Timely data delivery for heterogeneous IoT applications

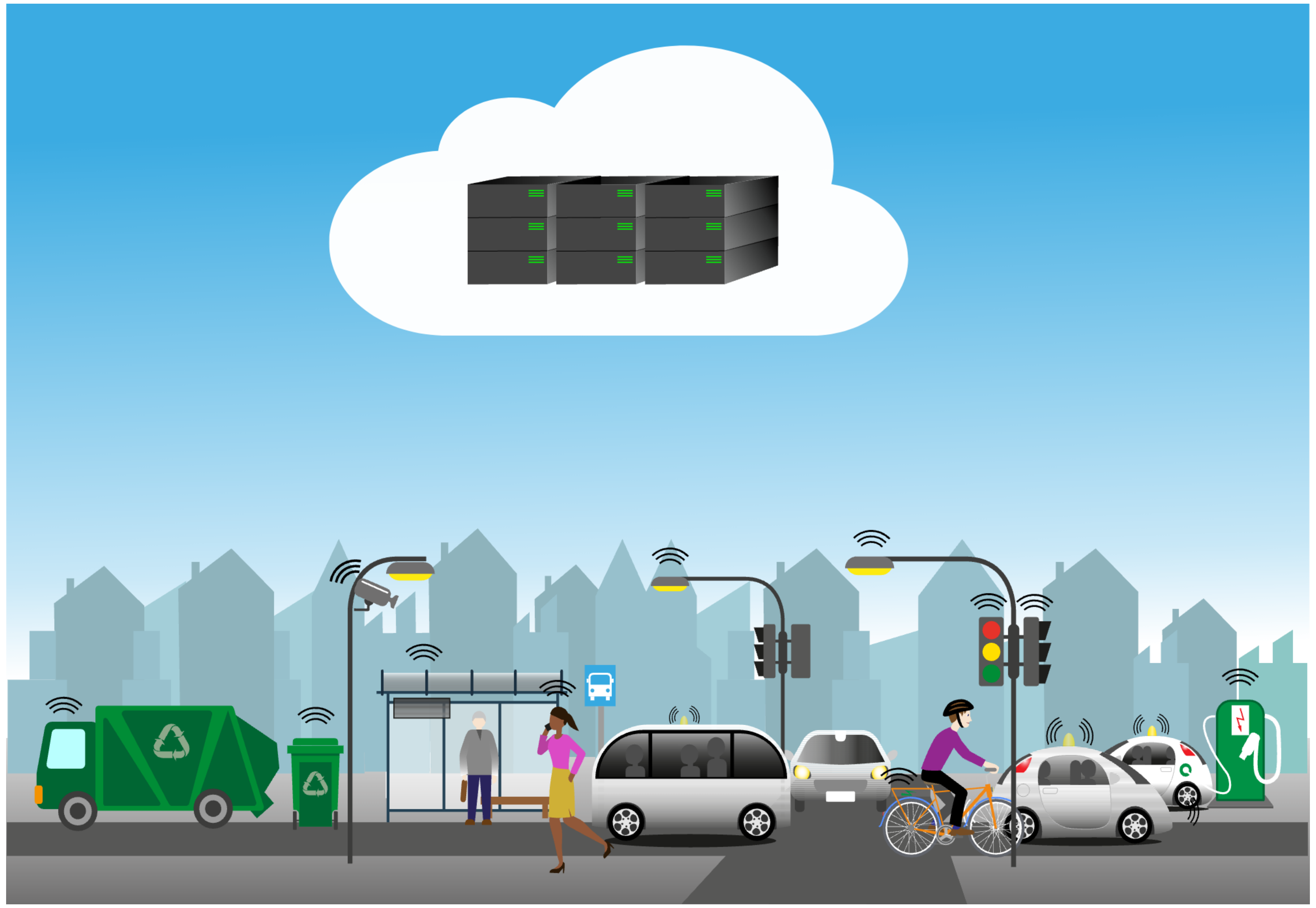
Verónica Toro-Betancur*^, Gopika Premsankar*, Lorenzo Corneo*, Mario Di Francesco*

†

* Aalto University, Finland ^ Nokia Bell Labs, Finland

May 28, 2025

Internet of Things (IoT)



- Data sent from sensors is aggregated in cloud or edge servers
- IoT applications make decisions based on sensed data

Aalto University

Timeliness of data in IoT

- Clients request data through **update requests** sent to cloud or edge server
- Sensed data change over time; data delivered to clients should be **timely**
- Challenges:
  - Diverse application types that rely on same underlying data but have different request patterns
  - Network issues with reliability and latency

Research objective

How to ensure timely delivery of updates for different IoT applications having distinct request patterns?

- How should the updates be generated such that the data received is timely?
- How to account for the **fairness** of data updates received by different clients requesting the data?

Contributions

- Describe timeliness in heterogeneous IoT scenarios
- Analytical solutions for optimal data update generation policies assuming different request patterns
- Performance evaluation of several generation policies with a real-world dataset of network latency

Related work

- Timely dissemination of updates to applications using caches and AoI-aware scheduling:

L. Corneo, C. Rohner, P. Gunningberg. "Age of information-aware scheduling for timely and scalable internet of things applications". INFOCOM 2019.

- Fairness of AoI in cache-enabled UAV network

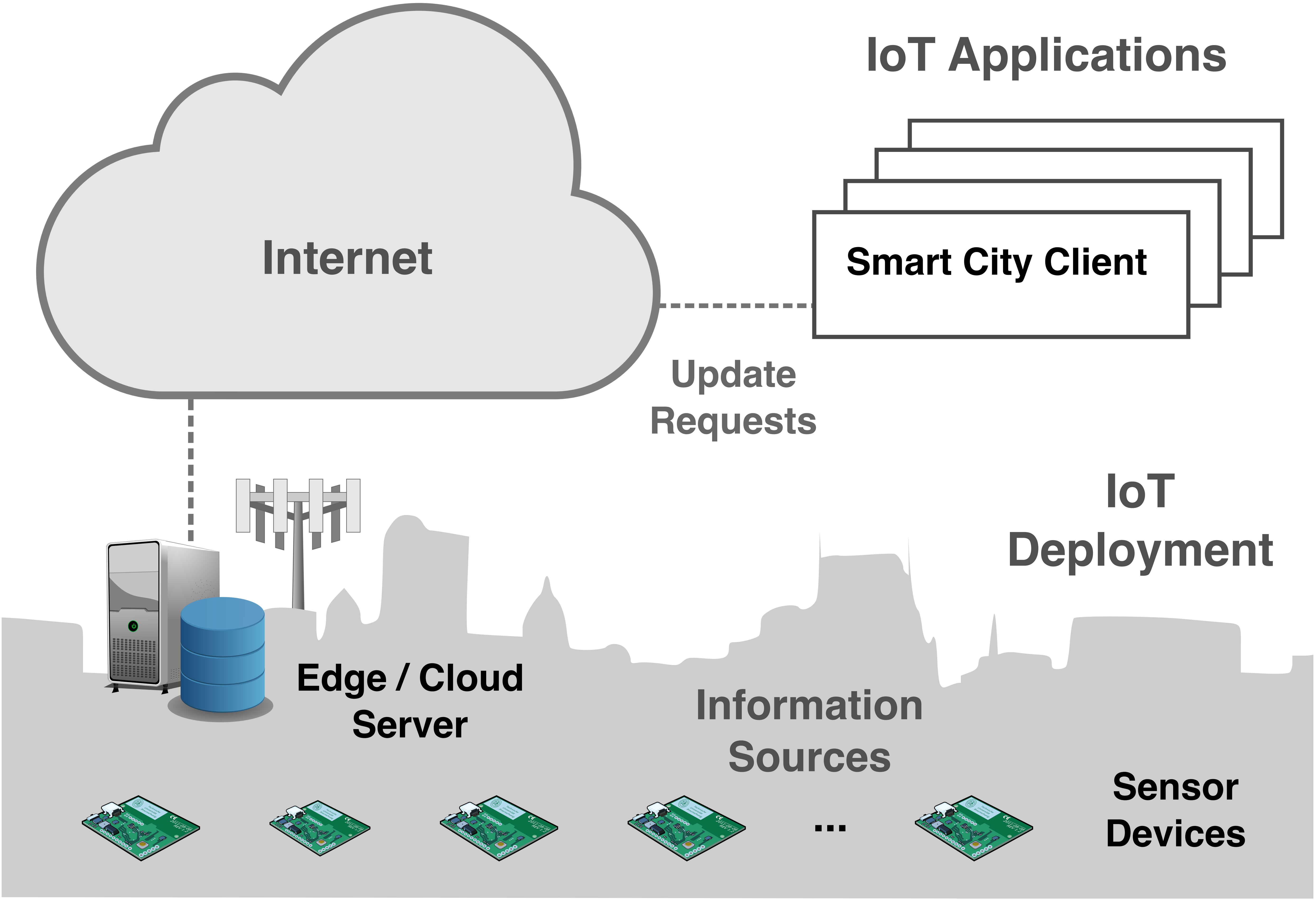
P. Yang et al. "Fresh, fair and energy-efficient content provision in a private and cache-enabled UAV network." IEEE Journal of Selected Topics in Signal Processing 2021.

Agenda

- System model
- Optimal data generation policies for different IoT scenarios
- Evaluation and results
- Conclusion and future work

System model

Scenario


IoT Applications
Internet
Smart City Client
Update
Requests
IoT
Deployment
Edge / Cloud
Server
Information
Sources
...
Sensor
Devices

Timeliness metrics

- **Update age***,* including network delay

$$\overline{\Delta}_i = \lim_{T \to +\infty} \frac{1}{T} \int_0^T \Delta_i(t)\, dt$$

- Instantaneous ***system update age*,** including packet success probability

$$\overline{\Delta} = \frac{1}{N} \sum_{i \in C} \overline{\Delta}_i \alpha_i + \tilde{\Delta}_i \left(1 - \alpha_i\right)$$

Optimization problem

Objective is to obtain update generation times for IoT applications that

- Minimizes system update age **(min-sys)**
- Maximizes update age fairness **(max-fairness)**
    - Fairness modeled using Jain's fairness index

Unconstrained optimization problem solved analytically for different request patterns

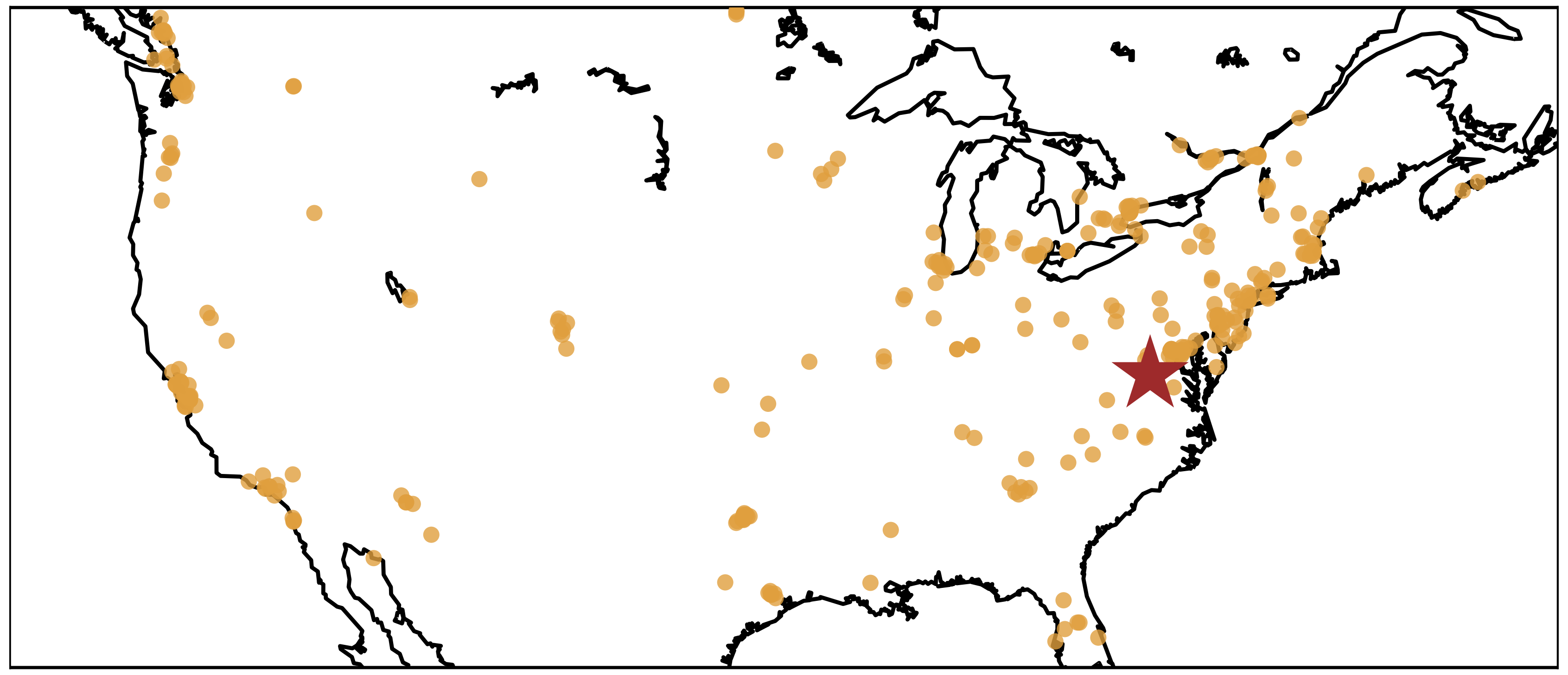
Optimal update generation time

- Three different probability distributions for client update requests' inter-arrival times: **uniform, exponential, and normal distribution**
- Solutions for update generation time for both min-sys and max-fairness obtained by numerically solving the related polynomials

Evaluation

Simulation setup

- 400 clients send requests with interarrival times from three probability distributions
  - Parameters obtained from literature for latency-critical IoT applications [Sch17]
- Network delay between clients and servers from real-world dataset [Cor21]



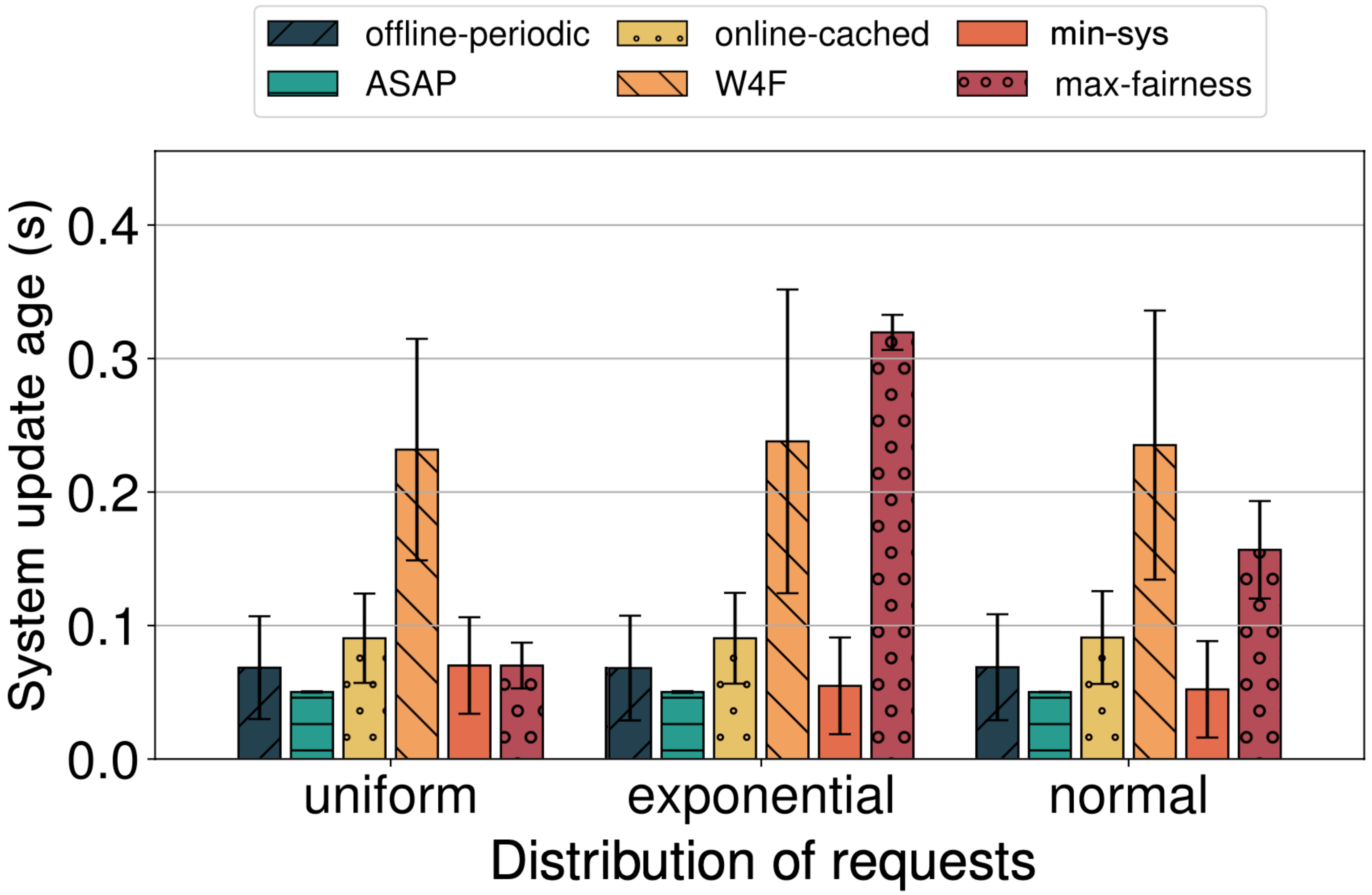
Aalto University

[Sch17] P. Schulz et al. "Latency critical IoT applications in 5G: Perspective on the design of radio interface and network architecture." *IEEE Communications Magazine* 55.2 (2017)
[Cor21] L. Corneo et al. "Surrounded by the clouds: A comprehensive cloud reachability study." *Proceedings of the Web Conference 2021*.

Evaluated policies

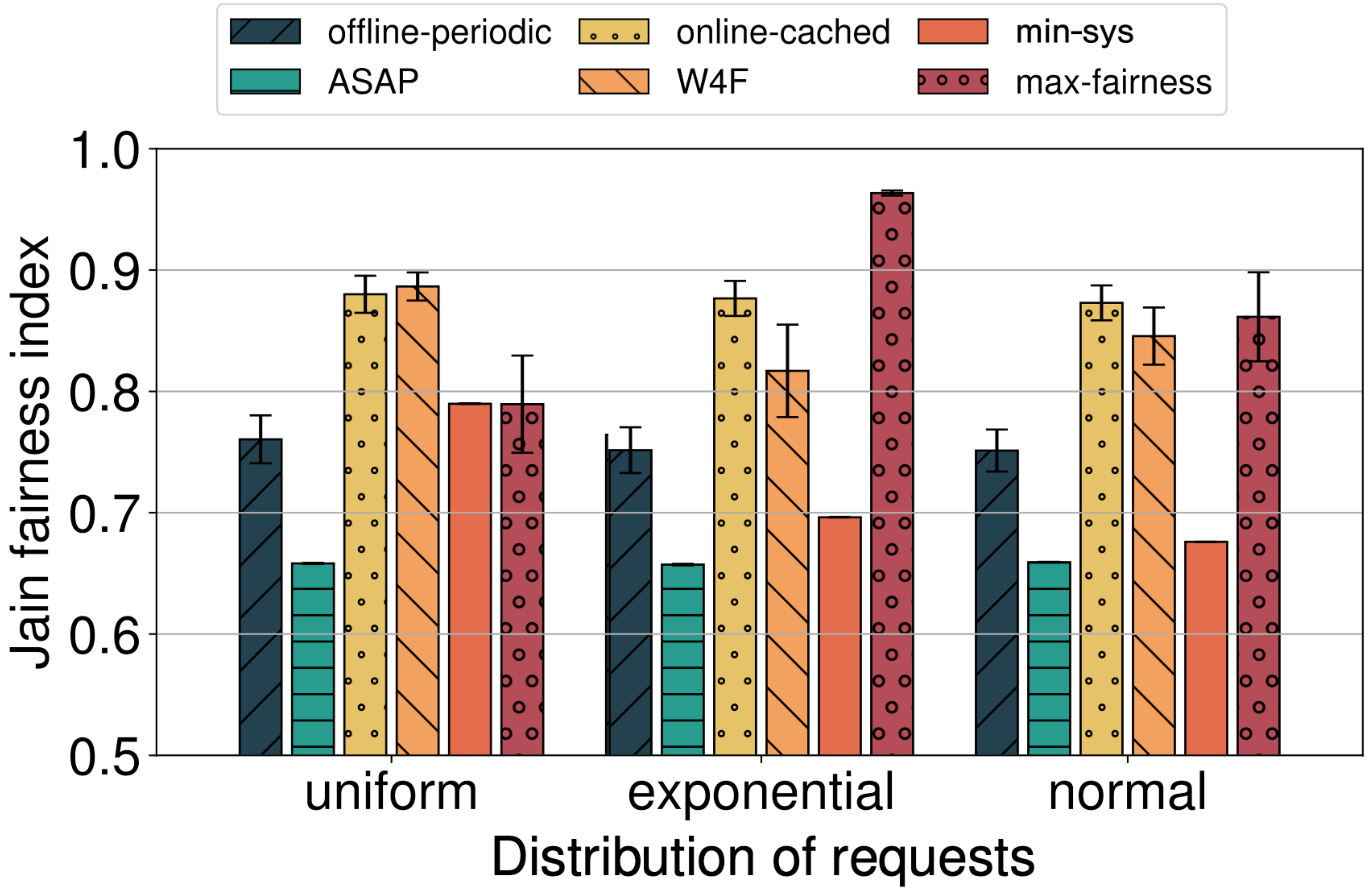
- **Min-sys**
- **Max-fairness**
- **As soon as possible (ASAP):** send update whenever data is generated
- **Wait for farthest (W4F):** send update based on client which has most network delay to the server
- **Offline-periodic:** generate updates every T seconds
- **Online-cached:** generate updates every T seconds and cache last *m* values locally, send cached or fresh updates to clients depending on network delay

Evaluation: system update age



- Lowest update age with **ASAP**
- **Min-sys** achieves 9.1% and 4% higher update age for exponential and normal distributions

Evaluation: fairness



- Lowest fairness with **ASAP,** higher fairness with **min-sys**
- **Max-fairness** achieves highest fairness for exponential distribution

Summary

Conclusion

- Analytical solutions for optimal data update generation policies assuming different request patterns
- Caching requests and serving different data to clients depending on the delay results in higher fairness and lower update age

Future work:

- Consider both fairness and update age together
- Consider more servers and more probability distributions

Thank you.

Paper:

Contact:

- veronica.toro_betancur@nokia-bell-labs.com
- gopika.premsankar@aalto.fi
- lorenzo.corneo@ericsson.com
- mario.di.francesco@aalto.fi

Aalto University

NOKIA BELL LABS

Research Council of Finland