Designing and modeling sustainable, autonomous, smart, and energy efficient Internet of Things systems, applied to precision beekeeping

PhD Defense

Hugo HADJUR — hugo.hadjur@ens-lyon.fr

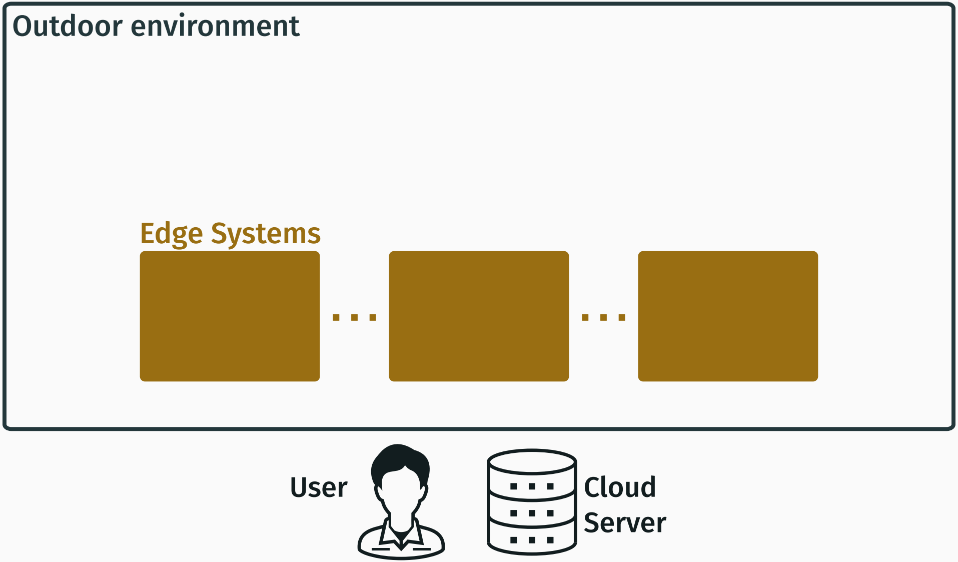
July 13, 2023

Laurent LEFÈVRE (Director) — Avalon Project Team of LIP, ENS de Lyon, Inria
Doreid AMMAR (Co-supervisor) — aivancity School for Technology, Business & Society Paris-Cachan

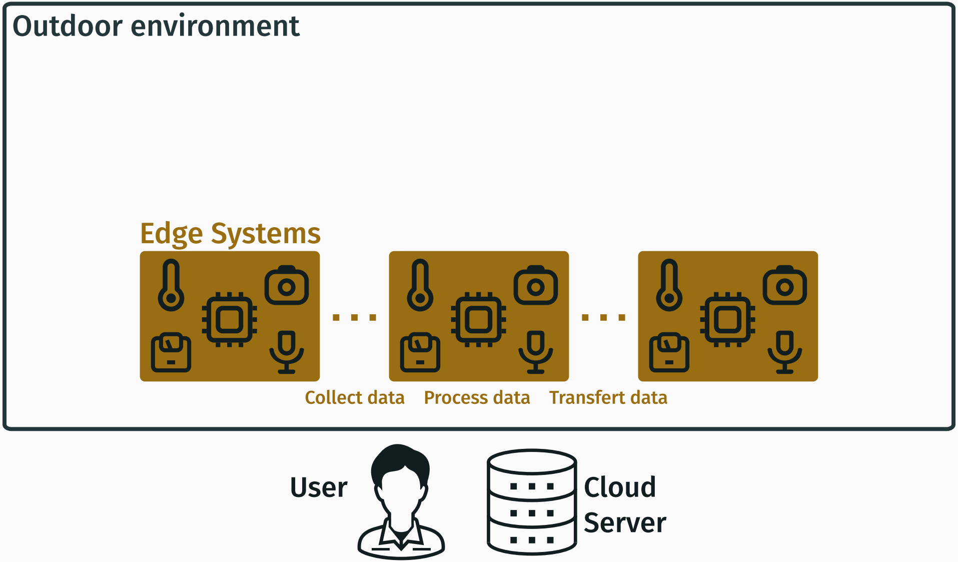
Table of contents

Context: IoT edge systems in outdoor environment and beekeeping

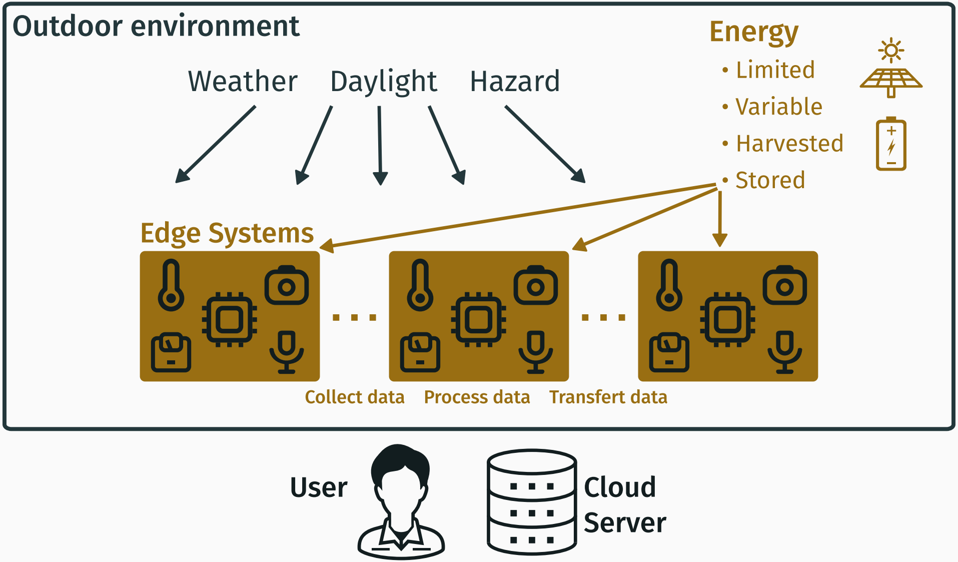
IoT edge systems in outdoor environment



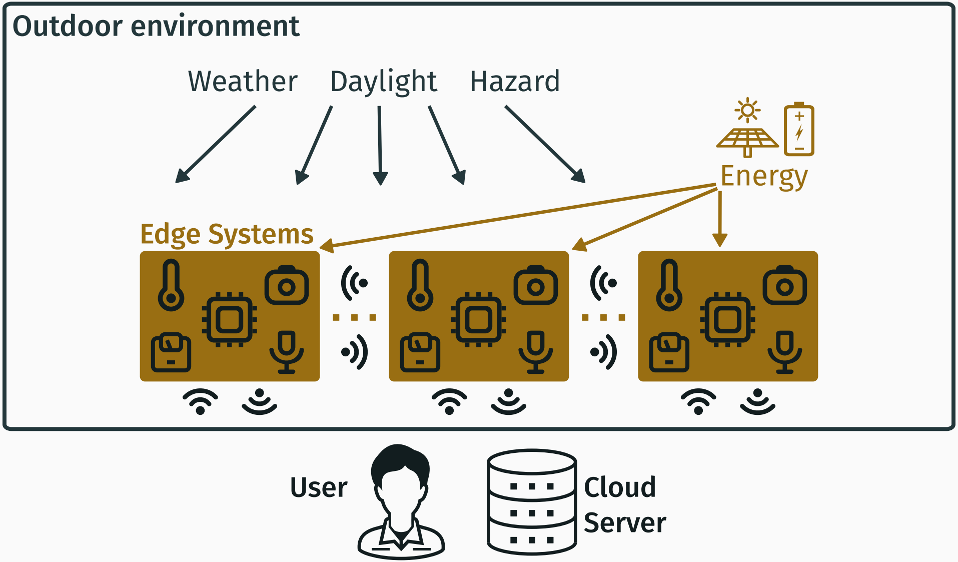
IoT edge systems in outdoor environment



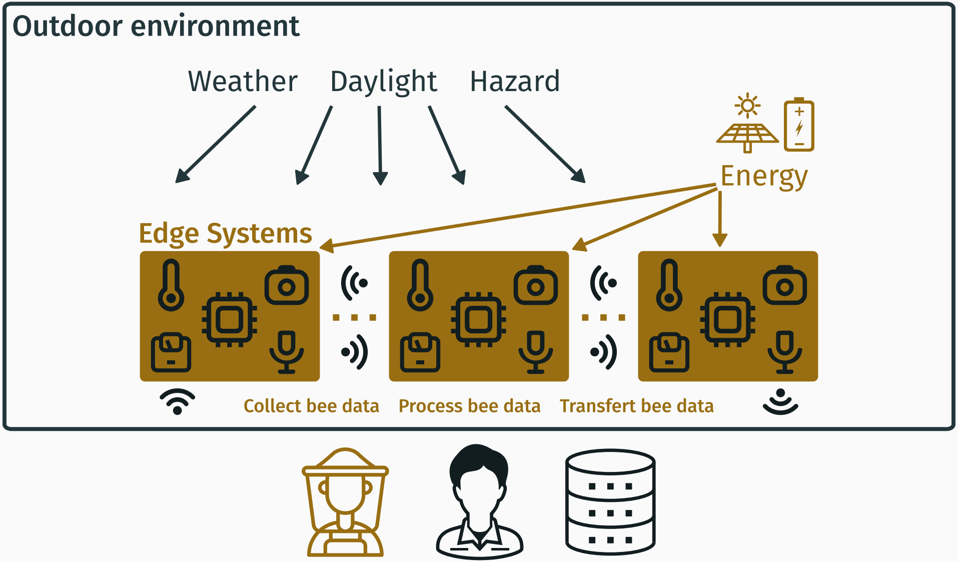
IoT edge systems in outdoor environment



IoT edge systems in outdoor environment



IoT edge systems in outdoor environment

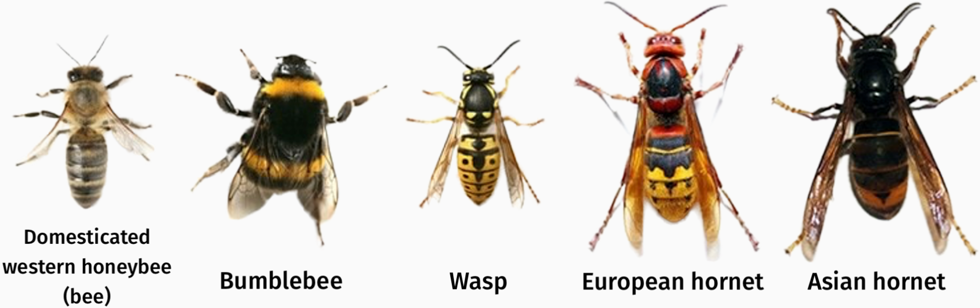


Bees

Bees are a **vector of pollination**.

They are **domesticated animals** as opposed to other flying insects.

1 beehive = 1 colony = 1 queen + thousands of female and male workers



Bees & their queen on a beehive frame

Threats to bees

Bee colonies are threatened by:

- Pesticides
- Varroa destructor mite & viruses
- Genetic constraints
- Habitat destruction
- Asian hornets

A bee infected by a Varroa mite

The combination of these constraints are labeled as the **Colony Collapse Disorder**.

Precision beekeeping

Definition of precision beekeeping

Precision beekeeping is a subfield of precision agriculture, both make use of the Internet of Things.

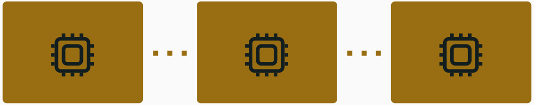
Precision beekeeping is an “**apiary management strategy** based on the monitoring of individual bee colonies to **minimize resource consumption and maximize the productivity of bees**” [Zacepins et al., 2012].

Most of the beekeepers using precision beekeeping solutions are **professional** (own more than 150 beehives).

Pictures of precision beekeeping systems found in the scientific literature

Precision beekeeping systems - Hardware

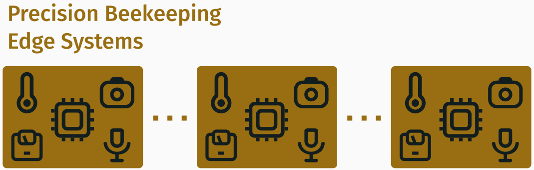
Precision Beekeeping
Edge Systems



A **connected beehive** consists of four main parts:

- **Microcontroller**: Raspberry Pi, Arduino, ESP32, etc.

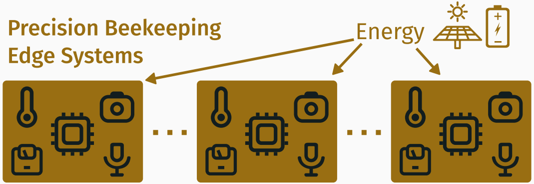
Precision beekeeping systems - Hardware



A **connected beehive** consists of four main parts:

- **Microcontroller**: Raspberry Pi, Arduino, ESP32, etc.
- **Sensors**: camera, microphone, accelerometer, thermometer, scale, etc.

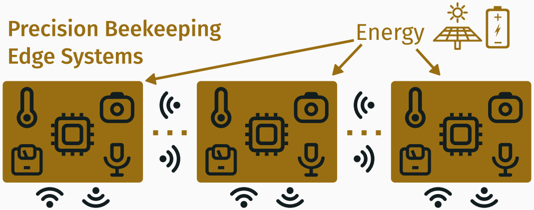
Precision beekeeping systems - Hardware



A **connected beehive** consists of four main parts:

- **Microcontroller**: Raspberry Pi, Arduino, ESP32, etc.
- **Sensors**: camera, microphone, accelerometer, thermometer, scale, etc.
- **Energy**: battery, solar panel, power supply

Precision beekeeping systems - Hardware



A **connected beehive** consists of four main parts:

- **Microcontroller**: Raspberry Pi, Arduino, ESP32, etc.
- **Sensors**: camera, microphone, accelerometer, thermometer, scale, etc.
- **Energy**: battery, solar panel, power supply
- **Network**: short-range (Bluetooth, Wi-Fi, etc.), cellular (2G, 3G, 4G, etc.), LPWAN (LoRa, Sigfox, etc.)

What are the services provided by a smart beehive?

Mostly **sound & vibration**-based:

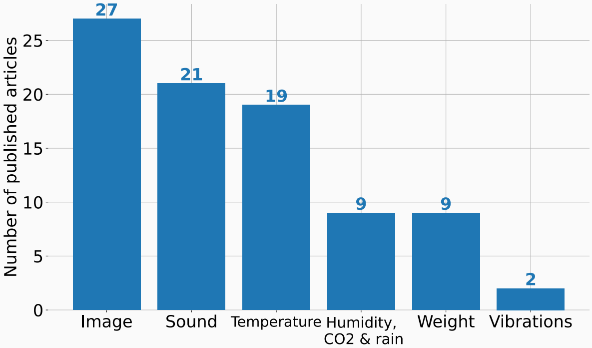
- Queen detection
- Swarm detection & prediction

Mostly **image**-based:

- Counting bees & population estimation
- Pest detection
- Pollen detection
- Tracking bees trajectories

Mostly using a **mix of temperature, humidity and weight**:

- Honey production estimation
- Health diagnosis



Distribution of the types of data used for creating services in precision beekeeping literature from 2008 to 2023

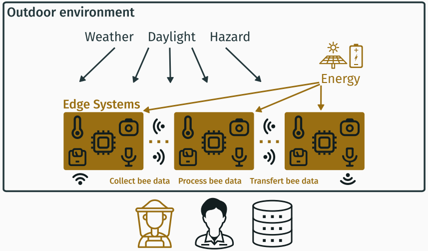
Objectives

Precision beekeeping:

- Propose an autonomous energy-aware **smart beehive architecture**
  To collect, process and transfer data to assist beekeepers and minimize the Colony Collapse Disorder
- **Collect multimodal bee data** and share it to the community

Energy conservation in IoT systems:

- **Collect energy data** of functioning IoT edge systems
- Study the **scalability of smart IoT edge systems** and the potential **value of cloud servers**
- **Maximize the value of data-based services in edge systems** that operate outdoors under limited, variable and harvested energy budget



Design and deployment of an energy-aware precision beekeeping system

Observation of the precision beekeeping literature

The articles published in the field of precision beekeeping show room for improvement in the following areas:

- Lack of **open dataset** that show longterm bee dynamics
- **Energy consumption analysis** of IoT edge systems and services

International journal article: Hadjur, H., Ammar, D., and Lefèvre, L. (2022). *Toward an intelligent and efficient beehive: A survey of precision beekeeping systems and services*. Computers and Electronics in Agriculture

Design of our precision beekeeping system

Designing and modeling sustainable, autonomous, smart, and energy efficient Internet of Things systems, applied to precision beekeeping.

Design of our precision beekeeping system

Designing and modeling **sustainable**, autonomous, smart, and energy efficient Internet of Things systems, applied to precision beekeeping.

- **Sustainable** → Rely on off-the-shelf hardware and open-source software, and share our collected data to the community

Design of our precision beekeeping system

Designing and modeling **sustainable**, **autonomous**, smart, and energy efficient Internet of Things systems, applied to precision beekeeping.

- **Sustainable** → Rely on off-the-shelf hardware and open-source software, and share our collected data to the community
- **Autonomous** → Be self-sufficient while being deployed outdoors and subject to weather and hazard

Design of our precision beekeeping system

Designing and modeling **sustainable**, **autonomous**, **smart**, and energy efficient Internet of Things systems, applied to precision beekeeping.

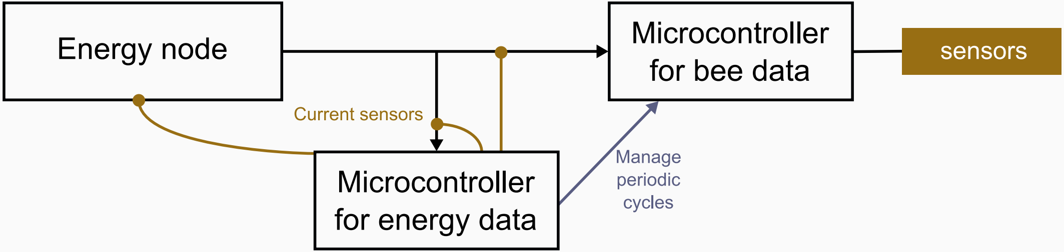
- **Sustainable** $\rightarrow$ Rely on off-the-shelf hardware and open-source software, and share our collected data to the community
- **Autonomous** $\rightarrow$ Be self-sufficient while being deployed outdoors and subject to weather and hazard
- **Smart** $\rightarrow$ Provide data-based services to assist beekeepers and protect bees

Design of our precision beekeeping system

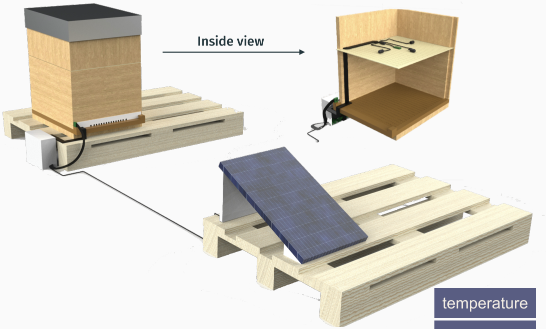
Designing and modeling **sustainable**, **autonomous**, **smart**, and **energy efficient** Internet of Things systems, applied to precision beekeeping.

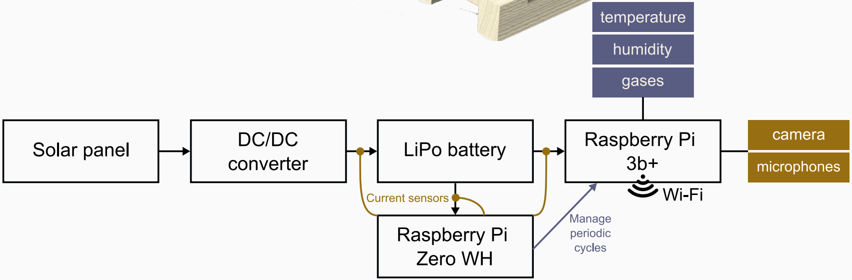
- **Sustainable** → Rely on off-the-shelf hardware and open-source software, and share our collected data to the community
- **Autonomous** → Be self-sufficient while being deployed outdoors and subject to weather and hazard
- **Smart** → Provide data-based services to assist beekeepers and protect bees
- **Energy-efficient** → Design frugal methods to provide the services

Development of our energy-aware precision beekeeping system



Deployment of our energy-aware precision beekeeping system





Deployment of our energy-aware precision beekeeping system

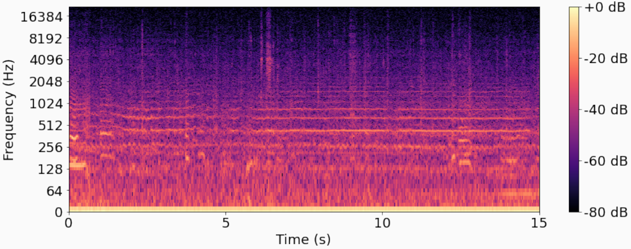
2 deployed systems and 4 unequipped hives in Cachan, France

Deployment of 3 systems in Lyon, France

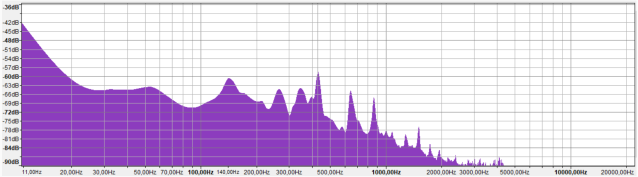
Images collected by our system




Example of audio collected by our system



Log-spectrogram of an audio sample



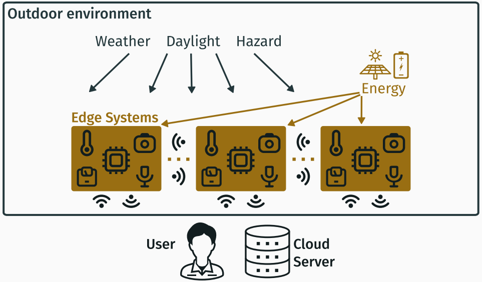
Frequency distribution of an audio sample

Modeling and simulation of resource allocation at the edge and in the cloud

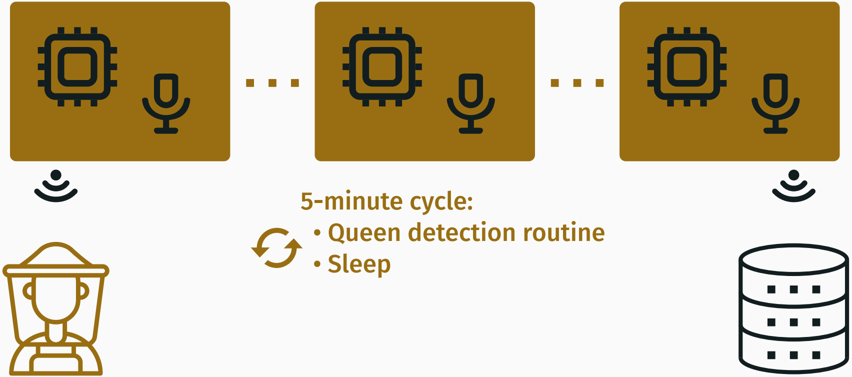
Objectives

Energy consumption analysis of IoT edge systems

- **Optimize the energy consumed** by our energy-aware precision beekeeping system
- **Model and simulate IoT edge systems** and **instantiate the simulation** thanks to our smart beehive's energy consumption data
- **Determine the value of a cloud server** as a complement to the edge device

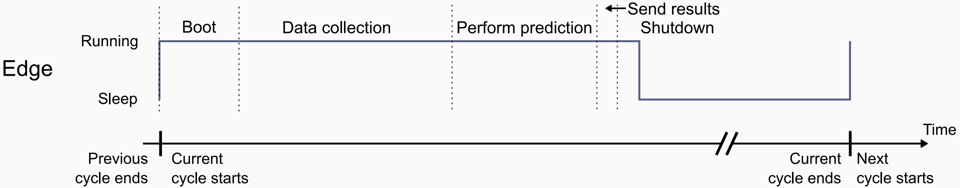


Simulation methods: by increasing the number of edge devices



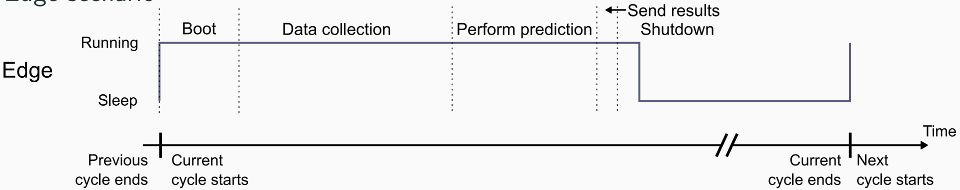
Two scenarios

- Edge scenario

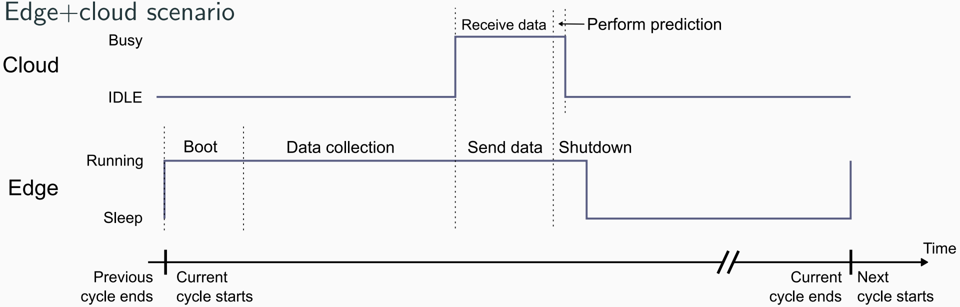


Two scenarios

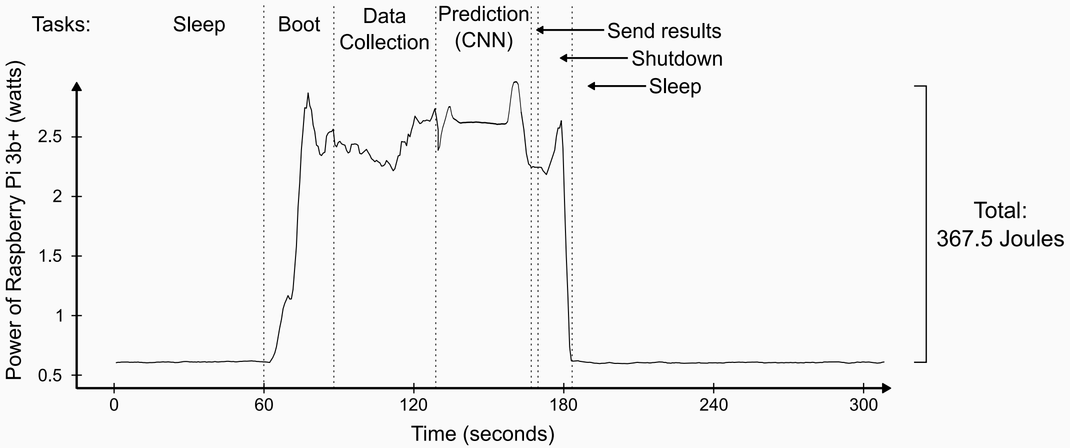
- Edge scenario



- Edge+cloud scenario

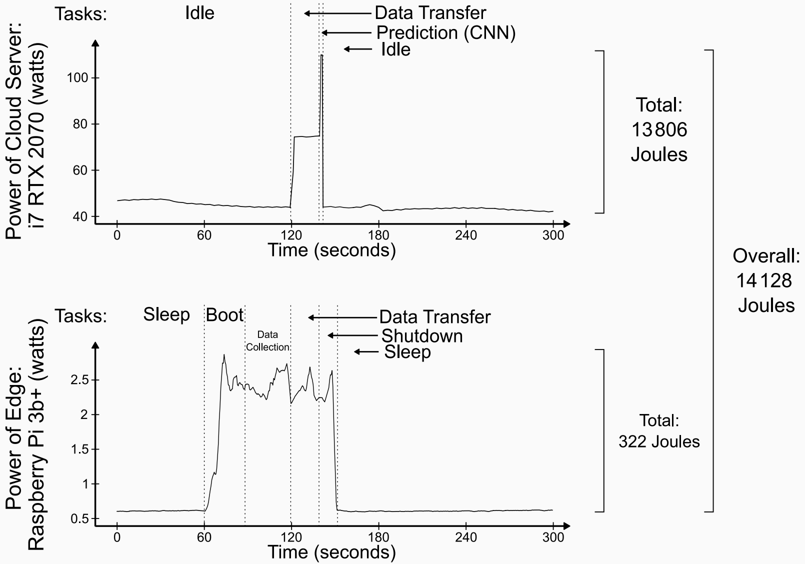


Edge scenario's measured energy consumption



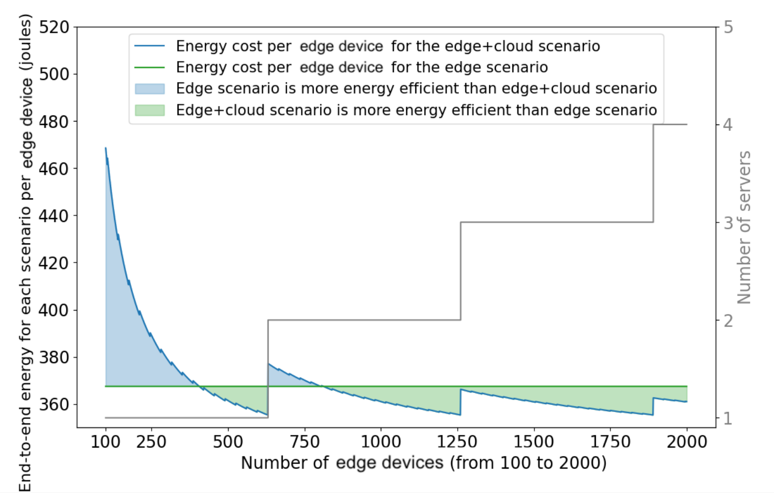
Energy consumption for the edge scenario

Edge+cloud scenario's energy consumption



Results of simulation in theoretical perfect conditions

Implementation of the simulation from scratch in Python.

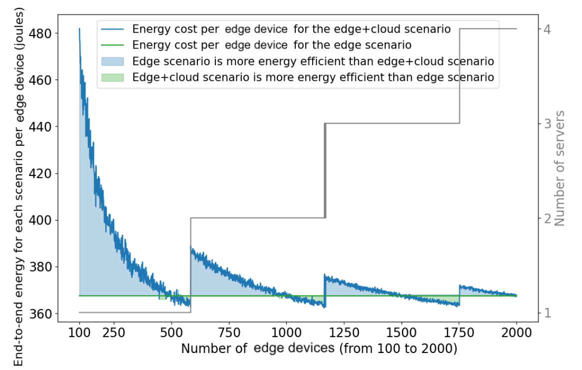


The use of a cloud server is always beneficial after 803 edge devices.

Results of simulation with losses

Addition of losses to replicate real conditions.

- Cloud server's capacity saturation → server uses more energy
- Network collision when many edges attempt to send data in parallel → transfer takes more time
- Edge devices sometimes fail to perform their routine → servers are not used at their full capacity

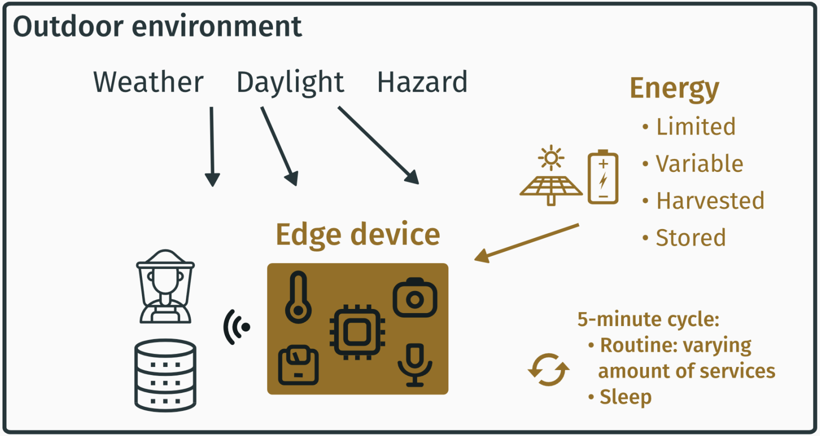


Hadjur, H., Ammar, D., Lefèvre L. (2023). *Services Orchestration at the Edge and in the Cloud for Energy-Aware Precision Beekeeping Systems*, PAISE 2023, 5th Workshop on Parallel AI and Systems for the Edge, IPDPS Workshop

A reinforcement learning approach for task allocation in IoT systems

Methods

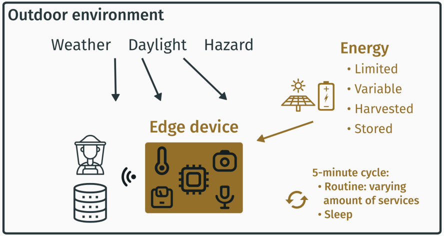
- Previously with the simulation: many edge devices with 1 service
- Now: **1 edge device with many services, taking into account the varying energy budget**

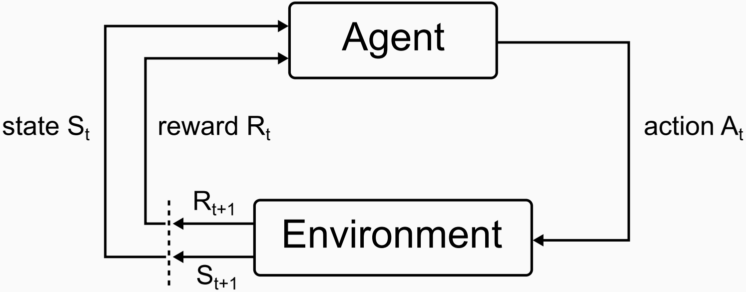


Reinforcement learning

Why reinforcement learning (RL)?

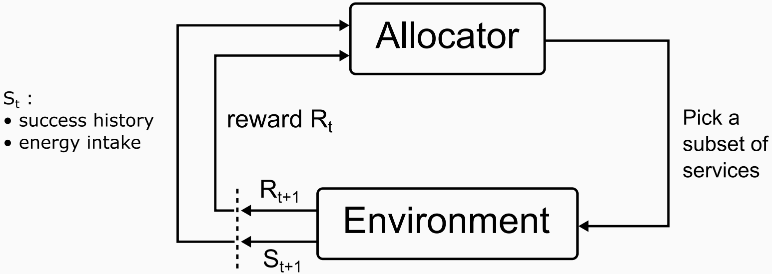
- Stochastic dynamic environment (outdoors)
- Partially observable states
- No training dataset available





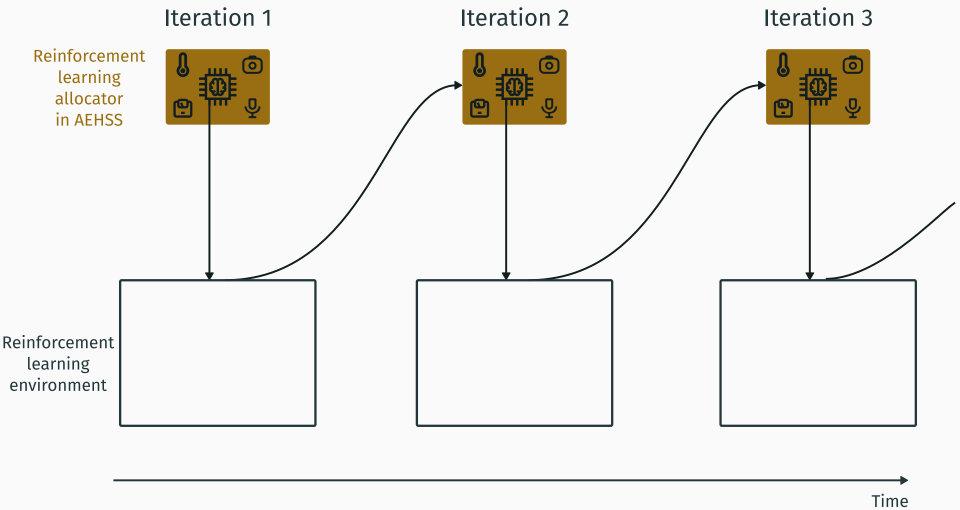
Objective: design a **task allocator** able to select a subset of services within a set of available services. Each service has a value and the allocator must **maximize the aggregated sum of values over time**.

Constraint: **stay within an acceptable energy budget** ($> 25\%$ of the total).



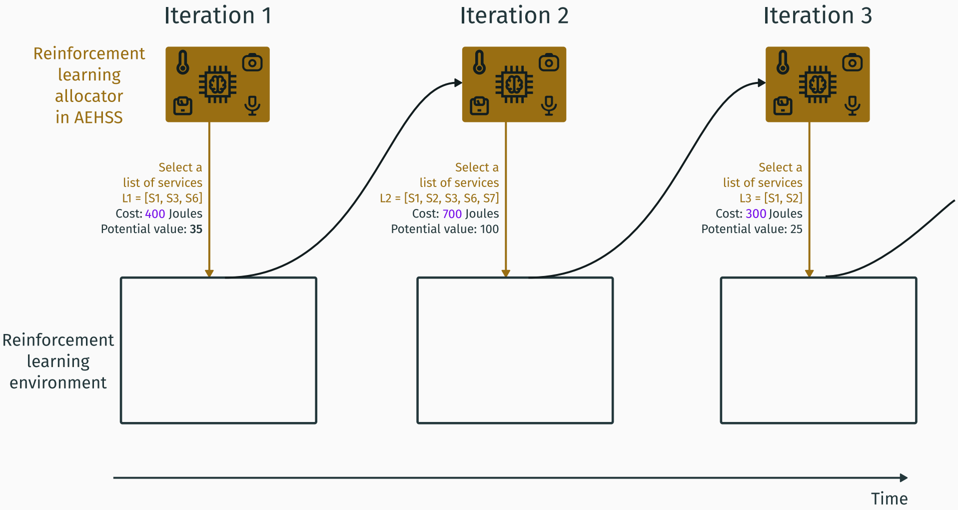
Schema of RL in IoT edge systems

Schema of RL iterations for energy optimization and value maximization in AEHSS systems

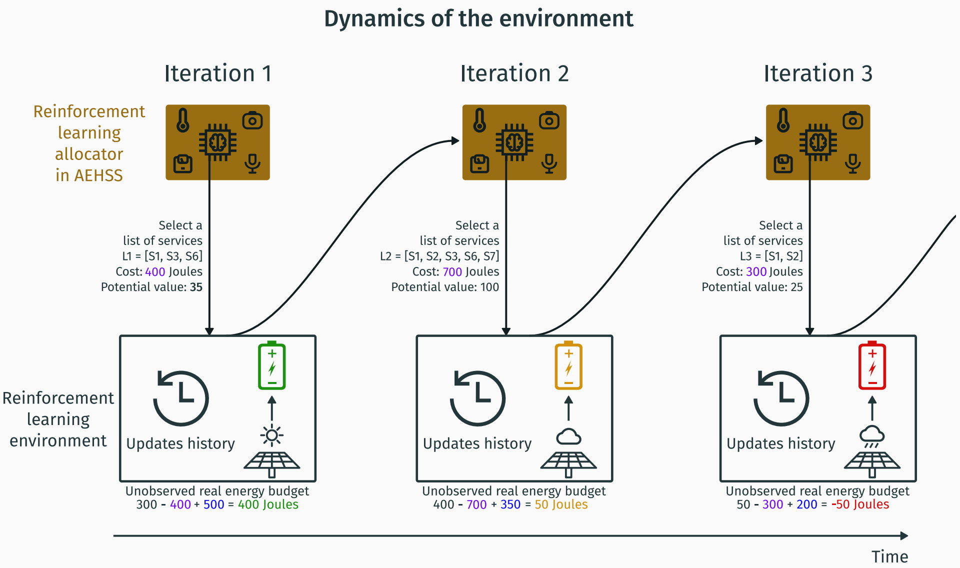


Schema of RL in IoT edge systems

Allocator's role

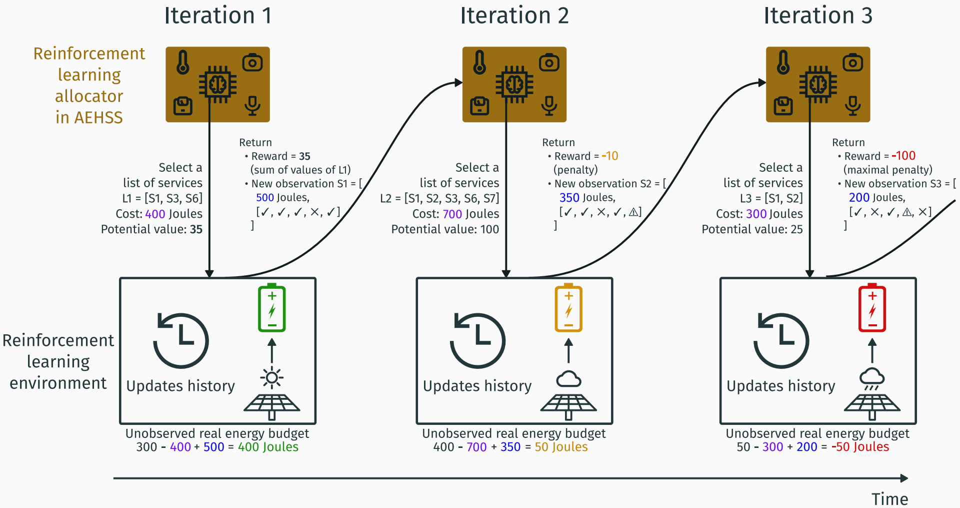


Schema of RL in IoT edge systems



Schema of RL in IoT edge systems

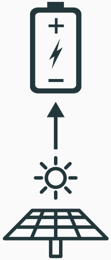
Schema of RL iterations for energy optimization and value maximization in AEHSS systems



Focus on the environment — Photovoltaic energy intake prediction

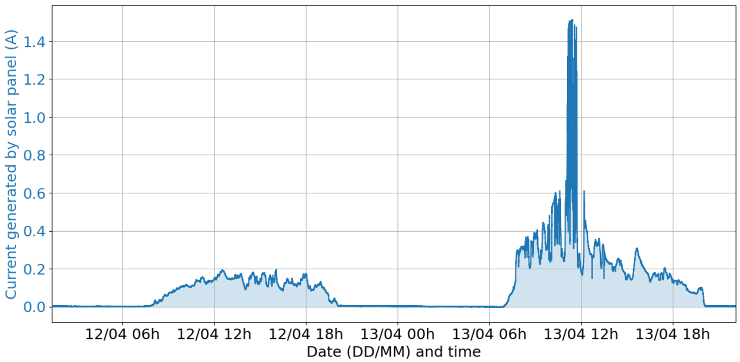
In deployed versions, the energy intake from the solar panel is not measured.

Challenge: **predict the energy intake thanks to external variables**.



External predictor variables:

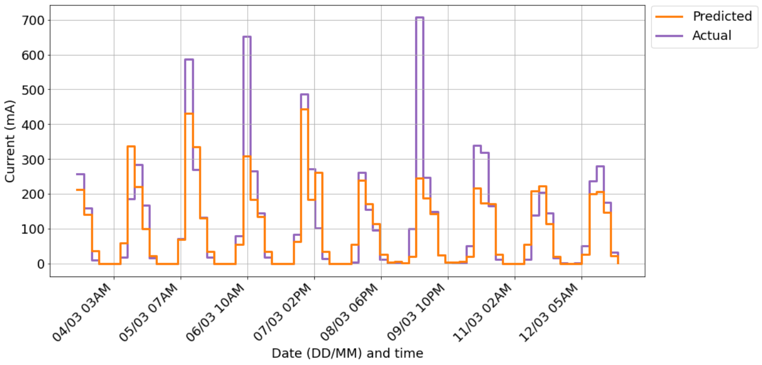
- Temperature
- Humidity
- Air pressure
- Last 3 hours precipitations
- Horizontal visibility
- Nebulosity
- Solar zenith
- Solar azimuth
- Number of seconds elapsed in the current day



Measured photovoltaic energy intake data (Lyon, France)

Prediction of energy intake thanks to external variable

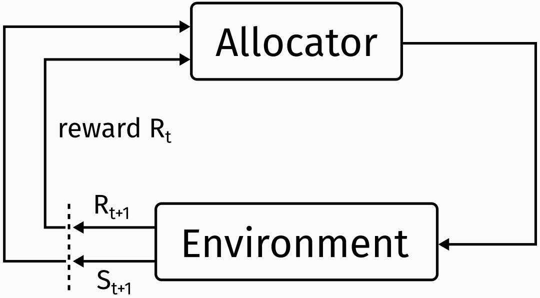
- Replication of the analysis from [Kraemer et al., 2020]
- Comparison of the performances of 22 versions of 8 machine learning models
- Confirmation that random forest regressors are the most accurate at predicting energy intake



Predicted and actual values of solar intake (test set)

Training and testing reinforcement learning models

S_t :
- success history
- energy intake



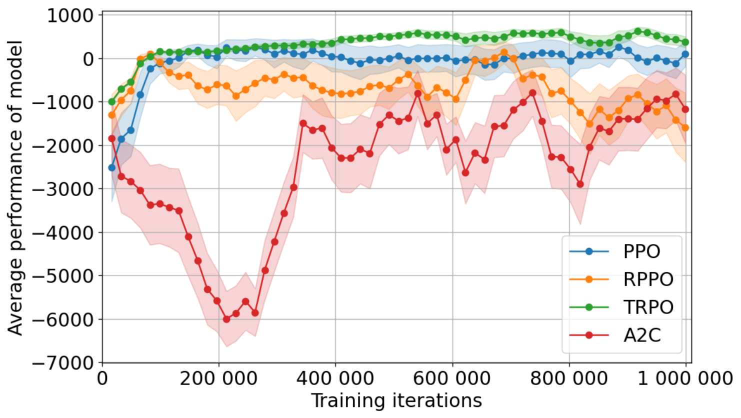
Training = exploration:
Pick subsets of services
Try different strategies

Inference = exploitation
Pick the best subset of services

Results of four reinforcement learning architectures

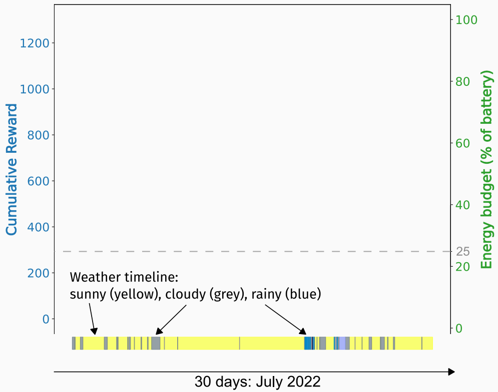
Comparison of 4 reinforcement learning methods

- Proximal Policy Optimization (PPO)
- Recurrent PPO (RPPO)
- Trust Region Policy Optimization (TRPO)
- Advantage Actor Critic (A2C)

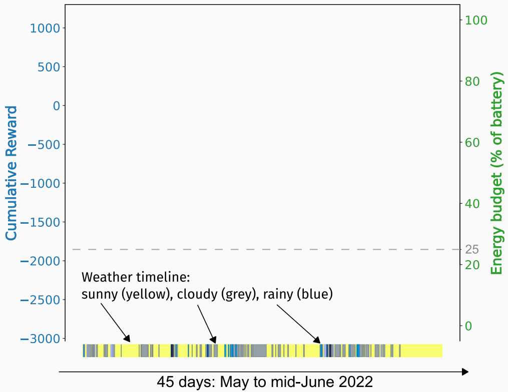


Plots of simulated runs and comparison with heuristics

Test experiment 1: 30 days (sunny weather)



Test experiment 2: 45 days (mixed weather)

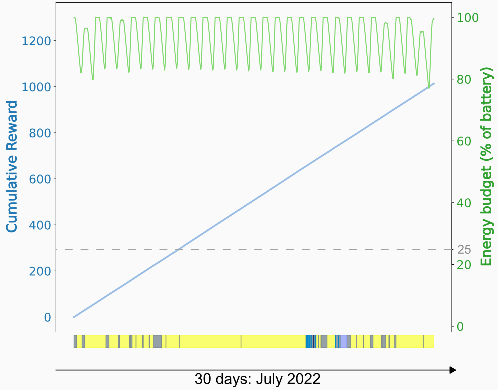


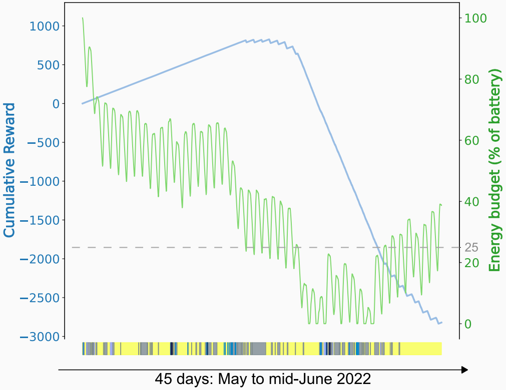
Plots of simulated runs and comparison with heuristics

Test experiment 1: 30 days (sunny weather)

Test experiment 2: 45 days (mixed weather)

Basic heuristic:



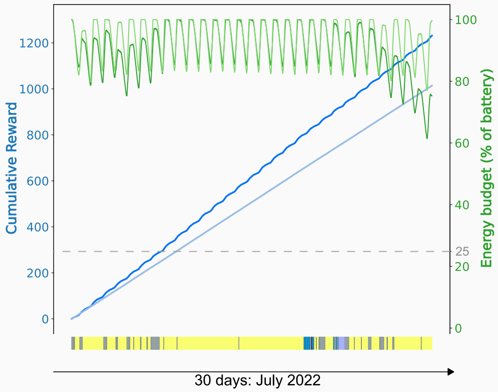


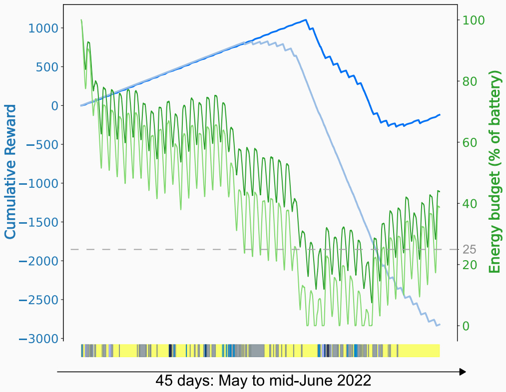
Plots of simulated runs and comparison with heuristics

Test experiment 1: 30 days (sunny weather)

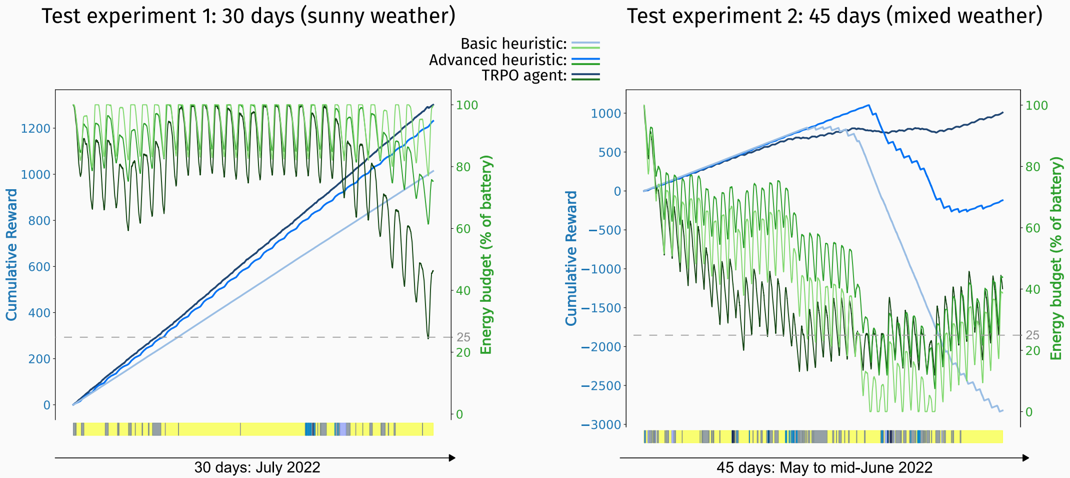
Test experiment 2: 45 days (mixed weather)

Basic heuristic:
Advanced heuristic:





Plots of simulated runs and comparison with heuristics



Conclusion & Future works

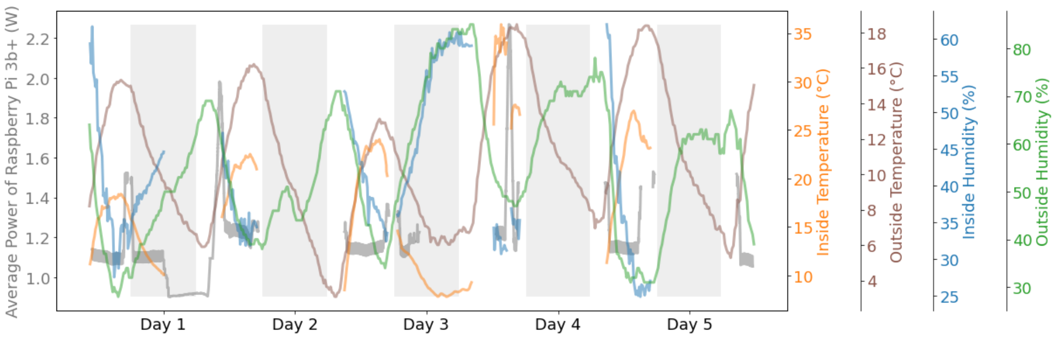
Contributions of the thesis

1. An autonomous energy-aware smart beekeeping system answering current precision beekeeping fields' challenges around the energy
2. Collection and publication of a multimodal (bee-related and energy) dataset
3. Large-scale simulation and analysis of autonomous energy-harvesting stationary IoT systems $\rightarrow$ Cloud servers do not considerably decrease the energy needed to perform a routine with one service
4. A reinforcement learning service placement model for task selection which outperforms rule-based agents

Lessons learned from experimental science

Challenges were met during the data collection period:

- Inconsistency of the battery power output → Holes in the dataset
- Unstable network in Lyon → Manual maintenance interventions
- Bee colonies did not make it to the winter → Fewer data (but insightful biological data)



The in-hive and external temperature and humidity, and the averaged Raspberry Pi 3b+ consumption during one week of Spring 2022 in Paris

Future works

- **Deployment and test** of the reinforcement learning task selection model
- Not only the usage's energy consumption analysis, but full **life cycle analysis**
- Optimization of the costs of AI models' **training** phase
- Go **from simulation to emulation** and explore **how many beehives should be equipped** to maximize the added value of services compared to the invested resources?
- Is **one Joule in the cloud "equal" to one Joule at the edge**?
- Could there be **rebound effects** for the large-scale deployment of our system?

Publications

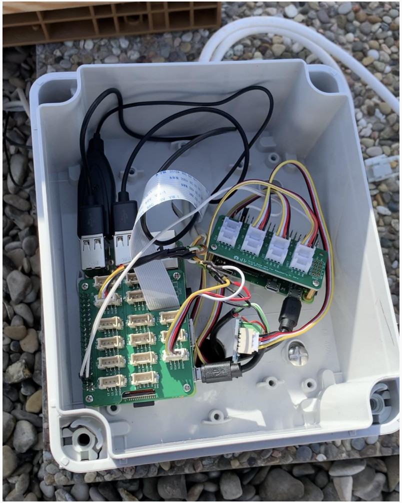
1. **International Conference: Hugo Hadjur**, Doreid Ammar, and Laurent Lefèvre. 2020. "Analysis of energy consumption in a precision beekeeping system" In Proceedings of the 10th International Conference on the Internet of Things (IoT '20). Association for Computing Machinery, New York, NY, USA, Article 20, 1–8. DOI:https://doi.org/10.1145/3410992.3411010
2. **International Journal: Hugo Hadjur**, Doreid Ammar, Laurent Lefèvre, "Toward an intelligent and efficient beehive: A survey of precision beekeeping systems and services", Computers and Electronics in Agriculture, Volume 192, 2022, 106604, ISSN 0168-1699, https://doi.org/10.1016/j.compag.2021.106604
3. **International Workshop: Hugo Hadjur**, Doreid Ammar, Laurent Lefèvre. 2023. "Services Orchestration at the Edge and in the Cloud for Energy-Aware Precision Beekeeping Systems", PAISE 2023, 5th Workshop on Parallel AI and Systems for the Edge, IPDPS Workshop
4. **Open-source Dataset: Hugo Hadjur**, Doreid Ammar, Laurent Lefèvre. 2023. "EAPBSdata: Energy consumption and bee dataset of five connected beehives of one high season in France" https://doi.org/10.5281/zenodo.7880085

The presentation's theme

Get the source of this theme and the demo presentation from

`github.com/matze/mtheme`

Appendix - Hardware for the deployment of our system

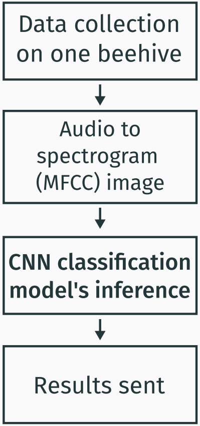


Hardware (microcontrollers, sensors) box

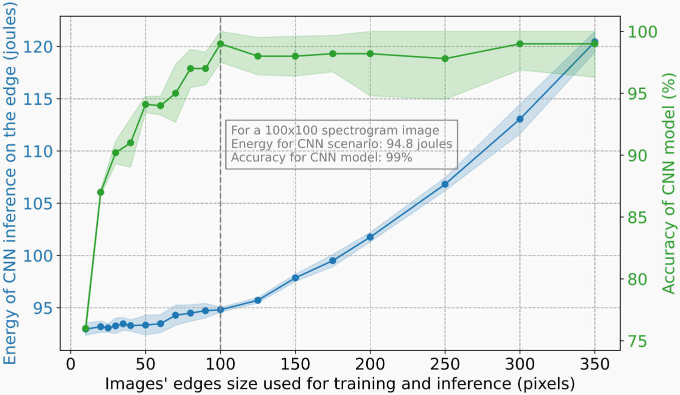
Energy (battery) box

Appendix - Energy consumption optimization of CNN queen detection model's inference

Service routine



Optimization of energy consumption



Appendix - Basic heuristic

Variables: Set of tasks: $T = \{T_1, T_2, ...\}$
Input : List of past outcomes of size n: $[S_{t-1}, S_{t-2}, ..., S_{t-n}]$
Output : Set of tasks $T_{selected}$

$confidence = \frac{\sum_{i=0}^{n-1} S_{t-i}}{2 * n}$
if $confidence < 0.5$ **then**
    return [ ]
end
else
    $T_{selected} = randomSample(T, 2)$
    return $T_{selected}$
end

Algorithm 1: Simple heuristic for selecting a subset of tasks

Appendix - Advanced heuristic

Variables: Set of tasks: $T = \{T_1, T_2, ...\}$
Their cost (energy) $E = \{E_1, E_2, ...\}$
The best-case solar-produced energy over a time interval: E_{max}

Input : Energy produced at time interval t: E_t
List of past outcomes of size n: $[S_{t-1}, S_{t-2}, ..., S_{t-n}]$

Output : Set of tasks $T_{selected}$

$R_{energy} = \frac{E_t}{E_{max}}$
$confidence = \frac{\sum_{i=0}^{n-1} S_{t-i}}{2 * n}$
shuffle(T)
$E_{selected} = 0$; $T_{selected} = [\,]$
if $confidence < 0.5$ **then**
    return $T_{selected}$
end
for T_i *in* T **do**
    if $\frac{E_{selected}}{\sum E_i} > (confidence - 0.2)^{1.2} * \min(1, (1.3 - R_{energy}))$ **then**
        break
    end
    $E_{selected} = E_{selected} + E_i$
    $T_{selected}.append(T_i)$
endFor
return $T_{selected}$

Algorithm 2: Advanced heuristic for selecting a subset of tasks at iteration t

Appendix - PPO's objective function

$$L^{CLIP}(\theta) = \hat{E}_t[min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)]$$