



EXPLORATORY RESEARCH PAPER

Editorial date: 23 October 2026

From Token Factories to Thermal Schedulers

LPU architecture, Saudi inference infrastructure, and a reproducible water–power–work trade-off experiment

Research status. This is a conceptual and computational working paper with an executed linear-programming experiment on invented data, a review of primary sources, and an analysis of reported operating cases. It is not a field study of Groq facilities, an independent benchmark of Groq chips, or an estimate of national water consumption.

Abstract

This working paper asks whether inference operations in Saudi Arabia can be managed as a joint optimization problem across valuable computational work, time, electricity, and cooling water rather than as a decision about accelerator counts alone. It separates three evidence layers: Groq’s publicly reported Saudi inference operations; technical claims about its SRAM-centric, statically scheduled Language Processing Unit; and a local optimization exercise with fully synthetic inputs. The operational starting points are Groq’s announced 2024 collaboration with Aramco Digital, its report that GroqCloud began serving customers from Dammam in 2025, its non-exclusive technology license with NVIDIA in December 2025, and the launch of the Groq 3 LPX platform in August 2026. Reported operations are kept distinct from expansion plans and vendor benchmarks.

The paper develops an operating model that distinguishes prefill from sequential decode, immediate requests from deadline-constrained batch work, on-site water from water associated with grid electricity, and raw token output from correct tasks completed within

their deadlines. A separable two-stage procedure first selects the lower-cost cooling mode for each hour under the declared water weight, then uses a 48-hour linear program to assign 48 flexible jobs subject to release and deadline constraints while interactive work remains fixed. The weather profiles are fictional and cooling costs are linear, with free mode switching. Every policy completes approximately **4,394.2 synthetic units of useful work** without deadline violations. In the synthetic hot-dry profile, increasing the water weight in the objective reduces site-water use by about 607 liters at the cost of about 8.26 additional kilowatt-hours relative to the minimum-energy solution. This is an operational existence proof for a multi-resource optimizer, not an observed facility effect or evidence that an LPU outperforms another architecture.

The paper concludes with a falsifiable field-test design. Its central research proposition is to combine phase-aware serving metrics, physical cooling telemetry, and completed-work quality in an auditable Saudi inference-measurement protocol. Vendor benchmarks and national announcements remain bounded by their operational and epistemic limits. Site consumption, water withdrawal, upstream water for electricity, real chip energy, network latency, and life-cycle effects all require independent facility data before any generalization.

Keywords: Groq; Language Processing Unit; LPU; static dataflow; SRAM; HBM; prefill; decode; KV cache; wet-bulb temperature; PUE; WUE; water consumption; inference; linear programming; batch-work scheduling; Saudi Arabia.

1. The research question and the limits of the claim

The central question is: **if an inference operator in Saudi Arabia serves both latency-bound immediate demand and delay-tolerant demand, and has reliable data on workload electricity, cooling methods, and weather, can it reduce water or electricity consumed per correct completed task without violating time and quality requirements?** A scientific answer cannot be “yes” until the response of real equipment and real workloads is known. This paper therefore offers three outputs that make a narrower claim and permit closer scrutiny:

1. A reconstruction of Groq’s Saudi case from company announcements, classifying each type of evidence and distinguishing an observed public event from an agreement, roadmap, or vendor test.
2. A causal and technical model linking input length, generation mode, memory, scheduling, server energy, heat rejection, weather, and water use.
3. An executed computational experiment on invented data that shows a trade-off surface and the failures that would make generalization invalid, followed by a protocol for replacing those invented functions with measurements from an actual facility.

We do not assume that **cheaper inference per token** means lower resource use per completed task, or that lower energy necessarily coincides with lower water consumption. We do not assume that a specialized chip performs best across every neural-network model. Nor do we attribute the 48-hour experiment to Groq or HUMAIN. It was created for this work and does not reproduce any known infrastructure belonging to either organization. That boundary is a condition of valid inference: an electrical and thermal system cannot be tested with numbers that merely pretend to come from a system we could not access.

2. An evidence ladder: what happened in the Saudi case?

This paper uses five evidence descriptions: **statement of intent**, **application demonstration**, **service-operation announcement**, **vendor measurement or benchmark on a specified platform**, and **independent reproducible measurement at a specific site**. These do not form a simple sequence. A company can have an independently tested accelerator in the United States before documenting that accelerator's operation elsewhere. The classification shows why grouping every headline under the single word "achievement" can mislead.

Date	Event	What the sources support	What they do not establish
12 September 2024	Groq and Aramco Digital announce an inference-center partnership, a voice-conversation demonstration, and the nawat marketplace	Cooperation announcement and interactive demonstration; estimates of future scale S01	That all planned capacity was installed or generating revenue
December 2024 and February 2025	A cluster reportedly installed in eight days; GroqCloud service announced from Dammam S02	Operator-reported service operation in a specified period and place	Electrical capacity, water use, detailed server count, or local revenue
February 2025	Saudi commitment of \$1.5 billion to infrastructure expansion S02	Announced financial commitment to expansion	Amount spent to date or number of racks purchased
May 2025	Groq announced as an inference provider to HUMAIN S03	Announced supply relationship	Groq's actual share of HUMAIN requests or the content of each purchase
24 December 2025	Groq grants NVIDIA a non-exclusive technology license; founder and engineering personnel move to NVIDIA S06	Announced technology agreement; Groq remains legally independent according to its statement	Sale of the entire company or termination of its services
24 August 2026	NVIDIA announces Groq 3 LPX in full production and Nebius as the first adopter S07	Announced product, manufacturing, and adoption status	That this product operates in the Dammam center
August 2026	Groq reports work with Dell on LPX deployment and plans to adopt the system S12	Supply arrangement and forward-looking deployment goal	Installation of the product in Saudi Arabia at the time of documentation

The apparent location conflict. Statements from February and May 2025 name Dammam, while Groq's current platform page lists **SA-1 — Riyadh** among 13 global centers. The more recent source does not explain this operational difference. It only names some sites. We record both sources without constructing a relocation story or inventing two separate centers. Direct evidence from the operator or facility documents would be required before mapping the cluster. [S02](#) [S03](#) [S08](#)

What does it mean for Groq to make its own chip? The vendor has said that the LPU and associated systems were designed, manufactured, and assembled in North America. "Inference factory in Saudi Arabia" therefore refers to **where computation and service occur**, not to ownership of the chip design or a domestic silicon fab. A Saudi-developed Arabic model can run on a US-designed chip in a Saudi data center through a global commercial interface. Measuring local value requires identifying the work performed at each layer rather than jumping from one layer to another.

3. Why token speed is not proportional to FLOPS

The first analytical tool from computer architecture is **arithmetic intensity**, $I = F/B$, where F is the number of operations and B the bytes required to fetch the data. If compute capacity is C operations per second and memory bandwidth is M bytes per second, a simplified theoretical performance ceiling is:

$$P_{\max} \leq \min(C, IM)$$

This is not sufficient to calculate the performance of a real model. It is a **Roofline** approximation that excludes kernel scheduling, network and small-operation calls, sharding across processors, repeated data movement, and synchronization delay. Its basic distinction nevertheless matters. When $IM < C$, increasing arithmetic capacity alone has little effect until memory movement or data reuse improves.

During **prefill**, many context positions can be processed in parallel, producing dense matrix operations. During **decode** for an autoregressive model, the next token depends on the preceding token. Many users permit requests to be grouped into batches, but they do not remove the dependency within one user's sequence. Higher-bandwidth memory can increase individual token-generation rates, but placing all model parameters and KV state in that memory remains a problem of distribution and capacity management.

For a unit check, consider a neutral **70-billion-parameter** model. The minimum raw weight sizes, before any additional object, are:

Weight representation	Calculation	Weights only
16 bit	70 billion $\times$ 2 bytes	140 decimal GB
8 bit	70 billion $\times$ 1 byte	70 decimal GB
4 bit	70 billion $\times$ 0.5 bytes	35 decimal GB

This is a **definitional calculation, not a benchmark**. It does not determine how many LPUs a model requires because distribution across chips depends on weight partitioning, the KV cache, other memory management, interconnect speeds, quantization, and the compiler. Nor does the announced 128 GB of aggregate SRAM across 256 LPUs in an NVIDIA Groq 3 LPX rack prove that a 70B model at 16-bit precision must be rejected. It shows that **capacity, distribution, and density** do not disappear behind a memory-bandwidth figure. Likewise, Groq's figure of more than 80 TB/s for an earlier LPU's memory cannot be compared with selected HBM configurations as if it were a controlled comparison of completed tasks on both devices. Model, precision, batch, context size, and KV state all have to be fixed. [S04](#) [S05](#) [S07](#) [S08](#)

4. Groq’s “determinism”: what the compiler guarantees, and what it does not

Groq describes a *programmable assembly line*: the compiler partitions data and instructions, then assigns work to execution units and transport links. Many movements between processors become **planned at known times**. This is internal determinism at the level of program operations, not a promise that every user will receive the same end-to-end response time. The distinction matters for service engineering. Control inside the chip reduces one source of runtime improvisation, while external variables remain: prompt length, service queues, contention among models, data conversion, speech processing, the internet, and API rate limits. [S04](#) [S05](#) [S09](#)

With deliberate simplification, the total time for a task can be expressed as:

$$T_{task} = T_{net} + T_{queue} + T_{prefill} + \sum_{k=1}^N T_{decode,k} + T_{tools} + T_{post}$$

This is an **operational sum that identifies components of elapsed time**, not an assumption that every component is measured in the same way. Some can overlap during streaming or parallel execution. It remains necessary to distinguish **TTFT**, **inter-token latency**, and **end-to-end latency**. The last includes the full output; a first word may arrive quickly even if the rest of the answer takes much longer. Groq documents Prometheus metrics for queueing, TTFT, and request completion that could in principle support experiments. This study did not use a live Groq interface, obtain credentials or Saudi customer data, or claim to have extracted those measurements. [S10](#)

Repeated interaction adds a further burden in agent systems. A short request with five consecutive generation rounds may depend more on inter-token and between-round time than a long request completed in one pass. Any benefit should therefore be weighted by its effect on **time to an accepted completed task**, not merely by a single-user token rate. The same point helps explain NVIDIA’s architecture: its description combines Rubin GPU and LPX resources across different phases. That should not be reduced to “the GPU failed and the LPU won,” or the reverse. The hybrid design itself indicates that optimizing different stages can require different architectures. [S07](#) [S12](#)

5. Defining the variable worth optimizing

We propose an output measure Y more exact than “number of tokens”:

$$Y = \sum_{j=1}^J \mathbf{1}\{\text{quality}_j \geq q_j\} \mathbf{1}\{\text{finish}_j \leq d_j\} v_j$$

Here, j denotes a task, q_j a quality threshold specified in advance, d_j a deadline, and v_j a value weight declared before execution. Setting $v_j = 1$ makes Y a count of quality-accepted tasks completed by their deadlines. That is the proposed field metric; it is not the source of the experiment’s reported 4,394.2 figure.

The synthetic computation instead reports an artificial work volume $W = \sum_{t=0}^{47} (U_t + \sum_j x_{j,t})$, where U_t is immediate work and $x_{j,t}$ is the batch work from job j executed in hour t . Both use the same deliberately normalized artificial work unit, although their invented electricity coefficients differ. Every policy completes the same $W \approx 4394.2$ and all 48 batch jobs. This convention permits an internal comparison of policies; it does not establish that an interactive unit and a batch unit have equal value in a real service. A field study must define v_j and its quality oracle in advance, because quantization or moving a task between models can change answer quality.

Real operations should read both resource-efficiency measures together:

$$\eta_E = Y/E_{site}, \quad \eta_W = Y/W_{site}.$$

These cannot be treated as mere inverses of PUE and WUE. The denominator of PUE is E_{IT} , and site WUE also uses E_{IT} , while the proposed economic and engineering output is a **successful task completed on time**. When server utilization falls, some facility ratios may remain stable even as the number of accepted tasks per hour changes sharply.

Four causal paths should be tested separately: (a) model choice, quantization, and batching create a quantity of computation and memory movement; (b) computation becomes heat; (c) cooling method and weather determine additional electricity and on-site water; and (d) the source of electricity adds off-site water. Combining site water and electricity-related water and calling the result WUE without defining the boundary would be an error. Recent water literature recommends specifying location, cooling method, electricity-related water factors, and actual workload. A 2025 review by Lei and colleagues found very large variation in workload-level water consumption across different conditions and technologies. [S11](#) [S13](#)

6. The physics of heat rejection: why temperature needs two names

Heat in a data center passes through layers: from transistor junctions to a thermal package, then a heat-spreading surface, a coolant or air stream, a heat exchanger, and finally an external medium. A liquid cold plate can operate with a nearly closed loop inside the rack. “Liquid cooling” alone therefore does not imply continuous evaporative consumption. In an evaporative configuration, some energy moves into water vapor, and make-up water is consumed alongside varying treatment, blowdown, drift, and leakage. [S11](#)

Two atmospheric variables are related but not interchangeable:

- **Dry-bulb temperature**, T_{db} : the sensible heat of the air, commonly shown in weather reports.
- **Wet-bulb temperature**, T_{wb} : the temperature of a wetted surface under evaporation, related to the air's capacity to accept additional water vapor; approaching it constrains theoretical evaporative cooling.

Two cities with the same T_{db} can have very different T_{wb} because of humidity. Heat-exchanger performance also changes with the approach between fluid temperature and air temperature, and with design constraints. “Run AI jobs at night” is therefore not a universal rule. Lower nighttime dry-bulb temperature does not always improve evaporation, guarantee cheaper electricity, or reveal the water consumed to generate that electricity.

An idealized phase-change calculation shows the scale. Evaporating one kilogram of water near 25°C requires about 2,442 kJ/kg of latent heat according to standard measurements, and 1 kWh equals 3,600 kJ. Thus $3600/2442 \approx 1.474$ kilograms of evaporated water per kWh of heat removed under **ideal complete evaporation**. At a density near one kilogram per liter, this is about 1.47 L/kWh of evaporated heat. It is not an absolute water limit for a system with multiple cooling mechanisms, total water withdrawal from a utility, or water attributed to a cloud service. [S14](#)

Related but distinct definitions:

Symbol	Definition	Unit	Conceptual boundary
E_{IT}	Energy for compute, memory, and qualifying network equipment inside the defined IT boundary	kWh	Can vary with quantization, idle load, and processor type
E_{site}	Facility energy including IT, cooling, and supporting infrastructure	kWh	Not necessarily equal to gross grid generation before losses
W_{site}	Water consumed on site within the experiment's boundary	liter	Separate from withdrawal and electricity-generation water
PUE	E_{site}/E_{IT}	dimensionless	Requires consistent time and measurement boundaries
WUE	W_{site}/E_{IT}	L/kWh	A site measure here; excludes upstream water
W_{elec}	Water consumption in electricity supply chains attributed to use	liter	Requires location- and time-specific grid data unavailable here
Y	Useful tasks that satisfy quality and deadline requirements	value or work units	Must be defined before the field study

At a broader boundary, the multi-resource objective could take the form $E_{site} + \lambda\{W_{site} + W_{elec}\}$. This work **does not calculate** W_{elec} because no hourly, location-specific marginal factor for Saudi electricity generation and water type was available. We therefore did not present it as zero. It remains **outside the model boundary**.

7. Synthetic experiment design: 48 hours without a city name

To separate a computational demonstration from field evidence, we created two 48-hour weather sets with an invented daily cycle repeated twice. T_{db} and T_{wb} each follow a phase-aligned sinusoidal curve with different coefficients in scenarios named `hot_dry` and `coastal_humid`. Those names describe two stylized regimes, not Saudi weather stations. The hours are computational periods. Hour zero has no calendar date and no corresponding National Center for Meteorology reading. The Center documents an API for historical observations, current conditions, and forecasts that requires credentials. We did not obtain a documented hourly series for this work, and annual climate averages cannot justify a claim about one real hour at a data-center site. [S18](#)

We then invented **48 batch jobs**, $j = 0 \dots 47$, one released in every hour and each due within a window of up to nine hours. A 48-hour interactive workload U_t is fixed to its arrival hour, preserving its urgency. Synthetic energy is set at 0.8 kWh per interactive-work unit and 1.0

kWh per batch-work unit. Each hour also carries 18 kWh of fixed IT electricity representing idle or standby load, under a ceiling of 150 kWh of IT energy per hour. A “work unit” has no direct meaning in tokens, Groq chips, or real-facility capacity. It is an abstract quantity used to analyze timing choices.

The synthetic electricity curve for each cooling method is $c_{m,t}$, a ratio to IT energy in hour t , where $m \in \{dry, evap\}$. Its water counterpart is $w_{m,t}$ in L/kWh of IT energy. In the archived code, **dry-cooling overhead** rises with air temperature, while **evaporative cooling** uses less electricity in some hours and more water. The dry option carries a small, nonzero synthetic water load of 0.035 L/kWh of IT energy to represent supporting uses. It must not be described as an entirely water-free process.

Input	Invented value	Why it is included	What it does not represent
Horizon	48 hours	Permits scheduling across fictional days	A full year of seasonality
Maximum capacity	150 kWh IT per hour	Creates plausible congestion	Commercial rack capacity
Standby electricity	18 kWh IT/hour	Prevents a claim that moving work eliminates consumption	Measured Groq idle draw
Interactive-unit electricity	0.8 kWh	Standardizes time-bound work	A token
Batch-unit electricity	1.0 kWh	Makes work shifting computable	A GPU or LPU measure
Batch flexibility	Up to 9 hours	Constrains workload shifting	Every Batch API workload
λ	0 to 0.12 kWh/L	Traces a mathematical trade-off	A water price or official policy weight

The reproducibility archive contains the literal commands that generate these inputs. Keeping the piecewise functions, including their `max` terms and cooling transition points, in executable form rather than only in a rounded table permits a reader to modify and rerun them.

8. The mathematical program and its solvability

Let $x_{j,t} \geq 0$ be the amount of batch job j completed in hour t . A variable exists only when the hour falls between release r_j and deadline d_j . If b_j is the required work for the job, every job must complete in full:

$$\sum_{t=r_j}^{d_j} x_{j,t} = b_j, \quad x_{j,t} \geq 0.$$

Under the formulation's constant capacity during each hour, the hourly IT-energy constraint is:

$$18 + 0.8 U_t + \sum_{j:r_j \leq t \leq d_j} x_{j,t} \leq 150.$$

Because batch work can be divided into fractional units, **linear programming** is appropriate for this exercise. Real jobs may be indivisible; KV state may have to move; model replicas may have to open; and batch sizes may be integral. Such conditions may require **mixed-integer linear programming**, or a staged online solution. We do not claim that a fractional model resolves those details.

For each hour, the cooling method is selected outside the linear program by minimizing $c_{m,t} + \lambda w_{m,t}$. After fixing m_t , the cost of one batch-work unit in that hour is $1 + c_{m_t,t} + \lambda w_{m_t,t}$, yielding:

$$\min_x \sum_{t=0}^{47} \sum_j x_{j,t} [1 + c_{m_t,t} + \lambda w_{m_t,t}]$$

Interactive and standby energy remains part of the **final evaluation** of every policy. It need not appear as a variable term in the scheduling objective because it is fixed with respect to x once the cooling method is chosen. Under the model's linear, separable costs, the per-hour cooling argmin followed by the workload LP is equivalent to the corresponding joint choice. It is not a realistic mixed operating model of a cooling plant. The model assigns no switching cost or minimum run time to a cooling mode, and assumes that a mode can change freely by hour. An industrial study must remove those assumptions. Actual facilities also differ in whether they can select dry or evaporative operation within an hour. Software cannot add a water tower to a building designed without one.

The implementation uses the **SciPy HiGHS solver through linprog**, with one equality constraint per job and one capacity constraint per hour. Two baseline policies run work at the first opportunity, or **FIFO**, with evaporative-only or dry-only cooling. Two additional policies combine FIFO with the lowest-electricity cooling mode in each hour, and optimized work shifting with cooling-mode selection inside each job window. Sensitivity analysis repeats the optimum at **eight water weights** from zero to 0.12. The archive contains every point, not only those selected for display. No GPU or LPU is represented in the energy functions, so none of the results can be read as a chip comparison.

9. Executed results and internal checks

The program was executed locally with SciPy, and the reproducibility archive contains the run documentation, inputs, and results. The table selects **seven policies** for each weather regime. Every result uses the same period and workload. **FIFO-energy** starts batch work when it

becomes available and selects the lower-electricity cooling method in each hour. LP_{λ} denotes the linear-program solution at a specified water weight. Energy and water change only because of the invented functions and carry no statistical standing outside the model. An independent rerun reproduced all aggregate values below to the displayed precision. Because the LP admits tied optima and the archived dependencies are not fully pinned, an equally optimal rerun can assign some batch units to different hours while preserving the objective, total work, deadlines, capacity, and aggregate electricity and water. The hourly schedule is therefore not claimed to be byte-identical across solver environments.

Synthetic weather	Policy	Total electricity, kWh	Site water, L	Dry-cooling hours
Hot-dry	FIFO–evaporative	5223.07	6818.11	0
Hot-dry	FIFO–dry	5490.52	165.20	48
Hot-dry	FIFO–energy	5192.71	4829.34	18
Hot-dry	LP, $\lambda=0$	5179.88	4166.87	18
Hot-dry	LP, $\lambda=0.02$	5188.14	3559.52	22
Hot-dry	LP, $\lambda=0.05$	5243.32	2222.32	30
Hot-dry	LP, $\lambda=0.12$	5407.56	165.20	48
Humid-coastal	FIFO–evaporative	5277.24	7076.90	0
Humid-coastal	FIFO–dry	5504.52	165.20	48
Humid-coastal	FIFO–energy	5261.37	5451.87	14
Humid-coastal	LP, $\lambda=0$	5245.08	4893.67	14
Humid-coastal	LP, $\lambda=0.02$	5263.52	3640.80	22
Humid-coastal	LP, $\lambda=0.05$	5313.90	2248.73	30
Humid-coastal	LP, $\lambda=0.12$	5438.69	165.20	48

Internal consistency check. Every policy completes approximately **4,394.2 work units**, including fixed immediate work and 48 batch jobs, with zero violations of release windows or completion deadlines and zero violations of the IT ceiling. The program checks every cost total, equality between partitioned and required job work, and the placement of each fraction inside its job window, and rejects an unsuccessful solver result. These assertions test **whether the model was implemented as written**. They do not test whether a thermal coefficient, customer behavior, or Groq device is realistic.

Moving from LP $\lambda=0$ to LP $\lambda=0.02$ in the hot-dry profile increases electricity by **8.26 kWh** and reduces water by **607.35 L** over 48 hours. In the humid-coastal profile, electricity rises by **18.43 kWh** and site water falls by **1,252.87 L**. These figures are exact outputs of an invented program; percentages derived from them must not be presented as economic or environmental returns for a Saudi plan. Comparing the two fictional worlds demonstrates a methodological point: **different cooling functions can change how an additional kilowatt-hour trades against liters of water avoided.**

Calculated electricity and site-water frontier for two invented weather sets

Figure 1. Every point comes from an executed solver at a declared value of λ . No point is a field measurement or uses a real electricity grid.

10. Why the computational exercise succeeds but remains incomplete before a real machine

In theory, moving a batch looks like a free exchange between time slots. In practice, its benefit can fail in four places.

First, static power and shutdown. If racks draw much of their power while idle, moving work will not reduce much energy. If an operator uses deep shutdown, wake-up will not be instantaneous and models may require minutes to load and refill memory. The model's fixed idle load prevents facility energy from collapsing into the sum of per-job energy. Its limitation is that it does not test shutdown, wake-up, or readiness for sudden interactive demand.

Second, model residency in SRAM or HBM. A batch moved between hours can encounter a memory-capacity constraint. A model rich in weights and KV state requires partitioning across chips. The cost of evicting one state and restoring another can consume the cooling benefit. *Model residency, KV-cache occupancy, swap traffic, compilation cache, and interconnect load* must be measured beside energy, rather than added after the experiment.

Third, service requirements and model mix. Congestion in the batch queue alone may not matter, but interactive users can share the same devices through another queue. The tail of TTFT and request-completion time must remain within each service class's limit; an average is insufficient. Variation in context and output lengths can radically change the prefill and decode mix. A real test must therefore sample request-length distributions and arrival waves rather than assume homogeneous units.

Fourth, thermodynamics across multiple loops. Cooling effectiveness depends on surface area, heat-exchanger type, pumping cost, server inlet temperature, precipitation, dust, make-up-water treatment, and more, not on an air thermometer alone. It is not scientific to assume that

an actual facility can switch freely between dry and evaporative operation every hour without reference to its design. A field study should use real *part-load performance, cooling approach temperature, fan curves, pump curves*, and switching costs.

This is the epistemic boundary between an **existence proof** and an **effect estimate**. The former shows that time flexibility can be converted into a multi-resource optimization under coefficients selected to permit it. The latter must establish that those coefficients represent real machines, weather, and workloads, then compare the policy against current operations with fair task allocation and measurement.

11. Prior work: what came before this proposal?

The scientific question did not begin with this project. The 2025 Berkeley review examined the relationships among server efficiency, electricity energy, grid water, cooling method, and climate variation. The US Department of Energy presents PUE and WUE as useful ratios with measurement-boundary cautions. The distinction between site resources and the broader resource chain therefore predates this study. [S11](#) [S13](#)

The 2026 preprint “**Balancing Bits and Drops: Stress-Adjusted Water Management for Data Centers**” examines spatial and temporal water stress and proposes water-aware workload management, including attention to dry cooling and off-site water. It directly refutes any claim that this paper invented the first water scheduler. The contribution proposed here is narrower: **study multi-stage inference on Saudi infrastructure while separating job classes, processor telemetry, cooling, and hours**. [S15](#)

The August 2026 preprint **PowerSlider** addresses instantaneous power constraints and incorporates prefill/decode asymmetry and service-quality times in a short-horizon repeated solution. Its authors report 78.3% experimental goodput at a 30% power reduction, compared with 47.6% for their strongest baseline in the tested environment. Those figures cannot be transferred to an LPU, Groq, or Saudi Arabia, and the paper’s electrical objective is not the same as the cooling-water objective here. It nevertheless shows that **phase-aware scheduling under power limits** is more than a conversational hypothesis: a published system, workloads, and experiments exist and can be challenged. [S16](#)

Vendor papers and claims also provide falsifiable metrics. NVIDIA reported a benchmark for Gemma 4 31B on the Groq 3 LPX platform and described high performance for agentic workloads. The fair treatment is to bind each result to its platform, model, context, and benchmark conditions rather than turn it into an average for all Saudi services. Fair comparison also requires distinguishing tokens for one user, aggregate tokens, and completed-task time. [S07](#)

12. A Saudi field experiment designed to reject the hypothesis if it fails

This section is a **proposal and has not been executed**. It is distinct from the synthetic program described above. It would require permission from a computing-facility operator, calibrated electricity and water measurements, and a way to replicate workloads without exposing user data or disrupting production. It need not disclose private architecture publicly; controlled statistical aggregates can still make the method and equations reviewable.

Sample and strata. Collect an operational trace over weeks that include variation in temperature, humidity, and request lengths. Pre-register three workload classes: live conversation with a hard latency limit, multi-round agent tasks with a reasonable completion-time limit, and batch work with real deadlines. Preserve the distribution of inputs, outputs, timestamps, and model versions under each policy. Exclude private text from the experimental store, or convert it into anonymous length measures under an explicit definition of what remains comparable.

Comparison arms. Use (a) the production scheduler as it stands; (b) scheduling aware of batch deadlines and GPU/LPU load only; (c) scheduling aware of time and electricity; (d) scheduling aware of time, electricity, and site water; and (e) an analytical arm adding an electricity-generation water factor when a reliable source exists. Apply the same feasible quality, service, and power ceilings to every arm, and create a safety stop if an arm begins to harm a real request. The final arm has no presumed advantage. If it does not improve useful work per unit of resource within the stated confidence interval, reject its claim.

Hardware layer. Compare a general GPU architecture with an LPU or hybrid platform only when **the same models can run at comparable precision and quality**. Record precision, quantization method, context length, KV capacity and model-residency locations, concurrent users, network conditions, and the tool sequence. A web demonstration on one accelerator cannot be compared with a production environment on another network and attributed to chip physics alone. If a model cannot run on a particular chip at the same quality threshold, report that as an eligibility result rather than deleting the case.

Environmental layer. Record T_{db} and T_{wb} at the same times as liquid supply and return temperatures, fan and pump power, make-up-water meters, and **water withdrawal and consumption as separate quantities**. Record water source and type when available. Read IT, PDU, UPS, and cooling electricity at explicit meter boundaries. Publish instrument accuracy, sampling intervals, and clock differences between electricity and request traces so that delayed measurements do not create a false temporal correlation.

Success measures. Lead with the count of correct tasks completed within their deadlines, facility energy per thousand accepted tasks, and site water consumed per thousand accepted tasks. Report PUE and WUE alongside them, plus TTFT and completion time at p50, p95, and p99. Report quality-failure and rerun rates, because a faster invalid answer is not better production. Include the time and operating cost of preparing and compiling several models. A full day's operational work should not collapse into one fast token.

Limits of causal identification. Comparing a mild week with a hot week is insufficient. Request mix, software versions, weather, available capacity, and electricity prices can all change together. A better design randomizes authorized test work across alternating windows with similar conditions, blocks carry-over such as rack uptime or model warm-up, or uses a crossover design with washout periods. Researchers should pre-register outcome distributions, report negative results and heterogeneous conditions, and only then consider conclusions at national or multibillion-project scale.

13. The next economics: from processor value to the value of converting electricity into valid time

A city with computing facilities can sell capacity on paper, sell raw token output, or sell quality-assured tasks under time commitments. These products transfer different levels of industrial capability. The first sells installed capacity. The second sells a narrower accounting unit. The third requires engineering across context length, model–processor fit, network bandwidth, scheduling, electricity and water meters, and experiments that establish whether the work is useful. Better workload allocation can raise output value without increasing accelerator count, while implementation cost and complexity can erase the gain. This is not a revenue claim. It is a **testable microeconomic hypothesis**.

Saudi Arabia could develop operational businesses adjacent to Groq's original service: **inference-workload adapters across heterogeneous architectures**, scheduling algorithms measured against local conditions, and benchmark libraries connecting model and context type to quality in Arabic and other languages. Platforms could measure **accepted tasks per joule** and **accepted tasks per site liter**, distinguishing work that requires fractions of a second from work that benefits from hours of flexibility. Sales and defensibility cannot be inferred from heavy global investment. The software's decision value has to be demonstrated against native NVIDIA and Groq solutions, academic work, and cooling suppliers. Otherwise, the exercise merely renames tools that already exist.

The period from 2027 to 2035 should be treated as a test tree, not a numerical forecast. If standardized cooling telemetry becomes usable and platforms manage heterogeneous processors with limited state-transfer cost, physical scheduling may deserve a place in the inference service's **control plane**. If immediate work remains dominant or state transfer

consumes more than timing saves, reducing energy per token through compilers, quantization, and electrical efficiency may deserve priority. If site-water savings conflict with the water used to produce electricity, no policy can be declared the winner before its accounting boundary is defined. These conclusions are conditional and can change with experimental evidence; they are not statements of fact about 2035.

14. Computational reproducibility and independence

The reproducibility archive contains executable code, input data, reviewable outputs, an evidence workbook, and both raster and vector versions of the synthetic figures. The computation depends on **NumPy, SciPy, and Matplotlib**, without API keys or calls to a proprietary product. The archive's weather and workload inputs are generated synthetically; its hourly, summary, and sensitivity tables are computed outputs. A run manifest records the configuration, and the evidence workbook exposes the result tables, calculations, and links between inputs. The figure register labels the figures as synthetic.

There is no **probability sample** of climates or companies in this paper, and no confidence-interval analysis that could be justified for an industrial outcome. HiGHS solves a written linear program; a valid solution under those assumptions is not a behavioral or national estimate. We did not test real queues from Saudi centers or establish consumption per token on a Groq chip. A reader can rerun the code, substitute authorized measured curves, and test the differences. The distinction between **aggregate computational reproducibility** and **independent empirical replication** is essential: the aggregate values reproduce to the displayed precision, although tied optima can yield different valid hourly allocations; empirical replication is a future scientific project and has not been performed.

15. What remains unknown, and what the figure must not pretend to know

The public sources reviewed here do not disclose the Dammam Groq center's electricity or water consumption, its heat-rejection design, the type and quantity of chips at each site, the tokens produced inside Saudi Arabia alone, or how much additional water capacity might accompany the announced investment commitment. No reviewed evidence links the newer NVIDIA Groq 3 LPX platform to a specific Saudi center, though Groq said it planned to be among the first adopters. We have not established an independent performance difference between Groq LPU and GPU systems under matched task conditions inside Saudi Arabia. "Inference provider to HUMAIN" is an announced commercial relationship rather than an invoice ledger, so it does not establish usage or revenue.

Nor did this work retrieve hourly National Center for Meteorology observations. “Coastal” in the experiment names only an invented weather curve. We did not measure WUE at any center, desalinated-water margins, or the water intensity of grid electricity. A real center may use a cooling system or operational restrictions that do not permit the choices assumed in the model. These are not editorial gaps to be covered with more recommendations. They are the exact limits of what the evidence can establish.

Falsifiable conclusion. Saudi Arabia has a real public signal of Groq inference operations and an opportunity to test software that improves **correct work completed on time per unit of electricity and water**. Scientific and economic success cannot begin with the assumption that an LPU is best at every task, or with a promise that dry cooling automatically equals sustainability. It begins with two operating questions: **what actually limits task speed—memory, compute, network, or queue—and how much electricity and water did that task require at this place and hour, measured at auditable boundaries?**

References

The sources below are institutional primary materials or research papers. Vendor announcements are treated as vendor statements rather than independent verification. The synthetic experiment is distinct from any real Groq or Saudi installation.

- **S01.** [Groq and Aramco Digital, partnership announcement, 12 September 2024.](#)
- **S02.** [Groq, Dammam operations and Saudi expansion commitment, 10 February 2025.](#)
- **S03.** [Groq, HUMAIN inference-provider announcement, 27 May 2025.](#)
- **S04.** [Groq, “The Groq LPU Explained,” 7 March 2025.](#)
- **S05.** [GroqChip Processor product sheet.](#)
- **S06.** [Groq, non-exclusive NVIDIA inference-technology license, 24 December 2025.](#)
- **S07.** [NVIDIA, Groq 3 LPX full-production announcement, 24 August 2026.](#)
- **S08.** [Groq platform and infrastructure page.](#)
- **S09.** [Groq production service tiers and Batch API documentation.](#)
- **S10.** [Groq Prometheus request-latency metrics.](#)
- **S11.** [US Department of Energy, *Cooling Water Efficiency Opportunities for Federal Data Centers*, 9 January 2019.](#)
- **S12.** [Groq, planned adoption of NVIDIA Groq 3 LPX and Vera Rubin, 2026.](#)
- **S13.** [Lei et al., “The water use of data center workloads,” 2025, DOI 10.1016/j.resconrec.2025.108310.](#)
- **S14.** [US National Bureau of Standards, measurements of water’s latent heat near 25°C.](#)
- **S15.** [Talukder et al., “Balancing Bits and Drops,” arXiv:2607.22617, preprint, 2026.](#)

- **S16.** Li et al., “PowerSlider: Exploiting Phase Asymmetry,” arXiv:2608.21719, preprint, 2026.
- **S17.** Reproducibility archive: code, synthetic inputs, computed outputs, and run documentation.
- **S18.** Saudi National Center for Meteorology, weather-data API documentation; the API was not queried for this work.