Optimizing Edge Computing Resources Towards Greener Networks and Services

XXXV cycle of Ph.D. Course in Information Engineering
Department of Information Engineering
University of Padova

Ph.D. candidate: Giovanni Perin

Supervisor: Prof. Michele Rossi
Co-supervisor: Prof. Tomaso Erseghe

giovanni.perin.1@unipd.it

IEEE ITS Ph.D. Thesis Award
Rome, June 5, 2023

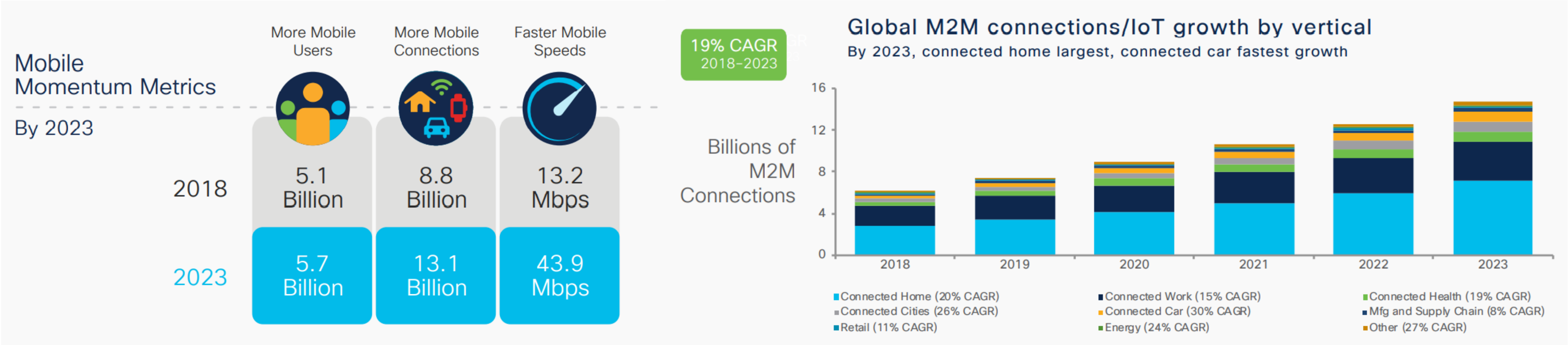
DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE
800 ANNI 1222·2022
UNIVERSITÀ DEGLI STUDI DI PADOVA
SIGNET
signet.dei.unipd.it

Introduction

- The growth of the Internet and the energy problem
- Multi-access edge computing
- Thesis objective, contributions and methods

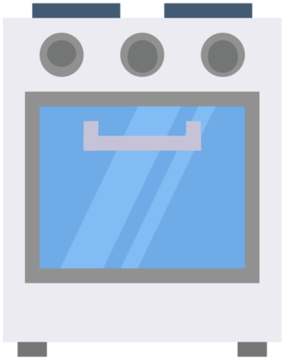
The growth of the Internet

2018		2023
50%	(users)	66%
2.4	(devices × person)	3.6
33%	M2M	50%



Source: Cisco Annual Internet Report, 2018–2023, February 2020

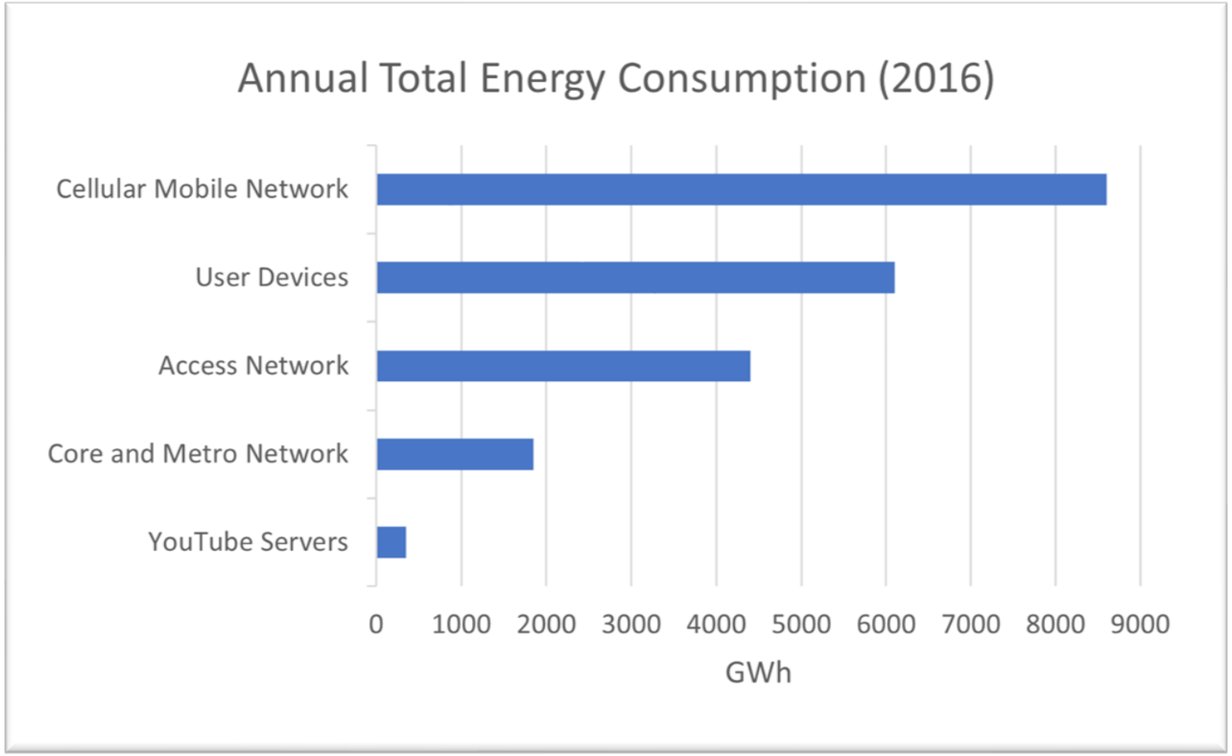
The energy problem



10 mins HD streaming ↔ 5 mins oven at 2 kW

Source: The Shift Project. 2019. Lean ICT – Towards Digital Sobriety. White paper report.

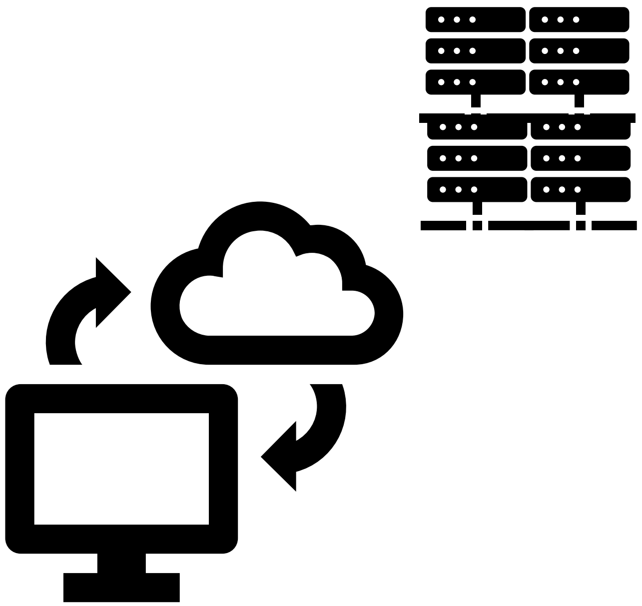
Video content: 80% of Internet traffic share (2020)
Expected to still be 80% in 2028 (Ericsson)



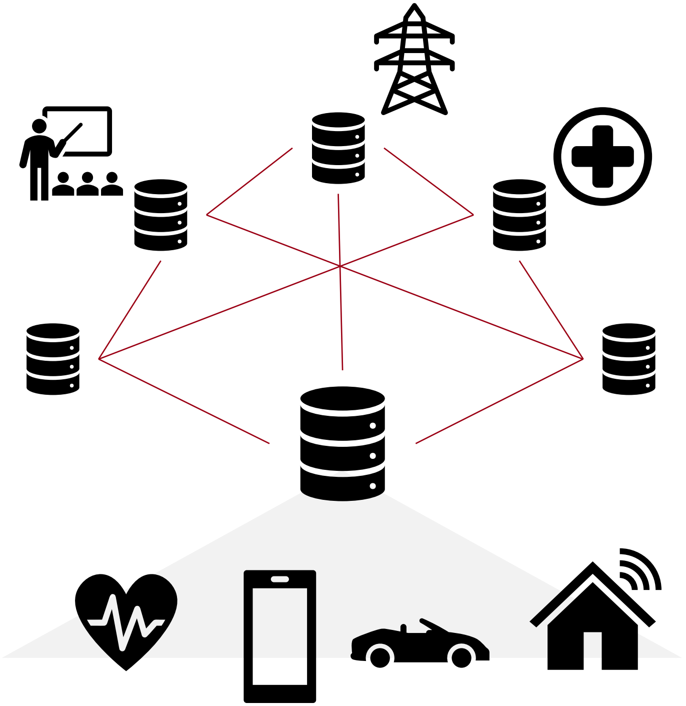
Source: C. Preist, D. Schien, and P. Shabajee. 2019. Evaluating sustainable interaction design of digital services: the case of YouTube. 2019 CHI Conference on Human Factors in Computing Systems.

Multi-access Edge Computing

Cloud computing



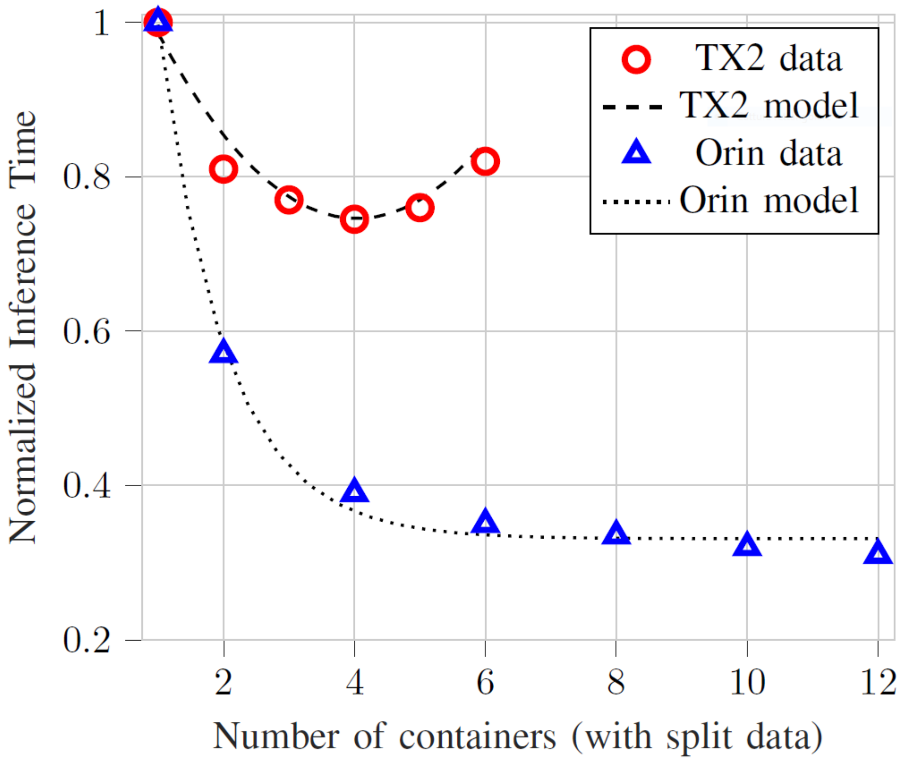
Edge computing



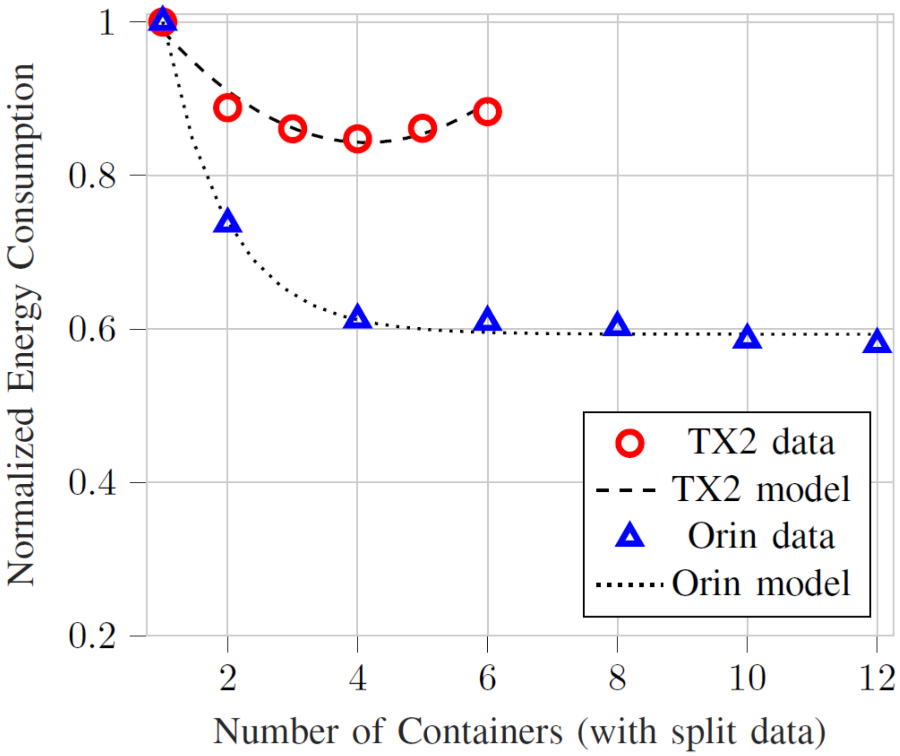
Around the city
Lower latency
Lower energy

Managing workload in edge devices

Execution time



Energy consumption

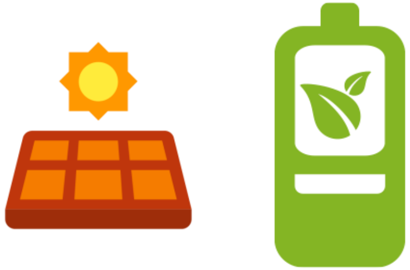


A. Khoshsirat, G. Perin, and M. Rossi, «Divide and Save: Splitting Workload Among Containers in an Edge Device to Save Energy and Time,» *IEEE ICC 2023 Second International Workshop on Green and Sustainable Networking (GreenNet)*, May 2023.

Thesis objective and contributions

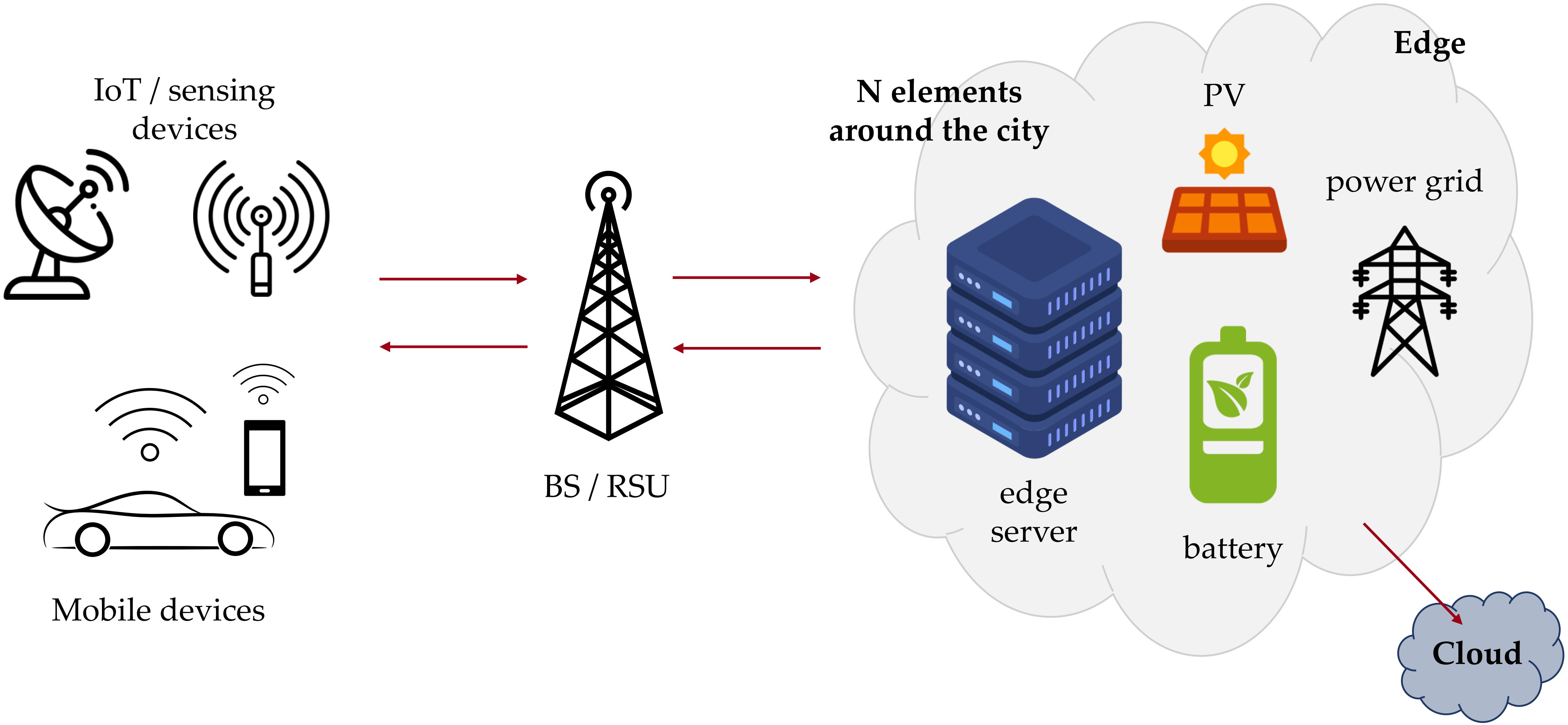
Design of **distributed network controllers** to manage the MEC platform

- MEC agent with
  - computing equipment to serve terminals
  - wireless comm. to the terminals + wired comm. among servers
  - power grid + **EH devices** (such as PVs)
- **Fully decentralized** and **model-based** controllers that
  - plan the local execution of jobs
  - decide the (portion of) jobs to offload
  - trade electrical energy with the power grid
- **Load balancing** vs **server consolidation**
- Study of **communication vs computation tradeoff**
- Objective: minimize the **carbon footprint**



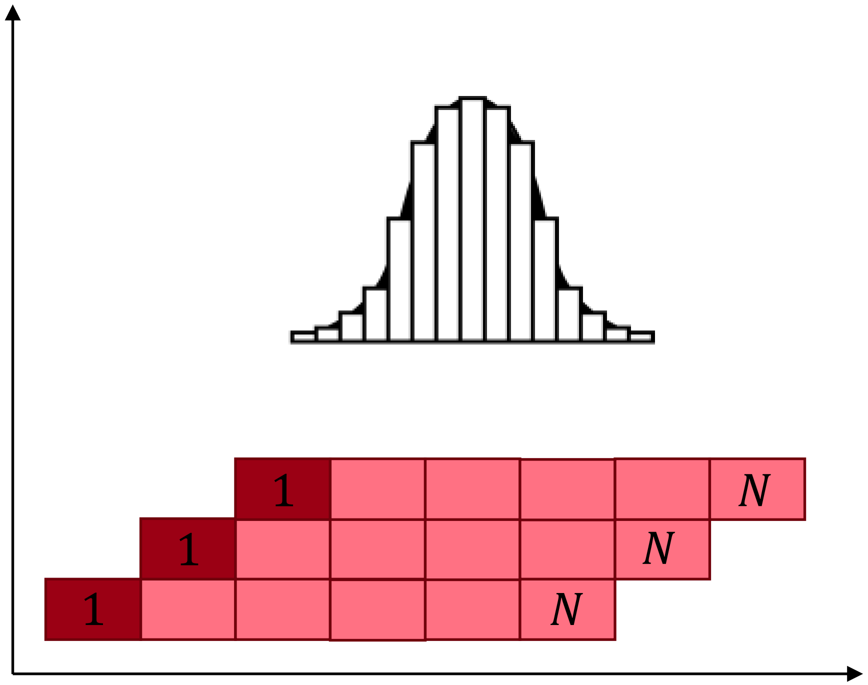
Source: R. S. Sutton, and A. G. Barto. 2018. Reinforcement learning: An introduction. MIT press.

Considered system



Model predictive control

- **Adaptive predictive controller** inspired by optimal control
- Control computed on the whole **predictive window *N***
- First control is applied, and procedure is repeated, sliding the window (***receding horizon***)
- The controller self-adapts to exogenous processes (e.g., job and energy arrivals)

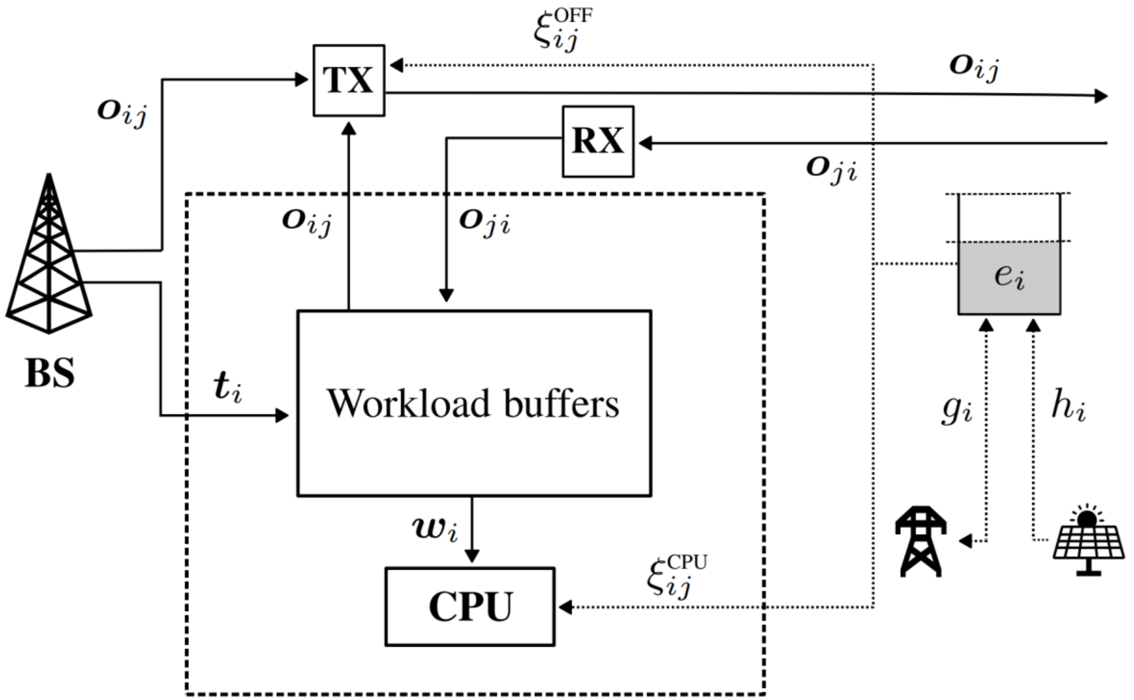


Offloading of tasks in edge computing (A)

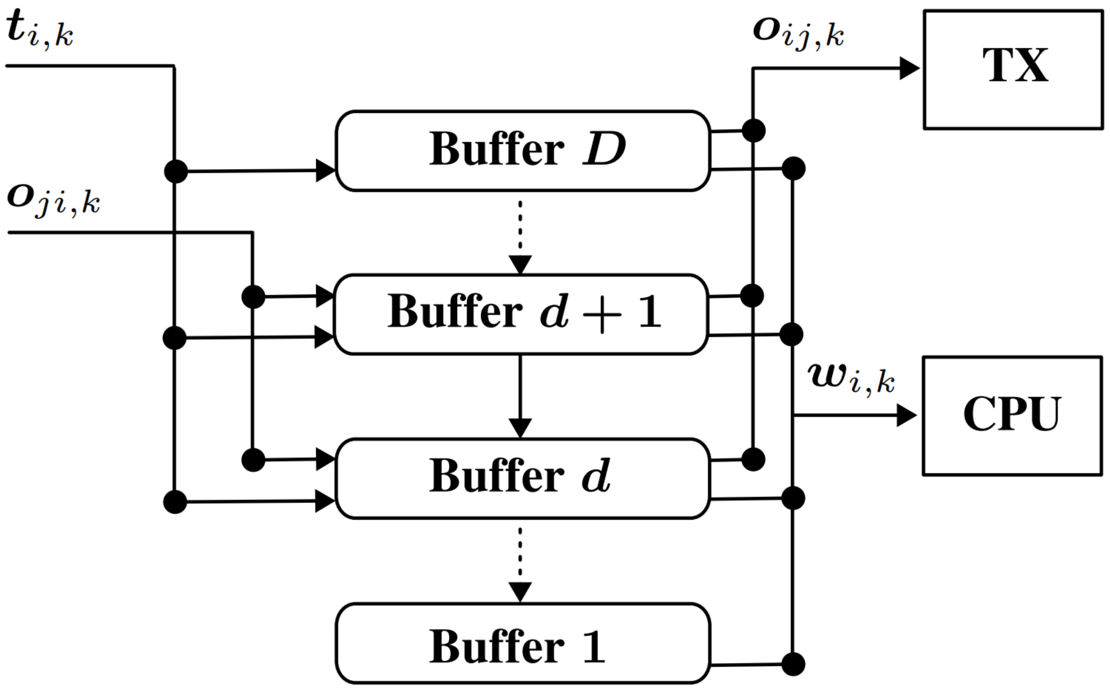
- Managing renewables and trade with the grid
- Load balancing vs consolidation

System model

Access node



Workload buffers detail



G. Perin, M. Berno, T. Erseghe, and M. Rossi, "Towards Sustainable Edge Computing Through Renewable Energy Resources and Online, Distributed and Predictive Scheduling," in *IEEE Transactions on Network and Service Management*, vol. 19, no. 1, pp. 306-321, March 2022, doi: 10.1109/TNSM.2021.3112796.

Evolution dynamics

$$s^{d-1}_{i,k+1} = s^{d}_{i,k} + \underbrace{t^{d}_{i,k}}_{\text{locally generated}} - \overbrace{w^{d}_{i,k}}^{\text{locally executed}} + \underbrace{\sum_{j \in \mathcal{N}_i} o^{d}_{ji,k}}_{\text{offloaded here}} - \overbrace{\sum_{j \in \mathcal{N}_i} o^{d}_{ij,k}}^{\text{offloaded elsewhere}}$$

- Workload buffer with deadline d
  - jobs arriving from the coverage area and from other BSs enter the local system
  - jobs processed and offloaded to other BSs exit the local system

$$e_{i,k+1} = \delta^{\mathrm{E}}_i e_{i,k} + h_{i,k} + g_{i,k} - \sum_{j \in \mathcal{N}_i, d \in \mathcal{D}} \xi^{\mathrm{OFF}}_{ij} o^{d}_{ij,k} - \sum_{d \in \mathcal{D}} \xi^{\mathrm{CPU}}_{i} w^{d}_{i,k}$$

- Battery of each server
  - natural capacity decay δ
  - harvested energy h
  - energy traded with grid g
  - transmission and processing consumption

Cost functions

Load balancing implicitly promoted by L2-norm on processing (and state)

buffer, energy trade, transmission, execution

$$J_1(\boldsymbol{s}, g, \boldsymbol{o}, \boldsymbol{w}) = \sum_{i=1}^{N}\sum_{k=1}^{K} \underbrace{\boldsymbol{s}_{i,k}^T Q_i \boldsymbol{s}_{i,k}}_{\text{buffer}} + \underbrace{\xi(g) g_{i,k}}_{\text{energy trade}} + \underbrace{\boldsymbol{o}_{i,k}^T R_{i,o} \boldsymbol{o}_{i,k} + \boldsymbol{r}_{i,o}^T \boldsymbol{o}_{i,k}}_{\text{transmission}} + \underbrace{\boldsymbol{w}_{i,k}^T R_{i,w} \boldsymbol{w}_{i,k}}_{\text{execution}}$$

Logarithm is superlinear in proximity of the zero, it induces **consolidation**

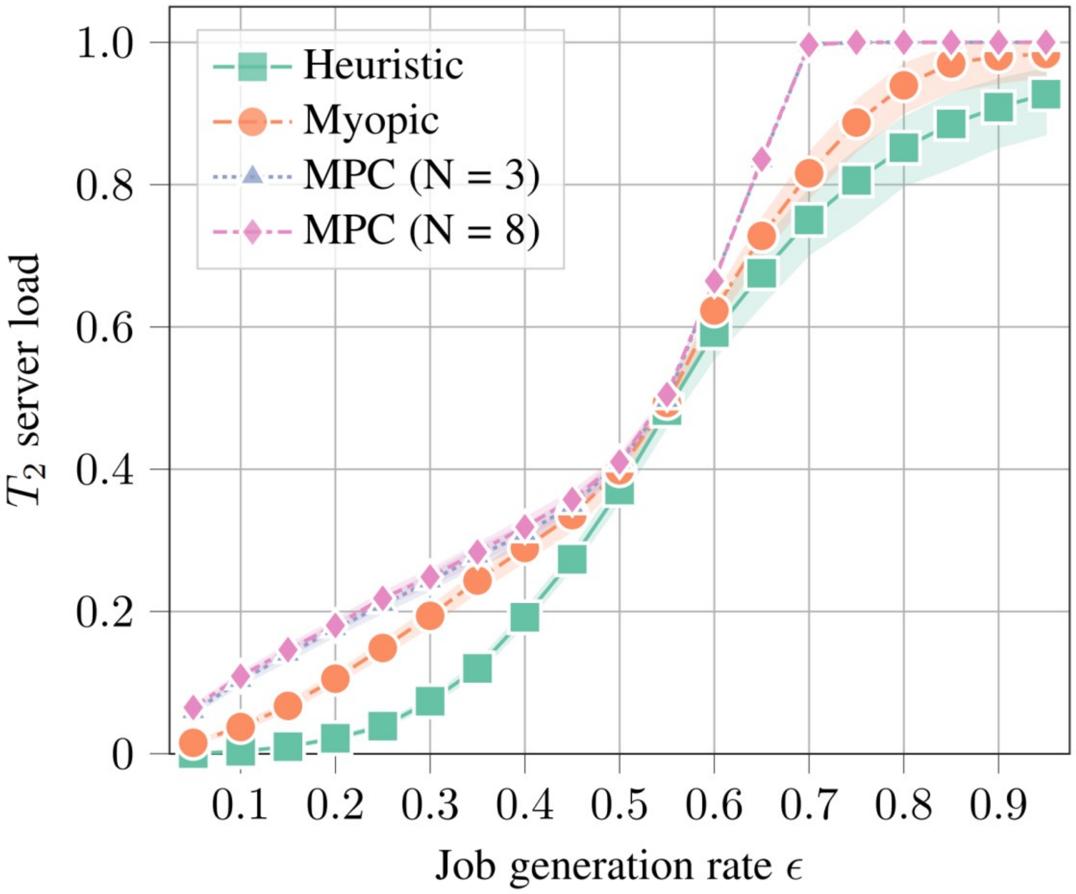
$$J_2(\boldsymbol{s}, g, \boldsymbol{o}, \boldsymbol{w}) = \sum_{i=1}^{N}\sum_{k=1}^{K} \boldsymbol{q}_i^T \boldsymbol{s}_{i,k} + \xi(g) g_{i,k} + \boldsymbol{r}_{i,o}^T \boldsymbol{o}_{i,k} + \log\left(\boldsymbol{r}_{i,w}^T \boldsymbol{w}_{i,k}\right)$$

$$\text{with} \quad \xi(g) = \begin{cases} \xi_1 & \text{if } g > 0 \\ -\xi_2 & \text{otherwise} \end{cases}$$

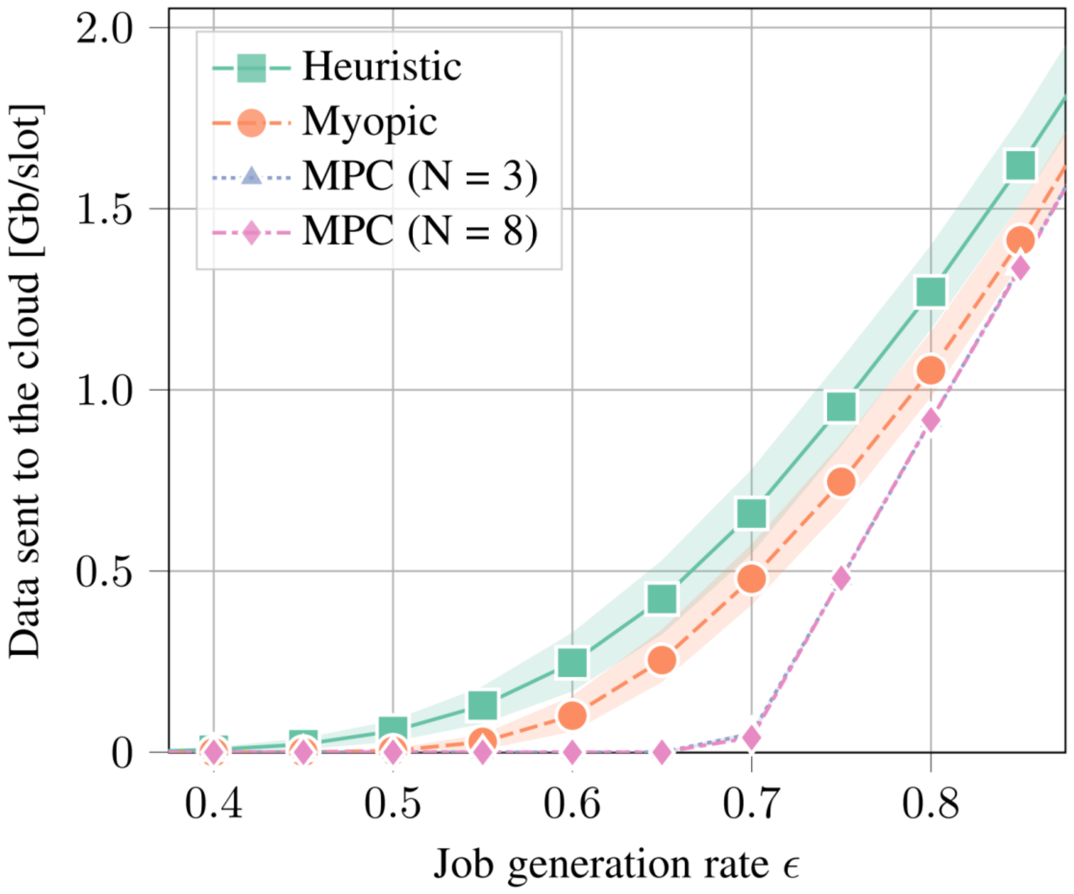
Presence of constraints: execution, deadline, transmission, battery

Executing jobs inside the edge

T2 servers load factor

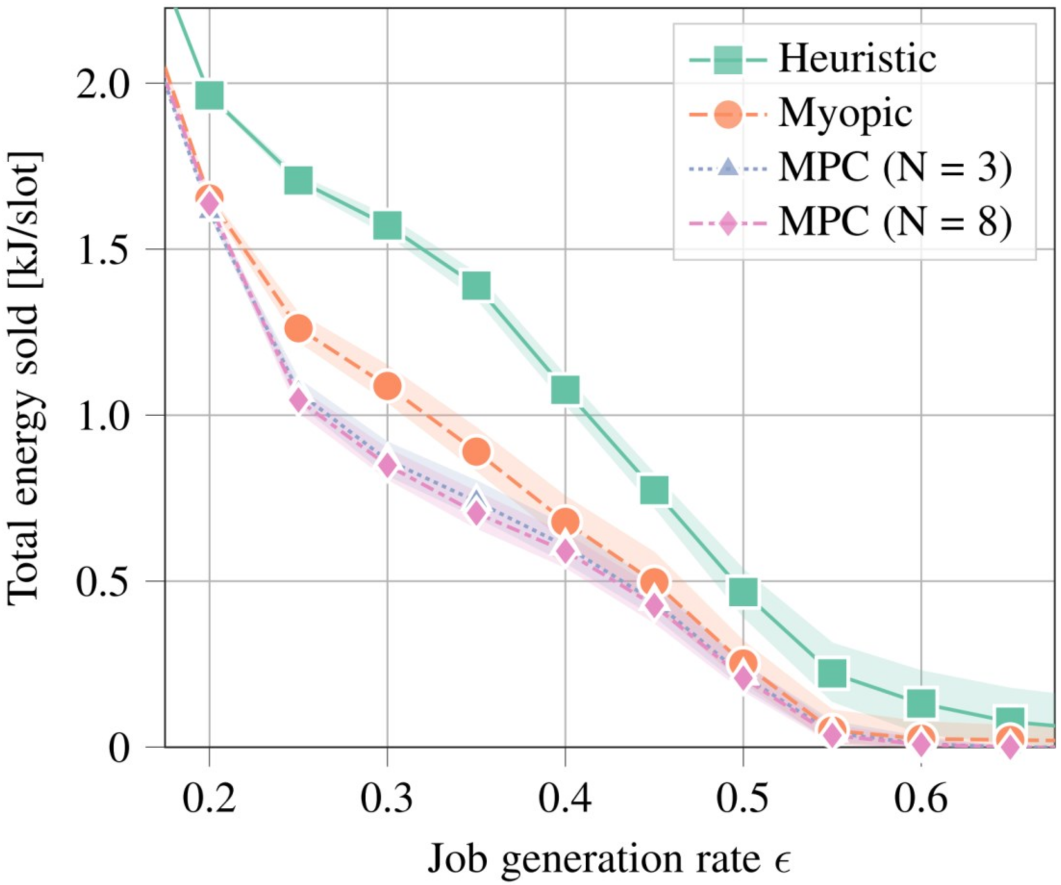


Data sent to the cloud

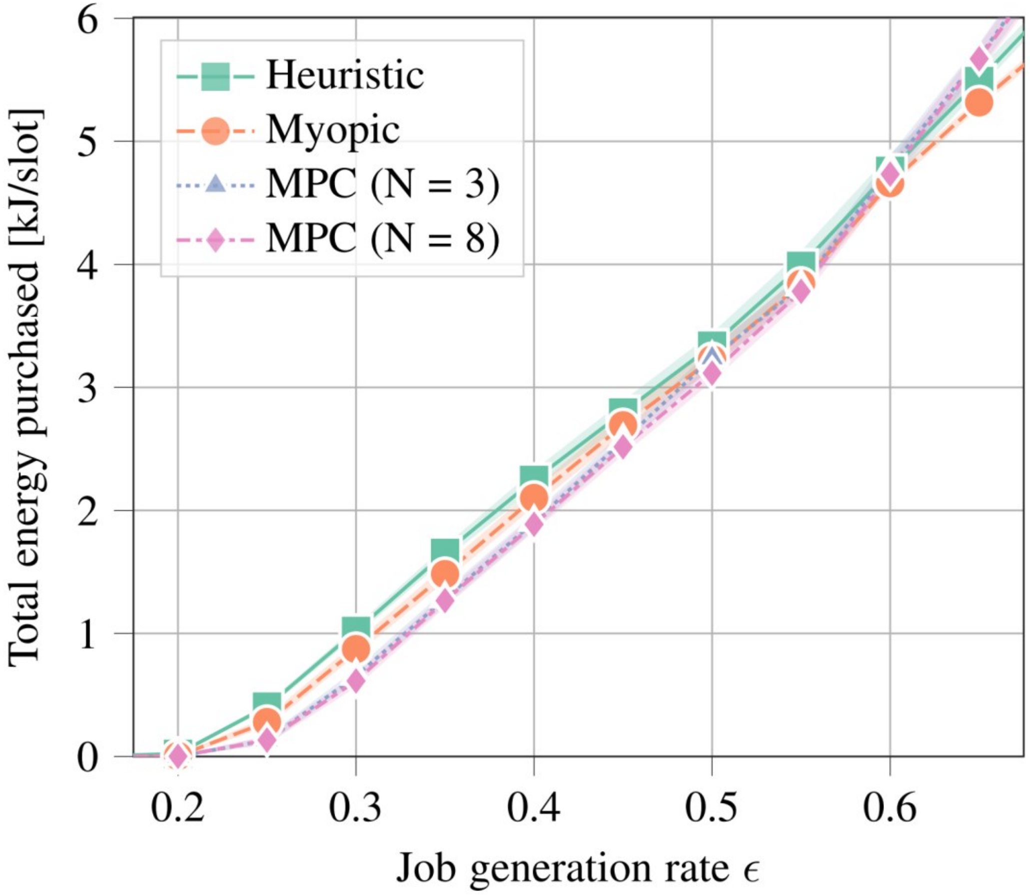


Energy traded with the power grid

Sold

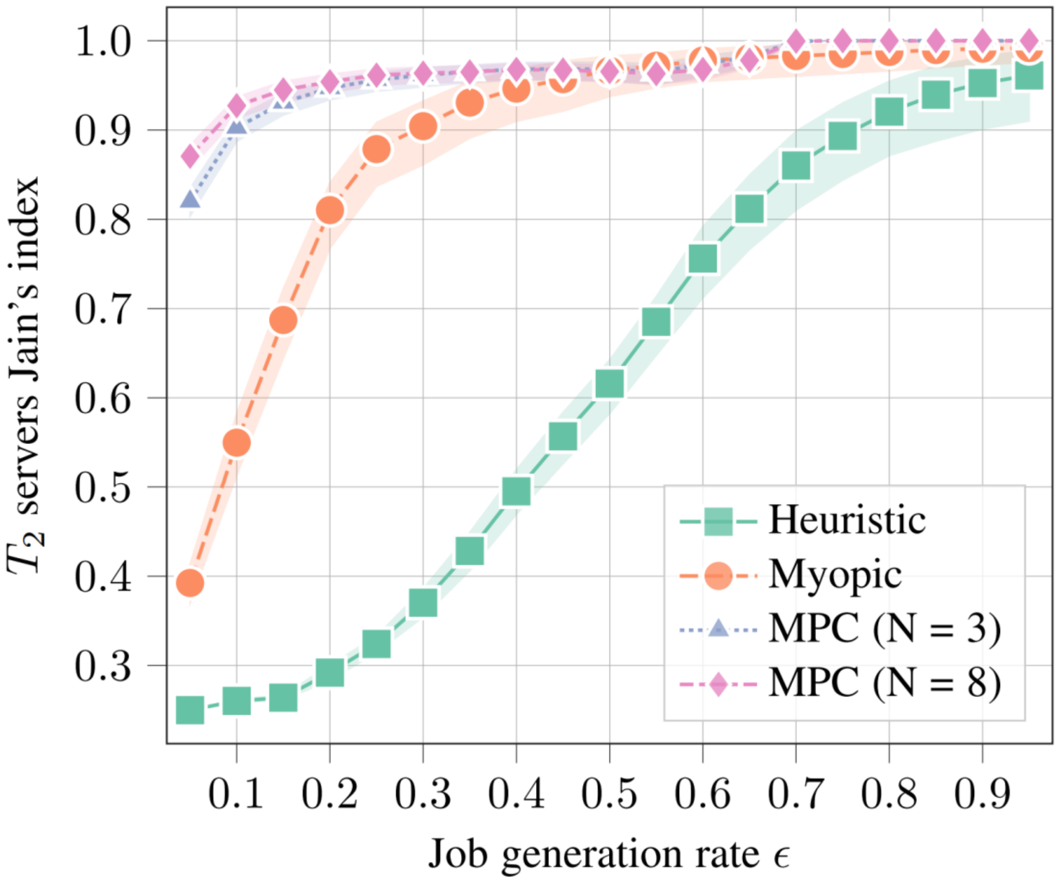


Purchased

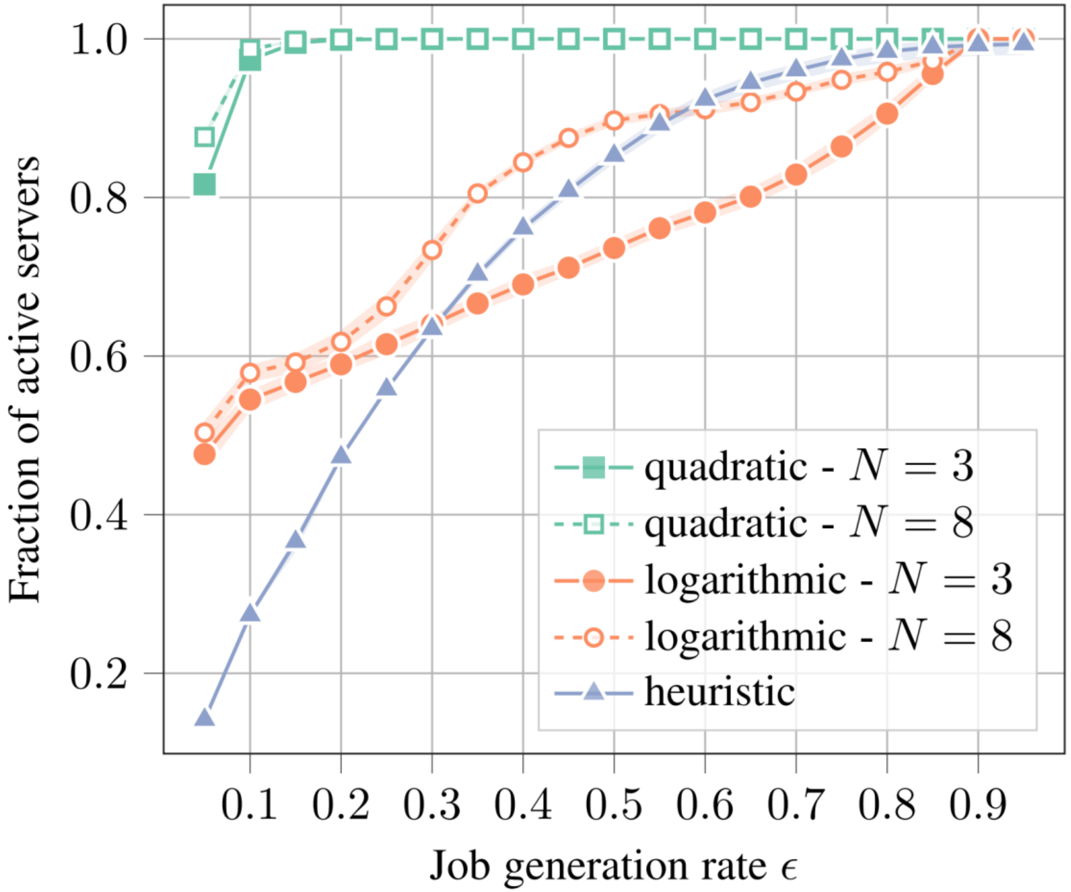


Load balancing vs consolidation

Jain's fairness index (load)



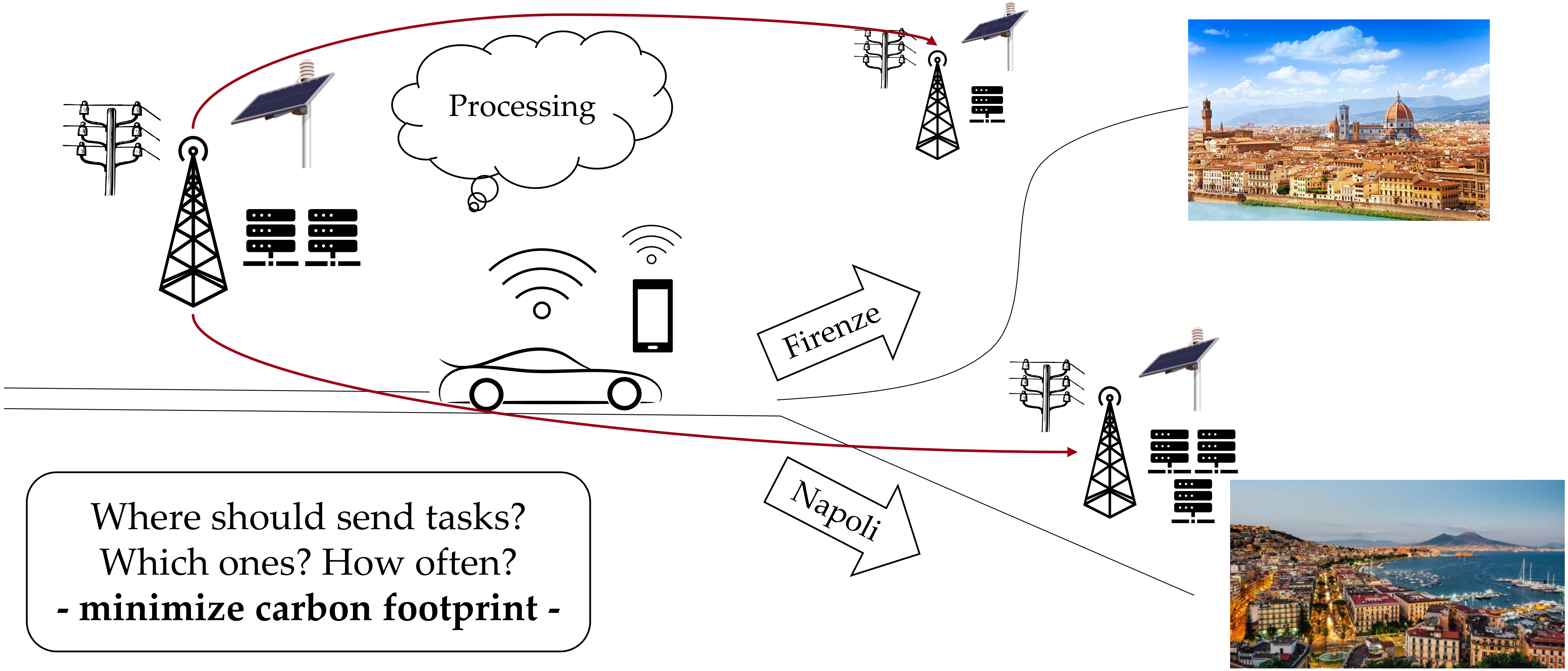
Fraction of active servers



EASE: job migration for vehicular networks (B)

- A different pipeline
- The problem of following users

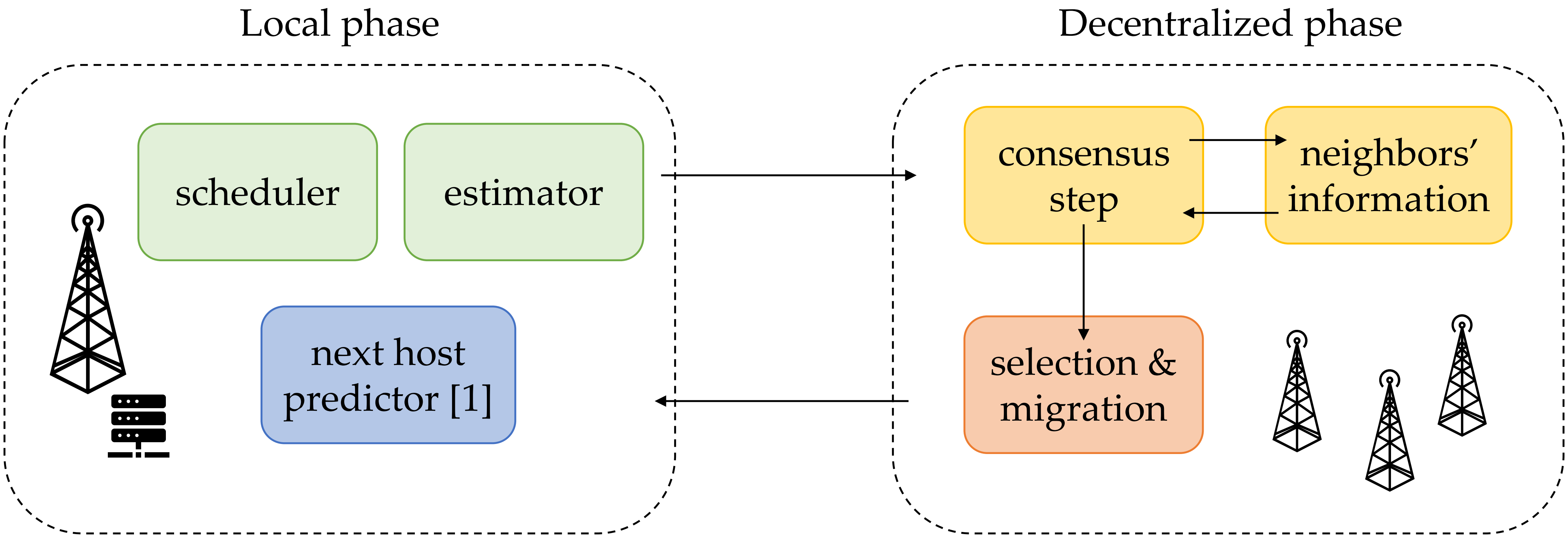
The problem



Where should send tasks?
Which ones? How often?
- minimize carbon footprint -

Working pipeline

[1] I. Labriji et al., "Mobility Aware and Dynamic Migration of MEC Services for the Internet of Vehicles," in IEEE Transactions on Network and Service Management, vol. 18, no. 1, pp. 570-584, March 2021, doi: 10.1109/TNSM.2021.3052808.



G. Perin, F. Meneghello, R. Carli, L. Schenato, and M. Rossi, "EASE: Energy-Aware Job Scheduling for Vehicular Edge Networks With Renewable Energy Resources," in *IEEE Transactions on Green Communications and Networking*, vol. 7, no. 1, pp. 339-353, March 2023, doi: 10.1109/TGCN.2022.3199171.

Scheduler + estimator – problem definition

Job k description

$$
\begin{cases}
\text{Intensity: } I_k(t+\tau) = I_k(t) - w_k(t) \\
\text{Deadline: } D_k(t+\tau) = D_k(t) - \tau \\
\text{Size: } S_k(t) = \frac{S_k(0)}{I_k(0)} I_k(t)
\end{cases}
$$

$w_k(t)$: Amount of workload of job k executed at time t

$$
\min_{\boldsymbol{w},\boldsymbol{\delta}} \sum_{t=0}^{T-1} \left(\gamma \sum_{k=1}^{K} \left(\frac{I_k(t)}{D_k(t)} \right)^2 + \max\{\boldsymbol{q}^T \boldsymbol{w} - h(t), 0\}^2 + c\boldsymbol{\delta} \right)
$$

$\boldsymbol{\delta}$: Penalty for soft constraints (memory + execution)

$h(t)$: Harvested energy

$\boldsymbol{q}$: Energy cost

State cost: proportional to residual intensity
inversely proportional to residual deadline

Control cost: zero if only renewables are used, linearly growing with slope $\boldsymbol{q}$ otherwise

Decentralized step

BS tracking

energy costs

$$\min_{\boldsymbol{x}} \sum_{i\in\mathcal{N}} \left(\frac{\rho}{2}\|\boldsymbol{x}_i - \boldsymbol{b}_i\|^2 + \max\left(\boldsymbol{q}_i^T\boldsymbol{x}_i - \hat{H}_i, 0\right)\right)$$

offloading variable

predicted
migration location

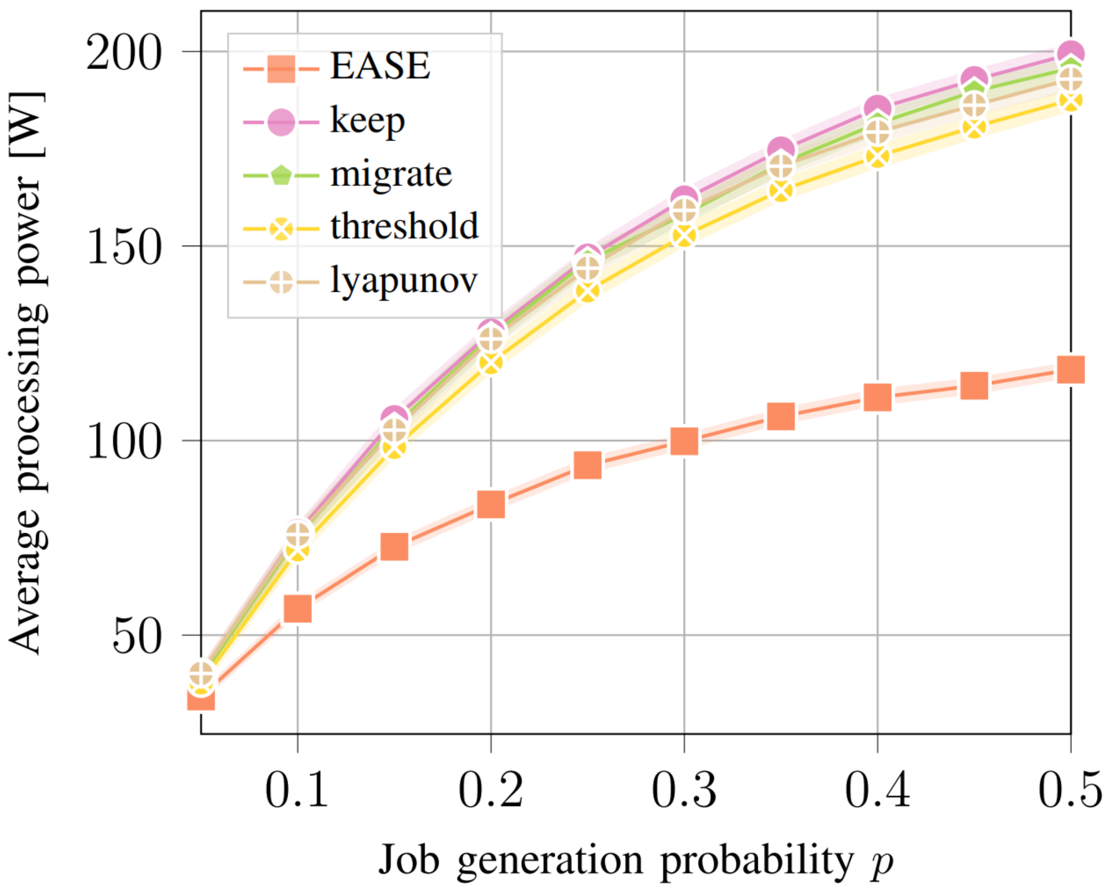
estimated available
harvested energy

Additional constraints on average power availability + consensus

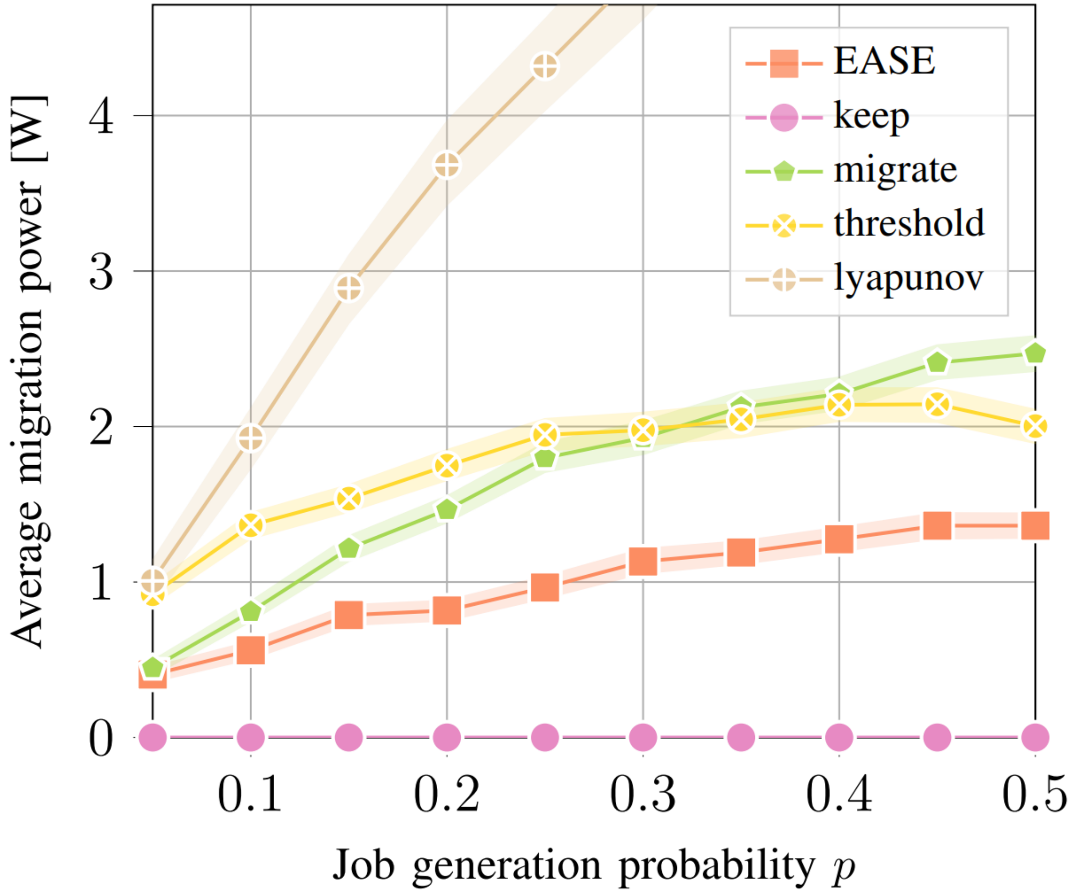
The problem can be solved through iterations of **closed-form** solutions with three possible cases, to handle the non-differentiability of the max operator

CPU and TX consumption

Processing



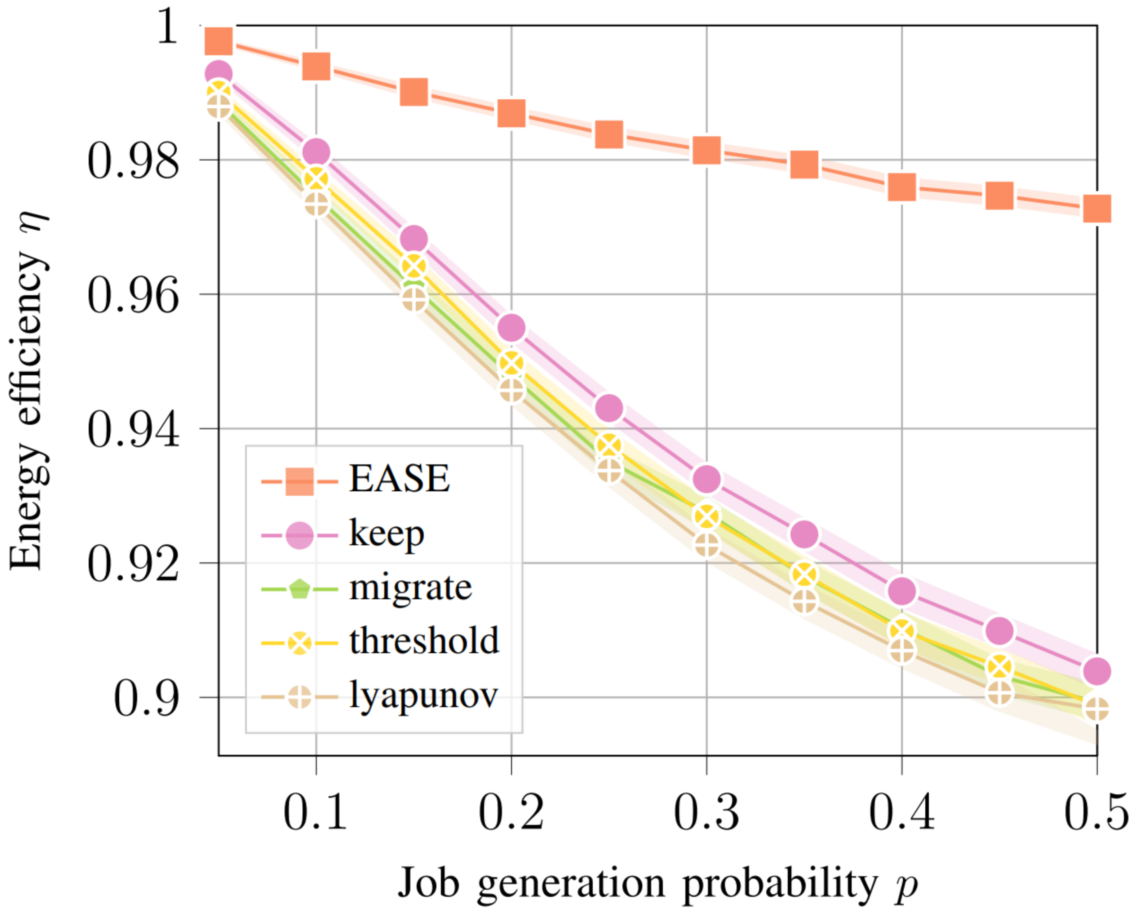
Transmission



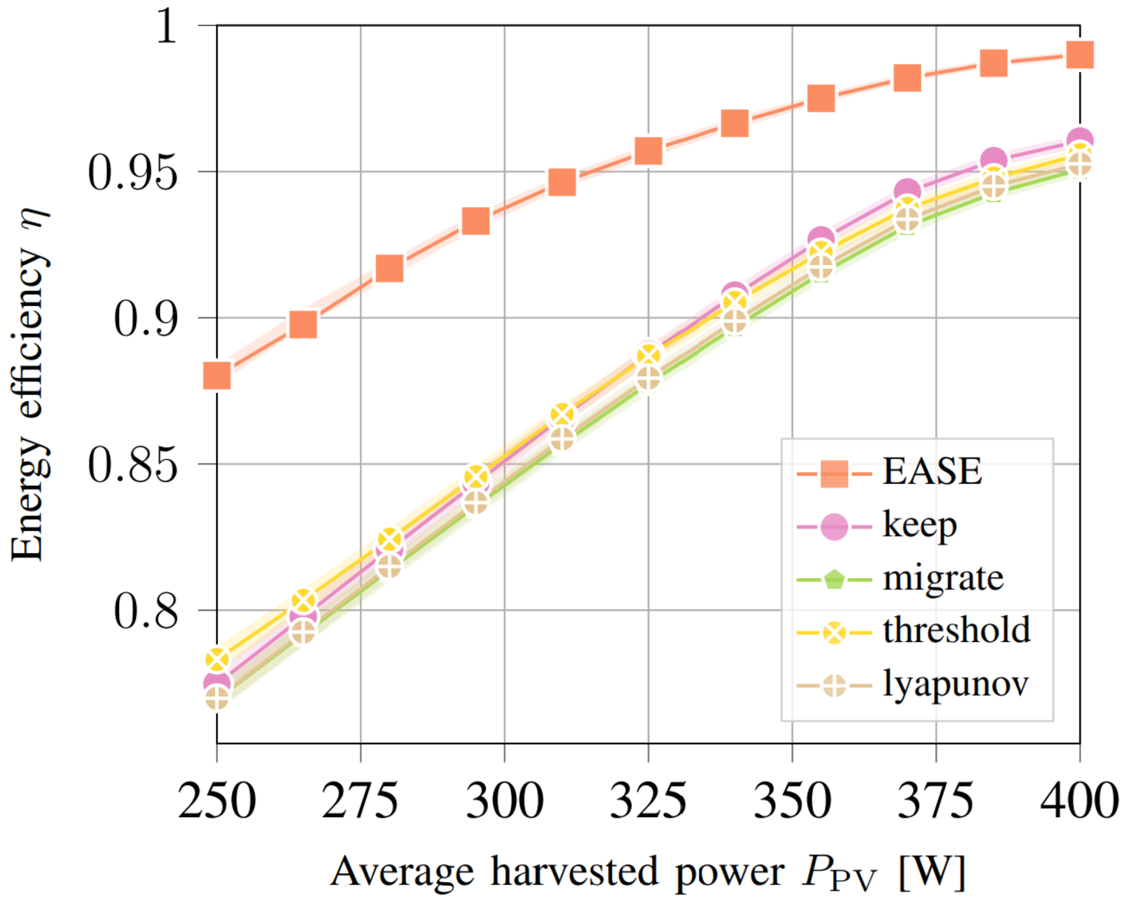
Energy efficiency

$$\eta = \frac{E_g}{E_{tot}}$$

vs load



vs available power



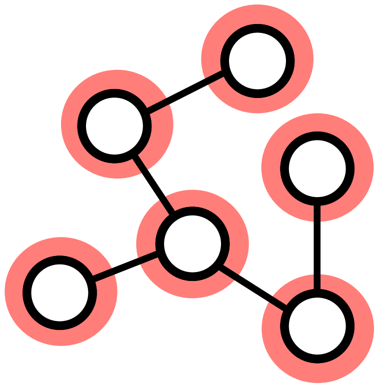
Comparison of the distributed solutions

- Intuition of message passing
- Convergence results

Distributed solution

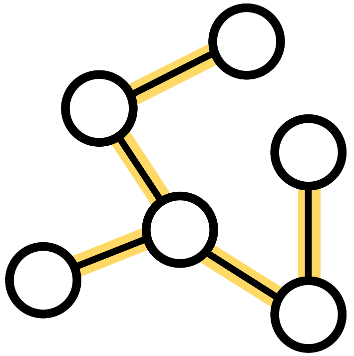
- The global cost function is a sum of separable local functions
- Minimizing the global cost can be done in a distributed way via ***message passing*** (with neighbors only) as a ***consensus problem***
- Nodes must agree on the value of the exchanged workload

Local problem



Local step
Every node solves a local sub-problem

Info broadcasting



Distributed step
Neighbors exchange a portion of the local solutions

Communications ~ 10^1
Constrained QP at every round

Communication burden (A)

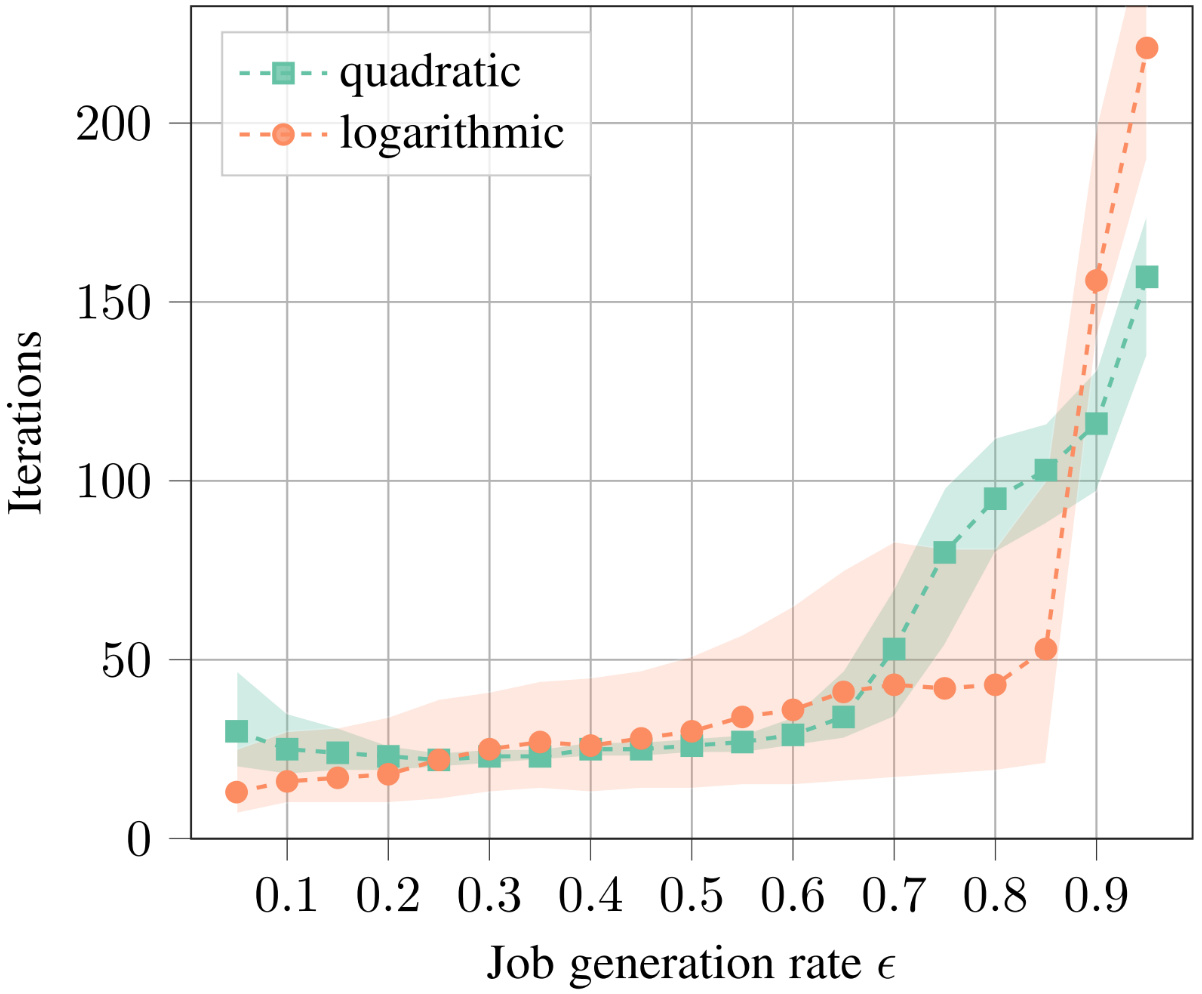
Main working region ($\epsilon < 0.7$) - both cost functions require < 50 iterations

Convex cost ~ 25 iterations with small variance

Variance is larger with log

For high ϵ iterations required by log explodes: the problem is ill-posed

Douglas-Rachford splitting with
Anderson acceleration of type II



Communications ~ 10^2
Closed form update at every round

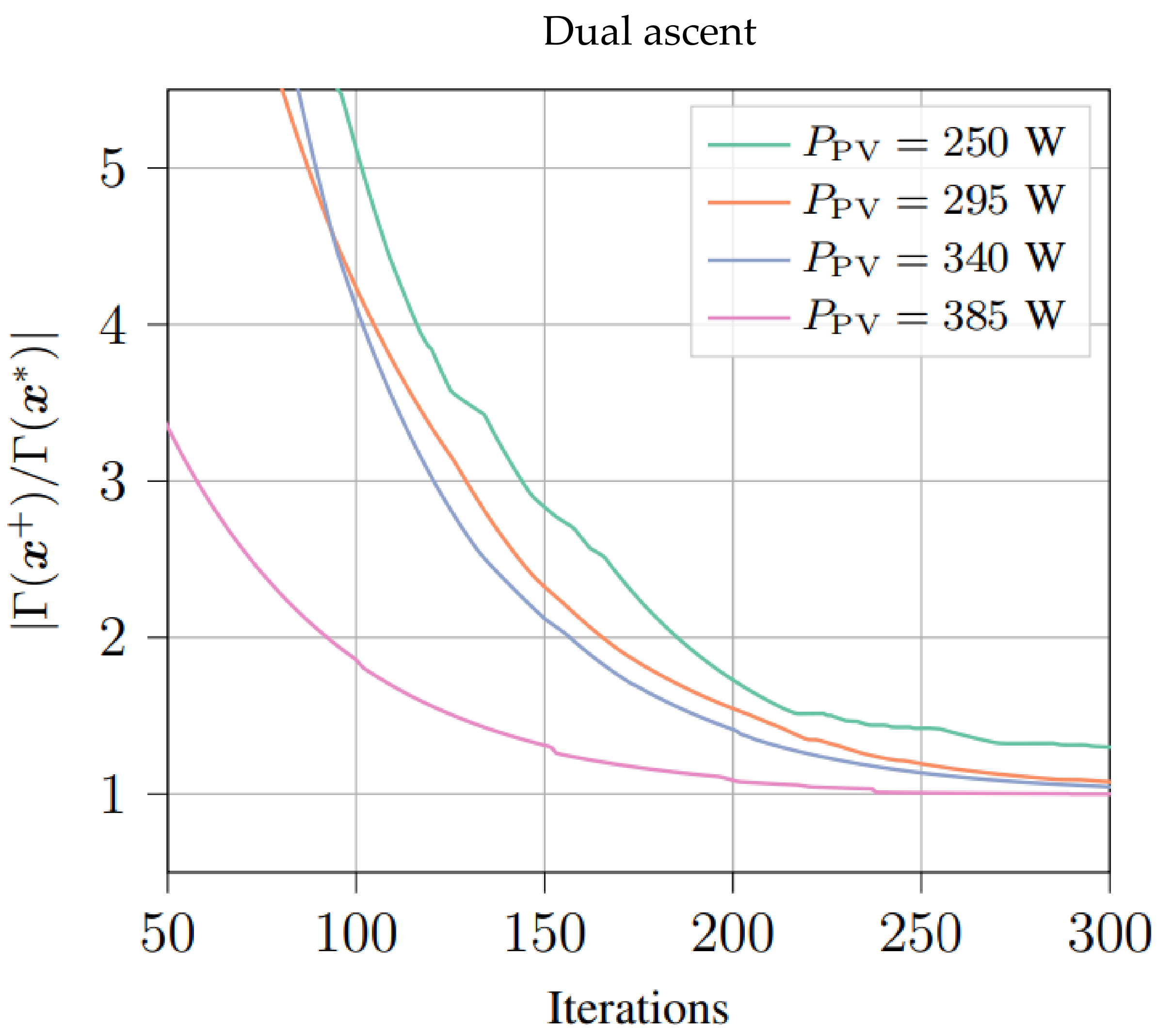
Communication burden (B)

Converg. relatively slow (90th perc.)

Overhead – 2 messages (e.g., 4 bytes) per iteration among neighbors

Depends on the «complexity» of the problem: constraint activation

Rounding step afterwards: when can we stop for a fair solution? – To be investigated



Take home messages and current/future work

- MEC is the enabler of real-time applications
  - it can also help reducing energy consumption of networks
- Managing a distributed architecture with growing connections is challenging
- Dual methods are candidates for **model-based distributed controllers**
- While **optimizing renewables** != minimizing energy, the designed controllers can **reduce the energy footprint**, obtaining **carbon neutrality** in a vast range of network conditions
- When **mobility** is involved other factors must be considered for QoS

- Current/future work
  - Optimize **tasks split** among multiple containers and across multiple edge devices
  - **Vehicular federated learning** at the edge: exploiting radio environment maps (REMs)

Relatable Publications

A. G. Perin, M. Berno, T. Erseghe, and M. Rossi, “Towards Sustainable Edge Computing Through Renewable Energy Resources and Online, Distributed and Predictive Scheduling,” in *IEEE Transactions on Network and Service Management*, vol. 19, no. 1, pp. 306-321, March 2022, doi: 10.1109/TNSM.2021.3112796.

B. G. Perin, F. Meneghello, R. Carli, L. Schenato, and M. Rossi, “EASE: Energy-Aware Job Scheduling for Vehicular Edge Networks With Renewable Energy Resources,” in *IEEE Transactions on Green Communications and Networking*, vol. 7, no. 1, pp. 339-353, March 2023, doi: 10.1109/TGCN.2022.3199171.

C. N. Shalavi, G. Perin, A. Zanella, and M. Rossi, “Energy Efficient Deployment and Orchestration of Computing Resources at the Network Edge: A Survey on Algorithms, Trends and Open Challenges,” submitted to *ACM Computing Surveys* (preprint available: arXiv:2209.14141).

D. A. Khoshsirat, G. Perin, and M. Rossi, “Divide and Save: Splitting Workload Among Containers in an Edge Device to Save Energy and Time,” *IEEE ICC 2023 Second International Workshop on Green and Sustainable Networking (GreenNet),* May 2023.

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

800 ANNI 1222·2022

UNIVERSITÀ DEGLI STUDI DI PADOVA

SIGNET
signet.dei.unipd.it

Optimizing Edge Computing Resources Towards Greener Networks and Services

XXXV cycle of Ph.D. Course in Information Engineering
Department of Information Engineering
University of Padova

Ph.D. candidate: Giovanni Perin

Supervisor: Prof. Michele Rossi
Co-supervisor: Prof. Tomaso Erseghe

giovanni.perin.1@unipd.it

IEEE ITS Ph.D. Thesis Award
Rome, June 5, 2023

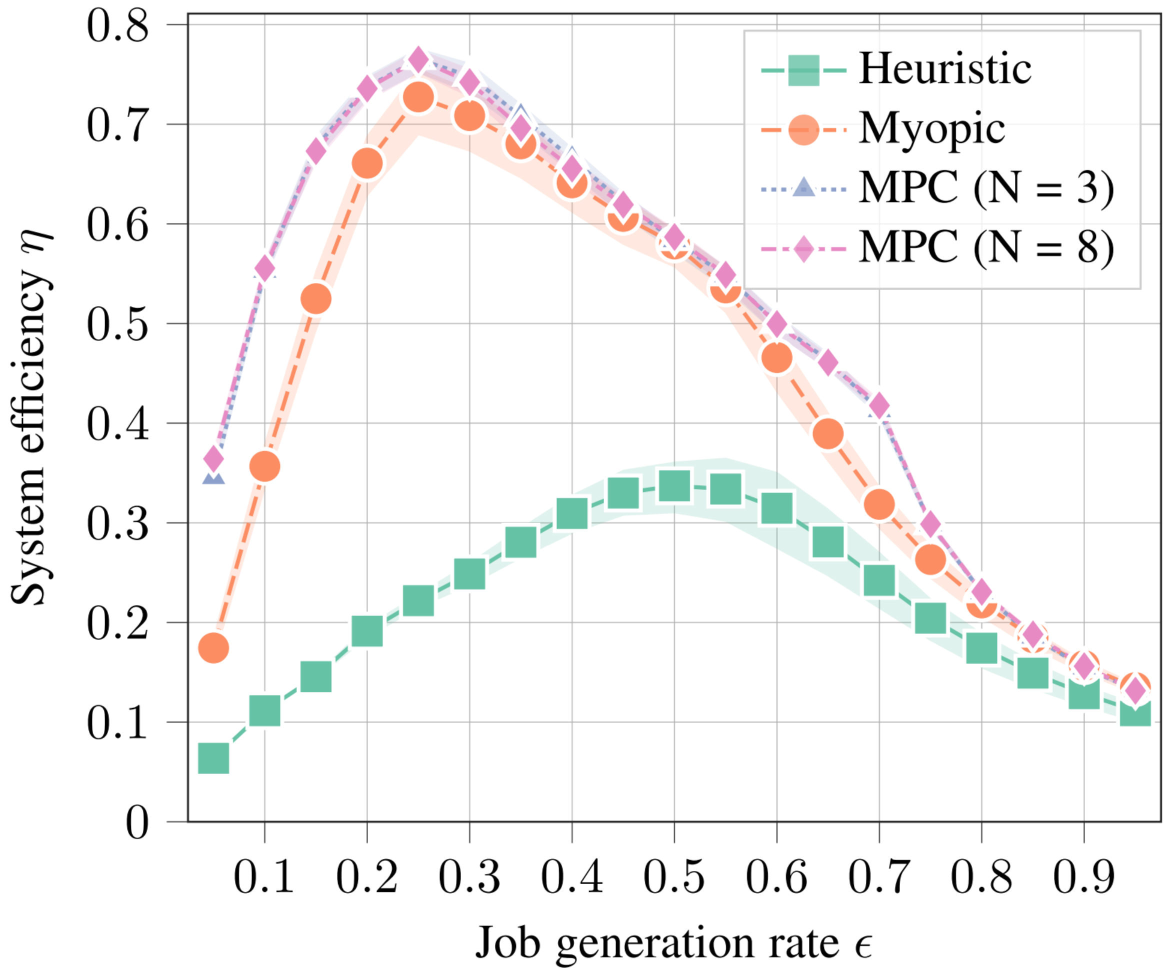
System constraints

- Processing capacity: $\sum_d w_{i,k}^d \leq W_i$
- Forced execution: $w_{i,k}^1 = s_{i,k}^1 + t_{i,k}^1 \rightarrow o_{ij,k}^1 = 0$ (unless j is the cloud)
- Transmission capacity: $\sum_{d \neq 1} o_{ij,k}^d \leq O_{ij}$
- Battery bounds: $b_{low} \leq e_{i,k} \leq B_{high}$
- Workload conservation: $w_{i,k}^d + \sum_j o_{ij,k}^d \leq s_{i,k}^d + t_{i,k}^d$

$$\eta = \frac{E_h}{E_h + E_p} \times \frac{\mu_1 W_e}{\mu_1 W_e + \mu_2 W_c} \times F(\phi)$$

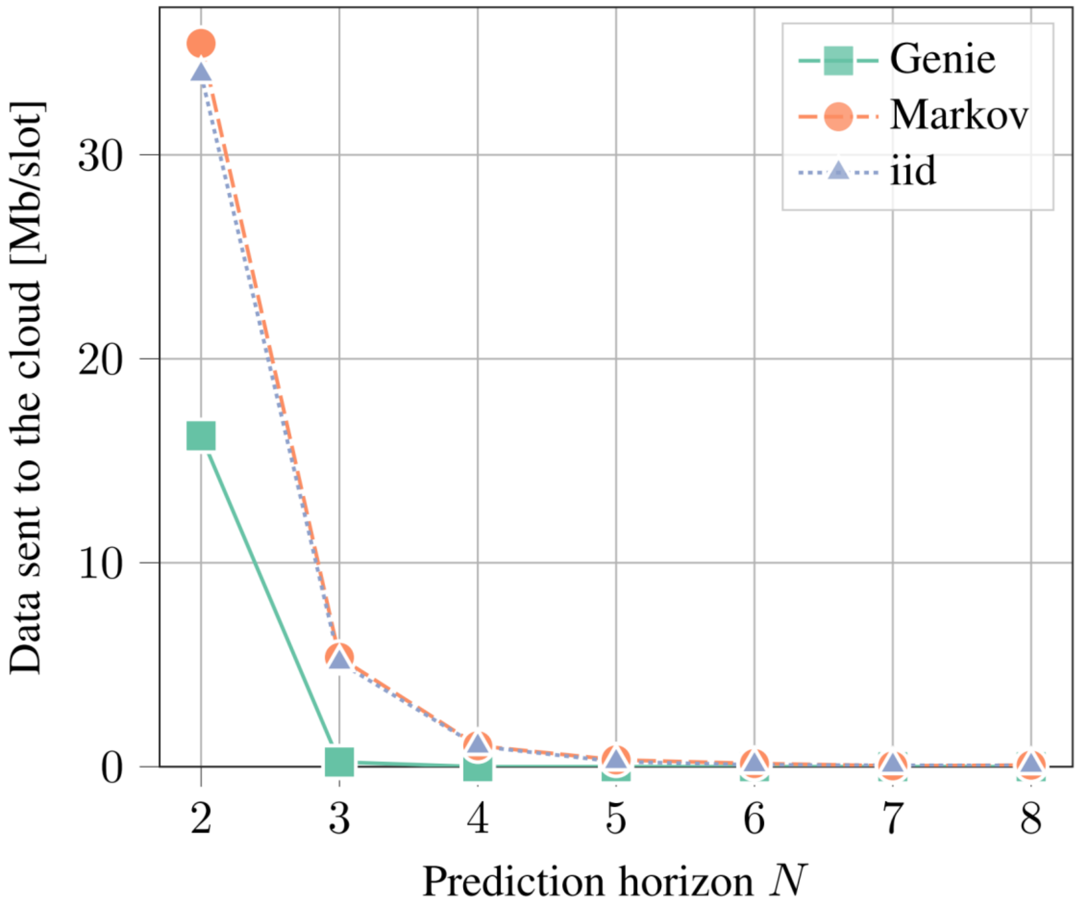
Efficiency

- Advantage for low ε, because load balancing is better induced
- For ε in $[0.6, 0.8]$, although the system is full (workers are all active), the advantage comes from a better exploitation of both harvested energy and edge resources

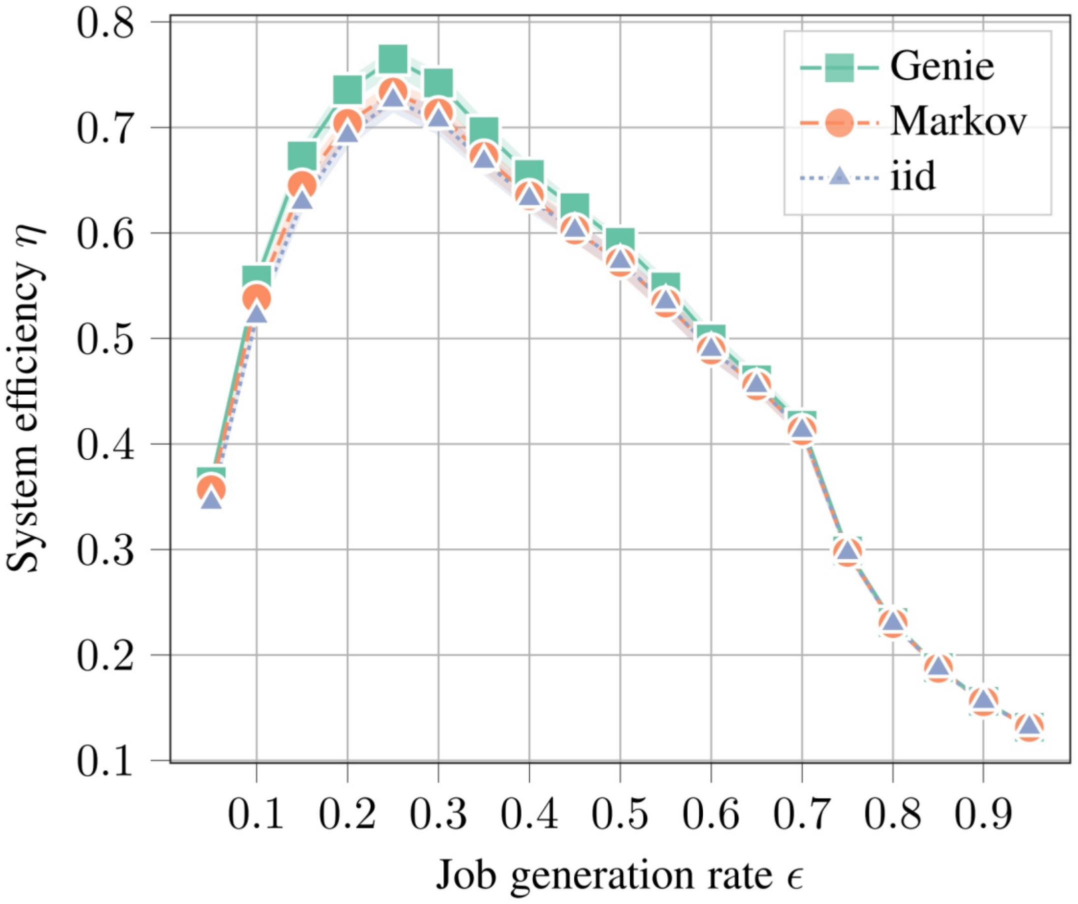


Effect of using different predictors

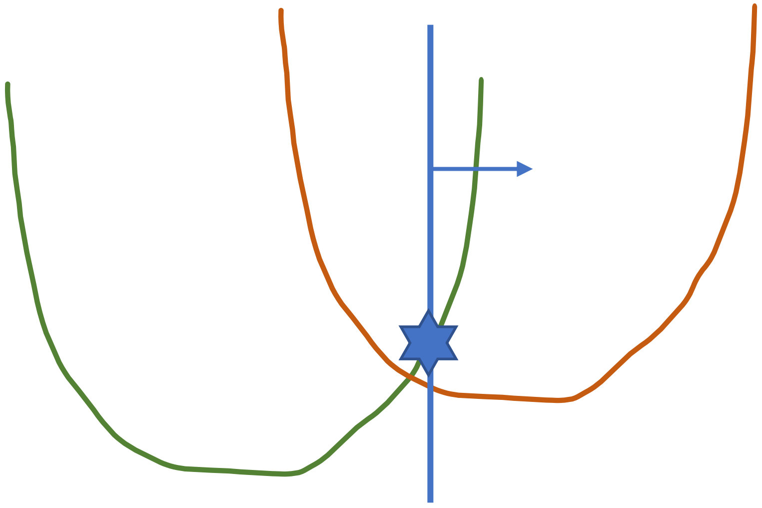
Data sent to cloud



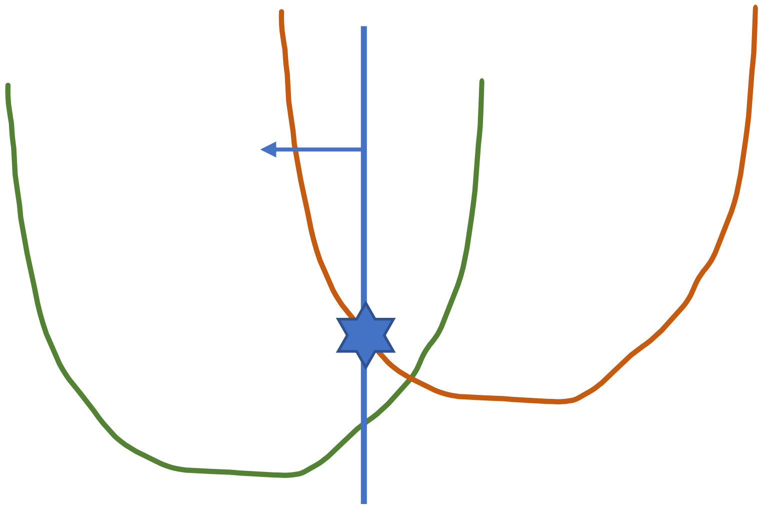
Efficiency – N = 8



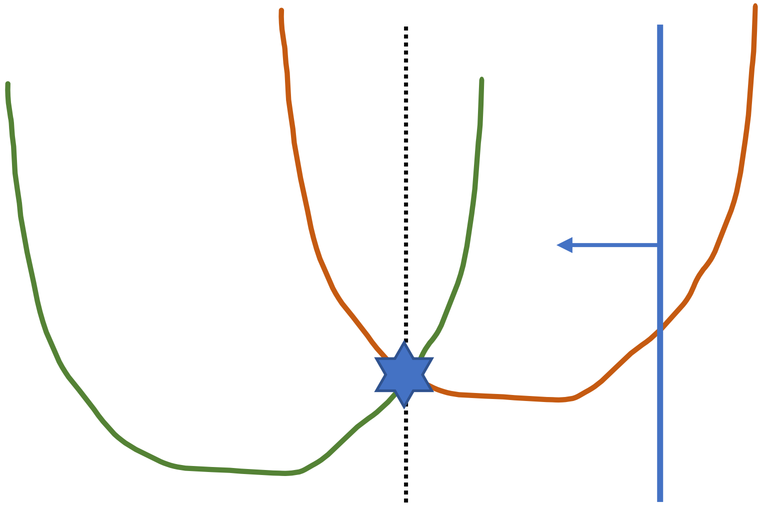
An intuition of the closed-form solution



Only one parabola is
in the feasible region
(always take the max)



OR both are
(partially) in the
feasible region



The solution is the
intersection point

Distributed optimization

Dual ascent (B)

$$\min f(x)$$
$$s.t.\ Ax = b$$

$$L(x,y) = f(x) + y^T(Ax - b)$$

$$x_{k+1} \leftarrow argmin_x\ L(x, y_k)$$

$$y_{k+1} \leftarrow y_k + \alpha_k(Ax_{k+1} - b)$$

Alternating direction method of multipliers (A)

$$\min f(x) + \mathrm{g}(\mathrm{z})$$
$$s.t.\ Ax + Bz = c$$

$$L(x,y) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\rho}{2}\ ||Ax + Bz - c||^2$$

$$x_{k+1} \leftarrow argmin_x\ L(x, z_k, y_k)$$

$$z_{k+1} \leftarrow argmin_z\ L(x_{k+1}, z, y_k)$$

$$y_{k+1} \leftarrow y_k + \rho\ (Ax_{k+1} + Bz_{k+1} - c)$$