LAPSE-B

FYP · BSC COMPUTER SCIENCE

Latency-Aware Predictive Optimisation for Real-Time Edge-Based Behaviour Analysis

Adaptive Inference Scheduling for Resource-Constrained Distributed Systems · WISDM & UCI HAR Datasets

Mihran Ali

Student ID: 23034784

Supervisor: Dr. Viraj Dawarka

B.Sc. Computer Science

THE CHALLENGE

Strict latency constraints

Edge devices must process continuous sensor data in real-time; cloud offloading introduces unacceptable delays.

Static pipelines fail

Existing systems do not adapt inference mode to workload pressure, causing deadline violations under load.

No unified approach

Frameworks treat latency, energy, and accuracy as separate objectives rather than a joint problem.

RESEARCH GAPS

GAP 1

No explicit latency-aware optimisation for real-time behaviour prediction at the edge.

GAP 2

Latency and resource constraints treated separately — no holistic cost function.

GAP 3

Decision overhead of the optimiser itself rarely modelled or bounded.

KEY CONTRIBUTIONS

1. Novel latency-aware scheduling algorithm with decomposed end-to-end latency model
2. Multi-objective cost function: (1−Acc) + λ·L with quadratic deadline penalty
3. WISDM & UCI HAR dataset integration for realistic HAR workloads
4. Real-time demonstration module + interactive analytics dashboard
5. Simulation-based evaluation — reproducible, hardware-agnostic

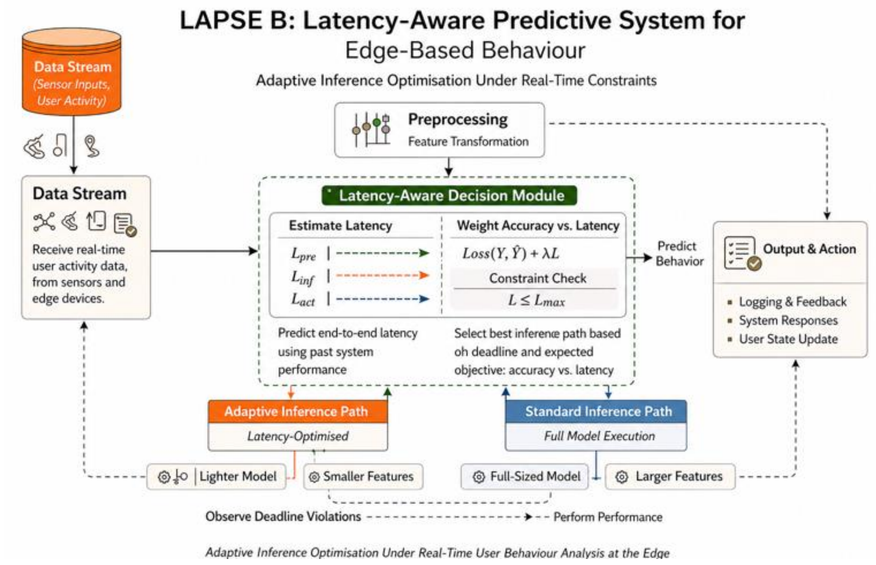
SYSTEM ARCHITECTURE & LAPSE-B DECISION ENGINE

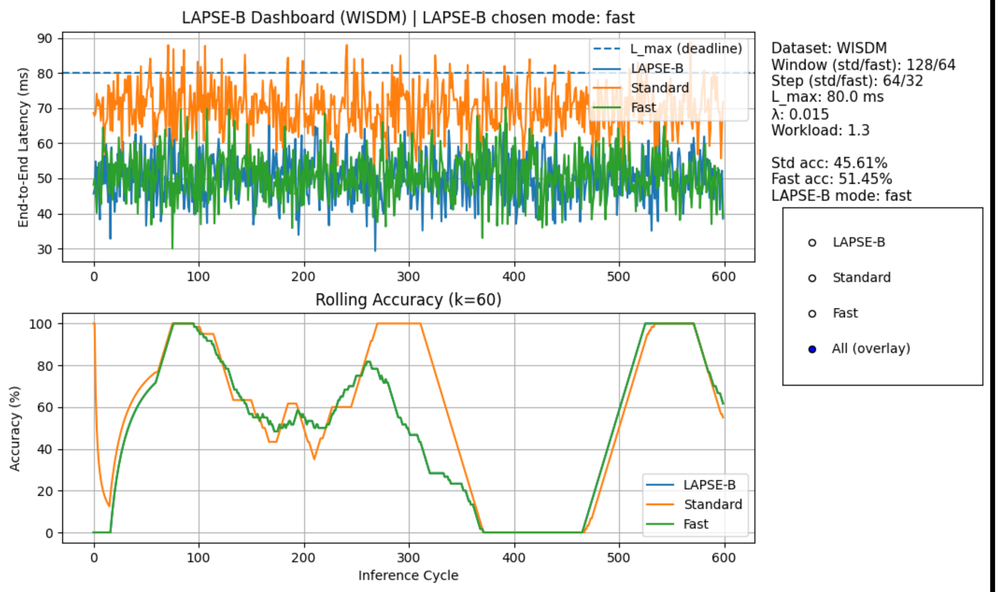
Sensors (WISDM / UCI) → Windowing (w=128 / 64) → Feature Extraction (L_{pre}) → LAPSE-B Engine (min obj + λL) → Inference + Output (L_{inf} + L_{act})

$L = L_{pre} + L_{inf} + L_{act}$ — End-to-end latency decomposition

$\min (1 - Acc) + \lambda \cdot L + Penalty$ — LAPSE-B optimisation objective

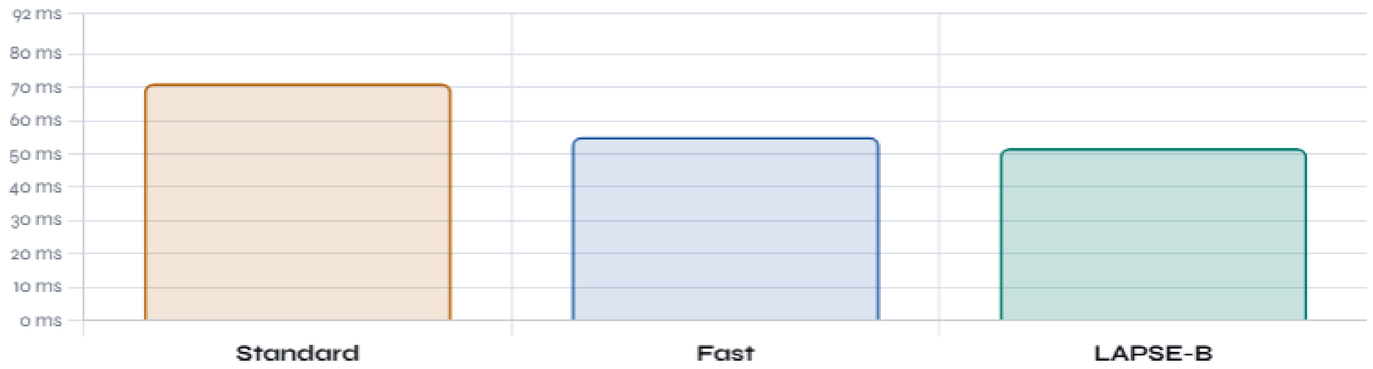
$L \le L_{max} = 80$ ms — Real-time deadline constraint





LATENCY DISTRIBUTION (MEDIUM WORKLOAD λ=1.3)

⚠ Deadline L max = 80 ms



Key result: LAPSE-B reduces deadline violations from 14.3% → 4.9% — a **66% reduction** under medium load (λ=1.3). Predictive scheduling pre-empts violations rather than reacting to them.

Pareto efficiency: LAPSE-B achieves lower latency than both baselines without proportional accuracy loss — demonstrating controlled, structured degradation rather than a naive speed-accuracy swap.

WORKLOAD ESCALATION

λ	LAPSE-B	Std	Fast
1.0	42.1 ms	65.3 ms	50.4 ms
1.3	51.8 ms	71.2 ms	55.2 ms
1.6	59.7 ms	88.4 ms	63.5 ms

SAVING vs STD (λ=1.6)

LAPSE-B saves **28.7 ms**

Std exceeds deadline **+8.4 ms**

◉ WISDM ○ UCI HAR

KEY METRICS

51.8 ms	4.9%
LAPSE-B mean latency	Deadline violations
56.4%	80 ms
Std mode accuracy	L_{max} deadline

INFERENCE MODES

Standard mode

Window	128 samples
Accuracy	56.43%
Latency	71.2 ms
Violations	14.3%

Fast mode

Window	64 samples
Accuracy	49.94%
Latency	55.2 ms
Violations	6.4%

LAPSE-B (adaptive)

Mode	Dynamic
Accuracy	~49.94%
Latency	51.8 ms
Violations	4.9%

KEY FINDINGS

Latency reduced

27% at λ=1.0, 28% at λ=1.3, 32% at λ=1.6 vs Standard

Violations halved

14.3% (Static) → 4.9% (LAPSE-B) under medium load

Pareto-efficient

Controlled accuracy degradation — not a naive speed-accuracy swap

Smooth degradation

Continuous objective avoids oscillatory mode-switching

CONCLUSION

LAPSE-B demonstrates that **explicit latency awareness** *within the inference decision process materially improves real-time performance at the edge — without catastrophic accuracy collapse.*

Future work: *real hardware deployment on TinyML / mobile platforms; energy-aware objective extension; reinforcement learning for scheduling policy.*