CaRE: Towards Carbon and Resource Efficient Orchestration at the Cloud-Edge Continuum

Georgia Christofidi *Francisco Álvarez Terribas* *Jesus Alberto Omaña Iglesias*

Nicolas Kourtellis *Thaleia Dimitra Doudali*

institute IMDEA software

Telefónica Innovación Digital

The problem of Carbon Emissions Reduction

Challenge: Increased Carbon Emissions due to **exponential growth** of Computing.

Key drivers:
- ML applications
- Generative AI
- Video streaming

AI Model	Carbon Impact of Training*	Real-word equivalent example
GPT-3	500 metric tons of CO2eq.[1]	500 round-trip flights from Madrid to New York for one passenger.
GPT-4	12,456 - 14,994 metric tons CO2eq (*estimated*).[2]	50-60 fully loaded Boeing 747 flights.

*Training only accounts for 43% of lifecycle carbon emissions. [1]

Solution: **Spatial** and **Temporal** Workload Shifting.

[3]

☑ Spatial Shifting

Fossil-fuel-heavy regions

↓ Workload Migration

Greener areas

☑ Temporal Shifting

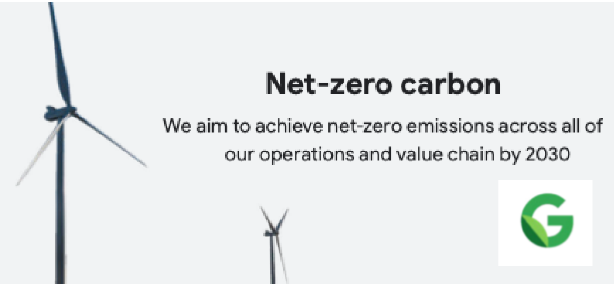
Pause with no strong latency requirements (e.g., batch jobs)

Resume when green energy available.

Sources [1]: Beyond Efficiency: Scaling AI Sustainably
[2]: https://towardsdatascience.com/the-carbon-footprint-of-gpt-4-d6c676eb21ae
https://app.electricitymaps.com/map/72h

The problem of Carbon Emissions Reduction

During the last 2 years existing systems are **redisigned** with the end goal of **reducing carbon emissions**.



Microsoft

Carbon negative

Our carbon negative commitment includes three primary areas: reducing carbon emissions, increasing use of carbon-free electricity, and carbon removal. We made meaningful progress on carbon-free electricity and carbon removal in FY23. Microsoft has taken a first-mover approach to supporting **carbon-free electricity** infrastructure, making long-term investments to bring more carbon-free electricity onto the grids where we operate.

Going Green for Less Green: Optimizing the Cost of Reducing Cloud Carbon Emissions

ASPLOS '24

Ecovisor: A Virtual Energy System for Carbon-Efficient Applications

ASPLOS '23

Carbon Explorer: A Holistic Framework for Designing Carbon Aware Datacenters

Meta

CARIBOU: Fine-Grained Geospatial Shifting of Serverless Applications for Sustainability

SOSP '24

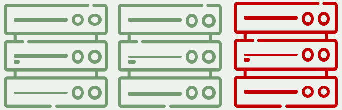
The problem of Carbon Emissions Reduction

Implications of CO_2 reductions on other aspects

Problem: **Resource**, **Performance,** and **Cost** are compromised when reducing CO_2.

Resource Awareness

Idle!



Temporal Shifting

- Resource Waste
- Energy Inefficiency
- Increased Cost

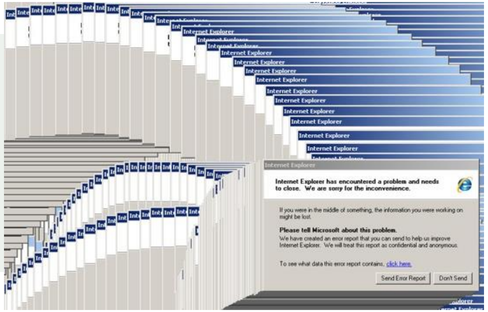
Cost Awareness

Spatial Shifting

Small national companies need **additional budget** to rent remote resources in greener regions.

Performance Awareness

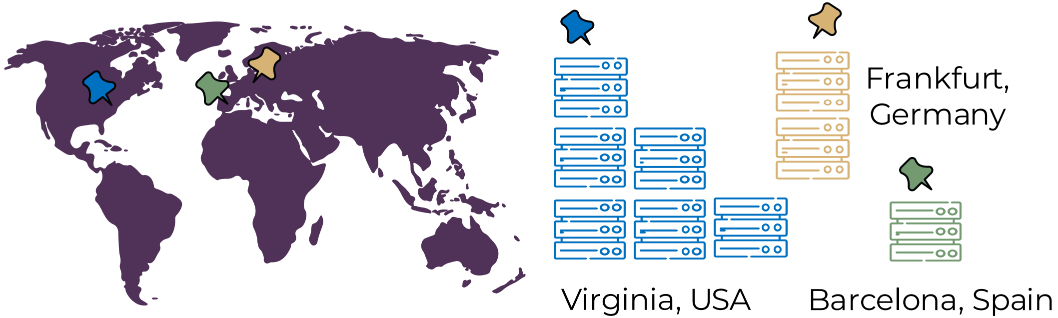
Only **specific types** of jobs can be shifted in time.



Not all workloads can wait!

Takeaway: Optimizing Carbon + Resource + Cost + Performance = Harder than it looks.

Applications across the Edge-Cloud Continuum



Heterogeneous resources + diverse applications = complex trade-offs

Real-World Conflicting Requirements

1. Movie Platform Recommendations

- Not time-sensitive.
- Global platform, resources worldwide.

Carbon Efficiency Focus

2. Small National Business in Spain

- Limited local resources.
- Renting resources elsewhere is costly.

Cost Constraints

3. Online Gaming

- Latency-critical application.
- Carbon efficience is secondary to user experience.

Performance Requirements

Takeaway: Each application across the cloud-edge continuum values carbon, resources, cost, and performance differently.

Motivation – Preliminary Results

1. Experimental Methodology

Usecase: Company with entire cloud-edge infrastructure deployed in Spain.

Location	Carbon Intesity
Spain ES	206 gCO2eq/kWh
Sweden SE	20 gCO2eq/kWh

The lower the better

Goal: Quantify the additional **cost ($)** to rent resources in Sweden to reduce the **carbon footprint**.

2. Experimental details

Applications (using the Microservices benchmark **DeathStarBench**)

Social Network	Media streaming
24 Microservices	32 Microservices
Users send requests to compose posts.	Movie platform where users can log in and upload movie reviews.

Workload

10 minutes

- 1,000 requests to each application
- Time steps follow a Poisson distribution, emulating multiple concurrent users

Motivation – Preliminary Results

1. Composing and uploading a movie review is **more computationally demanding** than creating a social media post.

Application	AVG Latency
Social Network	9.49 ms
Media Streaming	26.08 ms

2.89x

~10x ~2x

	App (Location)	Carbon (mgCO2eq)	Local ($/hr)*
ES	Social Network (Spain)	72.72	0.0912
SE	Social Network (Sweden)	7.06	0.0864
ES	Media Streaming (Spain)	166.17	0.0456
SE	Media Streaming (Sweden)	16.13	0.432

2. Running the applications in Sweden, is a much more **sustainable** solution.

*3. Hosting the media streaming in Sweden will lead to a **higher impact** in sustainability.*

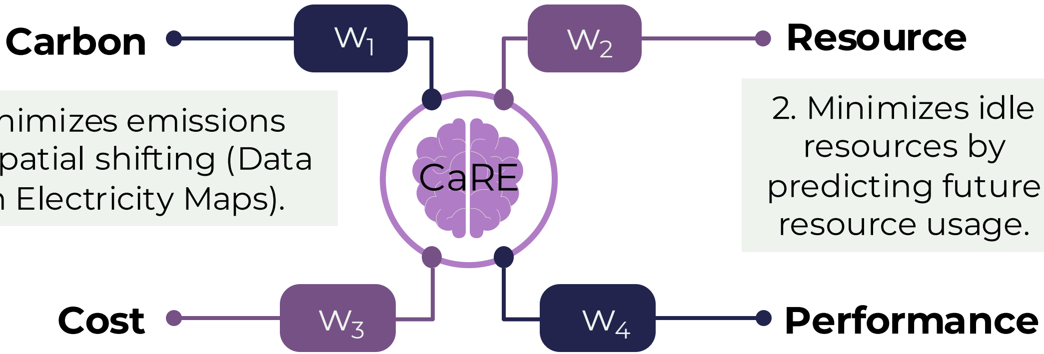
4. **Double the budget** is needed for similar infrastructure in a different country. Users from Spain will connect first to the closest DC → the application runs on both locations.

Takeaway: Become **greener** → More **money**. Choose wisely what to offload!

We need an **application-specific solution** for the **carbon – cost trade-off.**

***Source: Amazon EC2 On-Demand Pricing**. Hourly rate in the eu-south-2 region for Spain, eu-north-1 region for Sweden.

CaRE: A Carbon and Resource Efficient Orchestrator for the Cloud-Edge Continuum



1. Minimizes emissions with spatial shifting (Data from Electricity Maps).

2. Minimizes idle resources by predicting future resource usage.

3. Minimizes the overall cost of the infrastructure (Data from On-Demand Plans for Amazon EC2).

4. Minimizes network and request execution latency, enforcing the SLAs.

CaRE **prioritizes** the optimization metrics according to the **specific application requirements** and the user preferences.

Current Application: **Microservices**

Future Work: Extend to **serverless** applications.

Takeaway: CaRE jointly optimizes the **carbon**, **resource** and **cost** efficiency of the workloads, complying with **SLAs**.

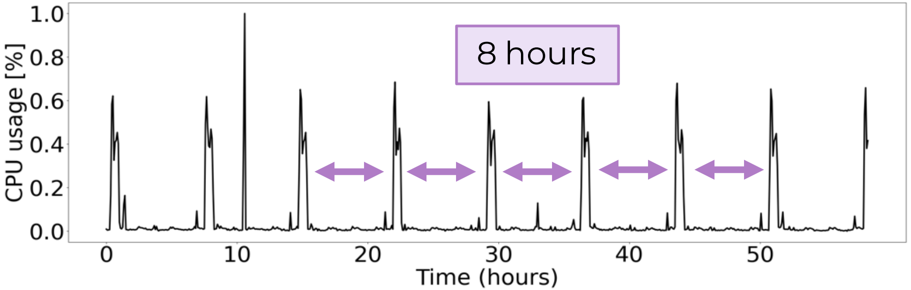
Challenge – Accurate Resource Usage Prediction

1. Proposed Approach: **Persistent Forecast**.

Assume **resource usage repeats itself periodically**.

User behaviours follow predictable cycles.

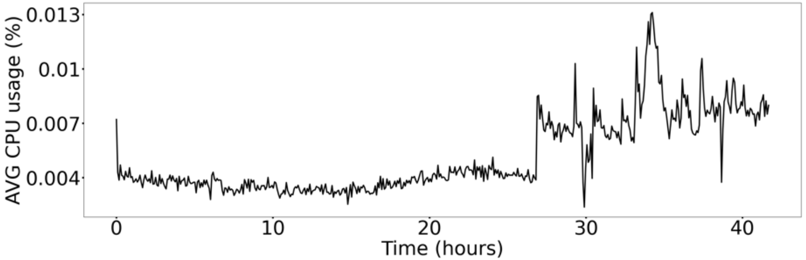
Cloud data is **highly correlated** in time.



Highly **accurate** on cloud data with average prediction error 7%. *

* Is Machine Learning Necessary for Cloud Resource Usage Forecasting? SoCC '23. G. Christofidi, K. Papaioannou, T. D. Doudali.

2. **Limitations** of the Persistent Forecast – **hard to predict patterns**.



Resource utilization is often **unpredictable**, even when everything is running correctly.

When unexpected usage occurs:
- Lower resource efficiency.
- Potential resource contention.
- Higher carbon footprint.

We deploy **anomaly detection techniques**, to predict highly dynamic resource usage.

Proposed Approach for Prediction

3. Handling **Anomalies** with a **Two-Model Approach**.

When the persistent forecast accuracy drops below a minimum accuracy threshold, we enter an **anomalous state**.

Fallback Mechanism that predicts:

- **Duration** of the anomaly.
- **Resource usage** during this time.

Primary model

Persistent Forecast:
✓ Overheads

Fallback Mechanism

Anomaly detection model:
✓ Accuracy
✓ Performance

For the **anomaly detection model** we will explore a variety of ML and non-ML methods commonly used for anomaly detection.

CaRE: A Carbon and Resource Efficient Orchestrator for the Cloud-Edge Continuum