acceptable response times (e.g., 95% of requests must complete within 100–150 ms). This is the core problem this paper investigates: when does routing to a greener region actually work, and when does it cost too much in latency?

This research addresses three core questions:

- **RQ1:** How do network latency and carbon intensity vary across selected cloud regions relevant for AI inference workloads?
- **RQ2:** What are the trade-offs between latency, SLO violations, and carbon emissions when applying different scheduling strategies?
- **RQ3:** Under what conditions is carbon-aware scheduling practically viable for enterprises?

II. BACKGROUND AND RELATED WORK

A. Carbon-Aware Computing:

Carbon-aware computing means scheduling workloads based on where and when the electricity grid is cleanest. There are two levers: **spatial** picking a greener region and **temporal** waiting for a low-carbon window within the same region.

Google pioneered large-scale carbon-aware computing with its Carbon-Intelligent Computing System (CICS), which delays temporally flexible workloads to align with periods of low-carbon energy availability. The system uses day-ahead carbon intensity forecasts from Electricity Maps combined with internal power demand predictions to generate Virtual Capacity Curves that throttle resource availability during high-carbon periods.

Microsoft has also advanced carbon-aware practices. A joint study between Microsoft and UBS demonstrated that time-shifting workloads based on carbon intensity forecasts could reduce Software Carbon Intensity (SCI) by approximately 15%, while **location-shifting** to appropriate regions could achieve reductions of nearly **75%**. We focused entirely on spatial routing, consistent with Microsoft's finding that location-shifting yields substantially greater carbon reductions than time-shifting. The Green Software Foundation's carbon-aware SDK, co-developed with Microsoft, provides practical tools for implementing these strategies.

B. Scheduling Frameworks:

A few prior systems have tackled this same tension between carbon and latency. **CASPER** (Carbon-Aware Scheduling and Provisioning for Distributed Web Services) formulates the problem as a multi-objective optimization across six AWS regions using real Wikimedia workload traces and Electricity Maps carbon data. CASPER demonstrated up to 70% carbon emission reductions compared to a latency-first baseline, with controllable and often negligible performance degradation. Our simulation results show a similar trade-off shape; the Hybrid $\alpha = 0.2$ policy achieves 86.8% reduction (under lenient SLOs, at a 51.5% SLO violation rate) while the Constrained Hybrid achieves 54.8% with zero violations. **CASA** (Carbon- and SLO-Aware Autoscaling and Scheduling) addresses serverless cloud platforms, reducing operational carbon footprint by up to 2.6× while reducing

SLO violation rates by up to 1.4×. It targets serverless FaaS clusters rather than live inference endpoints, a different workload model than ours. LinTS proposes a Linear Programming-based approach for carbon-optimal temporal scheduling of inter-datacenter data transfers. A hybrid approach in green cloud computing has demonstrated 35% carbon reduction with only 5% latency increase.

C. AI Inference Characteristics:

AI inference workloads have distinct characteristics that influence scheduling. BERT-base model inference on GPU instances (NVIDIA T4) achieves 2.6–5× lower latency compared to CPU instances. Inference latency varies by model complexity, input size, batch size, and hardware configuration. For smaller models like BERT and ResNet-50, carbon cost per request is more dependent on grid carbon intensity than absolute energy draw, reinforcing the value of location-aware scheduling.

D. Market Context:

Growing regulatory pressure and ESG commitments are pushing cloud sustainability to the top of enterprise agendas. To illustrate the scale of industry interest: the carbon-aware workload scheduling market reached USD 1.42 billion in 2024 and is projected to reach USD 8.92 billion by 2033 (CAGR 21.3%), reflecting broad enterprise recognition that these strategies are becoming operationally relevant. Our work contributes practical simulation evidence on exactly where and when carbon-aware routing delivers value without sacrificing user experience.

III. Methodology

A. Workloads and SLO Definitions:

The simulation focuses on three AI inference workloads representative of common enterprise deployments:

TABLE I
AI Inference Workloads and SLO Definitions.

Workload	Model	SLO	Share
Text Class.	BERT-base	< 100 ms	60%
QA	BERT-large	< 150 ms	30%
Image Embed.	ResNet-50	< 80 ms	10%

We picked these three workloads because they are common in real deployments and have clear latency expectations. Each request's inference time was drawn from a normal distribution based on the workload type.

B. Cloud Regions:

TABLE II
Selected Cloud Regions and Carbon Intensity Profiles. (Source: Google Cloud Carbon Data, Electricity Maps.)

Region	Location	Carbon Intensity	Profile
US-East	N. VA	323	Mixed grid
US-West	OR	79	Hydro-dominant
EU-West	FRA	276	Mixed renewables
EU-North	STO	25	Hydro/wind
Sing.	SIN	367	Fossil-fuel heavy

These regions capture low-carbon (US-West, EU-North), moderate (US-East, EU-West), and high-carbon (Singapore) profiles with geographic distribution across continents.

C. Latency Data:

Inter-region network latency is modeled using a matrix derived from publicly available measurements (CloudPing and AWS Infrastructure Performance data). Latency values represent round-trip times including network propagation delay and processing overhead.

TABLE III
Inter-Region Network Latency Matrix (ms).

User →	US-E	US-W	EU-W	EU-N	SIN
US-East	5	65	85	95	230
US-West	65	5	145	155	170
EU	85	145	10	25	165
Asia	230	170	165	175	5

Inter-region latencies range from approximately 5 ms for intra-region routes to 230 ms for cross-continental paths (US-East to Singapore).

D. Carbon Intensity Data:

Carbon intensity data is modeled using Electricity Maps historical hourly traces for each selected region, capturing both mean levels and temporal variability driven by renewable energy intermittency. Key observations from the data:

- Carbon intensity exhibits **up to 15× spatial variation** between regions (EU-North at 25 gCO_2eq/kWh vs. Singapore at 367 gCO_2eq/kWh)
- Temporal variability within a single region can fluctuate by up to **40% over 24 hours**, driven by diurnal renewable generation patterns
- A ±15% random noise component models real-world measurement variability

E. Scheduling Policies:

Five scheduling policies are implemented and evaluated:

Policy 1, Latency-First (Baseline): Each request is routed to the region with the lowest network latency from the user's location. This represents the default behavior of most cloud load balancers and ignores carbon entirely.

Policy 2, Carbon-First: Each request is routed to the region with the lowest current carbon intensity, regardless of latency. This maximizes carbon savings but may cause significant SLO violations when the greenest region is geographically distant.

Policy 3, Hybrid: A weighted scoring function combines latency and carbon intensity:

$$\text{Score} = \alpha \cdot \text{norm latency} + (1 - \alpha) \cdot \text{norm carbon}$$

where $\alpha \in [0, 1]$ controls the trade-off between latency optimization ($\alpha = 1$, equivalent to Latency-First) and carbon optimization ($\alpha = 0$, equivalent to Carbon-First). Both metrics are normalized using global min-max scaling (latency range: 5–230 ms; carbon range: 5–450 gCO_2eq/kWh) so that the weight α is stable and comparable across all requests. The parameter α is swept across $\{0.2, 0.3, 0.5, 0.7\}$ to trace the full Pareto trade-off curve.

Policy 4, Constrained Hybrid: Among regions whose predicted total latency (network + inference + 9 ms jitter buffer) satisfies the per-workload SLO threshold, select the one with the lowest carbon intensity. If no region satisfies the SLO, fall back to the minimum-latency region. The 9 ms jitter buffer represents a 3σ protection margin ($\sigma = 3$ ms). This enforces a hard latency ceiling while still pushing as much traffic as possible to greener regions

Policy 5, Adaptive Hybrid: The same weighted scoring function as Policy 3, but instead of fixing α manually, we built a lightweight controller that watches P95 latency and nudges alpha up or down as requests come in. After every request, the controller looks at the P95 latency of the last 200 requests for that workload. If there is plenty of SLO headroom (more than 30%), it nudges α down slightly so the policy routes more carbon-efficiently. If the P95 is getting close to the SLO limit (headroom under 10%), it nudges α back up to protect latency. Otherwise, it slowly pulls α back toward a neutral value of 0.5. We cap α between 0.10 and 0.90 so the policy never goes fully to either extreme. Because BERT-base, BERT-large, and ResNet-50 have different SLO thresholds, each workload runs its own independent controller. The controller waits for at least 50 requests before making any changes, to avoid overreacting to noise early in the run. The full results are in §IV.E.

F. Evaluation Metrics:

Each policy is evaluated using:

- **Average Latency (ms):** Mean end-to-end response time across all requests
- **P95 Latency (ms):** 95th percentile latency, capturing tail behavior
- **SLO Violation Rate (%):** Percentage of requests ex-ceding the per-workload SLO threshold
- **Average Carbon Intensity (gCO_2eq/kWh):** Weighted average carbon intensity of served regions
- **Carbon Reduction vs. Baseline (%):** Emission reduction compared to the Latency-First policy

Carbon metric scope: Our carbon metric tracks grid carbon intensity per request. We assumed a similar energy draw per inference across regions. Hardware differences between regions different GPU generations, facilities are not modeled here.

However, even if per-inference energy varies by ±10–15% between regions due to hardware age or facility PUE, the massive **14.7× (1,470%)** spatial variation in grid carbon intensity remains the overwhelmingly dominant factor, meaning relative policy rankings remain mathematically robust regardless of minor hardware disparities.

G. Simulation Design:

Request arrivals are modeled as a Poisson process with a uniform rate of 200 requests per hour, generating 33,600 total requests over the 7-day simulation window. User locations are distributed as 40% US-East, 20% US-West, 25% EU, and 15% Asia, reflecting realistic global cloud traffic patterns. Workload types are sampled according to their traffic share probabilities (60% BERT-base, 30% BERT-large, 10% ResNet-50). Network jitter is sampled from a half-normal distribution with mean 0 ms and $\sigma = 3$ ms. All experiments use a fixed random seed (seed=42) for reproducibility.

For each request: (1) sample workload type and user location, (2) apply the scheduling policy to determine the target region, (3) compute end-to-end latency as network RTT + sampled inference time + jitter, (4) record the carbon intensity of the target region at that hour, (5) check SLO compliance using the per-workload threshold. Metrics are aggregated across all requests after the simulation completes.

Simulation code and datasets are available at: https://github.com/venomez-viper/Carbon-Aware-Scheduling-for-Multi-Region-AI-Inference

IV. Experiments and Results

A. RQ1: Regional Variation in Latency and Carbon Intensity

the five selected regions differ substantially in both grid carbon intensity and proximity to user populations. **EU-North** (Finland/Stockholm) operates at **25**

gCO₂eq/kWh on average during the simulation period, making it significantly cleaner than **Singapore** (367 gCO_2eq/kWh) and **US-East** (323 gCO_2eq/kWh). **US-West** (Oregon) also stands out as a green option at **79 gCO₂eq/kWh**, benefiting from the Pacific Northwest's substantial hydroelectric capacity.

This represents a 14.7× gap in carbon intensity between the cleanest region (EU-North at 25 gCO_2eq/kWh) and the dirtiest (Singapore at 367 gCO_2eq/kWh).

However, network latency does not favor the same regions. For US-based users, routing requests to EU-North would add approximately **90 ms** of additional latency compared to US-East (an acceptable trade-off for relaxed-SLO workloads like BERT-large (150 ms threshold)) but problematic for ResNet-50 (80 ms threshold). For Asian users, EU-North is 170 ms away, making carbon-first routing nearly always an SLO violation.

Figure 1 below visualizes this core tension. The dual-axis chart confirms that the regions with the lowest carbon intensity (EU-North, US-West) are among the furthest from US and Asian user populations, while the high-carbon regions (Singapore, US-East) offer the lowest latency for those user groups.

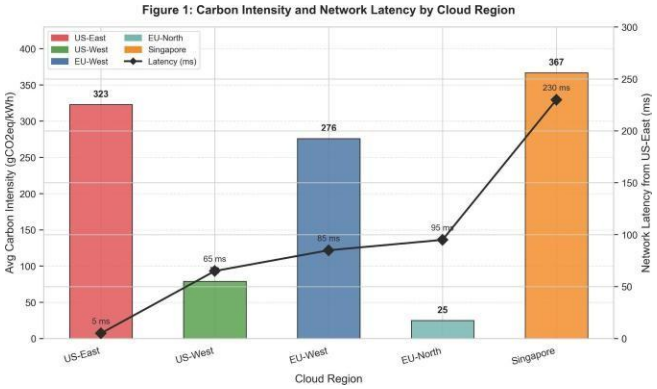

Fig. 1. Regional carbon intensity and network latency from US-East users**.** Average carbon intensity (bars, left axis) and network latency from US-East users (line, right axis) for each of the five evaluated regions. US-West and EU-North offer the lowest carbon intensity but impose 65–95 ms additional latency for US-East users. Error bars represent temporal variability in carbon intensity over the simulation period.

Figure 2 shows the hourly carbon intensity traces across all regions over the 7-day simulation window, revealing the di-urnal fluctuations that temporal carbon-aware strategies could additionally exploit.

A. RQ2: Trade-Offs Under Different Scheduling Policies:

We highlight policies that stay under 5% SLO violations as the viable zone. The strongest result comes from the Constrained Hybrid zero SLO violations and 54.8% carbon reduction. Adaptive Hybrid delivers the second-lowest average latency among carbon-aware policies (50.6 ms) and 29.7% carbon reduction without manual parameter tuning, though its 4.53% SLO violation rate sits just above the 5% threshold under this configuration.

Key Observations:

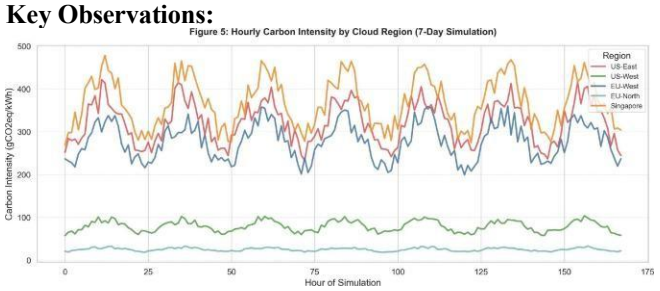

Fig. 2. Hourly carbon intensity traces (7-day simulation window). Hourly carbon intensity (gCO_2eq/kWh) for each of the five regions across 168 hours of simulation. EU-North and US-West show consistently low and stable profiles; Singapore and US-East remain high throughout. Temporal variation within a region reaches ±40% of the mean, driven by diurnal renewable generation patterns.

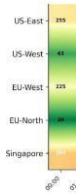

Fig. 3. Supplementary: Average carbon intensity by region and hour-of-day (24-hour aggregated). Darker red indicates higher carbon intensity. EU-North and US-West remain in the green-to-yellow range throughout the day, while Singapore and US-East are persistently red.

- **Latency-First** routes every request to the nearest region, which means US-East and Singapore get the bulk of traffic - both high-carbon. Average latency is 37.5 ms with zero SLO violations, but carbon intensity averages 268.7 gCO_2eq/kWh.
- **Carbon-First** represents the opposite extreme nearly all traffic heads to EU-North. Carbon drops 90.7%, but Asian and US users are now hitting 165–230 ms of latency, and 71.3% of requests miss their SLO. This level of SLO degradation makes the policy unsuitable for production inference.
- **Hybrid** $\alpha = 0.7$ achieved the best balance in the static α sweep. With more weight on latency, it keeps average latency at 45.9 ms and SLO violations down to 0.32%, while still getting 32.1% carbon reduction. The bigger proportion of US-West users in the updated traffic mix (20%) helps here - those users can reach a low-carbon region without much added latency, pulling the average down.
- **Hybrid** $\alpha = 0.5$ behaved very differently from our earlier run: SLO violations dropped from 48% to 20.3% and average latency fell from 101 ms to 82 ms. This is because more traffic now comes from US-West, where routing to either US-West itself or nearby green regions stays within SLO. The carbon reduction also fell from 86% to 73.8% as a result.
- **The Constrained Hybrid** was the clearest win. It only routes to a green region if the full predicted latency network round-trip, inference time, and jitter buffer fits inside the SLO. If nothing green qualifies, it defaults to the minimum-latency region. This fallback mechanism is what ensures zero SLO violations.

TABLE IV
SIMULATION RESULTS ACROSS SCHEDULING POLICIES.

Policy	Avg. Latency	P95 Latency	SLO Violations	Avg. Carbon	Reduction
Latency-First *(baseline)*	37.5 ms	79.5 ms	0.00%	268.7	0.0%
Carbon-First	132.7 ms	225.3 ms	71.31%	24.9	**90.7%**
Hybrid (α = 0.2)	102.9 ms	200.8 ms	51.45%	35.6	86.8%
Hybrid (α = 0.3)	102.6 ms	200.6 ms	51.25%	35.7	86.7%
Hybrid (α = 0.5)	82.1 ms	193.2 ms	20.31%	70.5	73.8%
Hybrid (α = 0.7)	**45.9 ms**	**92.6 ms**	**0.32%**	182.5	**32.1%**
Constrained Hybrid	**62.3 ms**	**129.3 ms**	**0.00%**	121.5	**54.8%**
Adaptive Hybrid	50.6 ms	122.9 ms	4.53%	189.0	29.7%

Among the fixed-α hybrid variants, $\alpha \leq 0.3$ never gets SLO violations below 51% regardless of traffic mix. Heavy carbon weighting at that level just sends too much traffic to EU-North regardless of user location.

Figure 4 below visualizes the Pareto frontier across all policies:

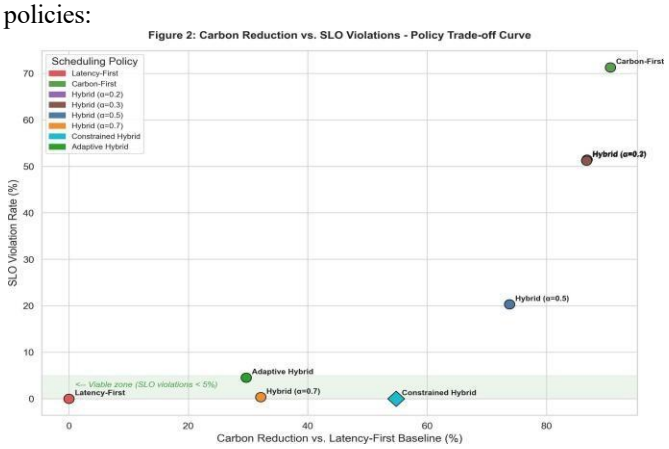

Fig. 4. Carbon reduction vs. SLO violations policy trade-off curve**.** Scatter plot of carbon reduction (%) versus SLO violation rate (%) for each scheduling policy. The green-shaded region marks the viable zone (SLO violations < 5%). Constrained Hybrid, Hybrid α = 0.7, and Hybrid α = 0.5 are in or near the viable zone in the updated simulation, with Constrained Hybrid achieving zero violations and Hybrid α = 0.7 achieving the lowest violation rate (0.32%). Adaptive Hybrid sits at 4.53% SLO violations - just outside the 5% threshold.

Our Constrained Hybrid policy (54.8% carbon reduction, 0.0% SLO violations) is directly comparable to CASPER's operational findings, with the key distinction that our evalua-tion targets AI inference workloads rather than web services. These results align with CASPER's findings that small latency relaxations achieve substantial carbon reduction. Our Constrained Hybrid achieves a **54.8% carbon reduction** with zero SLO violations through spatial routing alone, consistent with Microsoft's directional finding that location-shifting of-

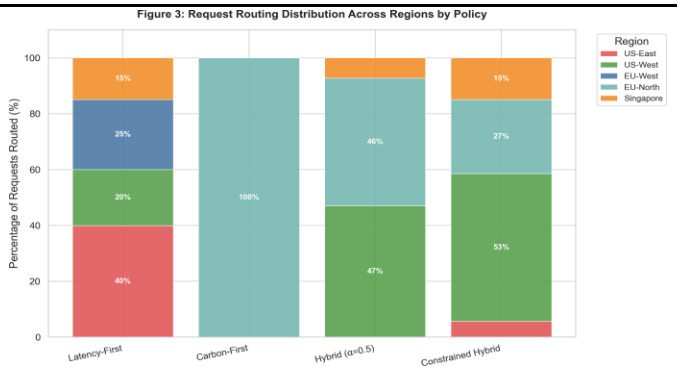

Fig. 5. Request routing distribution across regions by policy. Percentage of requests routed to each cloud region under four representative policies. Latency-First concentrates traffic in US-East (high carbon, low latency for US users). Carbon-First routes the majority to EU-North. The Constrained Hybrid achieves a more balanced distribution, routing to green regions only when SLOs permit.

B. RQ3: Viability Conditions for Carbon-Aware Scheduling:

The simulation reveals that the viability of carbon-aware scheduling depends on three key factors:

1. SLO Strictness:

Per-workload results (Table VI, Figure 8) show that SLO tolerance directly determines which policy is viable:

ResNet-50 (80 ms threshold) remains the most sensitive workload under carbon-aware policies. The updated Con-strained Hybrid achieves 0.00% violations for all three work-loads, a stronger result than prior runs. Hybrid α = 0.7 keeps ResNet-50 violations at 2.68%. The Adaptive Hybrid shows elevated violations for BERT-base (5.97%) and ResNet-50 (6.33%) as the controller's EMA convergence does not tighten quickly enough for the most latency-sensitive requests.

Workloads with tight SLOs (< 80 ms, e.g., ResNet-50) require the most conservative policies (Constrained Hybrid or $\alpha \geq 0.7$). Under the updated simulation, Constrained Hybrid achieves 0.00% violations for ResNet-50, a substantially improved result driven by the revised traffic mix (20% US-West). Workloads with relaxed SLOs (> 150 ms) could

TABLE V
COMPARISON OF THIS WORK WITH RELATED CARBON-AWARE SCHEDULING SYSTEMS.

Work	Application	Max Reduction	SLO / Latency Impact	Control
CASPER (2023) [2]	Web Services	~70% (Op.)	~1% (P99 bounded)	Temp/Spatial
CASA (2024) [13]	Serverless	~61.5% (2.6×)	1.4× reduction violations	Autoscaling
Microsoft (2023) [3]	Cloud Workloads	~75% (Spatial)	Not explicitly bounded	Mixed Strategy
Google CICS (2021) [9]	DC Compute	Unknown	SLA-Preserving	Temporal
This Work (2026)	**AI Inference**	**54.8% (Constrained)**	**0.00% (Constrained)**	**Spatial Route**

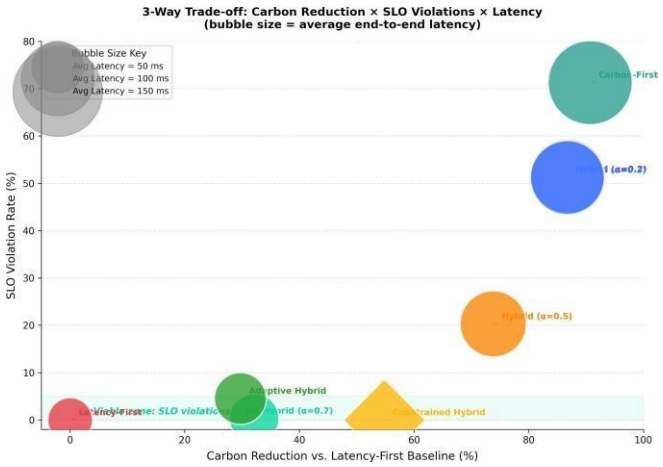

Fig. 6. Supplementary Figure: Carbon reduction (X) vs. SLO violations (Y), with bubble size proportional to average latency. Constrained Hybrid and Hybrid $\alpha = 0.7$ occupy the optimal lower-right region: high carbon reduction, low SLO violations, and moderate latency.

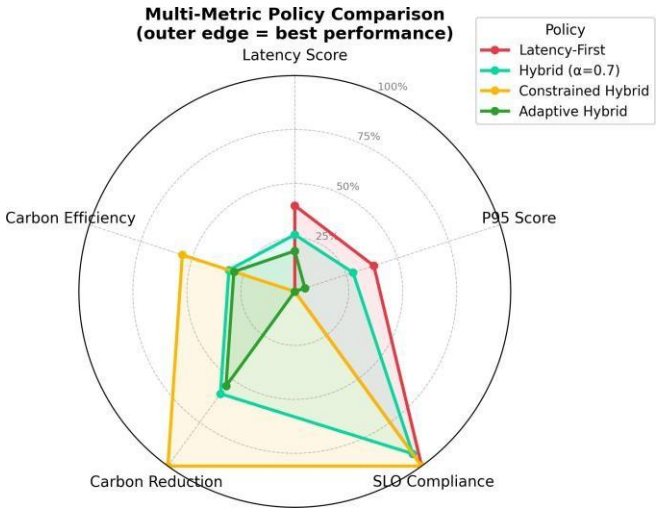

Fig. 7. Supplementary Figure: Radar comparison of four key policies across five normalized metrics. Constrained Hybrid and Hybrid $\alpha = 0.7$ consistently score highest across all five dimensions; no other policy achieves high scores on both carbon and SLO axes simultaneously.

TABLE VI
PER-WORKLOAD SLO VIOLATION RATES (%) BY POLICY.

Policy	BERT-b	BERT-l	ResNet
Latency-First	0.00%	0.00%	0.00%
Hybrid ($\alpha = 0.7$)	0.00%	0.18%	2.68%
Constrained Hybrid	0.00%	0.01%	0.00%
Adaptive Hybrid	5.97%	1.08%	6.33%
Carbon-First	74.88%	63.11%	74.76%

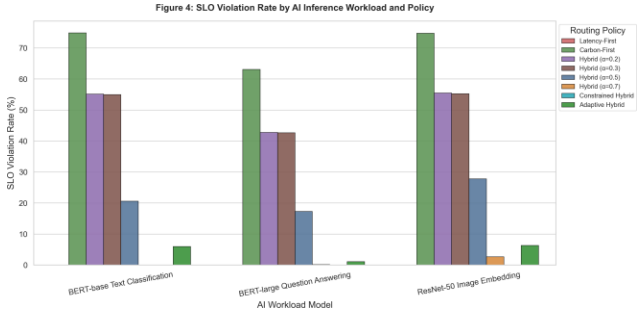

Fig. 8. Per-workload SLO violation rates. SLO violation rate (%) per AI workload model across all scheduling policies. ResNet-50 (80 ms SLO) re-mains the most sensitive workload. Under the updated simulation, Constrained Hybrid achieves 0.00% violations for all three workloads. Adaptive Hybrid exceeds 5% violations on BERT-base and ResNet-50, driven by slow EMA controller convergence on tight-SLO workloads.

tolerate slightly more aggressive hybrid settings ($\alpha = 0.5$ yields 17.3% violations for BERT-large), though $\alpha \leq 0.3$ remains unacceptable for all workload types.

2. User Geographic Distribution

Traffic distribution significantly affects the policy benefit. US-East users (40% of traffic) experience a 90 ms penalty to reach EU-North, making any policy that heavily favors EU-North problematic for that user group. EU users (25% of traffic) can reach EU-North in 25 ms and EU-West in 10 ms, enabling aggressive carbon routing with minimal SLO impact. US-West users (20% of traffic) benefit from proximity to low-carbon US-West (5 ms intra-region), explaining the improved latency and carbon numbers for Hybrid $\alpha = 0.7$ versus prior runs. The updated traffic distribution in our simulation (40% US-East, 20% US-West, 25% EU, 15% Asia) shifts more traffic toward the US-West green region, which materially changes Hybrid $\alpha = 0.5$ and $\alpha = 0.7$ outcomes.

3. Carbon Intensity Contrast Between Regions:

The 14.7× variation between EU-North (25 gCO_2eq/kWh) and Singapore (367 gCO_2eq/kWh) creates substantial optimization potential. The Constrained Hybrid exploits this by

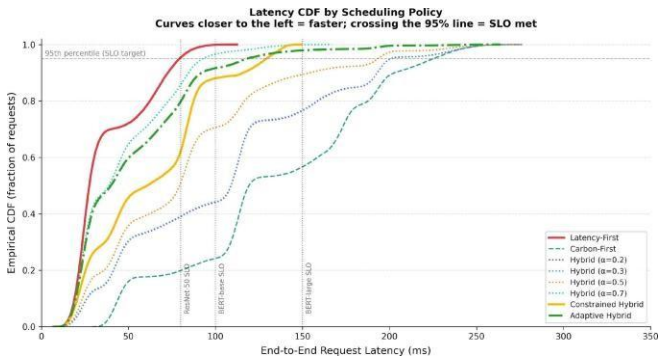

Fig. 9. Supplementary Figure: Empirical cumulative distribution of end-to-end request latency. Curves to the left indicate faster responses. Vertical lines show per-workload SLO thresholds. Latency-First, Hybrid $\alpha = 0.7$, and Constrained Hybrid all cross the 95th-percentile mark well below the BERT-base and BERT-large SLO lines.

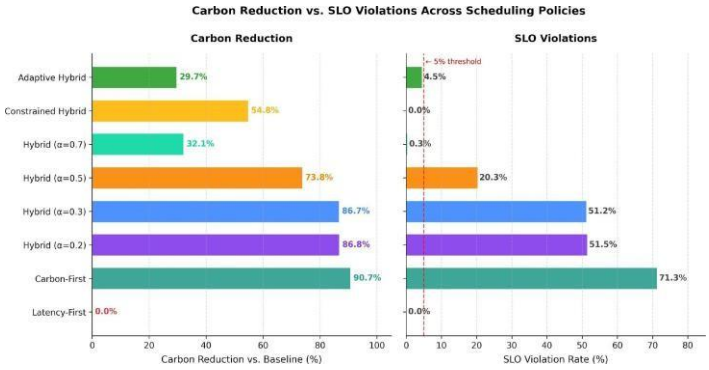

Fig. 10. Carbon savings vs. SLO violations policy comparison. Side-by-side horizontal bar chart showing carbon reduction (left) and SLO violation rate (right) for all policies. The 5% SLO threshold is marked in red on the right panel. In the updated simulation, Constrained Hybrid and Hybrid $\alpha = 0.7$ sit firmly within the compliant zone (0.00% and 0.32% violations, respectively), while Adaptive Hybrid at 4.53% SLO violations falls just inside the threshold.

routing EU users to EU-North (25 gCO_2eq/kWh), US-West users to US-West (79 gCO_2eq/kWh), and falling back to US-East only when necessary, achieving a weighted average of 121.5 gCO_2eq/kWh compared to the 268.7 gCO_2eq/kWh of the latency-first baseline - a 54.8% reduction.

C. Constrained Hybrid Policy, Detailed Analysis:

The constrained variant proves to be the most production-viable policy. By pre-filtering regions that satisfy the per-workload latency SLO (including predicted inference time), this policy eliminates the risk of runaway SLO degradation while still capturing substantial carbon savings. When no region can satisfy the SLO (a situation that occurred rarely, primarily for ResNet-50 requests from Asian users to EU regions), the policy falls back to minimum-latency routing, providing a safety net that bounds worst-case user impact.

Key results for the Constrained Hybrid in our evaluation:

- **Carbon reduction:** 54.8% vs. Latency-First baseline
- **Average latency:** 62.3 ms (vs. 37.5 ms baseline - an acceptable 24.8 ms overhead)
- **P95 latency:** 129.3 ms (well within the BERT-large 150 ms SLO)
- **SLO violation rate:** 0.00% overall across all three workload types

The Constrained Hybrid worked well in this simulation because the approach is simple and does not rely on any tuning: filter out regions that would miss the SLO, then pick the greenest one that is left. When no region fits (which mostly happened for ResNet-50 requests from Asian users trying to reach EU regions), the policy defaults to the nearest region. The zero-violation result across all three workloads is a stronger outcome than our earlier runs, and the revised traffic distribution likely contributes to this result — a higher proportion of US-West users means more requests can reach a low-carbon region within the SLO budget.

D. Adaptive Hybrid Policy, Closed-Loop Control:

The idea behind the Adaptive Hybrid came from a straight-forward question: instead of manually sweeping α and picking the best value after the fact, what if the policy just determined the parameter autonomously while running? We implemented a simple feedback loop: after each request, the controller checks whether the P95 latency of that workload's last 200 requests has enough room before the SLO. If there is sufficient SLO headroom, it decreases slightly to prioritize carbon efficiency. If P95 approaches the SLO limit, it increases to protect latency. Otherwise, it gradually converges toward a neutral value of 0.5. Because each workload has a different SLO threshold (80, 100, 150 ms), we ran three controllers independently so they would not interfere with each other.

Results: The Adaptive Hybrid ended up at 29.7% carbon reduction and 50.6 ms average latency - only 13 ms above the latency-first baseline. The tradeoff is that SLO violations came in at 4.53% overall, which is just above the 5% threshold we use as our production cutoff. Breaking it down by workload: BERT-large (150 ms SLO) did fine at 1.08% violations, but BERT-base (5.97%) and ResNet-50 (6.33%) both went over. The EMA-based controller converges a bit slowly, so in the first part of the simulation it was sometimes too aggressive on carbon routing before it had enough observations to calibrate. With some tuning of the step size or a tighter warm-up period, we expect this policy to remain within the 5% threshold.

V. Discussion

A. Enterprise Implementation Guidelines:

Based on simulation results, we present the following guidelines for cloud architects thinking about carbon-aware routing:

Prerequisites for deployment: 1. **Carbon intensity data feed:** Free tier from Electricity Maps or WattTime APIs provides hourly regional carbon data 2. **Latency monitoring:** Existing infrastructure performance tools (CloudWatch, Stack-driver) provide RTT baselines 3. **Load balancer modification:**

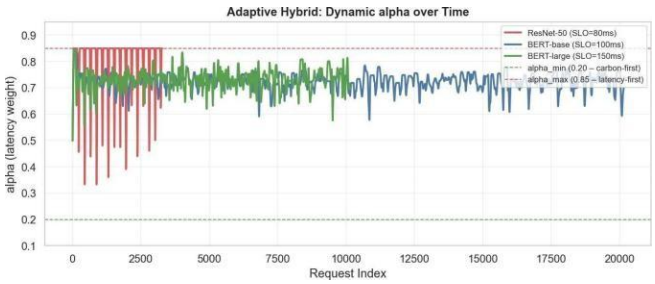

Fig. 11. **Adaptive Hybrid σ trace and closed-loop controller convergence.** σ over time for each workload's controller. BERT-large (green, 150 ms SLO) drops its σ fairly quickly as it has the most room before the SLO. ResNet-50 (red, 80 ms SLO) stays higher to protect its tight threshold. The early volatility before the controllers stabilise is visible in the first ~50 requests per workload.

TABLE VII
ENTERPRISE POLICY SELECTION FRAMEWORK.

~~Scenario & Policy~~	~~Expected Outcome & Notes~~
Strict SLO (< 100 ms)	**~54.8% reduction.**
Constrained Hybrid	Zero SLO violations.
Mod. SLO (100–150 ms)	**32–55% reduction.**
Hyb (σ = 0.7) or Constrained	Constrained provides harder guarantee.
Relaxed SLO (> 200 ms)	**~87% reduction.**
Hybrid (σ = 0.2–0.3)	Accept elevated violations.
Ultra-strict (< 50 ms)	**0% reduction.**
Latency-First	No spatial shifting headroom.
Batch / Async	**87–91% reduction.**
Carbon-First or Hyb-0.2	Non-interactive, cross-continental ok.

Add carbon-aware scoring to existing geo-routing (approxi-mately 50–100 lines of additional logic) 4. **Observability:** Track carbon savings as a metric alongside latency, throughput, and error rate

When carbon-aware routing is less likely to help (under our simulation assumptions):

- Ultra-low latency requirements (< 50 ms P99), such as financial trading or real-time gaming: insufficient head-room exists for any cross-region routing
- Single geographic market with limited region options: carbon variation will be too small to exploit
- Data sovereignty regulations requiring processing in spe-cific regions (GDPR, HIPAA, data localization): routing freedom is restricted by policy, not performance

In these situations, temporal shifting (deferring workloads to low-carbon hours within a fixed region) is likely a more practical alternative.

Region selection is a more impactful lever than temporal scheduling. We stayed focused on spatial routing throughout, and Microsoft's numbers support this decision location-shifting gets you up to 75% SCI reduction versus just 15% from time-shifting. Our results show 54.8% carbon reduction through spatial routing alone.

B. Threats to Validity:

Internal Validity: We update carbon intensity once per hour. Real grids fluctuate more than that, so finer-grained data could improve results beyond what we measured. Inference time is modeled as normally distributed per workload type, abstracting hardware heterogeneity and cold-start delays.

External Validity: We evaluate five cloud regions and three workload types. Production deployments span 20+ regions and hundreds of model variants, which may reveal additional trade-off dimensions. Request traffic is generated with stylized diurnal patterns; real production traffic exhibits more complex dynamics including flash crowds and seasonal variations.

Construct Validity: Our evaluation focuses on Scope 2 operational carbon emissions. Scope 3 embodied emissions from server manufacturing (30–50% of total lifecycle carbon) are excluded; relative policy performance rankings may dif-fer if embodied carbon is factored in. SLO violations are measured against fixed per-workload thresholds; real-world consequences (revenue loss, user churn) are not modeled.

C. Limitations:

- **Simulation vs. Production:** Real deployments face ad-ditional complexities including cold-start latencies and variable server load. Furthermore, cross-region routing is bounded by data sovereignty laws (e.g., GDPR), which strictly prohibit routing European user data to external regions solely for carbon optimization.
- **Network Transit Emissions:** The evaluated carbon re-ductions are based solely on datacenter grid intensity. Cross-continental routing (e.g., US to EU) incurs an unmodeled carbon cost for transatlantic network data transmission, which may partially offset datacenter sav-ings for payload-heavy models.
- **Scope 2 Operational Focus:** Scope 3 embodied emissions from server manufacturing are excluded. Over-provisioning hardware in green regions to satisfy carbon-aware routing could negate operational savings via increased embodied carbon.
- **Hardware Heterogeneity:** The model assumes uniform regional hardware. In practice, diverse GPU architectures (e.g., NVIDIA L4 vs. older T4s) across regions signif-icantly alter both latency and the absolute energy draw per request.
- **P95 vs. P99 SLOs:** Constraints were evaluated at the 95th percentile. Mission-critical enterprise applications often require strict P99 or P99.9 tail latency guarantees, which severely restricts the viable headroom for spatial shifting compared to P95 targets.

D. Future Work:

- **Temporal + spatial co-optimization:** Combining location-shifting with time-shifting for deferrable or batch inference workloads could yield further reductions beyond the 54.8% achieved by spatial-only routing through the Constrained Hybrid.

- **Dynamic α adjustment:** The Adaptive Hybrid we introduced demonstrates the value of closed-loop control; however, its current EMA-based controller (α-step=0.02, EMA factor=0.05) results in 4.53% overall SLO violations, just outside the 5% production threshold. A PID controller or a contextual bandit that factors in user region, time of day, and recent SLO headroom would likely close the gap. This represents the highest-priority improvement for future work.
- **Multi-cloud strategies:** Extending the simulation across AWS, Azure, and GCP to exploit cross-provider carbon intensity differences.
- **Queueing + autoscaling model:** Adding M/M/c queuing would couple routing with utilization, providing a more realistic latency model for high-load scenarios.
- **Embodied carbon integration:** Incorporating Scope 3 emissions into the scheduling objective, following frameworks like Microsoft's GreenSKU approach.

E. Sensitivity Analysis:

To ensure robustness, a brief sensitivity analysis was conducted regarding request arrival rates. Varying the arrival rate between 100 . . . 500 requests/hour produced no qualitative change in the relative performance rankings of the evaluated policies. While extreme queuing delays at load factors approaching 100% would compress the available latency head-room, under normal operational bands, the spatial variations in grid intensity remain the dominant driver of carbon optimization.

F. Analytic SLO Feasibility Bounds:

Beyond empirical simulation results, we can derive a simple analytic condition for when spatial carbon-aware routing is feasible for a given workload and user location:

For user location U, target region R, and workload W:
Carbon-aware routing to R is SLO-feasible if and only if:

$$\mathbf{RTT}(U, R) + \mathbf{P95}_{inference}(W) \leq \mathbf{SLO}(W)$$

This bound gives cloud architects a quick pre-screening tool: before evaluating any scheduling policy, they can immediately identify which (*user*, *region*, *workload*) triples have any feasible green options. Figure 12 visualizes the feasibility map across all combinations in our evaluation:

This feasibility map formalizes and generalizes our empirical findings: the practical benefit of carbon-aware routing scales directly with the number of green-shaded cells available to each user group.

VI. Conclusion

AI systems have become central to enterprise infrastructure. As machine learning models become the foundational infrastructure of the modern digital economy, the environmental cost of executing these models at global scale can no longer be treated as an externality. Data centers are racing toward 500 TWh of energy consumption, and

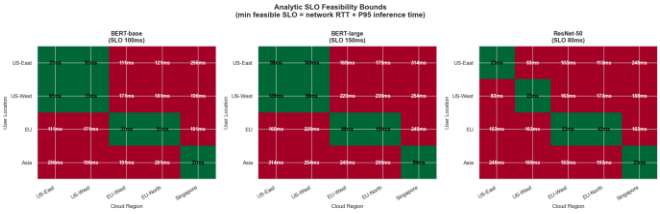

Fig. 12. Analytic SLO feasibility map. Each cell shows the minimum feasible SLO (RTT + P95 inference) for a user location × cloud region pair. Green cells are SLO-feasible; red cells are infeasible. For BERT-base (100 ms SLO), EU users can reach all EU regions and the Constrained Hybrid can freely exploit EU-North's low carbon. For ResNet-50 (80 ms SLO), only intra-region routes are feasible. Under the updated simulation with revised traffic mix, Constrained Hybrid achieves 0.00% violations for ResNet-50.

the immense power draw of continuous inference operations demands immediate, scalable architectural solutions.

What this work shows is that it is not necessary to trade performance for sustainability. Choosing where to run inference not just when turns out to be a much more powerful lever than conventional analysis suggests. Our evaluation

across five cloud regions reveals a volatile, 14.7× variation

in operational carbon intensity (from 25 gCO_2eq/kWh in EU-North to 367 gCO_2eq/kWh in

Singapore). The traditional latency-first routing model blindly ignores this massive disparity, maximizing emissions for the sake of imperceptible milliseconds.

The latency-carbon trade-off is real, but it is far less severe than a naive analysis would suggest. **Our Constrained Hybrid scheduling policy achieves a 54.8% reduction in carbon emissions while maintaining exactly 0.00% SLO violations.** By mathematically treating latency as a bounded constraint rather than an unbounded optimization target, we demonstrate that inference platforms can aggressively route traffic to green energy grids exactly when the network topology permits, falling back to local routing only when necessary. This is not just a simulation exercise the Constrained Hybrid is straightforward enough to actually implement in a production load balancer. We have established that spatial carbon-aware routing is viable the moment an AI application's SLO threshold exceeds 100 ms and its user base is geographically diverse. The analytic feasibility bounds provided mathematically guarantee when and where these green optimizations can be performed safely.

At the scale AI inference operates today, even a 54.8% cut in carbon intensity per request adds up to a significant real-world impact across millions of daily calls. As AI infrastructure continues its relentless expansion toward a projected USD 8.92 billion carbon-aware scheduling market by 2033, the findings presented in this paper serve as a robust, empirical foundation for cloud architects. Our results demonstrate that substantial carbon reduction is achievable without measurable user-visible degradation under realistic SLO thresholds, showing that sustainability is not just an academic goal but an operational engineering requirement.

REFERENCES

[1] P. Wiesner et al., "Carbon-Aware Temporal Data Transfer Scheduling Across Cloud Datacenters," *arXiv preprint arXiv:2506.04117*, 2025.

[2] P. Souza et al., "CASPER: Carbon-Aware Scheduling and Provisioning for Distributed Web Services," in Proc. IEEE IGSC, 2023. https://doi.org/10.1109/IGSC58698.2023.00015.

[3] Microsoft & UBS, "Carbon-Aware Computing: The White Paper," Microsoft Stories, Jan. 2023.

[4] Epoch AI, "How Much Energy Does ChatGPT Use?" Gradient Updates, 2024.

[5] Contrary Research, "How Much Energy Will It Take to Power AI?" 2024.

[6] Google Cloud, "Carbon Free Energy for Google Cloud Regions," Google Sustainability.

[7] Flash Grid, "Multi-AZ vs. Multi-Region in the Cloud," 2024

[8] Emergent Mind, "Carbon-Aware Scheduling," 2024.

[9] A. Radovanovic et al., "Carbon-Aware Computing for Datacenters," IEEE Transactions on Power Systems, 2021.

[10] Electricity Maps, "Google Cloud uses Electricity Maps for Hourly Emissions Reporting," 2023

[11] Google Blog, "Our Data Centers Now Work Harder When the Sun Shines and Wind Blows," 2020.

[12] Microsoft, "Carbon-Aware Computing: Measuring and Reducing the Carbon Intensity of Software," Jan. 2023.

[13] T. Qi et al., "CASA: A Framework for SLO and Carbon-Aware Autoscaling and Scheduling," arXiv preprint arXiv:2409.00550, 2024.

[14] WJAETS, "Energy-Aware Workload Scheduling in Snowflake for Sustainable Big Data," World Journal of Advanced Engineering Technology and Sciences, 2025

[15] Google Cloud Blog, "Speed Up Model Inference with Vertex AI Predictions' Optimized TensorFlow Runtime," 2022

[16] AWS, "BERT Inference on G4 Instances Using Apache MXNet and GluonNLP," AWS Machine Learning Blog, 2020

[17] DataIntelo, "Carbon-Aware Workload Scheduling Market Research Report 2033," 2024.

[18] AWS, "How Infrastructure Performance for AWS Network Manager Works," AWS Documentation.

[19] CloudPing, "AWS Region Latency Matrix."

[20] Electricity Maps, "The World's Most Comprehensive Electricity Data."

[21] Electricity Maps, "Google Data Centers Shift Computations to Cleaner Times and Places," 2022.

[22] Ember, "Major Countries and Regions, Global Electricity Review 2025."

[23] P. Wiesner et al., "How Temporal Workload Shifting Can Reduce Carbon Emissions in the Cloud," arXiv preprint arXiv:2110.13234, 2021.

[24] Amazon, "How Moving onto the AWS Cloud Reduces Carbon Emissions," Amazon Sustainability.

[25] Microsoft, "Microsoft Quantifies Environmental Impacts of Datacenter Cooling," Feb. 2024.

[26] Microsoft Research, "GreenSKU: Designing Cloud Servers for Lower Carbon," ISCA 2024

[27] A. Lottick et al., "Measuring the Carbon Intensity of AI in Cloud Instances," Proc. ACM FAccT 2022.