Hot AI in Cold Space: Thermal-Crosstalk-Aware Scheduling for Sustainable Orbital AI Clusters

Shuyi Chen
Southern University of Science and Technology
Shenzhen, China
chensy8@mail.sustech.edu.cn

Zhengchang Hua
Southern University of Science and Technology
Shenzhen, China
huazc@mail.sustech.edu.cn

Nikos Tziritas
University of Thessaly
Lamia, Greece
nitzirit@uth.gr

Georgios Theodoropoulos
Research Institute of Trustworthy Autonomous Systems
Southern University of Science and Technology
Shenzhen, China
theogeorgios@gmail.com

Abstract

Terrestrial AI training faces an unsustainable energy and water crisis, positioning Orbital Data Centers (ODCs) as a "zero operational carbon" alternative. However, the sub-10μs communication latency required for distributed Large Language Model (LLM) training forces ODCs into extreme physical density, triggering a critical "Proximity-Thermal Paradox." As these high-density systems scale into Monolithic Structures or Proximity Swarms, they suffer from intense thermal-fluid crosstalk (heat traps in shared cooling loops) and thermal-radiative crosstalk (mutual heating that blocks deep-space cooling radiators). If left unmitigated, this persistent heat stagnation not only triggers severe thermal throttling that degrades training throughput, but also induces severe thermal fatigue, drastically shortening hardware lifespans and generating premature space e-waste. To make orbital AI truly sustainable, this position paper challenges traditional uniform load-sharing. We propose the Thermal-Aware Heterogeneity Thesis, which treats spatial cooling variances as a primary resource management dimension. Building on this, we introduce Thermal-Load Balancing (TLB), a software framework that dynamically migrates LLM workloads to the coolest available units based on instantaneous fluid temperatures or absorbed radiation. Our analysis demonstrates that TLB resolves thermal bottlenecks to restore Model Flops Utilization (MFU), while simultaneously reducing physical thermal stress. Extending the operational lifespan of orbital hardware is crucial to amortize the massive embodied carbon of rocket launches, outlining a necessary pathway to scale orbital AI without accelerating e-waste.

CCS Concepts

• **Computing methodologies** → **Distributed computing methodologies**; *Artificial intelligence*; • **Hardware** → **Thermal issues**; *Impact on the environment*.

HotCarbon'26, Seattle, WA

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

Keywords

orbital data centers, sustainable computing, embodied carbon, thermal-aware scheduling, large language models, satellite swarms

ACM Reference Format:
Shuyi Chen, Zhengchang Hua, Nikos Tziritas, and Georgios Theodoropoulos. 2018. Hot AI in Cold Space: Thermal-Crosstalk-Aware Scheduling for Sustainable Orbital AI Clusters. In *Proceedings of HotCarbon'26.* ACM, New York, NY, USA, 6 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Terrestrial AI training faces an unsustainable energy and water crisis, making Orbital Data Centers (ODCs) an attractive "zero operational carbon" alternative [4, 19]. However, to truly offset the massive *embodied carbon* of rocket launches [20], ODCs must maximize their lifetime Model Flops Utilization (MFU) and avoid premature hardware failures. This presents a severe challenge for the most demanding workloads: distributed Large Language Model (LLM) training. LLMs, particularly those utilizing Tensor Parallelism, demand sub-10μs communication latencies [24]. This forces computing nodes into extreme high-density configurations (e.g., 3-meter physical proximity), triggering a critical *Proximity-Thermal Paradox*: the spatial compactness required for low-latency synchronization inevitably leads to intense thermal crosstalk.

Furthermore, distributed LLM training exhibits unique "All-or-Nothing" synchronization patterns (e.g., All-Reduce operations). A single thermally congested "straggler" node stalls the entire gigawatt-scale cluster [8, 13]. If unmitigated, this persistent heat stagnation not only degrades training throughput but also induces severe thermal fatigue, which drastically shortens hardware lifespans and generates premature "space e-waste."

Scaling these high-density ODCs leads to two distinct architectural paradigms, each presenting unique multi-scale thermal interferences. The first paradigm is *Monolithic Structures*: massive self-assembled units that rely on centralized fluid loops. The unique challenge here is *Thermal-Fluid Crosstalk*, where heat from high-density GPU tiles accumulates along the shared cooling flow path, creating downstream heat traps [2]. The second paradigm is *Proximity Swarms*: free-flying nodes (e.g., 20kW design power) with independent cooling maintained in tight formations [26]. These suffer from *Thermal-Radiative Crosstalk*, where geometric shadowing

and mutual heating block deep-space cooling windows, particularly for nodes trapped in the cluster's core [18].

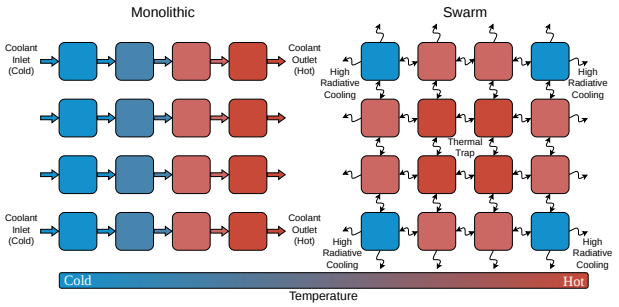

Figure 1: Two ODC architectural paradigms: Monolithic Structures with centralized cooling loops and Proximity Swarms of free-flying nodes.

In both paradigms, traditional load-balancing frameworks force structural and operational uniformity, which inevitably leads to synchronous bottlenecks. To make orbital AI physically and environmentally sustainable, we introduce the *Thermal-Aware Heterogeneity Thesis*. We argue that ODCs must reject uniform load-sharing and instead treat spatial and temporal cooling variance as a primary, schedulable computational resource.

Building on this, we propose the Thermal-Load Balancing (TLB) paradigm. TLB is a software orchestration framework that dynamically migrates LLM computation blocks to the coolest available units based on instantaneous fluid temperatures or absorbed radiation. By proactively shifting tasks, TLB actively resolves heat traps, prevents the emergence of thermal stragglers, and mitigates cyclical thermal stress on the silicon.

This paper outlines a pathway to scale orbital AI without accelerating space e-waste. Specifically, we make the following contributions:

- We define the *Proximity-Thermal Paradox* in ODCs, detailing how extreme density induces severe Thermal-Fluid and Thermal-Radiative Crosstalk.
- We introduce the *Thermal-Aware Heterogeneity* thesis, establishing that exploiting cooling variance is essential to preventing thermal fatigue and straggler-induced MFU degradation.
- We propose the *Thermal-Load Balancing* framework, providing workload migration strategies to resolve heat traps in both monolithic flow paths and swarm radiative fields.
- We formulate an evaluation methodology, modeling a dense multi-node mesh, to demonstrate that TLB restores MFU and extends hardware lifespans by reducing cyclic thermal stress, thereby amortizing the embodied carbon of ODCs.

2 Related Work

To contextualize the novelty of the TLB paradigm, we contrast it with the blind spots in two established research domains:

Space-Based Computing and Thermal Management. Recent studies have proposed ODCs for on-orbit edge computing [3, 4, 19, 22, 25], primarily targeting asynchronous tasks like remote sensing offloading [6, 7, 11]. However, supporting synchronous LLM training requires extremely dense spatial topologies to emulate the sub-10μs latency of terrestrial Optical Circuit Switches [10, 23], introducing unprecedented thermal congestion. Concurrently, spacecraft thermal management relies on radiative and fluid systems [14–16]. Yet, current designs assume isolated satellites or loosely formed constellations [5, 21]. They fail to address the multi-scale thermal crosstalk inherent to GW-scale monoliths or densely packed swarms, nor do they account for the rapid thermal fatigue induced by dynamic AI workloads.

Terrestrial AI Load Balancing. In terrestrial data centers, distributed AI load balancing optimizes for computational throughput or cooling energy expenditure (i.e., operational carbon) [1, 8, 12, 17, 27]. These models rely on active convective air-cooling and cannot be applied to the vacuum of space. Crucially, because ODCs are powered by solar arrays, operations are effectively zero-carbon. Thus, the sustainability objective must pivot from minimizing electricity usage to maximizing hardware lifespan to amortize the massive embodied carbon of space launches [20].

In summary, existing literature treats spatial topology, thermal dissipation, and AI scheduling as isolated domains. When conventional symmetric load balancers distribute synchronous AI tasks across ODCs, they ignore spatial cooling variances. Thermally congested nodes quickly hit temperature thresholds, triggering hardware throttling that stalls the entire cluster. This creates a catastrophic "straggler" effect [9] while simultaneously inducing severe thermal fatigue that shortens hardware lifespan. Our work bridges this exact gap by treating thermal heterogeneity not as a hardware limitation to be mitigated, but as a fundamental, software-schedulable resource.

3 System model

3.1 Orbital Data Center Model

We model ODCs as high-density satellite clusters and consider two distinct architectural paradigms. Let $\mathcal{N} = \{1, \ldots, N\}$ be the set of satellite nodes. Each node i is characterized by its thermal capacitance C_i, dynamic power limits (P_{idle} and peak P_{max}), and maximum processing speed S_{max}. Heat dissipation properties differ fundamentally depending on the architecture:

Monolithic Structures: The monolithic ODC features a rigid, large-scale (e.g., MW-to-GW scale) macro-structure shaped like a massive unified container, similar to conceptual tether-based orbital space stations (Starcloud-4) [2]. Power generation and heat dissipation for the entire ODC are managed globally by the primary structure via externally mounted large-area heat radiators and solar arrays. Standalone compute units are housed within this structure and managed by a centralized liquid cooling system.

As shown in the left part of Figure 1, multiple coolant pipes are routed through the data center, pumping coolant through the compute units before returning it to the heat exchanger. Along the pathway of each coolant loop, upstream nodes receive cold coolant, while downstream nodes receive pre-heated coolant, thereby creating an inherent thermal imbalance across the compute units.

Proximity Swarms: The proximity swarm ODC architecture consists of multiple independent satellite computing nodes flying

in a tight formation, similar to Google's Project Suncatcher concept [26]. These nodes form a dense constellation with inter-node distances typically ranging from hundreds of meters to a few kilometers, communicating via high-bandwidth FSO links, as shown in the right part of Figure 1.

Each satellite in the swarm is equipped with its own solar panels and heat radiators. Although each node utilizes independent passive radiative cooling, the nodes within the constellation are physically close enough to thermally influence one another. Due to view factor occlusion caused by neighboring nodes, their radiators experience varying effective radiating areas. Furthermore, warmer nodes may radiate waste heat directly toward their neighbors, establishing a dynamic thermal imbalance across the distributed swarm.

3.2 Heat Dissipation and Thermal Crosstalk Models

Let P_{in} and $T_{instant}$ denote the instantaneous chip power and temperature. We link computational load to heat generation non-linearly: $P_{in} = P_{idle} + (P_{max} - P_{idle})(S_{instant}/S_{max})^{\gamma}$. The transient thermal state is dictated by the net heat flow $P_{in} - P_{out}$, where heat dissipation P_{out} is architecturally dependent:

Thermal-Fluid Crosstalk (Monolithic): Downstream nodes are cooled by fluid pre-heated by upstream components ($T^{(i)}_{fluid} = T^{(i-1)}_{fluid} + \eta P^{(i-1)}_{out}$). Consequently, the downstream convective heat dissipation rate ($P^{(i)}_{out} \propto T^{(i)}_{instant} - T^{(i)}_{fluid}$) degrades, creating downstream heat traps.

Thermal-Radiative Crosstalk (Swarms): Dense formations rely on deep-space radiation governed by the Stefan-Boltzmann law ($P_{out} \propto T^4_{instant} - T^4_{amb}$). Mutual geometric shadowing severely restricts this, which we model via an Effective View Factor $\rho_i \in (0, 1]$. Peripheral nodes retain larger effective radiating areas (higher ρ_i), while core nodes suffer severe occlusion (significantly lower ρ_i), triggering immediate heat traps.

3.3 Throttling and Lifespan Degradation Models

To avoid hardware damage, nodes enforce a soft temperature threshold (T_{soft}) and a hard threshold (T_{hard}). When $T_{instant}$ exceeds T_{soft}, the processing speed $S_{instant}$ scales down linearly from S_{max}, capping at a minimum safe speed S_{min} once T_{hard} is reached.

Furthermore, we model thermal-induced hardware degradation using the standard Arrhenius equation, where the Mean Time To Failure (MTTF) is exponentially dependent on temperature: $MTTF \propto \exp(E_a/k_B T)$. Here, E_a is the activation energy (e.g., 0.685 eV) and k_B is the Boltzmann constant. Because of this exponential relationship, even minor reductions in peak operating temperatures yield disproportionately large extensions in hardware lifespan.

3.4 Workload Model with Asymmetric Data Parallelism

To accelerate training when memory is sufficient, we apply Data Parallelism. Assuming every computing node holds a full replica of the model, the training data batch is split across different nodes. Each node processes a certain volume of data, determined by the configured micro-batch size.

For distributed LLM training via Data Parallelism, the global batch (size B_{global}) is partitioned such that each node processes a micro-batch of b_i tokens. The computation time is directly proportional to b_i and inversely proportional to the thermally throttled speed $S_i(T)$.

Crucially, a global training step strictly requires gradient synchronization (e.g., All-Reduce). This means the total step latency is dictated entirely by the single slowest node in the cluster: $t_{step} = \max_i(t_{comp}(i) + t_{comm})$. This mathematical "max" operation establishes the severe consequence of the Proximity-Thermal Paradox: a single node suffering from intense thermal crosstalk will experience degraded $S_i(T)$, thereby stalling the entire ODC cluster and causing a global collapse in MFU.

4 The Thermal-Load Balancing (TLB) Framework

4.1 A Generalized Orchestration Architecture

TLB is designed as a generalized, closed-loop orchestration framework, fundamentally decoupling the load-balancing *mechanism* from any specific scheduling *policy*. It continuously operates through a three-phase control loop: 1) **Thermal Telemetry**: Aggregating real-time physical metrics, such as instantaneous silicon temperatures, local fluid pre-heating levels, and dynamic view factor occlusions; 2) **Capability Profiling**: Translating raw thermal data into an abstract capability score (w_i) that quantifies each node's real-time thermal margin; and 3) **Asymmetric Workload Slicing**: Dynamically adjusting the distribution of the AI workload (e.g., micro-batch sizes in Data Parallelism) to match the profiled capabilities.

This abstraction allows ODC operators to plug in arbitrarily complex decision engines, ranging from convex optimization solvers to Deep Reinforcement Learning (DRL) agents, without altering the underlying distributed AI communication backends.

4.2 Proof-of-Concept: Proportional Heuristic

For this position paper, rather than deploying a computationally expensive global optimization solver, we implement a lightweight, greedy heuristic to demonstrate the viability of the TLB paradigm.

Standard Data Parallel frameworks enforce a uniform batch distribution, allocating $B_{global}/|\mathcal{N}|$ tokens to each node. This severely penalizes the entire cluster when nodes with degraded cooling drop their processing speed $S_i(T)$. TLB breaks this uniformity by calculating the thermal capability score w_i. The workload b_i assigned to node i is strictly proportional to its score:

$$b_i = 1 + \left\lfloor (B_{global} - |\mathcal{N}|) \cdot \frac{w_i}{\sum_{k \in \mathcal{N}} w_k} \right\rfloor \quad (1)$$

where every node receives at least one baseline unit of work to maintain gradient synchronization participation, and any fractional rounding remainders are allocated to the highest-scoring nodes.

We define an aggressiveness multiplier α and compute the capability scores based on the architectural paradigm:

Monolithic Flow-Path Priority: We prioritize upstream nodes that receive the coldest fluid. Let L_{pipe} be the total number of nodes on a cooling pipe, and idx_i be the zero-based index of node i along

that pipe. The capability score is:

$$w_i = 1.0 + \alpha \cdot (L_{pipe} - idx_i) \tag{2}$$

Swarm Radiative Priority: We prioritize nodes on the periphery of the cluster with an unobstructed view of deep space. By directly utilizing the Effective View Factor (ρ_i) introduced in our thermal model, which inherently accounts for geometric shadowing, the capability score is:

$$w_i = 1.0 + \alpha \cdot \rho_i \tag{3}$$

By proactively shifting the heaviest computational burdens to nodes with high w_i, TLB inherently limits the heat generation of central or downstream nodes, preventing them from hitting T_{soft} and triggering the straggler effect.

4.3 Integration with the Distributed AI Stack

To transition TLB from a theoretical model to a deployable system, it must bridge the gap between hardware telemetry and the AI application layer. At the hardware level, TLB leverages standard space-grade baseboard management controllers (BMCs) to poll instantaneous silicon temperatures and coolant states. At the application layer, TLB interfaces with distributed AI frameworks (e.g., PyTorch DistributedDataParallel (DDP) or Megatron-LM). Instead of relying on a static `DataLoader` that evenly shards the dataset, TLB implements a *Thermal-Aware Data Sampler*. When the profiling engine updates the capability scores w_i, the sampler dynamically adjusts the micro-batch size b_i for the upcoming training steps.

A key practical challenge in dynamic workload slicing is memory pre-allocation. Terrestrial LLM training heavily relies on static computational graphs (e.g., XLA compilation) to maximize hardware utilization. Dynamically changing batch sizes mid-flight can trigger expensive graph recompilations. To mitigate this, TLB assumes the use of emerging dynamic-shape compilers (e.g., PyTorch 2.0 `torch.compile`) or pre-compiled computational graph buckets corresponding to a discrete set of allowable batch sizes. This ensures that thermal-induced workload shifts do not introduce prohibitive software overhead.

Triggering Policies and Overhead Mitigation. Executing the TLB control loop at every training step would introduce unacceptable synchronization delays. Therefore, TLB employs a hybrid policy: workload redistribution occurs at coarse-grained intervals (e.g., epoch boundaries) to handle slow-moving thermal dynamics like orbital eclipses. Conversely, an event-driven safety net triggers immediate asynchronous task evacuation if hardware telemetry detects a node rapidly approaching T_{soft}, guaranteeing thermal safety while bounding orchestration overhead.

5 Performance Evaluation

To validate the Thermal-Aware Heterogeneity thesis, we implement a proof-of-concept simulation of the TLB framework. Since this paper focuses on the necessity of a paradigm shift rather than proposing an optimal algorithmic solver, we benchmark a greedy variation of TLB against a standard homogeneous load balancer.

5.1 Simulation Methodology

To model the multi-scale thermal dynamics of ODCs, we developed a time-stepped thermal-compute co-simulator that evaluates distributed LLM training workloads executing via Data Parallelism.

For the **Monolithic** architecture, 64 nodes are distributed across 8 liquid cooling pipes (8 nodes daisy-chained per pipe). Downstream nodes process coolant that has been pre-heated by upstream nodes. For the **Proximity Swarm** architecture, 36 independent satellites fly in a dense 6×6 planar grid formation. We implement a radiative view factor model where nodes dynamically cast thermal shadows on each other; edge nodes retain larger effective radiating areas (higher ρ_i), while the core nodes suffer severe radiative occlusion (significantly lower ρ_i).

We compare two scheduling paradigms: 1) **Baseline (Uniform):** Shards the global micro-batch evenly across all nodes, mimicking standard PyTorch DDP behavior; 2) **TLB (Greedy):** Implements the proportional allocation heuristic. We track instantaneous hardware temperatures, dynamic processing speeds (throttling), and global step synchronization latency.

5.2 Mitigating Heat Traps (Spatial Analysis)

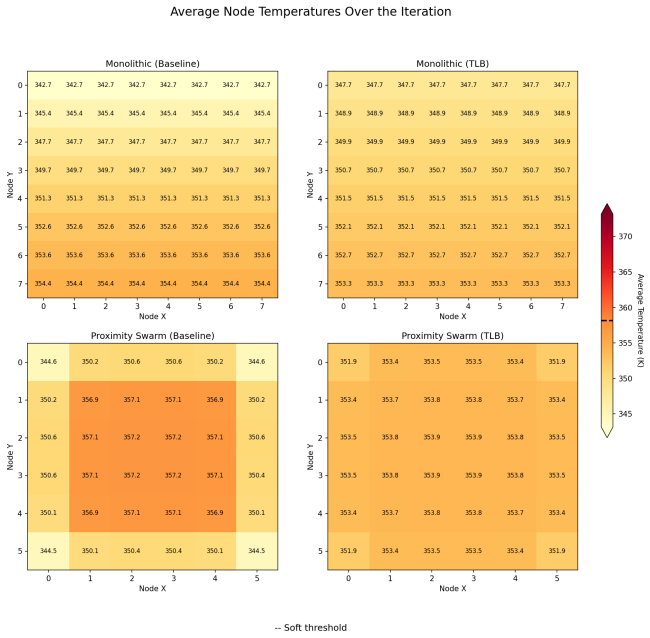

Figure 2: Average Node Temperatures Over the Iteration (Baseline vs. TLB)

The spatial thermal distribution reveals the fundamental flaw of homogeneous scheduling in ODCs. Figure 2 shows the average node chip temperatures in the Monolithic and Proximity Swarm architectures. Under the Baseline policy, the *Proximity-Thermal Paradox* manifests rapidly. In the Monolithic configuration, heat progressively accumulates along the cooling loops, pushing the downstream nodes (Row 7) into thermal saturation (up to 354.4 K / 81.3°C). In the Swarm configuration, the core nodes become severe

heat traps reaching 357.2 K (84.1°C) due to high radiative view factor occlusion, while the periphery satellites remain significantly cooler (344.5 K / 71.4°C).

TLB dynamically maps the thermal topology and deliberately shifts the heaviest computational burdens to the upstream Monolithic nodes and the periphery Swarm nodes. This asymmetric workload slicing neutralizes the cooling variance. For the Monolithic setup, the maximum temperature drops to 353.3 K (80.2°C). For the Swarm, the extreme thermal gradient is flattened, with core temperatures reducing to 353.9 K (80.8°C) and edge temperatures rising to 351.9 K (78.8°C), effectively eliminating spatial heat traps.

5.3 Restoring MFU and Hardware Lifespan

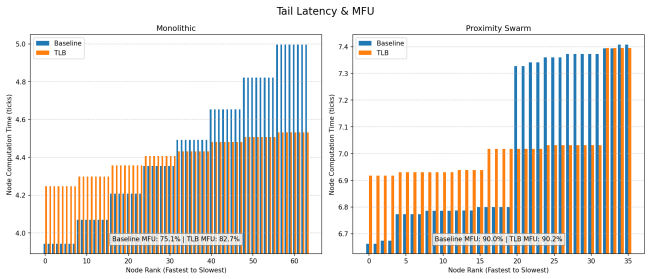

Figure 3: Tail Latency & MFU of node computation times (Baseline vs. TLB).

The resolution of spatial heat traps translates directly into measurable improvements in computational efficiency and hardware sustainability.

Latency Variance and MFU: In the Baseline scenario, nodes trapped in the thermal core exceed temperature thresholds and trigger hardware throttling. Due to the synchronization requirement of LLM training, the entire cluster must wait for the slowest overheated nodes. Figure 3 demonstrates this straggler effect: the Baseline Monolithic architecture exhibits severe step-like latency variance, capping the MFU at 75.1%. TLB compresses this tail latency spread, improving the Monolithic MFU to 82.7%. In the Proximity Swarm, while the Baseline MFU is already high at 90.0%, TLB perfectly equalizes the computation time across all 36 nodes (indicated by the flat TLB latency distribution), raising the MFU to 90.2% and completely eliminating synchronization wait times.

Extending Lifespan to Amortize Embodied Carbon: Using an apparent activation energy (E_a) of 0.685 eV, we calculate the cumulative thermal degradation to determine the relative hardware lifespan extension provided by TLB. Our analysis demonstrates that the TLB policy extends the MTTF of the most thermally constrained nodes, yielding a 6.15% lifespan increase for the core node in the Proximity Swarm and a 1.71% increase for the downstream outlet node in the Monolithic cluster, compared to the Baseline.

Under uniform loading, central nodes experience severe thermal fatigue due to relentless temperature oscillation and sustained peak temperatures (e.g., 357.2 K / 84.1°C). By smoothing out the thermal gradients and capping peak temperatures at safe operational limits, TLB prevents these nodes from entering cyclical throttling regimes. Flattening the temperature curve directly extends the physical durability of the hardware. In orbital computing, where the operational carbon footprint is zero, extending hardware lifespan is the primary mathematical pathway to amortizing the massive embodied carbon of rocket launches.

Limitations and Evaluation Scope. While the absolute throughput improvements (e.g., 7.6% MFU gain in the Monolithic cluster, 0.2% in the Swarm) may appear modest, this evaluation intentionally employs a rudimentary static heuristic. By ignoring secondary bottlenecks like multi-hop optical routing latency and transient fluid lag, this heuristic serves merely as a lower-bound baseline. Rather than presenting a theoretically optimal scheduler, our primary objective is to empirically validate the *existence* of the Proximity-Thermal Paradox and the *solvability* of spaceborne heat traps. Ultimately, extending hardware lifespans by mitigating thermal cycling, even alongside marginal MFU gains, is the fundamental prerequisite for amortizing the immense embodied carbon of space launches.

6 Discussion and Conclusion

Terrestrial AI's voracious appetite for energy and water is driving the industry toward ODCs. However, this transition fundamentally shifts the root cause of unsustainability: while operations become strictly zero-carbon, the environmental burden is entirely front-loaded into the massive *embodied carbon* of rocket launches and the impending accumulation of space e-waste. This paper highlights that the sub-10μs latency demands of LLM training force ODCs into extreme physical density, triggering a Proximity-Thermal Paradox that causes premature hardware failure and collapses MFU.

By introducing the Thermal-Aware Heterogeneity Thesis and the TLB framework, we demonstrated that resolving spatial heat traps through asymmetric software scheduling can restore synchronous throughput and mitigate cyclic thermal fatigue. Extending hardware lifespans via software orchestration is not merely a performance optimization; it is the indispensable prerequisite for amortizing the embodied carbon of space-based AI.

To fully realize sustainable orbital computing, the sustainable computing community must confront several critical open challenges:

1. Dynamic Network-Thermal Co-design: Terrestrial LLM clusters rely on dynamic network reconfiguration (e.g., Optical Circuit Switches [23]) to optimize all-to-all communication. In space, reconfiguring FSO links within a proximity swarm requires mechanical or optical steering. This directly alters a node's physical orientation, instantly changing its radiative view factors and thermal state. Co-optimizing dynamic network topologies with instantaneous orbital thermodynamics remains an entirely unsolved challenge.

2. The E-Waste vs. Carbon Trade-off: While software schedulers like TLB can extend hardware survival, orbital chips currently cannot be repaired. The community must conduct rigorous Life Cycle Assessments (LCAs) to answer a provocative question: does the "zero operational carbon" benefit of ODCs mathematically justify a paradigm of "disposable" high-end AI accelerators? Addressing this may eventually necessitate designing orbital hardware explicitly for in-space recycling.

3. Federated Thermal Telemetry for Heterogeneous Swarms: Future GW-scale ODCs will likely be multi-tenant constellations

composed of hardware from diverse vendors. Executing cluster-wide thermal scheduling requires nodes to share their physical statuses. Developing open standards to broadcast "thermal margins" without exposing proprietary silicon thermal layouts or payload capabilities is essential to building cooperative, sustainable orbital ecosystems.

References

[1] Rui Chen, Bo Liu, WeiWei Lin, JianPeng Lin, HuiWen Cheng, and KeQin Li. 2023. Power and thermal-aware virtual machine scheduling optimization in cloud data center. *Future Generation Computer Systems* 145 (2023), 578–589.

[2] Ezra Feilden, Adi Oltean, and Philip Johnston. 2024. Why we should train AI in space. *Lumen Orbit Inc* (2024).

[3] Ran Ginosar and David Steenari. 2025. Beyond Traditional Payload Data Handling: Micro-Datacenter in Space for Converged Software-Defined Storage and Payload Processing. In *2025 European Data Handling & Data Processing Conference (EDHPC)*. 1–7.

[4] Carlos Guimarães, Alessio Netti, Markus Sauer, Florian Zeiger, Hans-Peter Huth, and Elizaveta Boriskova. 2026. A Survey on Satellite Computing: Connecting the Dots Between Networks and Applications. *IEEE Communications Surveys & Tutorials* 28 (2026), 567–592. doi:10.1109/COMST.2025.3579525

[5] Mohamad Hnayno, Ali Chehade, Henryk Klaba, Hadrien Bauduin, Guillaume Polidori, and Chadi Maalouf. 2022. Performance analysis of new liquid cooling topology and its impact on data centres. *Applied Thermal Engineering* 213 (2022), 118733.

[6] Yifei Hu and Wenbin Gong. 2023. An On-Orbit Task-Offloading Strategy Based on Satellite Edge Computing. *Sensors* 23, 9 (2023). doi:10.3390/s23094271

[7] Qiangqiang Jiang, Lujie Zheng, Yu Zhou, Hao Liu, Qinglei Kong, Yamin Zhang, and Bo Chen. 2025. Efficient On-Orbit Remote Sensing Imagery Processing via Satellite Edge Computing Resource Scheduling Optimization. *IEEE Transactions on Geoscience and Remote Sensing* 63 (2025), 1–19. doi:10.1109/TGRS.2025.3528015

[8] Mingwei Li, Jilin Zhang, Jian Wan, Yongjian Ren, Li Zhou, Baofu Wu, Rui Yang, and Jue Wang. 2020. Distributed machine learning load balancing strategy in cloud computing services. *Wireless Networks* 26, 8 (2020), 5517–5533.

[9] Shigang Li, Tal Ben-Nun, Salvatore Di Girolamo, Dan Alistarh, and Torsten Hoefler. 2020. Taming unbalanced training workloads in deep learning with partial collective operations. In *Proceedings of the 25th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*. 45–61.

[10] Yingzhi Li, Baisong Chen, Haolun Du, Ziming Wang, Heming Hu, Xuetong Li, Huan Qu, Jie Li, Weipeng Wang, Min Tao, et al. 2026. Integrated Optical Wireless Communication Featured With Optical Phased Array Transceivers for Full-Duplex and NonLine-of-Sight Transmission. *Laser & Photonics Reviews* 20, 7 (2026), e00822.

[11] Yuejin Li, Mi Wang, Kai Hwang, Zhengdao Li, and Tongkai Ji. 2023. LEO Satellite Constellation for Global-Scale Remote Sensing With On-Orbit Cloud AI Computing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 16 (2023), 9369–9381. doi:10.1109/JSTARS.2023.3316298

[12] Jianpeng Lin, Weiwei Lin, Wentai Wu, Wenjun Lin, and Keqin Li. 2024. Energy-aware virtual machine placement based on a holistic thermal model for cloud data centers. *Future Generation Computer Systems* 161 (2024), 302–314.

[13] Rui Lu and Dan Wang. 2025. A Thermal-Aware Workload Scheduler for High-Performance LLM Inference in Cooling-Regulated Datacenters. *SIGENERGY Energy Inform. Rev.* 5, 2 (Aug. 2025), 98–104. doi:10.1145/3757892.3757906

[14] Yi-Gao Lv, Yao-Ting Wang, Tong Meng, Qiu-Wang Wang, and Wen-Xiao Chu. 2024. Review on thermal management technologies for electronics in spacecraft environment. *Energy Storage and Saving* 3, 3 (2024), 153–189. doi:10.1016/j.enss.2024.03.001

[15] Erdinç Mermer and Rahmi Ünal. 2023. Passive thermal control systems in spacecrafts. *Journal of the Brazilian Society of Mechanical Sciences and Engineering* 45, 3 (2023), 160.

[16] Jianyin Miao, Qi Zhong, Qiwei Zhao, and Xin Zhao. 2021. *Spacecraft thermal control technologies*. Springer.

[17] Sergio Moreno-Alvarez, Juan M Haut, Mercedes E Paoletti, Juan A Rico-Gallego, Juan C Diaz-Martin, and Javier Plaza. 2020. Training deep neural networks: a static load balancing approach: S. Moreno-Álvarez et al. *The Journal of Supercomputing* 76, 12 (2020), 9739–9754.

[18] Yunus Murat, Hatice Mercan, Nedim Sözbir, and Ahmet Selim Dalkilic. 2025. Thermal design for a communications satellite payload module. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 239, 18 (2025), 7629–7646.

[19] Shimaa Naser, Maryam Tariq, Raneem Abdel-Rahim, De Mi, Azzam Mourad, Hadi Otrok, Mahmoud Al-Qutayri, Ayman Elnashar, and Sami Muhaidat. 2026. From Connectivity to Multi-Orbit Intelligence: Space-Based Data Center Architectures for 6G and Beyond. arXiv:2603.18601 [cs.ET] https://arxiv.org/abs/2603.18601

[20] Robin Ohs, Gregory F. Stock, Andreas Schmidt, Juan A. Fraire, and Holger Hermanns. 2025. Dirty Bits in Low-Earth Orbit: The Carbon Footprint of Launching Computers. *SIGENERGY Energy Inform. Rev.* 5, 2 (Aug. 2025), 26–33. doi:10.1145/3757892.3757896

[21] Yeon-Kyu Park, Geuk-Nam Kim, and Sang-Young Park. 2021. Novel structure and thermal design and analysis for cubesats in formation flying. *Aerospace* 8, 6 (2021), 150.

[22] Cong Peng, Yuanzhi He, Shanghong Zhao, Lingyang Song, and Boyu Deng. 2023. Integration of Data Center into the Distributed Satellite Cluster Networks: Challenges, Techniques, and Trends. *IEEE Network* 37, 3 (2023), 52–58. doi:10.1109/MNET.105.2100614

[23] Leon Poutievski, Omid Mashayekhi, Joon Ong, Arjun Singh, Mukarram Tariq, Rui Wang, Jianan Zhang, Virginia Beauregard, Patrick Conner, Steve Gribble, et al. 2022. Jupiter evolving: transforming google's datacenter network via optical circuit switches and software-defined networking. In *Proceedings of the ACM SIGCOMM 2022 Conference*. 66–85.

[24] Jesmin Jahan Tithi, Hanjiang Wu, Avishaii Abuhatzera, and Fabrizio Petrini. 2025. Scaling intelligence: Designing data centers for next-gen language models. *arXiv preprint arXiv:2506.15006* (2025).

[25] Guoping Wang, Gang Wan, Zhijuan Su, Yang Wang, Yutong Jia, Gong Li, and Shi Liang. 2025. High-Performance On-Orbit Intelligent Computing and Real-Time Services for Remote Sensing Satellites Based on Large-Scale Computing Power in Space. *IEEE Access* 13 (2025), 92114–92133. doi:10.1109/ACCESS.2025.3573932

[26] Blaise Agüera y Arcas, Travis Beals, Maria Biggs, Jessica V Bloom, Thomas Fischbacher, Konstantin Gromov, Urs Köster, Rishiraj Pravahan, and James Manyika. 2025. Towards a future space-based, highly scalable AI infrastructure system design. *arXiv preprint arXiv:2511.19468* 4 (2025).

[27] Qing Ye, Yuhao Zhou, Mingjia Shi, Yanan Sun, and Jiancheng Lv. 2022. DLB: A dynamic load balance strategy for distributed training of deep neural networks. *IEEE Transactions on Emerging Topics in Computational Intelligence* 7, 4 (2022), 1217–1227.