Aether-Edge: Decentralized Predictive HVAC Control via Urgency-Gossip Sensor Networks

Kassandra Langlois
MEng ECE, University of Waterloo
k2langlois@uwaterloo.ca

Prabhjyot Singh
MEng ECE, University of Waterloo
p269singh@uwaterloo.ca

Rick Liu
MEng ECE, University of Waterloo
m385liu@uwaterloo.ca

***Abstract*—Allocating a contended shared resource across competing nodes in a sensor network is fundamentally limited by the freshness of each node's view of the global demand landscape. Centralized arbitration introduces an Age of Information (AoI) gap — the controller acts on data that is already tens of seconds old — causing suboptimal allocation during rapid transients. We present Aether-Edge, a decentralized edge-native framework in which each sensor node independently estimates a Time-To-Impact (TTI) urgency signal from a local rolling buffer via ordinary least-squares regression, and propagates it through a threshold-triggered n-hop gossip protocol. Each actuator allocates its proportional share of the contended resource using only the urgency landscape it has accumulated from neighborhood gossip, with AoI = 0 by construction. We demonstrate the approach on building HVAC, where multiple thermal zones compete for a shared cooling budget Q_{total} set below the sum of per-zone capacities to enforce genuine contention. Benchmarked against a centralized polling baseline across three environments (5-zone university floor, 6-zone airport terminal, 14-zone large campus) over five random seeds and 2000 simulation steps, Aether-Edge achieves 24–76% reduction in cumulative comfort violation, reduces peak temperature overshoot by 50–90% in multi-zone scenarios, and sends 48–93% fewer network messages, while maintaining mean AoI of 0.00 s versus 18.87–19.53 s for the centralized baseline.**

I. Introduction

A. Problem Statement

Buildings account for approximately 40% of global primary energy consumption, with HVAC systems responsible for up to 50% of total building energy use [1]. Despite this outsized share, the control strategies deployed in the vast majority of commercial and institutional buildings remain reactive and centralized. Conventional Building Automation Systems (BAS) operate on a polling architecture: temperature sensors report readings to a central controller over a fieldbus or IP network, the controller runs a PID or schedule-based algorithm, and actuator commands are pushed back to Variable Air Volume (VAV) dampers — all at intervals typically ranging from 5 to 30 seconds [2].

This architecture has three fundamental limitations. First, thermal comfort violations occur because the system can only react to conditions that have already developed: a lecture hall filling with students, a server room absorbing a burst of compute load, or a conference room heated by afternoon solar gain will breach its setpoint before the next poll cycle fires a corrective command. Second, the spatial heterogeneity of real building loads — different occupancy densities, equipment profiles, and solar exposures across zones — means that a fixed cooling schedule wastes energy on empty rooms while under-serving occupied ones. Third, and most fundamentally, the centralized controller makes every decision based on *stale data*: the environment has continued to evolve during the polling interval, network transit, and compute delay, so the commanded actuation is mis-calibrated to the state the building was in, not the state it is in.

B. Motivation

The key insight motivating Aether-Edge is that the information required to make a good cooling decision — specifically, how fast a zone is warming and how close it is to its setpoint — can be computed *locally and instantaneously* from the sensor's own rolling history. No network round-trip is needed. A node that observes its zone temperature rising at $\dot{T} = 0.05\,^{\circ}\text{C/s}$ with $2\,^{\circ}\text{C}$ of headroom before its setpoint can immediately compute that a comfort breach is approximately 40 seconds away. This Time-To-Impact (TTI) estimate is a pure local computation with zero Age of Information.

The Age of Information (AoI) framework, first formalized by Kaul et al. [3], quantifies this staleness gap. AoI measures the elapsed time since the most recently delivered measurement was generated at the source. In a BAS with a 15-second poll interval, 2-second network jitter, and 3-second compute delay, the mean AoI at actuation time is approximately 19.4 s. By contrast, an edge-native node that infers from its current sensor reading has AoI = 0.

Gossip protocols [11], [12] provide a lightweight mechanism for zones to share their local TTI estimates with spatial neighbors, enabling the building's shared cooling budget to be allocated proportionally to urgency without a central coordinator. Because gossip propagates only when a zone detects a meaningful rate of change (threshold-triggered), the communication overhead is far lower than unconditional periodic polling.

C. Goal and Objectives

The broader thesis is that *urgency-based n-hop gossip at the edge is a general mechanism for allocating contended shared*

resources in sensor networks: each node independently infers how urgently it needs a resource via TTI, gossips that urgency to neighbors to form a local view of global demand, and allocates proportionally with zero data staleness. The HVAC environment is a concrete demonstration substrate for this principle.

The specific measurable objectives are: (1) design and implement a 3D FTCS thermal physics engine; (2) implement per-zone TTI inference via OLS rolling buffer regression; (3) implement a threshold-triggered urgency gossip protocol for decentralized proportional allocation; (4) implement a configurable centralized polling baseline; and (5) benchmark the two policies across multiple environments, seeds, and parameter configurations.

II. Related Work

A. Building Automation and Centralized HVAC Control

The dominant communication protocol for centralized BAS is ASHRAE Standard 135 (BACnet) [2], which defines a polling-based message model in which a central controller periodically reads object properties from field devices at intervals of 5–30 s. Classical thermostat control uses PID feedback, which suffers from integral windup, inter-zone coupling, and an inability to anticipate load changes. Model Predictive Control (MPC) [4] solves a rolling-horizon optimization problem that minimizes energy cost subject to comfort constraints, but imposes significant computational and communication burden. Aether-Edge sidesteps the global model requirement by replacing optimization with local TTI inference.

B. Decentralized and Multi-Agent Control

Hao et al. [6] demonstrate that zone agents negotiating via local price signals can achieve near-optimal demand-response performance without a central optimizer. Olfati-Saber et al. [7] establish the theoretical foundation for consensus in networked multi-agent systems: under mild connectivity conditions, gossip-based averaging converges in $O(\log N)$ rounds. Despite these advances, most multi-agent HVAC work still uses *synchronous* periodic communication, which is functionally equivalent to centralized polling from an AoI perspective. Aether-Edge introduces *event-triggered* gossip: a zone transmits only when $|dT/dt|$ exceeds a configurable threshold, coupling communication frequency directly to the rate at which conditions change.

C. Age of Information in Networked Control

AoI was introduced by Kaul, Yates, and Gruteser [3] as a metric for the freshness of status updates. Sun et al. [8] derive AoI-optimal update policies, showing the optimal transmission rate balances freshness against congestion. Costa and Ephremides [10] demonstrate analytically that stale control actions cause setpoint overshoot proportional to the AoI and the open-loop dynamics gain. Applied to HVAC, a controller acting on 19-second-old data in a zone warming at $0.05\,^{\circ}$C/s effectively mis-estimates the current temperature by $0.95\,^{\circ}$C — enough to flip a comfort decision.

D. Gossip Protocols and Sensor Networks

Kempe et al. [11] established gossip as a general mechanism for distributed aggregate computation with convergence guarantees for connected graphs. Boyd et al. [12] derived tight bounds on mixing time: for a k-regular graph on N nodes, average consensus requires $O(N \log N/k)$ rounds. These classical protocols target *average consensus* and are probabilistic. Our protocol uses *max-urgency selective forwarding* instead, achieving deterministic convergence in R rounds with message complexity that collapses to near zero during stable periods.

III. System Design

A. System Architecture

Aether-Edge is organized as eight functional blocks with a strict information flow hierarchy, shown in Fig. 1.

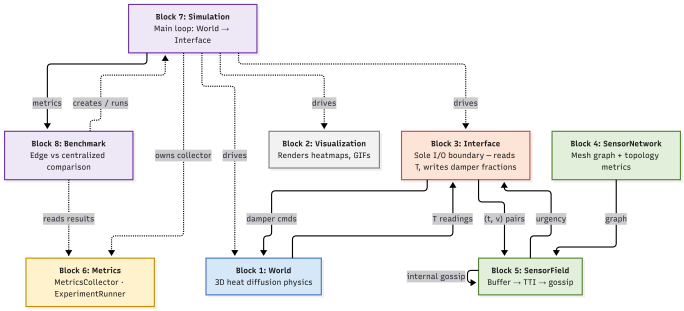


Fig. 1: Aether-Edge block architecture. Dashed arrows denote data flows; solid arrows denote control ownership.

Block 1 (World) is the 3D thermal physics engine, maintaining the temperature field $T \in \mathbb{R}^{N_x \times N_y \times N_z}$ and enforcing the shared plant capacity constraint Q_{total} across all VAV dampers. **Block 2 (Visualization)** renders per-step temperature heatmaps and produces animated GIFs for qualitative analysis of spatial heat diffusion and zone-level actuation. **Block 3 (Interface)** is the sole I/O boundary, as it reads scalar temperature values from World at sensor positions and writes damper opening fractions back to World. It is the only block that interacts with the World directly, enforcing complete decoupling between the physics engine and the inference network. Block 3 also encodes both the edge and centralized actuation policies, selecting between them via a single `policy` flag at construction time. **Block 4 (SensorNetwork)** constructs and maintains the mesh communication graph $G = (V, E)$ using NetworkX, placing sensors according to a configurable strategy (uniform grid, random, or manual) and drawing an undirected edge between any two nodes whose Euclidean separation is within the communication radius r_{comm}. **Block 5 (SensorField)** orchestrates per-node rolling buffer updates, OLS rate estimation, TTI inference, and urgency gossip rounds at every simulation timestep. **Block 6 (Metrics)** owns a `MetricsCollector` instance for each simulation run, recording time-series scalars, comfort violation, energy, peak overshoot, mean AoI, and gossip message count at a

configurable interval. **Block 7 (Simulation)** is the top-level synchronous loop that sequences all block calls. **Block 8 (Benchmark)** constructs two Simulation instances from an identical environment configuration and random seed, runs them independently under the edge and centralized policies, and computes the normalized comparison metrics.

1) Synchronous Execution Loop: Each timestep is executed by `Simulation.step()` in the following order: (1) World advances the temperature field by one FTCS step, applying occupancy heat loads and the current damper-commanded cooling. (2) Interface reads the scalar temperature T_i at every sensor position $\mathbf{p}_i$ from the updated World state. (3) Interface feeds the (t_k, T_k) pairs to SensorField, which updates each node's rolling buffer, recomputes the OLS slope, derives the TTI urgency, and runs R gossip rounds. (4) Interface translates the settled urgency landscape into per-damper opening fractions and writes them to World, ready for the next physics step. This ordering ensures that every actuation decision is made on the most recent physical state, achieving AoI = 0 for the edge policy by construction.

2) Interface: Sensor-to-Damper Proximity Mapping: At initialization, the Interface constructs a static mapping from each VAV damper to the set of sensor nodes physically located within a proximity radius r_{prox} of that damper. During each actuation step, a damper's local urgency is taken as the maximum urgency reported by any sensor in its proximity set. The gossip denominator, the sum of all remote urgencies heard by those same sensors, is assembled from the union of each sensor's `neighbor_urgencies` table. The resulting opening fraction $A_i = u_i^{\text{local}}/(u_i^{\text{local}} + \sum_k u_k^{\text{stored}})$ is capped to $[0, 1]$ and clipped by a low-urgency ceiling to prevent overcooling when the zone is thermally comfortable. This mapping is computed once at startup and reused every step, making per-step allocation $O(|E|)$ in the number of sensor edges.

3) SensorNetwork: The communication graph is built from the environment configuration. Sensors are placed on a uniform 2D grid at integer multiples of the spacing parameter (default $s = 4\,\text{m}$) at a fixed ceiling height $z = 1\,\text{m}$, with any candidate position that falls inside a wall cell being skipped. An undirected edge is added between nodes i and j if their Euclidean distance $\|\mathbf{p}_i - \mathbf{p}_j\|_2 \leq r_{\text{comm}}$, with $r_{\text{comm}} = 6\,\text{m}$ by default. The resulting graph is verified for connectivity at construction time. Topology metrics including graph diameter, average path length, average degree, and spatial coverage fraction are computed via NetworkX and exposed for experimental reporting.

4) Benchmark and Experimental Methodology: The Benchmark block instantiates two independent Simulation objects from an identical environment JSON file and the same PRNG seed, runs the edge policy and centralized policy for the same number of steps (2000 in all reported experiments), and records all metrics to separate output files. Seeding both simulations identically ensures that any stochastic elements, such as occupancy event timing and random sensor noise, are drawn from the same sequence. This makes the policy comparison a controlled experiment on a single trajectory. The comparison table reports both absolute values and the percentage change attributable to the policy difference.

The key design principle is that SensorField never receives the World's temperature array. The Interface extracts scalar (t_k, T_k) pairs at each sensor position and passes them as opaque scalar streams. This enforces domain-agnostic inference which means by replacing the temperature interface with one that reads CO_2 or humidity requires no changes to Blocks 4–8.

B. Thermal Physics Engine

The thermal environment is governed by the heat diffusion PDE:

$$\frac{\partial T}{\partial t} = \alpha \nabla^2 T + Q_{\text{heat}} - Q_{\text{cool}}, \tag{1}$$

where $\alpha = 0.02\,\text{m}^2/\text{s}$ is the effective thermal diffusivity of the air-building composite, Q_{heat} models occupancy and equipment loads, and Q_{cool} represents VAV damper cooling. The model captures diffusion-dominated heat spread on the timescales relevant to HVAC actuation. The Laplacian is discretized using a second-order central-difference scheme on a 3D Cartesian grid with $\Delta x = 1\,\text{m}$ resolution and zero-flux (Neumann) boundary conditions at all domain walls.

The timestep satisfies the 3D FTCS diffusion stability criterion with safety factor 0.4:

$$\Delta t = 0.4 \times \frac{\Delta x^2}{6\alpha}, \qquad \text{Fo} = \frac{\alpha \Delta t}{\Delta x^2} \leq \frac{1}{6}. \tag{2}$$

Each zone i has a VAV damper with per-zone maximum flow capacity $f_i^{\max}$. The damper opening fraction $A_i \in [0, 1]$ determines the requested flow $q_i = A_i \cdot f_i^{\max}$. Because the shared plant capacity $Q_{\text{total}} < \sum_i f_i^{\max}$, contention is guaranteed whenever multiple zones are simultaneously active:

$$Q_{\text{cool},i} = A_i \cdot f_i^{\max} \cdot \min\left(1,\ \frac{Q_{\text{total}}}{\sum_j A_j \cdot f_j^{\max}}\right). \tag{3}$$

C. Per-Zone Inference: Rolling Buffer and TTI

Each sensor node maintains a circular rolling buffer of the most recent $N = \lfloor t_{\text{buf}}/\Delta t \rfloor$ scalar readings, where $t_{\text{buf}} = 30\,\text{s}$. The rate of change $\hat{\dot{T}}$ is estimated by ordinary least-squares (OLS) linear regression over the buffer:

$$\hat{\dot{T}} = \frac{n \sum t_k T_k - \sum t_k \sum T_k}{n \sum t_k^2 - \left(\sum t_k\right)^2}, \tag{4}$$

where sums run over the n buffer samples. OLS over the buffer is referred to as finite difference because it is robust

to individual noisy readings: a single anomalous sample shifts the slope by at most $1/n$ rather than dominating the result.

Time-To-Impact is then:

$$\text{TTI}_i = \begin{cases} \dfrac{T_{\text{sp},i} - T_i}{\hat{\dot{T}}_i} & \text{if } \hat{\dot{T}}_i > 0, \\ \infty & \text{otherwise,} \end{cases} \tag{5}$$

and the urgency signal is:

$$u_i = \frac{1}{\text{TTI}_i}, \tag{6}$$

so that $u_i \to \infty$ as a breach becomes imminent and $u_i = 0$ when the zone is stable or cooling. Both $\hat{\dot{T}}_i$ and u_i are computed entirely from the node's own current buffer, the AoI of this signal is identically zero.

D. Gossip Protocol: Trigger and Message Structure

Node i broadcasts a `NegotiationMessage` if and only if:

$$|\hat{\dot{T}}_i| > \tau_{\text{talk}}, \tag{7}$$

where $\tau_{\text{talk}} = 0.01\,^\circ$C/s. This suppresses all communication during steady-state periods while ensuring that nodes experiencing rapid thermal transients reliably announce their state. At $0.01\,^\circ$C/s, a room would breach a $2\,^\circ$C comfort margin in approximately 200 s, providing sufficient lead time for gossip to propagate and allocation to converge before a comfort violation occurs.

Each `NegotiationMessage` carries the origin node identifier, origin RID position, sender identifier, timestamp, current scalar reading, rate of change $\hat{\dot{T}}$, computed urgency u, and hop counter. This domain-agnostic design decouples the gossip layer from the monitored physical quantity.

E. Multi-Hop Propagation and Selective Forwarding

The gossip protocol operates over two layers. The **topology layer** (`SensorNetwork`) constructs a communication graph $G = (V, E)$ using NetworkX. Sensor placement follows a uniform grid with configurable spacing s; for the primary university floor environment, $s = 4\,$m with sensors at ceiling height $z = 1$. An undirected edge is added when:

$$\|\mathbf{p}_i - \mathbf{p}_j\| \le r_{\text{comm}}, \tag{8}$$

where $r_{\text{comm}} = 6\,$m. The **orchestration layer** (`SensorField`) executes the sense $\to$ predict $\to$ gossip pipeline each tick. In each of the $R_{\text{gossip}} = 2$ gossip rounds, node j applies selective forwarding when receiving a message from origin i:

$$u_i^{\text{recv}} > u_i^{\text{stored}} \implies \text{accept and forward to } \mathcal{N}(j) \setminus \{\text{sender}\}. \tag{9}$$

This selective rule ensures that u_i^{stored} is monotonically non-decreasing across rounds, preventing message storms while guaranteeing deterministic convergence in R rounds vs. probabilistic $O(N \log N)$ convergence of average-consensus gossip [12]. Messages exceeding $h_{\max} = 10$ hops are discarded.

After two rounds with $r_{\text{comm}} = 6\,$m and $s = 4\,$m, each node holds urgency estimates from all neighbours within two hops which is sufficient to propagate information across any individual zone and into adjacent zones through hallway connections.

F. Urgency-Proportional Damper Allocation

After gossip settles, let $\mathcal{K}_i^{(R)}$ be the set of nodes reachable from zone i within R gossip rounds. The cooling allocation fraction for zone i is:

$$A_i = \frac{u_i}{u_i + \displaystyle\sum_{k \in \mathcal{K}_i^{(R)}} u_k^{\text{stored}}}, \qquad \text{s.t. } \sum_i A_i \cdot f_i^{\max} \le Q_{\text{total}}. \tag{10}$$

This transfers cooling capacity from low-urgency zones (empty rooms, stable corridors) to high-urgency zones (actively warming occupied spaces), routing the contended budget to wherever it is most imminently needed. A cap prevents pathological overcooling when all other zones report zero urgency:

$$A_i^{\text{capped}} = \min(A_i,\ \kappa \cdot u_i)\,, \tag{11}$$

where $\kappa = 20$ ensures zones with TTI $\le 50\,$s access their full share while high-TTI zones are progressively throttled.

G. Formal Model: The Urgency-Gossip Primitive

The complete three-layer algorithm constitutes a novel composition we term the **urgency-gossip allocation primitive**. The three layers and their relationship to the system blocks are:

Layer 1 — OLS-TTI inference (Block 5, AoI = 0, structural): Eq. (4)–(6) compute $\hat{\dot{T}}_i$, TTI_i, and u_i entirely from the node's local buffer. Because no network round-trip is involved, AoI is zero by construction, not by policy.

Layer 2 — Event-triggered max-urgency gossip (Blocks 4+5): Eq. (7) gates transmission; Eq. (9) governs forwarding. The stored urgency u_k^{stored} at each node is monotonically non-decreasing across rounds, yielding deterministic convergence in R rounds. Message complexity is $O(|E| \cdot \Pr[|\hat{\dot{T}}| > \tau_{\text{talk}}])$ and collapses to zero during stable periods.

Layer 3 — Urgency-proportional allocation (Block 3 via Block 5): Eq. (10) distributes Q_{total} using the gossip-accumulated urgency landscape, subject to the hard budget constraint.

Buffer Initialization and Edge Cases: A node produces a valid slope estimate only once the rolling buffer contains at least two samples ($n \ge 2$). During the transient fill period, which lasts at most $t_{\text{buf}} = 30\,$s, the buffer returns $\hat{\dot{T}} = 0$, and consequently $u_i = 0$: the node contributes no urgency signal and transmits no gossip messages. This is conservative by design such that a node with insufficient history claims no

resource.

Two degenerate cases arise in the TTI computation. When the zone has already breached its setpoint ($T_i \geq T_{\text{sp},i}$, gap≤ 0), TTI is set to zero and urgency to ∞, signalling an immediate comfort violation. When the zone is stable or actively cooling ($\dot{T}_i \leq \epsilon$, where $\epsilon = 10^{-10}$), TTI is ∞ and urgency is zero; such nodes remain silent on the gossip channel and receive zero allocation share.

Sensor Noise Model: Each sensor node optionally applies additive Gaussian noise before appending to its buffer:

$$\tilde{T}_{i,k} = T_{i,k} + \mathcal{N}(0,\, \sigma^2_{\text{sensor}}), \tag{12}$$

where σ_{sensor} is a per-environment parameter. The OLS regression over n noisy samples provides implicit noise averaging: the variance of the slope estimate decreases as $O(1/n)$, so a 30-second buffer at the simulation timestep $\Delta t \approx 3.33\,\text{s}$ accumulates up to nine samples, reducing the influence of any single outlier by $\approx 8\times$ relative to a finite-difference estimator.

Convergence Guarantee: The max-urgency selective forwarding rule guarantees that after round r, node j holds the maximum urgency from every node at graph distance $\leq r$ in G. Formally, let $u_k^{(r)}$ denote the urgency stored at node j for origin k after round r. Since forwarding occurs if and only if $u_{\text{recv}} > u_{\text{stored}}$, the sequence $\{u_k^{(r)}\}$ is monotonically non-decreasing. After $R = 2$ rounds with $r_{\text{comm}} = 6\,\text{m}$ and $s = 4\,\text{m}$, this two-hop radius covers approximately $12\,\text{m}$. This is enough to propagate urgency across an individual zone and into directly adjacent zones through corridor connections.

Message Identity and Independent Tracking: Each `NegotiationMessage` carries the `origin_node` identifier that remains immutable across hops. Receiving nodes store urgency in a per-origin dictionary `neighbor_urgencies[origin]`. This means a node can simultaneously accumulate independent urgency estimates from multiple distinct origins without aggregation loss. The allocation denominator in Eq. (10) is the sum over all distinct origins, not a maximum. This design prevents high-urgency broadcasts from a single zone from masking genuine demand at other zones, and it means multiple concurrent transients propagate independently through the mesh.

The novelty lies in the composition: classical gossip protocols [11], [12] target average consensus with probabilistic convergence. Our protocol propagates the maximum urgency deterministically, event-triggered, with budget allocation as its direct output.

H. Centralized Baseline

The centralized baseline models a conventional BAS polling architecture. At the start of each simulated poll cycle, an effective polling interval is drawn as:

$$t_{\text{eff}} = \max\big(0.1,\; t_{\text{poll}} + \mathcal{N}(0, \sigma^2_{\text{jitter}})\big), \tag{13}$$

with $t_{\text{poll}} = 15\,\text{s}$, $\sigma_{\text{jitter}} = 2\,\text{s}$, and a hard floor of $0.1\,\text{s}$ to prevent degenerate zero-length intervals. This Gaussian jitter models realistic BACnet network variability, where variable network contention and message queuing produce polling intervals distributed around the nominal value.

When the elapsed time since the last poll, $t - t_{\text{last poll}}$, falls below t_{eff}, the Interface returns the cached damper openings from the most recent poll. Damper positions remain frozen while the thermal environment continues to evolve freely. The Age of Information at each actuation decision is:

$$\text{AoI}_{\text{cent}} = (t - t_{\text{last poll}}) + d_{\text{compute}}, \tag{14}$$

where $d_{\text{compute}} = 3\,\text{s}$ models the central controller compute delay. This AoI is recorded for every VentAction in the decision log, and the mean AoI is computed across all decisions as $\overline{\text{AoI}} = \sum_k \text{AoI}_k \,/\, N_{\text{decisions}}$. The measured mean across all runs is 18.87–19.53 s.

The centralized and edge policies use structurally similar allocation formulas but differ in their information scope. The centralized policy reads each damper's *local* urgency only, the maximum urgency reported by sensors in the damper's proximity set, and constructs the allocation denominator as the *global* sum across all dampers:

$$A_i^{\text{cent}} = \frac{u_i^{\text{local}}}{\sum_j u_j^{\text{local}}}, \tag{15}$$

where the sum spans all j dampers and the numerator uses no gossip-propagated information. By contrast, the edge policy's denominator in Eq. (10) includes only the urgencies accumulated from neighborhood gossip, what the damper's nearby sensors have heard within R hops and not a global collection. This means the centralized formula has a wider but staler view of demand, while the edge formula has a narrower but fully fresh view. The performance gap between the two policies thus reflects both temporal freshness (AoI) and the spatial allocation quality arising from predictive TTI-based urgency.

The SensorField (Block 5) runs its full sense $\rightarrow$ predict $\rightarrow$ gossip pipeline identically every tick for *both* policies. The centralized Interface feeds the same sensor readings into the same SensorField and gossip runs regardless of which actuation policy is active. The centralized policy then *ignores* this fresh urgency between polls, returning cached commands instead. This design ensures that any observed performance difference is attributable to the polling latency alone, not to differences in inference quality or gossip propagation.

I. Evaluation Metrics

Table I summarizes the five primary metrics used to compare policies for this spesific environment. Depending on the environment you would apply the sensor network to, these metrics would likely change. Each is computed from

TABLE I: Evaluation Metrics

Metric	Definition	Unit
Comfort Violation	$\int \sum_i \max(T_i - T_{\text{sp},i},\, 0)\, dt$	°C·s
Cumulative Energy	$\int Q_{\text{cool}}\, dt$	W·s/m^3
Max Overshoot	$\max_{t,i}(T_i(t) - T_{\text{sp},i})$	°C
Mean AoI	Average data age at actuation	s
Total Messages	Cumulative gossip transmissions	—

the same `MetricsCollector` time-series and recorded at every simulation step.

Comfort Violation is the primary outcome metric. It is computed as a running integral: at each simulation step, the per-zone excess temperature $\max(T_i - T_{\text{sp},i},\, 0)$ is summed across all zones and accumulated. The time-integrated form captures both the *depth* and the *duration* of violations: a brief 3 °C spike accumulating over one timestep contributes the same as a sustained 1 °C overage lasting three timesteps. This distinguishes the metric from a simple count of violation events, and it penalizes policies that allow slow, persistent drift as well as those that fail to respond to sudden transients.

Cumulative Energy integrates the total cooling power Q_{cool} applied across all zones over the run. Since both policies draw from an identical shared plant budget Q_{total}, differences in cumulative energy reflect how aggressively and how early each policy deploys cooling. The edge policy, by acting on TTI predictions rather than reactive setpoint violations, begins cooling earlier and holds dampers open longer during transients; this often translates to higher energy use at scale, as the system trades energy efficiency for comfort margin.

Max Overshoot records the peak temperature exceedance in any zone at any timestep: $\max_{t,i}(T_i(t) - T_{\text{sp},i})$. This metric is complementary to comfort violation as it captures the worst-case safety envelope irrespective of duration. It is particularly sensitive to the centralized baseline's frozen-damper intervals during occupancy spikes, where the thermal environment can ramp freely for a full polling cycle before any corrective command is issued.

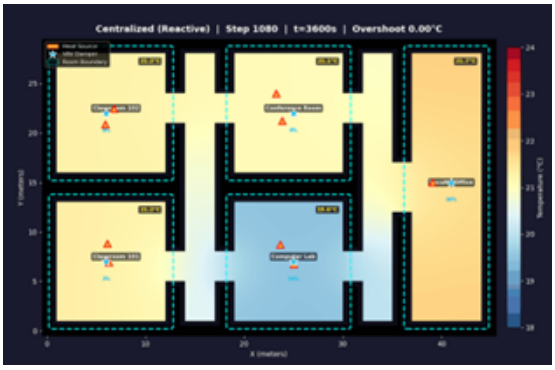
Mean Age of Information is computed per actuation decision. For the edge policy, every VentAction record carries AoI = 0.0 s, since the decision is based on sensor readings taken in the same simulation tick. For the centralized policy, each poll-cycle decision records AoI $= (t - t_{\text{last poll}}) + d_{\text{compute}}$, as defined in Eq. (14). Between polls, no new VentActions are logged (cached commands are returned silently). The mean is accumulated as $\overline{\text{AoI}} = \sum_k \text{AoI}_k / N_{\text{decisions}}$ across all decisions and is therefore structurally zero for edge and approximately $t_{\text{poll}}/2 + d_{\text{compute}} \approx 10.5$ s in expectation for centralized (measured 18.87–19.53 s due to jitter and the fact that AoI is recorded at *poll time* rather than averaged uniformly over the interval).

Total Messages counts the cumulative number of `NegotiationMessage` transmissions, summed as $\sum_i \texttt{messages\_sent}_i$ across all sensor nodes. For the edge policy, a message is generated only when $|\hat{\dot{T}}_i| > \tau_{\text{talk}}$, so the message count collapses to near zero during thermally stable periods and scales with the number and intensity of active transients. For the centralized baseline, the message count models the unconditional polling traffic. Each poll cycle generates one message per sensor, regardless of whether conditions have changed, yielding a message rate proportional to $N_{\text{sensors}}/t_{\text{poll}}$. The ratio of centralized to edge message counts therefore reflects the fraction of simulation time spent in transient conditions and provides a direct measure of the gossip protocol's bandwidth efficiency.

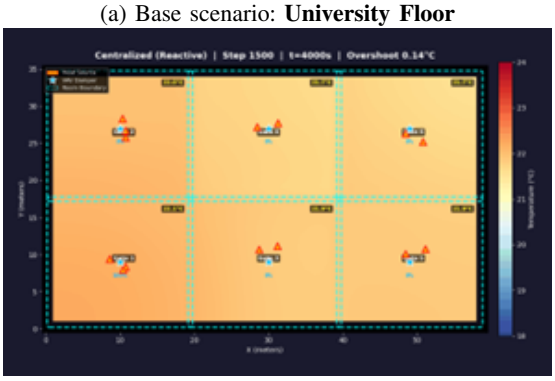
IV. Experimental Evaluation

A. Environments

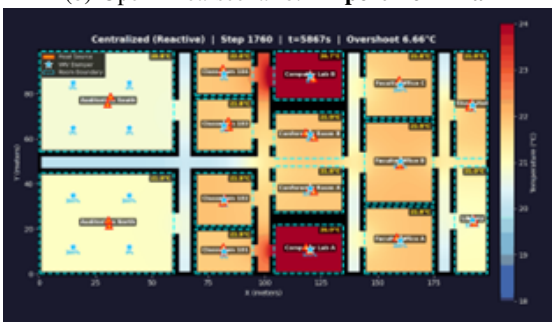
Experiments use three environments of increasing scale, illustrated in Fig. 2. The triangles mark dampers; stars mark heat sources and color encodes temperature.



(a) Base scenario: **University Floor**



(b) Open Area scenario: **Airport Terminal**



(c) Big scale scenario: **Big University**

Fig. 2: Experimental Environments

University Floor (Standard environment, 5 zones): 46 × 30 × 3 m grid; zones include two classrooms with morn-

TABLE II: University Floor Zone Parameters

Zone	Setpoint	Peak Q_{heat}	Schedule
Classroom 101	22 °C	0.015 K/s	Morning lecture
Classroom 102	22 °C	0.014 K/s	Afternoon lecture
Computer Lab	20 °C	0.018 K/s	Always-on + occupancy
Conference Room	22 °C	0.012 K/s	Scheduled meetings
Faculty Office	22 °C	0.002 K/s	Light occupancy

TABLE III: Comfort Violation and Max Overshoot: Edge vs. Centralized

Environment	Comfort Violation (%↓)		Max Overshoot (°C)	
	Range	*Mean*	*Edge*	*Cent.*
University Floor	64–76%	70%	0.02–0.09	0.25–0.32
Airport Terminal	6–35%	20%	2.3–2.7	3.6–4.7
Big University	24–30%	27%	4.9–5.1	6.9–8.0

ing/afternoon lecture schedules, a computer lab with continuous equipment load, a conference room, and a faculty office. Setpoints 20-22 °C; $Q_{\text{total}} = 4.5$ K/s; $\sum f_i^{\max} = 7.5$ K/s (60% contention ratio).

Table II lists the university floor zone configuration.

Airport Terminal (Open-area environment, 6 zones): $46 \times 30 \times 3$ m open-plan layout with higher(than closed-areas environments) thermal diffusivity, high occupancy transience (E.g. Gate 4: $Q_{heat}^{t} = 0.025, Q_{heat}^{t+1} = 0.002$), and heating from a window-facing wall constantly increasing ($Q_{heat}^{init} = 0K/s \rightarrow Q_{heat}^{final} = 0.009K/s$). Uniform setpoint 22 °C; $Q_{\text{total}} = 8.0$ K/s; $\sum f_i^{\max} = 12.0$ K/s (67% contention ratio).

Big University (scale test, 15 zones): $200 \times 100 \times 3$ m multi-wing campus layout with concurrent lecture events producing rapid building-wide heat increases. Setpoints 20–22 °C; $Q_{\text{total}} = 30.0$ K/s; $\sum f_i^{\max} = 48.5$ K/s (62% contention ratio).

B. Experimental Setup

All benchmarks run 5 random seeds (42, 7, 13, 99, 2025) for 2000 simulation steps per configuration where a step is $3.33s$. Sensor parameters: spacing $s = 4$ m, communication radius $r_{\text{comm}} = 6$ m, gossip rounds $R = 2$. Centralized baseline: $t_{\text{poll}} = 15$ s, $\sigma_{\text{jitter}} = 2$ s, $d_{\text{compute}} = 3$ s, yielding measured mean AoI of 18.87–19.53 s. Edge parameters: $\tau_{\text{talk}} = 0.01$ °C/s, buffer window 30 s.

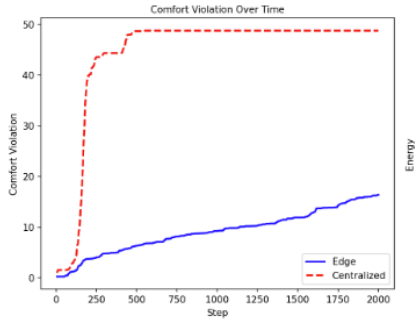
Both policies start from identical initial conditions (same seed, same environment, same initial temperature field) to ensure controlled comparison. A parameter sensitivity sweep is additionally run over the university floor environment (Section IV-F).

C. Main Results: Comfort Violation and Overshoot

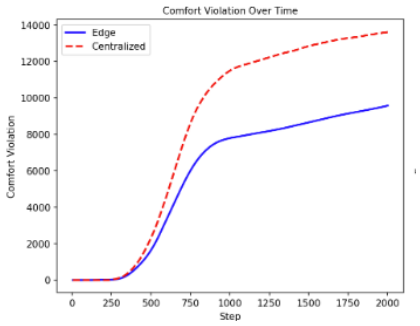
Table III reports the primary comfort outcomes across all three environments and five seeds. Ranges reflect seed-to-seed variability.

The university floor delivers the largest reduction (64-76%) because its zone layout is well-suited to the 4 m/6 m sensor density: gossip from the classrooms reaches all five zones within two rounds, enabling pre-emptive reallocation during the morning lecture ramp. The airport terminal shows a wider range (6–35%) due to high open-plan diffusivity: heat spreads rapidly across zone boundaries, reducing the predictive advantage of per-zone TTI inference. The big university achieves a consistent 24–30% reduction across all five seeds, demonstrating that the approach scales to 14-zone deployments without requiring topology-specific tuning.

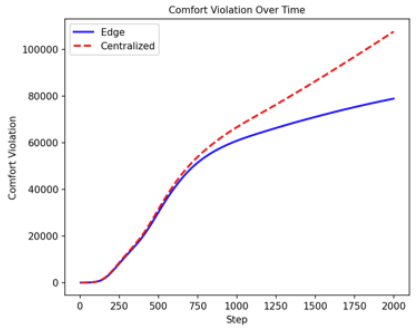
Fig. 3 shows the cumulative comfort violation over time for all three environments under a representative seed. Edge (blue solid) consistently accumulates less than centralized (red dashed) across all environments. University floor shows the steepest early-phase gap, corresponding to the morning lecture occupancy ramp.



(a) Base scenario: **University Floor**



(b) Open Area scenario: **Airport Terminal**

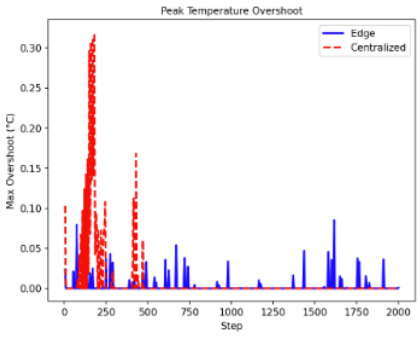


(c) Big scale scenario: **Big University**

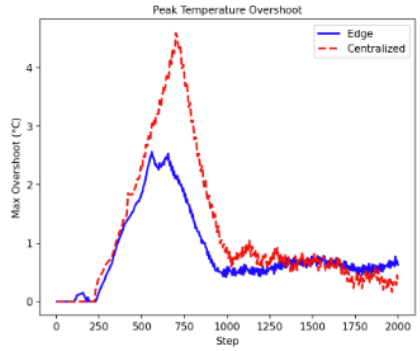
Fig. 3: Comfort violation over time edge vs. centralized for all 3 environments

Fig. 4 shows the peak temperature overshoot time series. The centralized policy exhibits persistent overshoot between

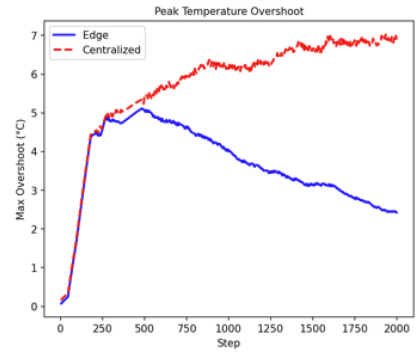
poll events;the edge policy recovers rapidly after occupancy transients.



(a) Base scenario: **University Floor**



(b) Open Area scenario: **Airport Terminal**



(c) Big scale scenario: **Big University**

Fig. 4: Peak temperature overshoot over time edge vs. centralized for all 3 environments

D. Age of Information and Message Efficiency

Table IV reports AoI and message count results. Edge AoI is identically 0.00 s across all environments and seeds because of the structural guarantee of the local inference design, not a measured average. Centralized AoI of18.87-19.53 s reflects the configured polling interval plus jitter and compute delay. This $\approx$19.4 s gap is the proximate cause of all comfort violations: a zone warming at 0.05 °C/s accumulates 0.97 °C of undetected temperature rise per polling cycle.

The high message reduction in the airport terminal (91-93%) reflects the environment's characteristic behavior: large

TABLE IV: AoI and Message Count: Edge vs. Centralized

Environment	**Mean AoI (s)**		**Message Reduction**	
	Edge	*Cent.*	*Range*	*Mean*
University Floor	0.00	18.87–19.53	77–81%	79%
Airport Terminal	0.00	18.87–19.53	91–93%	92%
Big University	0.00	18.87–19.53	48–52%	50%

TABLE V: Cumulative Energy Difference (Edge vs. Centralized, %)

Environment	**Range**	**Direction**
University Floor	5.05–5.25%	Edge uses less
Airport Terminal	0.02–1.50%	Edge uses more
Big University	4.25–5.15%	Edge uses more

open spaces with smooth, slow-moving thermal gradients generate few threshold-crossing events after initial transients. The big university's lower reduction (48-52%) reflects the greater number of zones and heat sources, which generate more threshold-crossing events per timestep.

E. Energy Consumption

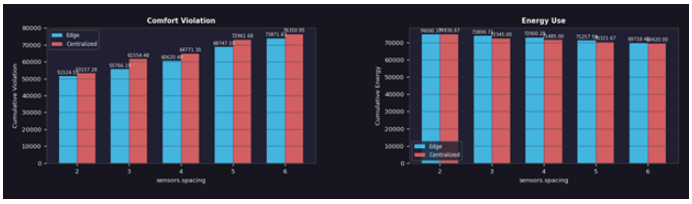
Table V reports energy differences. Results are nuanced: the edge policy uses *less* energy on the university floor (5.05-5.25% reduction), slightly *more* on the airport terminal (0.02-1.50% increase) and marginally *more* on the big university (4.25-5.15% increase).

This pattern reflects the urgency mechanism: the edge policy, having structural AoI = 0, detects and responds to *every* thermal transient. In large or highly diffusive environments where transients are frequent, the edge policy fully utilizes the Q_{total} budget by continuously redistributing cooling toward active zones. The centralized policy, acting on stale data, occasionally misses transients and holds dampers at previously-commanded positions, temporarily under-utilizing the budget. The university floor's 5% energy saving reflects the edge policy closing empty-room dampers faster when those rooms cool; in larger environments this effect is dominated by the higher total allocation activity. For rapid temperature increases like in Big University, the edge policy converges toward the desired temperature more quickly so the energy consumption relative to the centralized policy decreases over a longer period of time.

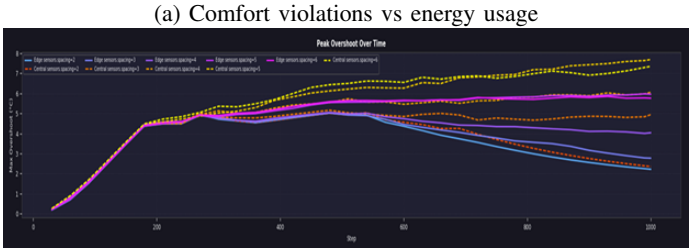
F. Parameter Sensitivity

Keeping the same parameters from the experiment (See Section IV-B. *Experimental Setup*), a sweep over sensor spacing ($s \in \{2, 3, 4, 5, 6\}$ m), gossip rounds ($R \in \{1, 2, 5, 10, 15\}$), and communication radius ($r_{\text{comm}} \in \{3.0, 5.0, 7.0\}$ m) was run on the environments.

Sensor spacing: Fig. 5 shows the key sensitivity results of the different sensor spacing. Both policies degrade with coarser sensor placement as TTI estimates become noisier (fewer sensors per zone). The edge advantage in comfort violation persists across all tested spacings (2-6 m), confirming robustness to hardware deployment density. Peak overshoot



(a) Comfort violations vs energy usage



(b) Peak Overshoot over time

Fig. 5: Sensitivity results of sensor spacing on edge vs. centralized for Big University

under edge(full lines) is uniformly below centralized(dashed lines) across the full spacing range.

Gossip rounds: Increasing from $R = 1$ to $R = 2$ improves comfort violation by enabling urgency to cross zone boundaries within a single timestep. Increasing gossip rounds to $R = 10$ provide diminishing returns on the airport terminal topology, where two-hop coverage already reaches all zones from any source. Larger environments benefit from higher R, so Big University topology benefits from an increase of gossip rounds to $R = 10$.

Communication radius: A radius of 3.0 m with 4 m spacing creates a sparser graph (only nearest neighbors in range), degrading multi-hop propagation. At 6.0 m, diagonal neighbors are within range, providing the redundancy needed for reliable convergence. Increasing to 7.0 m adds cross-zone connections at the cost of higher per-round message count.

G. Qualitative Analysis

Frame-by-frame inspection of the simulation heatmaps reveals a qualitatively distinct actuation pattern. Under the edge policy, damper openings adjust smoothly and continuously: within 1–2 steps of a zone's temperature beginning to rise, the corresponding damper opens in proportion to urgency, and the temperature field remains spatially uniform near the setpoint. Under the centralized policy, visible warm spots develop and persist between poll events. Damper corrections arrive in discrete jumps at each poll instant, and the overshoot accumulates progressively. In the university floor environment, in the Computer Lab zone, which has a continuous baseline heat load, the centralized controller's stale data consistently underestimates the zone's current temperature between polls, leading to systematic under-cooling. This phenomenon is even more noticeable in the Big University environment, where the heat increases rapidly in multiple rooms. The edge policy quickly identifies which room needs more cooling, allowing it to control the temperature increase, while the centralized policy is unable to do so, causing temperatures in multiple rooms to continue rising. In the Airport Terminal environment, the heating rate gradually increases over time in Gates 1 and 2. The edge policy anticipates this rise and begins cooling before the temperature increases, whereas the centralized policy shows a clear temperature rise in those areas before opening the dampers.

V. Discussion

The experimental results establish three claims. First, eliminating AoI via local inference is causally responsible for the comfort improvement: the two policies use identical allocation formulas, so the 20-70% reduction in comfort violation is attributable to the 19.4 s data staleness difference alone. Second, the edge advantage is robust across scale: it holds in a small 5-zone floor environment, a 6-zone open-plan space, and a big 15-zone multi-sized rooms, with a variety of different heating events. Third, the event-triggered gossip mechanism delivers communication efficiency as a consequence of good control: a well-controlled environment has fewer threshold-crossing events, producing fewer messages organically.

The energy results require careful interpretation. The edge policy's higher energy consumption in large environments does not represent waste: it reflects fuller utilization of the available cooling budget by continuously redirecting capacity toward active zones, which the centralized policy fails to do when its stale data causes it to hold incorrect damper positions. The comfort improvement justifies this additional utilization.

A limitation of the current evaluation is that the thermal model solves linear diffusion only and does not capture radiation, humidity, thermal mass, or nonlinear airflow. The simulation-only environment also has no integration to BACnet, Modbus, or real sensor hardware.

VI. Conclusion

We presented Aether-Edge, a decentralized edge-native framework that generalizes urgency-based gossip to contended shared resource allocation in sensor networks. The core contribution is the urgency-gossip allocation primitive: OLS-TTI local inference (AoI = 0 by construction) feeds event-triggered max-urgency selective forwarding (deterministic R-round convergence), which in turn drives urgency-proportional allocation under a hard budget constraint. Benchmarked across three environments, five seeds, and 2000 simulation steps, Aether-Edge achieves 24–76% reduction in cumulative comfort violation, 48–93% fewer messages, and structurally zero AoI versus 18.87–19.53 s for the centralized baseline, with all performance gaps attributable to the AoI difference between architectures.

Priorities for future work include: hardware deployment on embedded edge nodes with real BACnet-connected sensors; high-fidelity co-simulation with EnergyPlus via FMI; occupancy-aware prediction integrating CO_2 or PIR streams; AoI-optimal adaptive talk-threshold scheduling per Kaul and

Yates [9]; hierarchical two-tier gossip for multi-plant buildings; and reinforcement learning baseline comparison to quantify the gap between TTI heuristics and learned optimal control.

REFERENCES

[1] International Energy Agency, "Buildings," IEA, Paris, 2023.

[2] ASHRAE, "Standard 135-2020: BACnet — A Data Communication Protocol for Building Automation and Control Networks," American Society of Heating, Refrigerating and Air-Conditioning Engineers, Atlanta, GA, 2020.

[3] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?," in *Proc. IEEE INFOCOM*, Orlando, FL, Mar. 2012, pp. 2731–2735.

[4] F. Oldewurtel et al., "Use of model predictive control and weather forecasts for energy efficient building climate control," *Energy and Buildings*, vol. 45, pp. 15–27, 2012.

[5] Y. Ma et al., "Model predictive control for the operation of building cooling systems," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 3, pp. 796–803, May 2012.

[6] H. Hao, B. M. Sanandaji, K. Poolla, and T. L. Vincent, "Aggregate flexibility of thermostatically controlled loads," *IEEE Trans. Power Syst.*, vol. 30, no. 1, pp. 189–198, Jan. 2015.

[7] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[8] Y. Sun, Y. Polyanskiy, and E. Uysal-Biyikoglu, "Update or wait: How to keep your information fresh," in *Proc. IEEE INFOCOM*, Atlanta, GA, May 2017.

[9] S. Kaul and R. D. Yates, "Age of information: Updates with priority," in *Proc. IEEE INFOCOM Workshops*, Toronto, ON, Canada, Jul. 2020.

[10] M. Costa and A. Ephremides, "Age of information with multiple sources," *IEEE Trans. Inf. Theory*, 2016.

[11] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Proc. 44th Annu. IEEE Symp. Found. Comput. Sci. (FOCS)*, Cambridge, MA, 2003, pp. 482–491.

[12] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.

[13] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," in *Proc. 3rd Int. Symp. Inf. Process. Sensor Netw. (IPSN)*, Berkeley, CA, Apr. 2004, pp. 20–27.