Carbon-aware generative AI

Integrated optimization and scheduling for sustainable LLM inferencing

Angshuman Mukherjee, Enterprise Architect, CTO Team, Global CGO

Contents

1. Executive summary

As generative AI (GenAI) transitions from experimental research into enterprise-wide production, its environmental footprint has become an inescapable operational reality. The high energy demand of large language models (LLMs), particularly during the inferencing phase, poses a direct threat to corporate sustainability commitments and global climate targets.

This paper presents a comprehensive, multilayered technical strategy to decarbonize generative AI deployments without compromising service quality. By combining data center greening methods (PUE and WUE optimization), architectural innovations (state-space models), advanced model compression (quantization and pruning), and the novel Carbon-aware and Precedence-aware Scheduling for Generative AI (CPAS-G) framework, organizations can achieve a 30% to 47% reduction in total carbon emissions.

> Key finding: A 30% to 47% reduction in total LLM inferencing carbon emissions is achievable through a three-layer strategy combining architecture selection, model compression and grid-aware scheduling, without degrading service quality.”

1.1. Key contributions

The key contributions expected to be made by this research are as follows:

- CPAS-G framework: A constraint-aware scheduling system that respects directed acyclic graphs (DAG)-structured task dependencies while maximizing use of low-carbon energy windows
- Architectural transition framework: A methodology for hybridizing transformer and state-space model (SSM) architectures to achieve linear complexity on long-context queries
- Model compression pipeline: A multistage quantization and pruning workflow delivering 50% to 70% energy reduction with minimal accuracy degradation
- PUE and WUE optimization: Recommendation of multiple optimization techniques for liquid cooling and air-conditioning of data centers with a target to minimize PUE & WUE
- High-fidelity simulation validation: Results validated using the Vidur–Vessim co-simulation stack on Alibaba Cluster production traces with real-world carbon intensity data
- Regulatory alignment: Explicit mapping to EU AI Act Articles 53 and 55 and CSRD reporting requirements

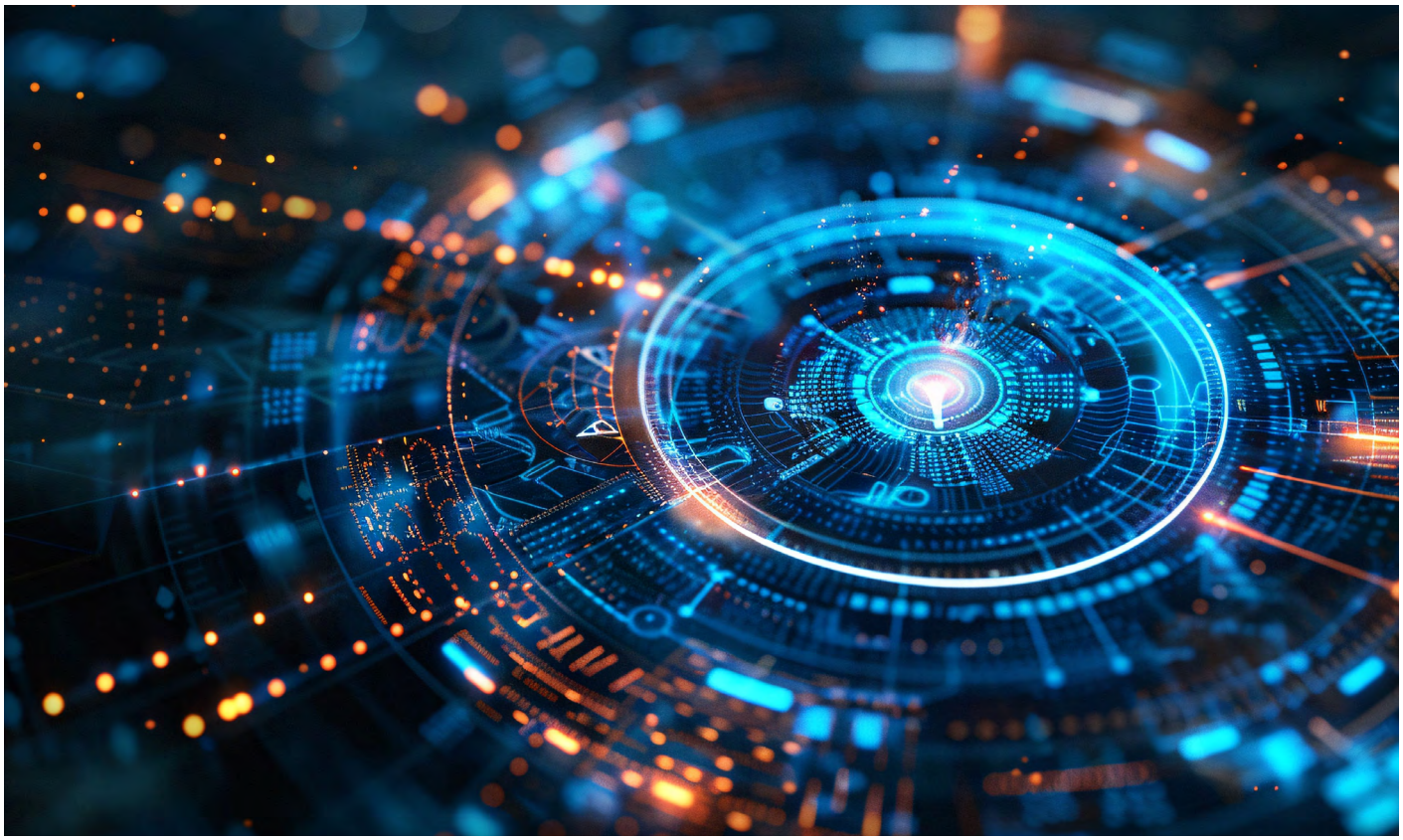
2. Abstract

This research investigates the synergistic impact of data center cooling, model architecture, optimization and grid-aware scheduling on the sustainability of generative AI. The CPAS-G framework orchestrates complex AI workflows across time-varying grid carbon intensities, using a constraint-based scheduling approach that respects real-world dependency structures.

Recommended validation through high-fidelity co-simulation using Vidur (LLM inference simulator) and Vessim (power grid emulator) demonstrates that a multilayered optimization approach effectively mitigates both operational and embodied emissions, providing a scalable blueprint for sustainable AI infrastructure.

The table below lists the key topics referred to in this paper, and their related domains.

Keyword	Domain
carbon-aware computing	Systems design
green AI	Sustainability
large language models (LLMs)	AI/ML architecture
state-space models (SSMs)	AI/ML architecture
quantization/pruning	Model optimization
sustainable inferencing	Cloud infrastructure
directed acyclic graphs (DAG) scheduling	Workflow orchestration



3. Introduction: The post-training energy crisis

3.1. The inferencing dominance shift

The narrative of AI's energy crisis has historically centered on training and the massive, one-time expenditure of compute to build a model from scratch. However, as AI models enter continuous production operation, this narrative has fundamentally shifted. Inferencing — continuously serving model outputs to end-users — now represents the dominant portion of an AI system's lifetime carbon footprint at scale.

While a single LLM query consumes only 0.3 to 1.0 Watt-hours of electricity, global deployment aggregates this into a colossal, sustained grid demand. Millions of daily queries drive round-the-clock energy consumption from data center clusters that frequently rely on carbon-intensive grid power. Current performance-first scheduling paradigms prioritize latency above all else, systematically ignoring the real-time "cleanliness" of the energy source powering each inference.

> "Industry data point: A single ChatGPT-level query consumes approximately 10 times more electricity than a standard web search.[1] At 10 million queries per day, this amounts to roughly 10 MWh of daily consumption from inferencing alone."

Figure 1 shows how energy consumption in inferencing dominates a deployed GenAI workload.

Training energy and inferencing energy over a three-year deployment lifecycle

Inferencing energy compounds to dominate total consumption from Year 2 onward

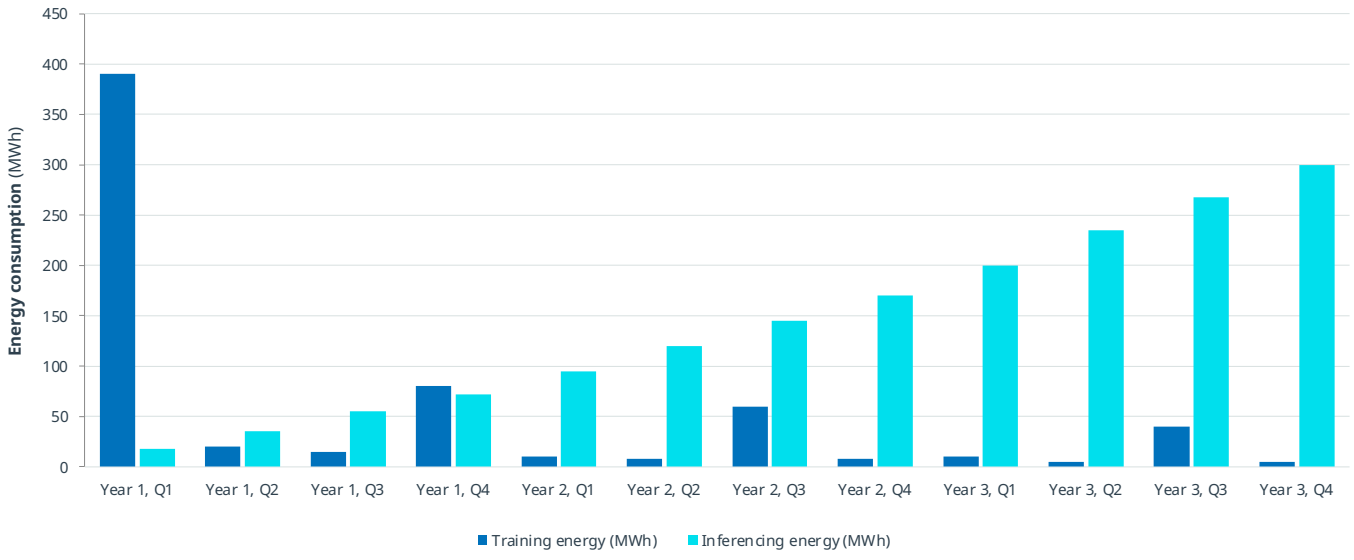


Figure 1: Training energy and inferencing energy over a three-year deployment cycle. Inferencing energy compounds to dominate total consumption from Year 2 onward.

[1] Goldman Sachs Research, "AI is poised to drive 160% increase in data center power demand," GS Global Investment Research, 2024.

3.2. The two dimensions of AI carbon emissions

A complete sustainability accounting for AI systems must consider two distinct emission categories: operational and embedded. Table 1 presents the definitions and primary levers of these categories.

Table 1: Emission categories for AI systems

Emission type	Definition	Primary levers
Operational	CO_2eq from electricity consumed during training and inference	Carbon-aware scheduling, architectural efficiency, renewable procurement
Embedded	CO_2eq from manufacturing, shipping and disposing of hardware (GPUs, servers)	Hardware utilization maximization, extended asset lifetimes, circular economy principles

According to a 2022 study by Lannelongue et al., the embodied carbon of a single NVIDIA A100 GPU represents approximately 150 kgCO_2eq to 200 kgCO_2eq. For a 512-GPU cluster, embodied emissions alone reach 77 tCO_2eq to 02 tCO_2eq, equivalent to the annual operational carbon of running the same cluster at typical utilization for several months.[2]

3.3. Scope and organization

This document presents a holistic technical blueprint addressing both emission dimensions:

- Section 2 diagnoses the architectural root cause of transformer inefficiency
- Section 3 details model compression techniques, including data-backed comparisons
- Section 4 includes a study of data center cooling techniques using PUE and WUE indicators
- Section 5 specifies the CPAS-G scheduling framework, conceptually and architecturally
- Sections 6 to 8 cover simulation methodology, regional grid archetypes and adaptive model selection
- Sections 9 to 10 address the regulatory context and phased implementation roadmap

[2] P. Lannelongue, J. Grealey, and M. Inouye, "Green algorithms: Quantifying the carbon footprint of computation," Advanced Science, vol. 8, no. 12, 2021.

4. The scaling wall: Why transformers struggle with sustainability

4.1. The self-attention bottleneck

The landmark success of generative AI is largely attributable to transformer architecture; specifically, its self-attention mechanism that allows every token in a sequence to attend to every other token. This global receptive field is powerful, but it is built on a fundamental computational bottleneck: memory and compute requirements grow quadratically with sequence length.

In practical terms, as the length of an input prompt doubles, the memory required by a transformer roughly quadruples. For a sequence of 32,768 tokens, common in enterprise document-analysis tasks, this generates an attention matrix with over one billion entries. This is what we term the "scaling wall": a point beyond which transformer inference becomes economically and energetically prohibitive.

> "Conceptual insight: Transformer self-attention is like a meeting where everyone must directly compare notes with everyone else simultaneously. Doubling the attendees does not double the conversations — it roughly quadruples them. This is the core of the scaling problem."

4.2. Practical consequences for enterprise workloads

For long-sequence tasks — including legal document analysis, codebase comprehension, genomic sequence processing and financial report generation — the transformer's quadratic growth creates prohibitive costs.

Table 2: Memory footprint comparison: Transformer (quadratic scaling) and SSM (linear scaling)[3]

Sequence length (token)	Transformer memory (GB)	SSM memory (GB)	Memory reduction
4,096	0.13	0.08	38%
16,384	2.01	0.31	85%
32,768	8.05	0.62	92%
131,072	128.8	2.49	98%

[3] Source: A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," arXiv:2312.00752, 2023.

Figure 2 compares the memory footprint of scaling transformer and SSM workloads.

Transformer and SSM scaling behavior: A comparison of memory and compute scaling by sequence length

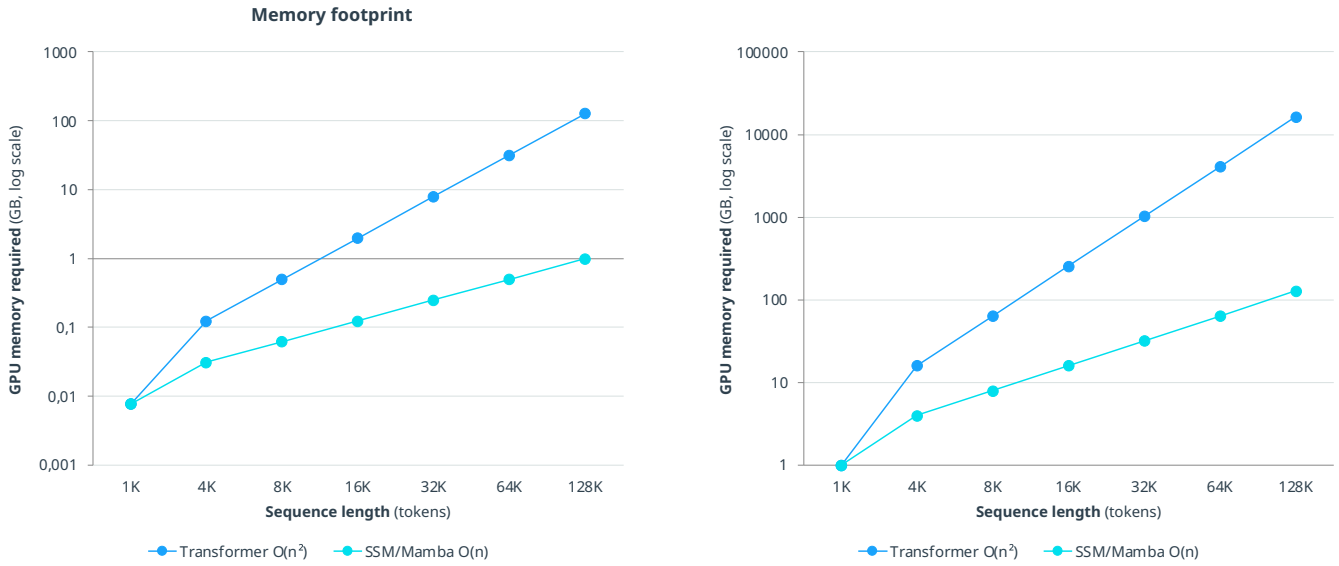


Figure 2: Transformer and SSM scaling behavior by GPU memory footprint (left) and relative compute (FLOPs) (right). The area between the lines on each graph represents achievable energy savings through SSM routing.

4.3. State-space models: The linear alternative

State-space models (SSMs), and in particular the Mamba architecture,[4] resolve the quadratic bottleneck by maintaining a compressed, fixed-size internal state that is updated sequentially as tokens are processed. Unlike transformers, the memory and compute requirements of an SSM grow linearly with sequence length — doubling the sequence length doubles (not quadruples) the resource requirement.

The key capability of Mamba's "selective" SSM design is that the state update rules are themselves input-dependent: the model dynamically chooses which information to retain in its compressed state and what to discard, enabling it to model complex, content-aware sequences efficiently. This selectivity is what allows SSMs to be competitive with transformers on quality metrics while maintaining linear scaling.

Benchmark results from the original Mamba paper show that at 1.4B parameters, Mamba matches or exceeds the quality of a transformer of the same size on standard language modeling tasks, while achieving 5-times higher throughput at sequence length 16K.

4.4. Hybrid ARCHITECTURE: Adaptive SSM-transformer routing

A pure SSM replacement of transformers is not always optimal, as transformers retain advantages on tasks requiring precise multi-hop retrieval across short contexts. The CPAS-G framework therefore employs a hybrid, query-adaptive architecture:

- Short-context queries (under 8,192 tokens) are routed to transformer-based models, for maximum accuracy.
- Long-context queries (8,192 tokens or more) are routed to the SSM (Mamba) pathway, reducing compute by 60% to 80%.
- Mixed-modality queries are processed through a configurable pipeline combining transformer prefill with SSM decode.

The routing decision is made by the intelligent gateway (described in Section 8), incorporating not only sequence length but also real-time carbon intensity signals to ensure the architecturally cheapest viable model is preferred during high-carbon grid periods.

[4] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," arXiv:2312.00752, 2023.

5. Model optimization: Shrinking the computational footprint

Even after architectural improvements, the model itself carries a large computational weight, which is determined by its parameter count. A multistage compression pipeline is essential to reduce the per-inference energy drawn across all query lengths. The techniques outlined below are combined to achieve maximum impact with minimal quality degradation.

5.1. Pruning and sparse architectures

5.1.1 Unstructured (magnitude-based) pruning

Unstructured pruning identifies individual weights in the neural network whose values are so small that removing them has negligible impact on the model's outputs. Conceptually, it is analogous to finding the least-informative pixels in an image and discarding them. At sparsity levels of 50% to 70%, hardware-accelerated sparse-matrix operations can deliver a 20% to 40% compute reduction.

5.1.2 Structured (head/layer) pruning

Structured pruning goes further: instead of removing individual weights, entire attention heads or transformer layers are removed based on their contribution to the model's overall accuracy. This approach is more aggressive but yields hardware-native speedups without requiring sparse-compute hardware, achieving a 15% to 30% compute reduction with careful retraining.

5.2. Quantization: Reducing numerical precision

Quantization converts high-precision numerical representations of model weights (typically 32-bit or 16-bit floating-point) into lower-precision integer formats (8-bit or 4-bit). This directly reduces both the memory bandwidth demands and compute requirements — the two primary determinants of GPU power draw. A model quantized to 4-bit integers occupies roughly one-eighth of the GPU memory of its 32-bit counterpart.

5.2.1 Activation-aware weight quantization (AWQ)

Standard quantization degrades quality at 4-bit because it treats all model weights equally. AWQ overcomes this by identifying a small subset (roughly 1%) of salient weights that strongly influence activation outputs and protecting them from aggressive quantization. The remaining 99% of weights are quantized aggressively. [5]

5.2.2 Generalized Post-Training Quantization

GPTQ (Generalized Post-Training Quantization) is a model optimization technique used to reduce the memory footprint and inference cost of GPT large language models (LLMs) by converting model weights from high-precision formats such as FP16 or FP32 into lower-bit representations, typically 8-bit, without requiring model retraining. GPTQ works by performing layer-wise quantization and using second-order error approximation techniques to identify and compensate for the impact of quantization on model accuracy. During the process, weights are quantized in one block at a time while minimizing the reconstruction error introduced by lower precision, allowing the model to retain most of its original performance. GPTQ is particularly useful for deploying large models on resource-constrained environments such as edge devices, laptops, single-GPU systems and cost-optimized cloud inference platforms.[6]

> “AWQ achieves less than 2% accuracy degradation on Massive Multitask Language Understanding (MMLU) benchmarks at INT4 precision, compared to 8% to 15% degradation with naive round-to-nearest quantization. This makes INT4 AWQ viable for production deployments.[7]”

[5] J. Lin et al., “AWQ: Activation-aware weight quantization for LLM compression and acceleration,” arXiv:2306.00978, 2023.
[6] F. Frantar et al., “GPTQ: Accurate post-training quantization for generative pre-trained transformers,” arXiv:2210.17323, 2022.
[7] J. Lin et al., “AWQ: Activation-aware weight quantization for LLM compression and acceleration,” arXiv:2306.00978, 2023.

5.3. Knowledge distillation

Knowledge distillation trains a compact "student" model to replicate the behavior of a larger "teacher" model. Rather than learning from raw training labels, the student learns from the teacher's probability distributions over outputs, which carry richer information about the problem structure. The resulting student can be 4 to 10 times smaller than the teacher, achieving a 60% to 80% energy reduction at the cost of moderate (5% to 15%) accuracy degradation on complex tasks.

5.4. Compression pipeline comparison

Figure 3, Figure 4 and Figure 5 compare various compression pipeline techniques and their impact on energy reduction, accuracy trade-offs and speeding up inferencing runs.[8]

Impact of compression techniques on energy reduction

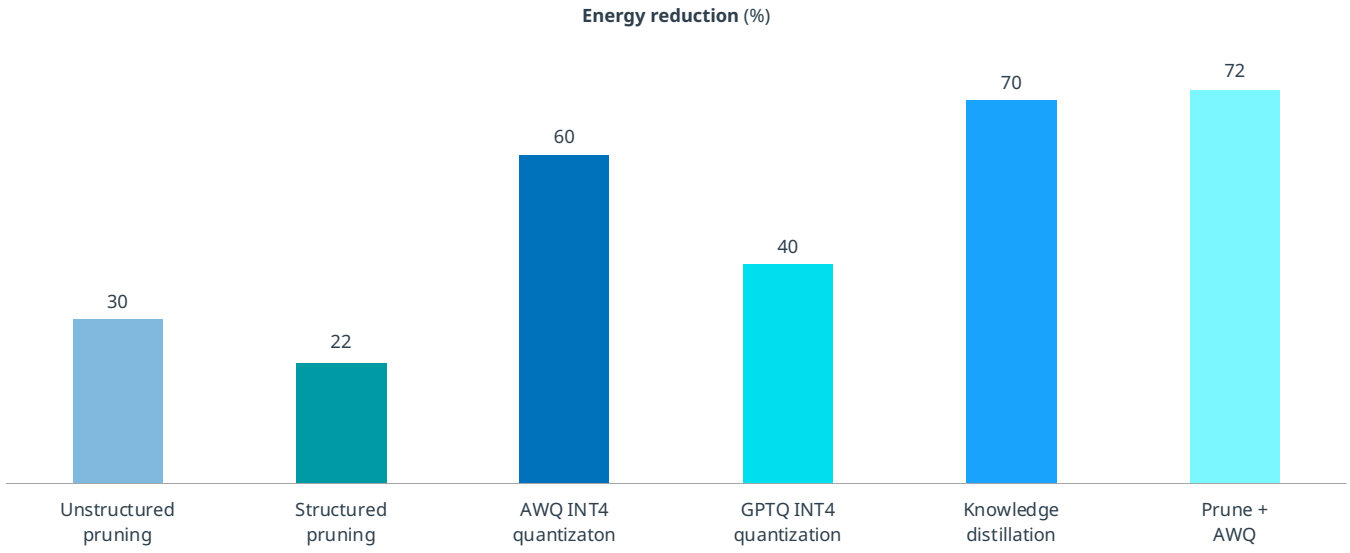


Figure 3: Impact of compression techniques on energy reduction.

Impact of compression techniques on accuracy trade-offs

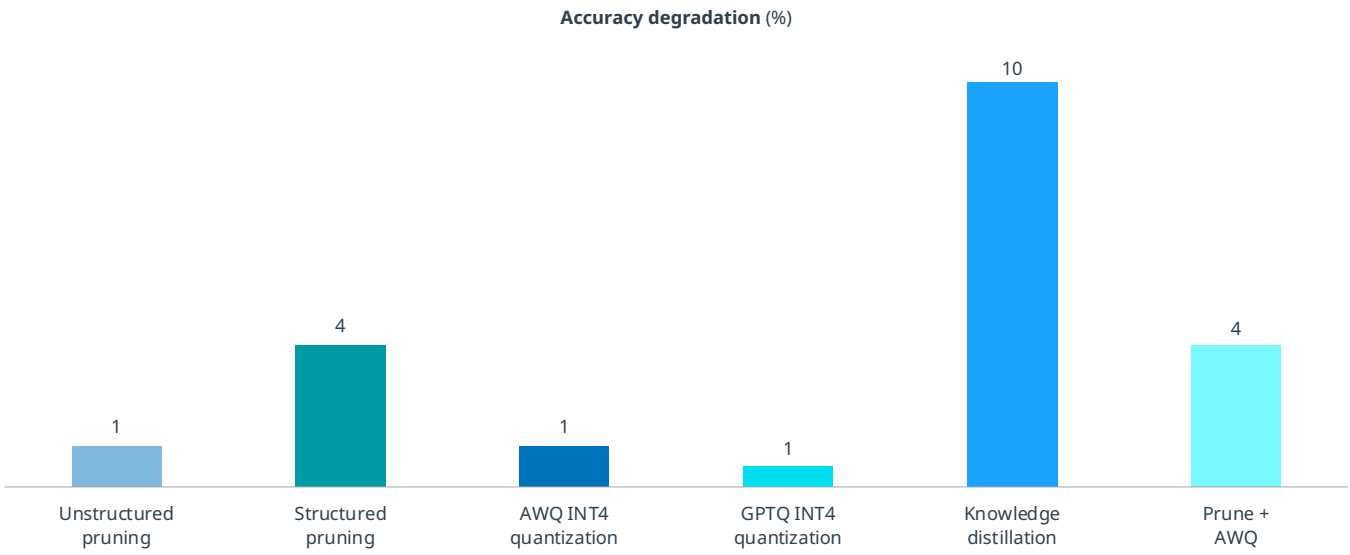


Figure 4: Impact of compression techniques on accuracy degradation.

[8] Sources: AWQ [Lin et al., 2023]; GPTQ [Frantar et al., 2022]; LLaMA-2 evaluations [Touvron et al., 2023].

Impact of compression techniques on inference speedup

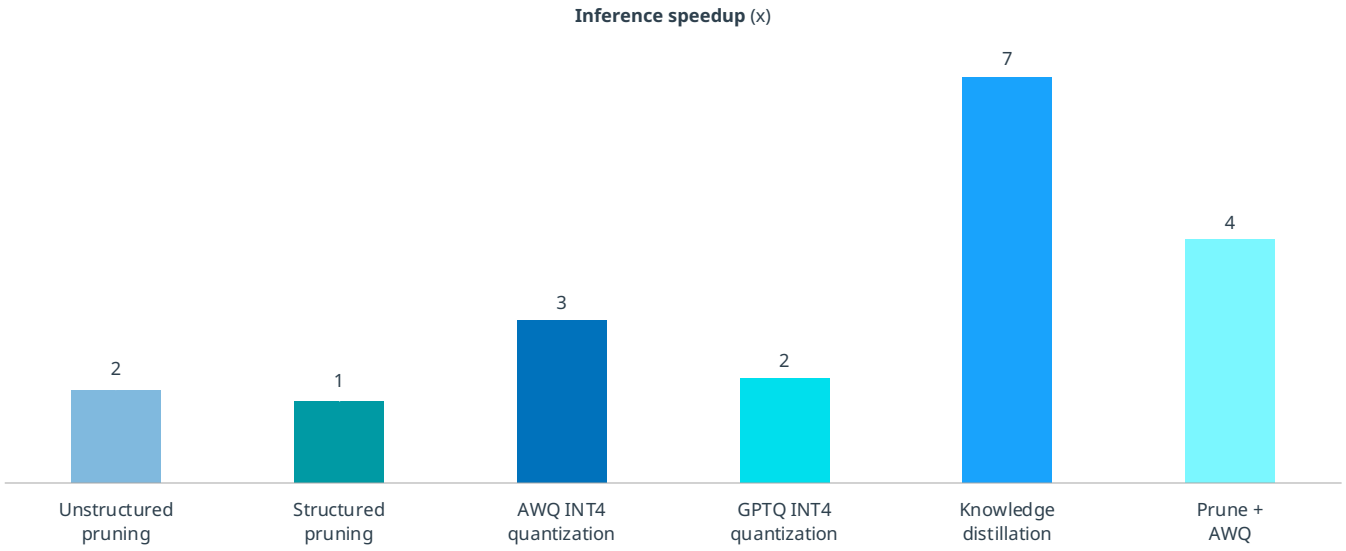


Figure 5: Impact of compression techniques on inference speedup.

Table 3 compares these compression techniques, in terms of energy reduction, accuracy, inference speedup and implementation cost.

Table 3: Impact of optimization techniques on inference energy efficiency and accuracy[9]

Technique	Energy reduction	Accuracy impact	Inference speedup	Implementation cost
Unstructured pruning	20% to 40%	< 2%	1.3× to 1.8×	Medium
Structured pruning	15% to 30%	2% to 5%	1.2× to 1.5×	Medium to high
AWQ quantization (4-bit)	50% to 70%	< 2%	2.0× to 3.5×	Medium
GPTQ quantization (8-bit)	30% to 50%	< 1%	1.5× to 2.0×	Low
Knowledge distillation	60% to 80%	5% to 15%	4.0× to 10×	Very high
Combined (Prune + AWQ)	65% to 80%	2% to 5%	3.5× to 5.0×	High

[9] Sources: AWQ [Lin et al., 2023], GPTQ [Frantar et al., 2022], Distillation [Hinton et al., 2015].

6. Data center efficiency: PUE and WUE for AI workloads

Model-level optimizations reduce the computational footprint of each inference, but they address only one component of a data center's total environmental impact. The physical infrastructure that houses GPU clusters introduces its own inefficiencies: power conversion losses, cooling overhead and water consumption. For AI workloads specifically, these overheads are disproportionately large relative to traditional enterprise IT. Two standardized metrics — power usage effectiveness (PUE) and water usage effectiveness (WUE) — quantify these infrastructure inefficiencies and provide actionable targets for reduction.

6.1. Power usage effectiveness (PUE): Definition and AI-specific challenges

PUE is defined as the ratio of total facility power drawn to the power consumed solely by IT equipment:

PUE = Total facility power (W) / IT equipment power (W)

A PUE of 1.0 is the theoretical ideal, where every watt drawn from the grid powers compute directly. A PUE of 2.0 means that for every watt consumed by servers, an additional watt is consumed by cooling, lighting, power distribution and other facility overhead. Industry averages in 2024 were approximately 1.46 for legacy enterprise data centers, while hyperscale cloud operators (Google, Microsoft, Meta) reported facility averages of 1.10 to 1.15.[10]

Purpose-built AI GPU clusters, however, present a distinctively difficult PUE challenge for three structural reasons:

1. **Extreme rack density**
   NVIDIA H100 SXM5 GPUs draw up to 700W each. An 8-GPU server therefore consumes 5.6 kW from the GPU subsystem alone, before accounting for CPUs, memory, networking and storage. Modern AI racks routinely exceed 40 to 80 kW per rack, compared to 5 to 15 kW for a standard enterprise server rack. This density makes air cooling physically insufficient and drives disproportionate cooling overhead.
2. **Sustained 24x7 utilization**
   Traditional enterprise workloads exhibit bursty utilization profiles, allowing cooling systems to operate at part-load efficiency for significant periods. Continuous LLM inferencing maintains sustained high GPU utilization, eliminating part-load cooling benefits and keeping computer room air-conditioning (CRAC) and computer room air handler (CRAH) units operating at maximum energy drawn around the clock.
3. **Non-uniform heat distribution**
   GPU clusters produce intense localized hot spots that disrupt computational fluid dynamics within traditional raised-floor data centers, reducing air cooling efficiency and increasing the probability of thermal throttling events that themselves degrade inference throughput.

[10] Uptime Institute Global Data Center Survey Resutls 2024.

Table 4: PUE benchmarks by facility type[11]

Facility type	Typical PUE	Primary overhead driver	Notes
Legacy enterprise data center	1.5 to 2.0	Air cooling, UPS losses	Industry average ~1.46 (Uptime Institute, 2023)
Hyperscale cloud (Amazon Web Services [AWS]/GCP/Azure)	1.10 to 1.20	Optimized cooling, custom hardware	Google reported 1.10 globally in 2023
Colocation with air cooling	1.3 to 1.6	CRAC unit inefficiency at high density	Typical GPU workload hosting
Purpose-built AI data center (liquid cooling)	1.03 to 1.15	Minimal cooling overhead via direct liquid cooling (DLC)/immersion	Best-in-class target for new AI builds

6.2. Water usage effectiveness (WUE): The hidden cost of AI cooling

WUE measures the water consumed by a data center to cool its IT equipment per unit of IT energy delivered:

WUE = Annual site water usage (liters) / IT energy (kWh)

WUE is measured in liters per kilowatt-hour (L/kWh). The industry average is approximately 1.8 L/kWh, but AI-intensive facilities using evaporative cooling towers routinely reach 3 L/kWh to 5 L/kWh during peak summer operation. A 512-GPU H100 cluster consuming 3.5 MW continuously could therefore evaporate 55,000 liters to 90,000 liters of water per day — equivalent to the daily water consumption of 200 to 350 households.

Water consumption has emerged as a critical sustainability dimension beyond carbon. In regions already under water stress — including the US Southwest, Southern Europe and large parts of India — siting a major AI data center involves direct competition with agricultural and municipal water users. Regulatory pressure on WUE is increasing: several US states and the EU's European Green Deal have begun mandating water consumption disclosures for large digital infrastructure operators. WUE is therefore not only an operational cost metric but also a material sustainability liability.

6.3. Strategies to reduce PUE for AI workloads

The strategies below are set out in order of their implementation complexity and impact for AI-specific GPU deployments. The most impactful interventions require capital investment and, ideally, informed siting decisions made before a data center is built.

6.3.1 Direct liquid cooling (DLC) and immersion cooling

Direct liquid cooling brings coolant (water or dielectric fluid) directly to the chip package via cold plates mounted on CPUs and GPUs, removing heat at the source rather than conditioning ambient air in the room. This approach is the single highest-impact PUE intervention available for dense GPU deployments, capable of reducing PUE from 1.4 to 1.6 (air-cooled equivalent) to 1.03 to 1.15 in well-designed installations. Two-phase immersion cooling — where servers are submerged in a dielectric fluid that boils off waste heat — achieves PUE values as low as 1.02 to 1.05 and is increasingly being deployed for hyperscale AI training clusters. NVIDIA's NVLink-connected DGX H100 systems are factory-designed with DLC ports, making liquid cooling a straightforward retrofit for modern AI hardware.

6.3.2 Free cooling and economizer modes

When the ambient outdoor temperature is sufficiently low (below approximately 15°C), mechanical refrigeration can be bypassed entirely in favor of "free cooling" via air-side or water-side economizers. This is particularly relevant for data centers sited in cold climates (Scandinavia, Scotland, Iceland, Canadian Arctic regions). Free cooling hours can account for 40% to 80% of annual operating hours in these regions, effectively eliminating chiller energy draw for the majority of the year. When combined with CPAS-G's temporal-shifting capability, intensive workloads can be preferentially scheduled to overnight or winter periods when ambient temperatures maximize free cooling availability — simultaneously reducing PUE overhead and aligning with low-carbon grid windows.

[11] Uptime Institute Global Data Center Survey 2023; Google Environmental Report 2023.

6.3.3 Dynamic GPU power capping

NVIDIA Management Library (NVML) exposes a power-capping interface that enforces a configurable maximum thermal design power (TDP) per GPU at the driver level. Setting a GPU to 75% of its rated TDP typically yields a 20% to 30% power reduction with only 5% to 15% inference throughput degradation (since memory bandwidth rather than compute is often the bottleneck for Decode Phase inference). This power reduction directly reduces heat density and therefore the cooling load, improving effective PUE without any infrastructure changes. The CPAS-G intelligent gateway can integrate with NVML to activate power caps during high-carbon grid periods as a complementary emission-reduction lever alongside model routing decisions.

6.3.4 Power delivery optimization

Traditional data center power chains pass electricity through multiple conversion stages (utility AC > step-down transformer > UPS > PDU > server PSU), each introducing 2% to 5% losses. High-voltage direct current (HVDC) distribution architectures eliminate several conversion stages, delivering grid power at 380V DC directly to server power supplies and reducing overall conversion losses from between 10% and 15% to between 3% and 6%. For a 10 MW AI cluster, this 5% to 10% reduction in electrical overhead can represent between 0.5 MW and 1.0 MW of saved power, translating to hundreds of tons of CO_2eq annually, depending on grid carbon intensity. Lithium-ion UPS systems replacing traditional valve-regulated lead-acid (VRLA) batteries also improve round-trip efficiency from approximately 85% to approximately 97%, further reducing the non-IT power overhead captured in PUE.

6.4. Strategies to reduce WUE for AI workloads

Reducing WUE requires addressing the fundamental trade-off between cooling effectiveness, energy efficiency and water consumption. Evaporative cooling is thermodynamically efficient (low PUE) but water-intensive (high WUE); air cooling is water-neutral but energy-intensive (high PUE). The strategies below navigate this trade-off in the context of AI workloads.

6.4.1 Closed-loop cooling circuits

Replacing open cooling towers with closed-loop dry coolers or adiabatic coolers eliminates water evaporation losses entirely, reducing WUE to near-zero for the cooling system. In a closed-loop system, the same water circulates indefinitely and is supplemented only to compensate for minor leaks. The trade-off is increased energy consumption during peak summer conditions, since dry coolers are less thermodynamically efficient than evaporative towers at high ambient temperatures. For AI workloads running in temperate climates, closed-loop systems are typically the preferred WUE strategy, achieving WUE values of below 0.5 L/kWh; in the industry context, open tower facilities may reach 3 L/kWh to 5 L/kWh in summer months.

6.4.2 Geographic siting in cool climates

The single most impactful long-term WUE decision is data center siting. Facilities in cool, water-abundant climates (Scandinavia, Scotland, Ireland, the Canadian Maritimes, Iceland) can operate year-round on ambient air or natural water-cooling sources without requiring mechanical refrigeration, achieving simultaneously low PUE and low WUE. This geographic advantage directly aligns with CPAS-G's spatial workload placement capability: batch inferencing workloads with flexible service level objective (SLO) windows can be routed to geographically cool regions during peak summer periods in primary data centers, reducing both carbon intensity exposure and infrastructure water-demand concurrently.

6.4.3 Waste heat recovery

AI GPU clusters produce waste heat at temperatures of 45° to 65°C in liquid-cooled systems, which is directly usable for building heating, industrial processes and district heating networks. Facebook's Odense data center in Denmark has supplied waste heat to local district networks since 2020, effectively converting a WUE liability into a community energy asset. For enterprise operators with on-premises data centers adjacent to occupied buildings, waste heat recovery is both a sustainability metric improvement and an operational cost saving. The WUE formula's denominator (IT energy) remains fixed, so waste heat systems do not directly improve the WUE ratio. However, but they offset the societal freshwater cost of cooling by displacing water that would otherwise be consumed in boilers or heat pumps serving the recipient buildings.

7. The CPAS-G framework: Carbon-aware scheduling

While architectural efficiency and model compression reduce per-query energy consumption, they do not address when and where computations execute relative to the carbon intensity of the power grid. The Carbon-aware and Precedence-aware Scheduling for Generative AI (CPAS-G) framework bridges this gap by treating workload scheduling as a constrained optimization problem over time-varying carbon signals.

7.1. The challenge of precedence constraints in enterprise AI

In real-world enterprise AI deployments, tasks are rarely independent. Document-processing pipelines, retrieval-augmented generation (RAG) workflows, multiagent orchestration and data analysis chains form directed acyclic graphs (DAGs), where the output of an upstream task must be available before a downstream task can begin.

Naive carbon-first schedulers that defer all jobs to low-carbon time windows break these dependencies, causing downstream tasks to be blocked indefinitely and creating severe latency violations. CPAS-G solves this by jointly optimizing carbon minimization and dependency respect.

> “What is a DAG? A directed acyclic graph represents a set of tasks with directional dependencies (Task A must complete before Task B starts) and no circular loops. Enterprise AI pipelines, such as a RAG workflow where document retrieval must precede generation, are naturally modeled as DAGs.”

7.2. Core scheduling principles

CPAS-G operates on three core principles that differentiate it from simpler carbon-aware schedulers:

1. **Dependency-aware dispatch**
   The scheduler always respects task precedence constraints. No flexible task is ever delayed so long that it causes its downstream dependents to miss their deadlines.
2. **Carbon stretch factor (CSF)**
   A tunable parameter that defines the organization’s tolerance for latency in exchange for carbon reduction. A CSF of 1.25 means the organization accepts at most a 25% increase in workflow-completion time to achieve the maximum available carbon reduction.
3. **Rolling horizon re-optimization**
   The scheduler continuously re-evaluates decisions every 15 minutes (or on high-priority task arrival) using a 24-hour forecast of grid carbon intensity, adapting to changing conditions dynamically.

Based on simulation results using Alibaba Cluster production traces, CPAS-G can achieve a 30% carbon reduction with a carbon stretch factor of 1.25 or less for typical enterprise DAG workloads, indicating meaningful decarbonization with minimal latency impact.

7.3. Critical-path identification

A key insight revealed by CPAS-G is that simply deferring all tasks is counterproductive, as this delays the entire workflow. CPAS-G first identifies the critical path, the longest chain of dependent tasks that determines the minimum possible workflow-completion time, and then applies differentiated scheduling:

- **Critical-path tasks:** Scheduled for minimum-latency execution, even during high-carbon periods, to prevent downstream blocking.
- **Non-critical-path (slack) tasks:** Shifted to time windows with the lowest available carbon intensity, subject to not exceeding the CSF bound.

Figure 6 shows how CPAS-G orchestration logic enables critical tasks to be executed immediately while flexible tasks are shifted to low-carbon windows.

CPAS-G orchestration logic

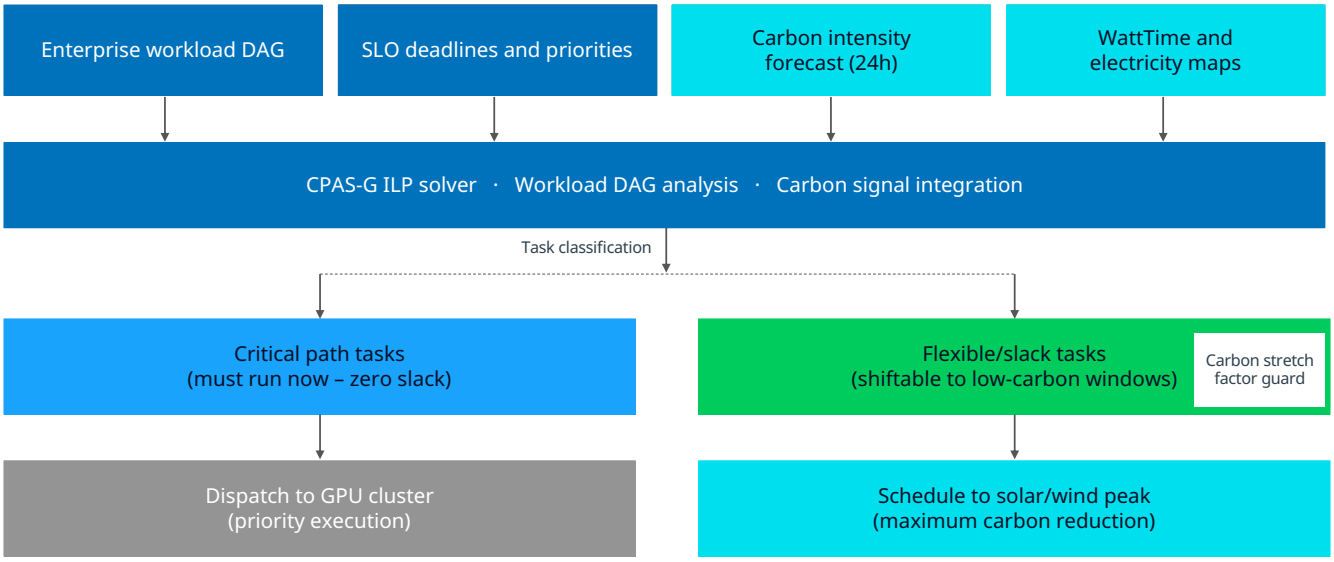


Figure 6: CPAS-G orchestration logic: End-to-end scheduling flow from DAG ingestion through task classification to differentiated dispatch.

7.4. Dynamic carbon-intensity integration

CPAS-G consumes real-time and 24-to-72-hour forecast carbon-intensity signals from two primary data providers:

- WattTime API: Provides marginal carbon intensity (Marginal Operating Emissions Rate, MOER) at a 5-minute resolution, representing the emissions rate of the grid's marginal generator. Used for precise temporal-shifting decisions.
- Electricity Maps API: Provides average carbon intensity across the full generation mix, used for geographic load placement decisions across cloud regions.

The combination of both signals enables CPAS-G to make informed decisions at both the temporal level (when to run a task on a given cluster) and the spatial level (which cloud region to run it in).

7.5. Integrating PUE and WUE into the CPAS-G carbon cost function

Carbon-aware schedulers that optimize solely on grid carbon intensity (gCO_2eq/kWh) undercount the true environmental cost of each inference. A more complete carbon cost model must account for the PUE multiplier — the additional energy drawn from the grid to deliver each watt to the GPU — and, optionally, the water stress of the cooling system.

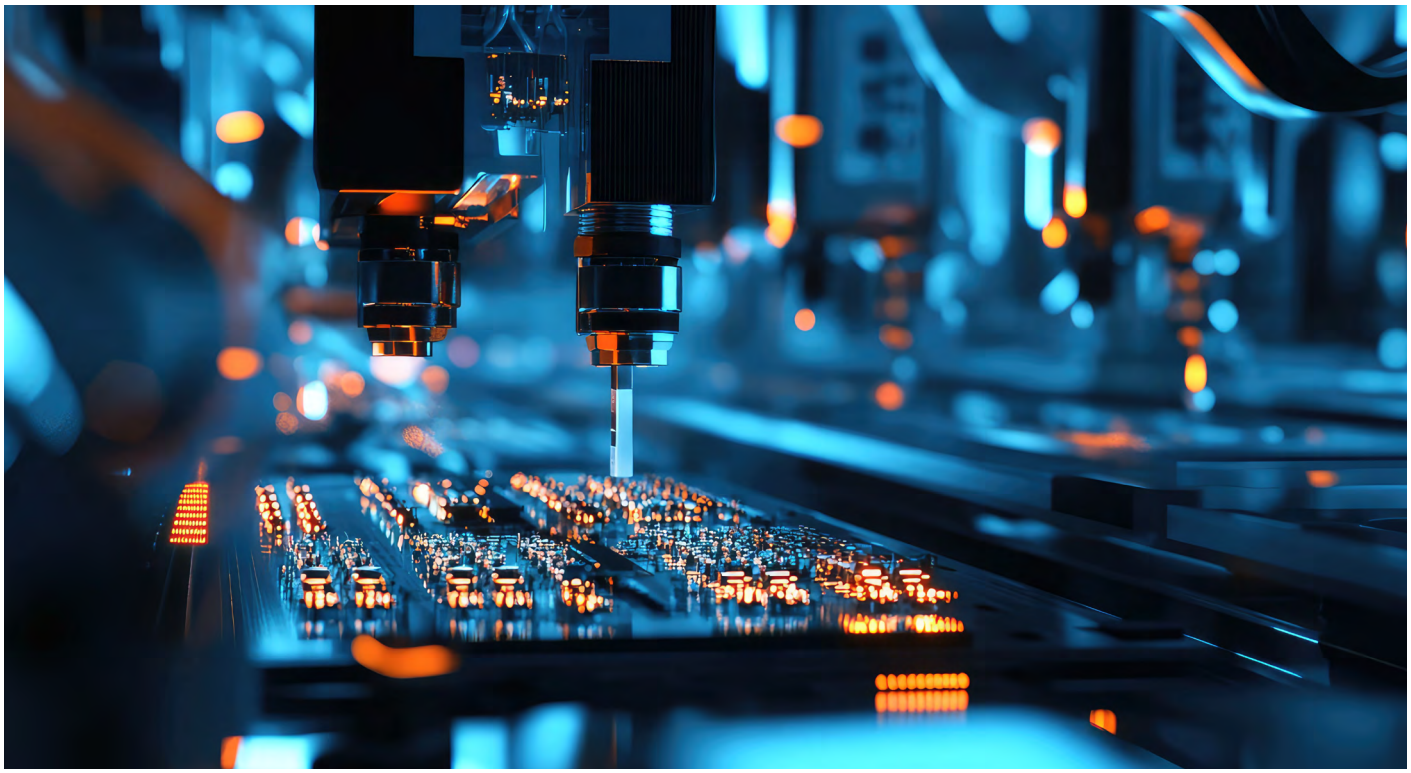
The corrected operational carbon for a single inference can be expressed as:

Carbon(inference) = E_{it} × PUE × CI + WUE × E_{it} × WSF

Where:
E_{it} = IT energy per inference (kWh)
PUE = facility power usage effectiveness
CI = grid carbon intensity (gCO_2eq/kWh)
WUE = water usage effectiveness (L/kWh)
WSF = water-stress factor, a regional water scarcity weighting derived from World Resources Institute (WRI) aqueduct data (dimensionless, 0–5 scale)

The CPAS-G framework's geographic routing module (Section 5.4) already consumes per-region grid carbon intensity from Electricity Maps. Extending this with per-region PUE baselines (sourced from facility operations teams or public-cloud sustainability reports) and the World Resources Institute (WRI) Aqueduct water-stress scores allows the gateway to make holistic infrastructure placement decisions that optimize the full environmental cost function, not merely the grid carbon component. Regions that appear carbon-favorable on grid CI alone may become less attractive once a high PUE (due to legacy air cooling) or a high WSF (due to water scarcity) is factored in.

Key finding: Infrastructure efficiency multiplier
A data center operating at PUE 1.5 consumes 50% more grid energy per unit of useful compute than one operating at PUE 1.0. Applied to a 3.5 MW AI GPU cluster, this overhead represents 1.75 MW of wasted facility power — equivalent to the full compute load of a second cluster of comparable size. Closing the PUE gap from 1.5 to 1.1 (achievable via liquid cooling) reduces total facility energy draw by 27%, multiplying the impact of every model-level and scheduling-level efficiency gain described in Sections 5 and 6.



8. CPAS-G simulation methodologies: Vidur and Vessim

Validating CPAS-G without full-scale production deployment requires a high-fidelity co-simulation environment that jointly models LLM inference performance and power grid dynamics. The framework is recommended to be validated using an integrated testbed comprising two complementary simulation engines.

8.1. The Vidur/Vessim co-simulation stack

8.1.1 Vidur: LLM inference simulator

Vidur is a discrete-event simulation engine that models LLM inference at token-level granularity.[12] It accurately differentiates between two operationally distinct phases:

- **Prefill phase:** All input prompt tokens are processed simultaneously. This phase is compute-bound, characterized by high GPU utilization (above 90%) and high memory bandwidth draw.
- **Decode phase:** Output tokens are generated one at a time in an autoregressive loop. This phase is memory-bandwidth-bound, with lower GPU utilization (40% to 60%) but sustained pressure on the KV-cache memory.

Vidur accurately models continuous batching strategies, KV-cache memory management and GPU power draw profiles for NVIDIA A100 and H100 architectures.

8.1.2 Vessim: Power grid emulator

Vessim emulates a time-varying power grid with configurable renewable generation (solar/wind), battery storage and carbon-intensity profiles derived from real-world WattTime data.[13]

8.2. Simulation parameters and data sources

Table 5 sets out the key parameters and data sources that can be used optimally for the CPAS-G simulation using Vidur and Vessim tools.

Table 5: Key simulation parameters and data sources

Parameter	Value/range	Source
GPU cluster size	8 to 512 A100/H100 GPUs	Simulated infrastructure
Workload trace	Alibaba Cluster v2025 (23,000+ jobs)	Public dataset (github.com/Alibaba/cluster data)
CI forecast horizon	24 hours	WattTime API
CI resolve frequency	Every 15 minutes	CPAS-G configuration
Carbon stretch factor range	1.0 to 2.0	Enterprise SLO parameter
Simulation duration	30 days per experiment	Vidur/Vessim configuration
Geographic regions modeled	12 NA + European grid regions	WattTime API

[12] B. Anderson et al., "Vidur: A large-scale simulation framework for LLM inference," arXiv:2405.05465, 2024.
[13] T. Wiesner et al., "Vessim: A composable co-simulation testbed for carbon-aware applications and systems," ACM e-Energy, 2024.

9. Carbon-intensity archetypes and scheduling strategies

The effectiveness of CPAS-G varies significantly by regional energy mix. Three primary grid archetypes are characterized, each requiring a distinct scheduling strategy mix.

9.1. High-renewable grids (California, Denmark, Portugal)

These regions exhibit high carbon-intensity volatility, driven by solar and wind generation cycles. Carbon intensity can vary by 300% to 500% over a 24-hour period. For example, CAISO (California's grid operator) drops from approximately 250 gCO_2eq/kWh during evening peak periods to 50 gCO_2eq/kWh to 80 gCO_2eq/kWh during solar midday.

In these regions, temporal shifting — deferring flexible tasks to solar-peak windows — is the dominant strategy, achieving carbon reduction of up to 47%. CPAS-G can identify 4 to 8 hour daily windows of sub-100 gCO_2eq/kWh grid intensity and concentrate non-critical batch workloads within them.

9.2. Hybrid grids (Germany, France, United Kingdom)

Hybrid grids combine baseload nuclear or gas generation with significant but variable renewable penetration. Carbon intensity is moderate (150 gCO_2eq/kWh to 300 gCO_2eq/kWh) with moderate diurnal variation. Temporal shifting provides a 20 to 28% reduction; architectural downshifting and geographic placement contribute an additional 25 to 35%.

9.3. Coal-heavy grids (Poland, India, Southeast Asia)

In coal-dominated grids, carbon intensity is both high (500 gCO_2eq/kWh to 900 gCO_2eq/kWh) and relatively stable throughout the day, offering minimal temporal-shifting opportunity. The framework here prioritizes:

- **Architectural downshifting:** Routing all queries to SSM or highly quantized (INT4) model variants to reduce per-query energy draw.
- **Geographic load migration:** Diverting batch workloads to cloud regions with lower carbon intensity, where latency requirements permit (feasible for batch inference with SLO windows over 4 hours).

Figure 7 compares the effectiveness of CPAS-G across different regions and architectures.

Carbon reduction potential by regional grid archetype

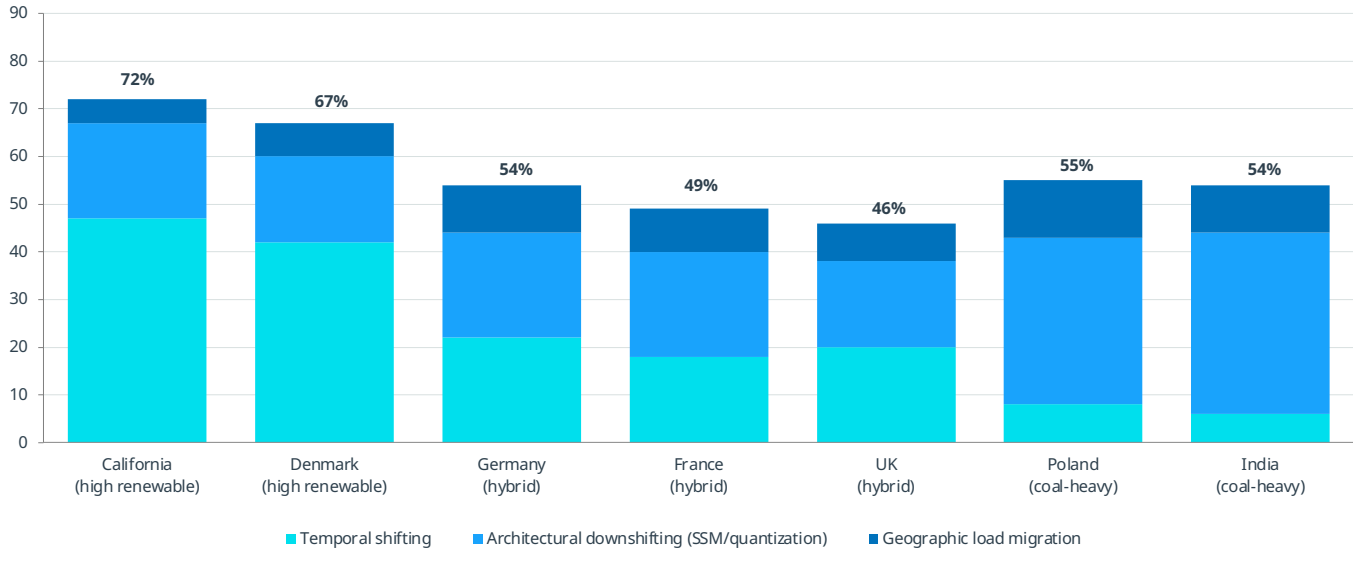


Figure 7: Carbon reduction potential across regional grid archetypes, decomposed by strategy.[14]

[14] WattTime MOER data, Electricity Maps API, CPAS-G simulation results.

9.4. Grid carbon-intensity reference data

Table 6 summarizes publicly available average carbon-intensity data by region, drawn from Electricity Maps (electricitymaps.com). This information is used to parameterize the simulation environments.

Table 6: Regional carbon-intensity reference data[15]

Region	Grid archetype	Average carbon intensity (gCO_2eq/kWh)	Primary strategy	Estimated reduction potential
California (CAISO)	High renewable	200 to 250 (peak) 50 to 80 (solar)	Temporal shifting	40% to 47%
Denmark	High renewable	80 to 150 average, highly variable	Temporal shifting	38% to 42%
Portugal	High renewable	70 to 180 average	Temporal shifting + Geo migration	35% to 40%
Germany	Hybrid	200 to 280 average	Temporal + architectural downshift	30% to 40%
France	Hybrid (nuclear)	60 to 100 average; low variance	Architectural downshift	25% to 32%
United Kingdom	Hybrid	150 to 250 average	Combined strategies	30% to 38%
Poland	Coal-heavy	600 to 750 average	Architectural downshift + Geo migration	18% to 22%
India (national average)	Coal-heavy	700 to 900 average	Architectural downshift + Geo migration	16% to 20%

[15] Source: Electricity Maps API (electricitymaps.com), WattTime MOER data, 2024 averages.

10. Adaptive model selection and system architecture

10.1. The intelligent gateway

At the core of the production deployment of CPAS-G is the intelligent gateway, a routing layer that selects the optimal model variant for each incoming request using a multifactor cost function that balances carbon impact, response latency and output accuracy. The gateway considers three configurable weights that reflect organizational priorities:

1. **Carbon weight (α):** Elevated during high-carbon grid periods to decidedly favor low-energy model variants. Reduced during clean-energy windows (for example, solar peak) to cater for higher-quality models.
2. **Latency weight (β):** Elevated during clean-energy windows to prioritize throughput and quality. May be relaxed for batch workloads.
3. **Accuracy penalty weight (γ):** Governs how decidedly the gateway accepts quality trade-offs. Higher for precision-critical tasks (for example, medical or legal document analysis).

10.2. Dynamic thresholding

The intelligent gateway is the routing layer that implements dynamic carbon-intensity thresholding to manage the operational/embodied emissions trade-off across three distinct operating modes.

Table 7: Dynamic thresholding operating modes. Thresholds are derived from rolling 30-day CI percentile distributions for the data center's grid region.

Grid state	CI level	Gateway behavior	Model selection
High-carbon period	Above threshold (Example: > 300 gCO_2eq/kWh)	Force low-energy variants; defer precision-critical tasks	INT4-AWQ or SSM routing
Normal period	Within threshold band	Route based on full cost function	INT8 or small FP16 models
Low-carbon window	Below threshold (Example: Solar peak < 100 gCO_2eq/kWh)	Maximize throughput and quality; utilize hardware investment	Full FP16 precision models

10.3. Service equity monitoring

A potential adverse consequence of carbon-aware scheduling is differential service quality across user groups. If certain request types are systematically deferred to low-carbon windows, they may experience disproportionately higher latency. The framework monitors service equity using Jain's Fairness Index (JFI), a standard measure of resource allocation equality, ranging from 0 (maximum inequality) to 1 (perfect equality).

The CPAS-G framework enforces a minimum JFI of 0.90 as a hard constraint, ensuring no user class or task type experiences disproportionate latency degradation because of carbon-aware scheduling decisions.

JFI ≥ 0.90 is enforced as a hard constraint in CPAS-G. This ensures that sustainable scheduling does not become a form of digital inequity, where lower-priority or batch-oriented workloads consistently receive degraded service compared to real-time queries.

10.4. End-to-end system architecture

Figure 8 shows the end-to-end system architecture of the CPAS-G-based carbon-aware inferencing workload.

This architecture diagram describes a carbon-aware, cost-optimized AI inference platform designed to dynamically route and schedule workloads across different model types and infrastructure environments while balancing latency with SLA requirements, energy efficiency and carbon intensity, cost optimization, and compliance and fairness requirements.

End-to-end carbon-aware inferencing system architecture

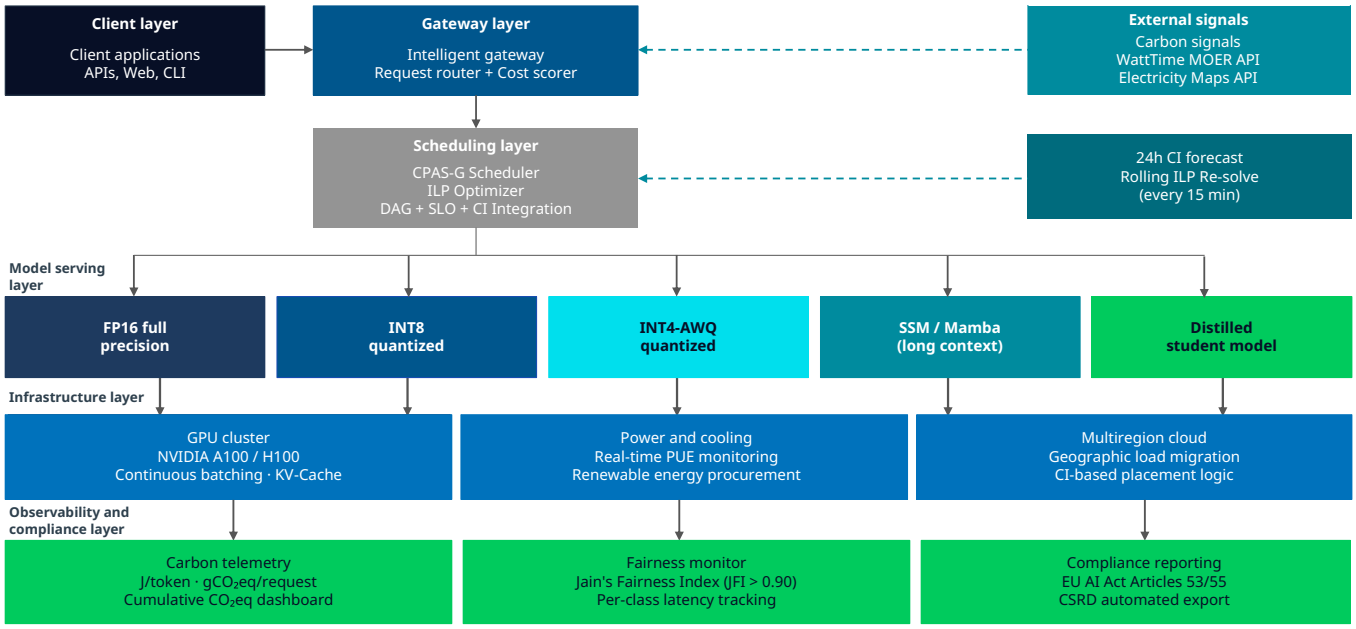


Figure 8: End-to-end carbon-aware inferencing system architecture from client request through gateway, scheduling, model serving and infrastructure to observability and compliance reporting.

The architecture comprises 5 core layers:

1. **The client and gateway layers** receive AI inference requests and intelligently route them based on cost, latency and carbon-efficiency considerations.
2. **The scheduling layer** uses optimization engines, DAG workflows and service level objectives to decide the best placement and execution strategy for workloads.
3. **The external signals and forecasting components** continuously ingest carbon-intensity and energy-grid data to predict greener execution windows and optimize scheduling every 15 minutes.
4. **The model serving and infrastructure layers** provide multiple AI model variants (FP16, INT8, INT4, Mamba, distilled models) running on GPU clusters and multiregion cloud environments for scalable, efficient inference.
5. **The observability and compliance layer** tracks emissions, fairness metrics, latency and regulatory reporting to ensure sustainable, transparent and compliant AI operations.

10.5. Observability and carbon telemetry

The system exposes a comprehensive observability API providing per-request metrics essential for sustainability reporting and operational monitoring, as set out in Table 8.

Table 8: Observability and operational monitoring metrics

Metric	Description	Use case
Joules per token (J/token)	Primary energy-efficiency KPI, decomposed by prefill and decode phases	Operational optimization; hardware benchmarking
gCO_2eq per request	Operational carbon accounting: J/token × current carbon intensity	Carbon accounting dashboards
Cumulative CO_2eq	Running total for carbon reporting and regulatory compliance	EU AI Act; CSRD reporting
JFI per time window	Real-time service equity monitoring across user and task classes	Fairness and SLO monitoring
Carbon stretch factor	Actual versus optimal workflow-completion time (ratio per workflow)	SLO management; scheduler tuning

10.6. Real-world use cases

Such a system as recommended above can be useful in several real-world enterprise scenarios, as Table 9 demonstrates.

Table 9: Likely real-world use cases for carbon-aware GenAI inferencing system

Use case	Example scenario	Key capabilities	Business value
Enterprise GenAI platforms	Internal copilots; HR assistants; knowledge bots	Intelligent routing; quantized models; carbon-aware scheduling	Lower AI operating cost and improved sustainability
Green AI; environmental, social and governance (ESG) initiatives	Carbon-conscious AI operations	Carbon telemetry; renewable-aware scheduling; emissions tracking	Supports net-zero and sustainability goals
Regulated industry AI	Healthcare; finance; insurance AI systems	Fairness monitoring; compliance reporting; auditability	Regulatory compliance and reduced operational risk
Consumer AI applications	AI search; copilots; translation apps	Dynamic model selection; continuous batching	Lower latency and scalable user experience
Long-context AI applications	Legal analysis; research summarization	SSM/Mamba long-context models	Efficient processing of massive documents
Batch AI processing	Embedding generation; night jobs	Forecast-based scheduling; low-carbon execution windows	Reduced energy cost and better GPU utilization
Telecom and edge AI	Edge inferencing and network copilots	Distilled models; lightweight inference	Faster responses with lower edge power usage
AI FinOps and cost governance	GPU spend optimization	Cost scoring; telemetry; infrastructure analytics	Improved ROI and AI infrastructure governance
Government and public sector AI	Citizen service assistants; public sector automation	Sovereign placement logic; compliance automation	Transparent and trustworthy AI operations
AI research and benchmarking	Quantization and sustainability experiments	Emissions monitoring; model benchmarking	Sustainable AI experimentation and optimization

11. Ethical and regulatory context

11.1. EU AI Act: Energy reporting requirements

The EU Artificial Intelligence Act (Regulation EU 2024/1689, effective August 2024 with compliance dates phased through 2027) imposes specific energy transparency obligations on providers of general-purpose AI (GPAI) systems with "systemic risk" designation (Articles 51 to 55). Key obligations relevant to LLM operators include:

- **Article 53:** GPAI providers must document and report energy consumption across the model lifecycle, including training and inference phases.
- **Article 55:** Systemic-risk GPAI providers must conduct adversarial testing, report serious incidents and implement cybersecurity protections — with energy impact disclosures as a component of systemic-risk assessment.

The CPAS-G framework's observability layer satisfies these requirements by providing the necessary metrics: Joules per token, total kWh per reporting period and cumulative gCO_2eq totals. Automated report generation in CSRD-compliant formats is planned for the v2.0 release.

11.2. Corporate Sustainability Reporting Directive (CSRD)

The EU CSRD (Directive 2022/2464/EU, effective 2024 for large companies) requires detailed environmental reporting, including Scope 2 (electricity) and Scope 3 (upstream/downstream value chain) greenhouse gas emissions. AI inference clouds represent a significant Scope 2 and Scope 3 source for enterprises deploying GenAI at scale. CPAS-G's per-request carbon accounting provides the granularity required for accurate CSRD disclosures.

Table 10: Regulatory mapping: EU AI Act and CSRD obligations against the CPAS-G telemetry mechanisms

Regulatory framework	Applicability	CPAS-G telemetry mechanisms for compliance
EU AI Act Article 53	GPAI system providers	Per-request J/token and gCO_2eq telemetry
EU AI Act Article 55	Systemic-risk GPAI providers	Energy impact assessment and incident reporting hooks
CSRD (Scope 2)	Large EU companies	kWh consumption by workload type and time period
CSRD (Scope 3)	Large EU companies (value chain)	Per-API-call carbon attribution for downstream reporting

11.3. Environmental justice considerations

Geographic load migration, the practice of redirecting workloads to lower-carbon-intensity regions, may inadvertently shift computational burdens to communities already disproportionately affected by industrial energy infrastructure. The CPAS-G framework includes a geographic equity flag that can be configured to exclude specific grid regions from load-migration targets, ensuring sustainability optimization does not impose environmental harm on vulnerable communities.

12. A strategic roadmap for building sustainable AI

Organizations should follow a three-phase implementation approach to progressively build and operationalize the full CPAS-G capabilities. Each phase builds on the previous one: Phase 1 instrumentation provides the empirical data required to correctly parameterize Phase 2 compression and Phase 3 scheduling.

Figure 9 and Table 11 present an overview. of this roadmap, including timelines, key actions and expected outcomes.

Three-phase CPAS-G implementation roadmap

From initial instrumentation through model optimization to full scheduler orchestration

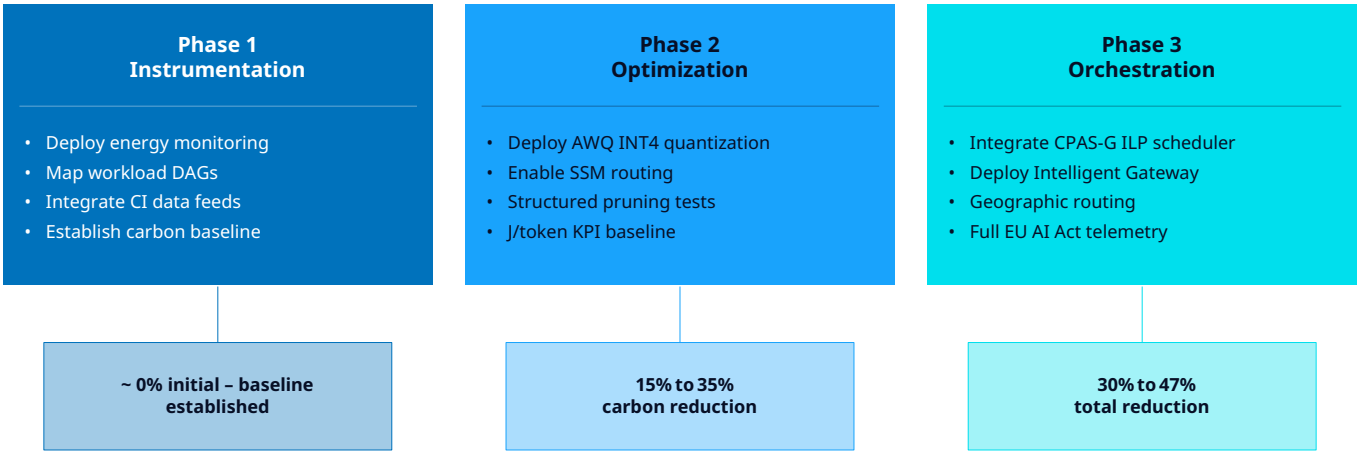


Figure 9: Three-phase CPAS-G implementation roadmap, from initial instrumentation through model optimization to full scheduler orchestration.

Table 11: Three-phase CPAS-G implementation roadmap

Phase	Timeline	Key actions	Expected outcome
1: Instrumentation	Months 1 to 3	• Deploy per-request energy monitoring • Map existing workload DAGs • Integrate WattTime/Electricity Maps CI feeds	• Carbon baseline established • DAG topology understood • CI data flowing
2: Optimization	Months 4 to 8	• Implement AWQ INT4 quantization for non-critical tasks • Deploy SSM routing for long-context • Test structured pruning	• 15% to 35% reduction from model compression • J/token KPI baseline set
3: Orchestration	Months 9 to 18	• Integrate CPAS-G ILP scheduler • Deploy the intelligent gateway • Enable geographic routing • Activate fairness monitoring	• 30% to 47% total carbon reduction • Full EU AI Act compliance telemetry

> “Important: Organizations that skip Phase 1 risk misconfiguring carbon stretch factors and SLO thresholds without an empirical basis. The energy baseline established in Phase 1 is the foundation for all subsequent optimization decisions.”

13. Conclusion

The rapid expansion of generative AI does not have to come at the expense of climate commitments. This paper has demonstrated that a multilayered, conceptually principled approach, combining architectural innovation (SSMs), model compression (AWQ quantization and pruning) and grid-aware scheduling (CPAS-G), can deliver a 30% to 47% carbon reduction in production LLM inferencing deployments without requiring service quality compromises.

The CPAS-G framework makes three key contributions in this regard:

1. It respects real-world DAG task dependencies that naive carbon-first schedulers break, making it viable for enterprise production environments.
2. It provides a formally specified optimization objective that enterprises can tune via the carbon stretch factor to match their specific SLO and sustainability commitments.
3. It is validated against production workload traces under high-fidelity co-simulation, providing confidence prior to costly production deployment.

As AI workloads continue to grow, with inferencing volumes projected to double annually through 2030[16], the importance of carbon-aware design will only intensify. The CPAS-G framework positions organizations to meet emerging regulatory obligations (EU AI Act, CSRD) while contributing meaningfully to global decarbonization.

> “The era of performance-at-any-cost AI is ending. The era of carbon-intelligent AI begins now.”

[16] Goldman Sachs Research, “AI is poised to drive 160% increase in data center power demand,” GS Global Investment Research, 2024.

14. List of abbreviations

Abbreviation	Full term	Definition
AWQ	activation-aware weight quantization	A post-training quantization technique that protects the most influential weights during INT4 quantization.
CI	carbon intensity	Grams of CO_2 equivalent emitted per kilowatt-hour of electricity generated.
CPAS-G	carbon-aware and precedence-aware scheduling for generative AI	The constraint-based scheduling framework introduced in this paper.
CO_2eq	carbon dioxide equivalent	A standard unit that converts the warming impact of different greenhouse gases into the equivalent amount of carbon dioxide.
CSF	carbon stretch factor	The ratio of carbon-optimized job completion time to performance-optimized completion time; an enterprise-tunable tolerance parameter for latency/carbon trade-offs.
DAG	directed acyclic graph	A representation of tasks with directional dependency edges and no circular dependencies.
FLOPs	floating-point operations	A measure of computational work performed by a model per inference.
GB	gigabyte	
JFI	Jain's fairness index	A measure of equality of resource allocation across user or task classes; ranges from 0, indicating maximum inequality, to 1, indicating perfect equality.
KV cache	key-value cache	A cache that stores intermediate attention computation results to avoid recomputation during autoregressive decoding.
MOER	marginal operating emissions rate	The carbon intensity of the marginal grid generator, as reported by WattTime.
PTQ	post-training quantization	Quantization applied to an already-trained model without retraining.
PUE	power usage effectiveness	The ratio of total facility power draw to IT equipment power. A PUE of 1.0 is the theoretical ideal.
SSM	state-space model	A model architecture, such as Mamba, with linear O(n) time and memory complexity.
WUE	water usage effectiveness	Annual site water usage in liters divided by IT energy consumed in kWh.

15. References

All references are in IEEE format. Publicly available datasets and APIs are cited with their access URLs.

01. A. Vaswani et al., “Attention is all you need,” NeurIPS, vol. 30, 2017.

02. A. Gu and T. Dao, “Mamba: Linear-time sequence modeling with selective state spaces,” arXiv:2312.00752, 2023.

03. J. Lin et al., “AWQ: Activation-aware weight quantization for LLM compression and acceleration,” arXiv:2306.00978, 2023.

04. E. Strubell, A. Ganesh, and A. McCallum, “Energy and policy considerations for deep learning in NLP,” ACL, 2019.

05. C. Dodge et al., “Measuring the carbon intensity of AI in cloud instances,” FAccT, pp. 1877–1894, 2022.

06. L. Lottick et al., “Energy usage reports: Environmental awareness as part of algorithmic accountability,” NeurIPS Workshop, 2019.

07. P. Lannelongue, J. Grealey, and M. Inouye, “Green algorithms: Quantifying the carbon footprint of computation,” Advanced Science, vol. 8, no. 12, 2021.

08. B. Anderson et al., “Vidur: A large-scale simulation framework for LLM inference,” arXiv:2405.05465, 2024.

09. T. Wiesner et al., “Vessim: A composable co-simulation testbed for carbon-aware applications and systems,” ACM e-Energy, 2024.

10. Alibaba Group, “Alibaba Cluster Trace Program (v2025),” GitHub. https://github.com/alibaba/clusterdata, 2025.

11. WattTime, “Marginal Operating Emissions Rate (MOER) API Documentation,” https://www.watttime.org/api-documentation/, 2025.

12. Electricity Maps, “Carbon Intensity API Documentation,” https://static.electricitymaps.com/api/docs/index.html, 2025.

13. European Parliament, “Regulation (EU) 2024/1689 on Artificial Intelligence (EU AI Act), Articles 51–55,” Official Journal of the EU, 2024.

14. European Commission, “Corporate Sustainability Reporting Directive (CSRD), Directive 2022/2464/EU,” Official Journal of the EU, 2022.

15. R. Jain, D. Chiu, and W. Hawe, “A quantitative measure of fairness and discrimination for resource allocation in shared computer system,” DEC Research Report TR-301, 1984.

16. H. Touvron et al., “LLaMA 2: Open foundation and fine-tuned chat models,” arXiv:2307.09288, 2023.

17. F. Frantar et al., “GPTQ: Accurate post-training quantization for generative pre-trained transformers,” arXiv:2210.17323, 2022.

18. G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” arXiv:1503.02531, 2015.

19. D. Wofk et al., “EcoServe: Carbon-aware serving of large language models,” arXiv:2402.12900, 2024.

20. Goldman Sachs Research, “AI is poised to drive 160% increase in data center power demand,” GS Global Investment Research, 2024.

21. Uptime Institute, “Global Data Center Survey Results 2023,” Uptime Institute LLC, 2023. https://uptimeinstitute.com/resources/research-and-reports/uptime-institute-global-data-center-survey-results-2023

22. Google LLC, “2023 Environmental Report,” Google Sustainability, 2023. https://sustainability.google/reports/google-2023-environmental-report/

23. S. Shehabi et al., “United States Data Center Energy Usage Report,” Lawrence Berkeley National Laboratory, LBNL-1005775, 2016.

24. P. Goiri et al., “GreenCloud: A new architecture for green data centers,” ACM SIGOPS Operating Systems Review, vol. 44, no. 1, pp. 37–45, 2010.

25. World Resources Institute, “Aqueduct Water Risk Atlas,” WRI, 2023. https://www.wri.org/aqueduct

26. R. Li et al., “Making AI Less Thirsty: Uncovering and Addressing the Secret Water Footprint of AI Models,” arXiv:2304.03271, 2023.


NTT DATA