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

Georgia Christofidi
Telefónica Research, Universidad Politécnica de Madrid

Francisco Álvarez Terribas
Telefónica Research

Jesus Alberto Omaña Iglesias
Telefónica Research

Nicolas Kourtellis
Telefónica Research

Thaleia Dimitra Doudali
IMDEA Software Institute

***Abstract*—The growing demand for low-latency and high-performance applications has resulted in the development of the cloud-edge continuum, where workloads are dynamically managed across cloud and edge resources to balance performance and resource efficiency. However, the expansion of the infrastructure to edge nodes introduces significant carbon emissions, largely due to the absence of energy-efficient optimizations, which are traditionally available in cloud data centers. Although recent advancements in carbon-aware resource orchestration propose shifting workloads to regions with lower carbon intensity, these solutions assume globally distributed infrastructures, which may not be the case for companies operating within a single country. In this paper, we investigate the carbon and resource cost implications of deploying microservice applications across geographically distinct regions. Our findings reveal a trade-off between carbon and cost efficiency that emphasizes the need for intelligent, selective microservice migration within the cloud-edge continuum. Future work will focus on developing CaRE, a Carbon and Resource Efficient orchestrator for optimizing workload placement, balancing sustainability, resource costs, and performance across heterogeneous cloud-edge infrastructures.**

I. Introduction

Recent advances in computing, such as microservices and machine learning workloads, have driven demand for responsive computation near users, introducing the cloud-edge continuum [1], [2]. This approach integrates cloud and edge resources, with edge computing enabling application deployment closer to end users to reduce latency and enhance real-time performance [1], [3]–[5]. However, edge environments face challenges, such as limited resources, device heterogeneity, and unreliable networks, and require new workload management approaches [6], [7]. Various resource orchestrators have been developed to manage workloads across this continuum, meeting QoS requirements, such as latency, throughput, and availability, through advanced scheduling and resource allocation techniques [1], [4], [6], [8], [9]. Many approaches focus on resource-aware strategies that predict future resource usage patterns to enhance resource utilization and system performance, while minimizing resource waste [1], [10]–[16]. Furthermore, anomaly detection techniques further enhance performance by identifying and addressing deviations from normal system behavior [17]–[19].

Work done while at Telefonica Research. Georgia Christofidi is currently a PhD student at IMDEA Software Institute and the Universidad Politécnica de Madrid and Nicolas Kourtellis is at Keysight.

The importance of carbon. The expansion of edge computing increases its environmental impact, as distributed edge data centers consume significant energy and contribute to carbon emissions. Unlike large cloud centers, smaller data centers often lack energy-saving optimizations, like liquid or location-based cooling, making carbon-conscious strategies essential. Recent carbon-aware solutions [20]–[24], such as spatial and temporal shifting, reduce emissions by adjusting workloads based on the carbon intensity of electricity generation. Spatial shifting moves tasks to regions with greener energy sources, while temporal shifting delays non-urgent workloads until energy grids are greener. These methods enable sustainable cloud-edge operations without sacrificing performance.

Combining carbon and resource awareness. Existing carbon-aware optimizations make the bold assumption that all cloud-edge infrastructures span across many countries with varying carbon intensity. This does not apply to national companies whose infrastructure resides within one country. To reduce their carbon footprint, such companies would need to buy or rent resources in greener regions, to implement spatial and temporal shifting. This approach increases costs ($) and resources deployment complexity, which may limit the effectiveness of current orchestration solutions. Therefore, carbon awareness does not necessarily result in resource efficiency, as it raises deployment costs and resource needs.

In this paper, we provide preliminary experiments that capture the additional costs for existing microservice applications to run in greener locations, where the total carbon emissions are significantly decreased. Our observations show that the costs of moving entire applications are not trivial, motivating the need to have a novel cloud-edge resource orchestrator that will make an intelligent selection of which microservices to be migrated, to allow for reduced carbon emissions, high resource efficiency and reduced operational costs, all at the same time.

II. Preliminary Results

As a motivational experiment, we assume that a company deploys its entire cloud-edge infrastructure in Spain [25], a country with a moderate carbon intensity of 206 grams of CO2 equivalent per kilowatt-hour (gC02eq/kWh), utilizing 68% low-carbon and 48% renewable energy sources. Recent works [20]–[23] suggest offloading workloads to 'greener' locations, such as Sweden [26], where carbon intensity is only

20 gC02eq/kWh, with 100% low-carbon and 71% renewable energy sources. Our experiment aims to capture the difference in the total carbon footprint and the cost ($) of deploying a workload in these 2 countries, as well as the additional cost ($) needed for moving the workload from Spain to Sweden.

Performance of microservice applications. We experiment with microservices, using the DeathStarBench [27] benchmark, which exemplifies a complex application that could be deployed in the cloud-edge continuum. In particular, we deploy the social network application with 24 microservices, where users send requests to compose posts. Also, we deploy the media streaming application with 32 microservices that implements a movie platform where users can log in and upload movie reviews. Every application runs on a separate node, with Kubernetes as a cluster manager and Prometheus for monitoring. We deploy a workload that sends 1,000 requests to each application at time steps that follow a Poisson distribution, emulating multiple concurrent users for a duration of 10 minutes, as also done in Sinan [28]. Table I summarizes the performance results of the 2 different microservice applications. We report the average, P95 and P99 latency (as collected by the Jaeger Tracing Platform [29]), as well as the storage requirements for the containers and the memory footprint of the workload. We observe that the media streaming application has $2.89\times$ higher average latency than the social network one, while requiring less memory and similar storage capacity. The performance difference arises because composing and uploading a movie review is more computationally demanding than creating a social media post.

Carbon footprint. Next, we capture the difference of running the applications in the two distinct countries mentioned above. Table II captures the total carbon emissions measured in milligrams of CO2 equivalent (mgC02eq). To calculate this number, we aggregate the total energy consumption during the 10-minute duration of the workload, collected via the Prometheus monitoring platform, and multiply it with the country's total carbon intensity, as suggested in [30]. We observe that running the applications in Sweden, makes for a much more environmentally sustainable solution. In particular, there is an order of magnitude difference in carbon emissions, which aligns with the difference in the carbon intensity of the two countries, as mentioned in the beginning of this section. Finally, we see that the media streaming application generates more than $2\times$ higher carbon emissions than the social network, due to its longer latency, as reported in Table I. Thus, hosting the media streaming application in Sweden will lead to a higher impact in environmental sustainability.

Resource cost. To realize a greener deployment, a company located in Spain needs to expand its infrastructure with additional resources in Sweden, leading to additional resource costs. To quantify those, we use the Amazon EC2 On-Demand Pricing [31] and consider the `t3.large` instance with 8GB of memory and 2 CPU cores for the social network application and the `t3.medium` instance with 4GB of memory and 2 CPU cores for the media streaming application, to accommodate the difference in memory footprint, as shown in Table I. Table II reports the on-demand hourly rate in the `eu-south-2` region for Spain vs. the `eu-north-1` region for Sweden. In addition, we calculate the cost of moving the entire application from Spain to Sweden by multiplying the storage size with the data transfer cost per GB from Spain to Sweden in AWS [31]. We observe, that deploying both applications in Sweden requires a cost similar to deploying them in Spain. Thus, if an application is moved within Europe, then double the budget is needed to acquire similar infrastructure in a different country, because the application needs to run on both locations, as users that are in Spain will always need to connect first to the closest datacenter. Also, there is additional cost associated for the actual migration, which is proportional to the application's storage needs. Finally, we see that the cost for deploying the media streaming application in Sweden is half of that of the social network one.

Application	AVG Lat	P95/P99 Lat	Storage	Memory
SocialNet	9.49 ms	42.12/89.16 ms	0.85 GB	5.15 GB
MediaStream	26.08 ms	70.65/127.52 ms	0.75 GB	3.71 GB

TABLE I: Application performance regardless of location.

App(Location)	Carbon (mgCO2eq)	Local($/hr) + Move ($)
SocialNet(**ES**)	72.72	0.0912 + 0.02
SocialNet(**SWE**)	7.06	0.0864 + 0.02
MediaStream(**ES**)	166.17	0.0456 + 0.02
MediaStream(**SWE**)	16.13	0.0432 + 0.02

TABLE II: Comparison of carbon and local operational cost per location, plus the cost of moving the app from ES to SWE.

III. Summary and Future Work

The experiment above indicates significant cost associated with expanding a cloud-edge infrastructure across countries to make for an environmentally sustainable deployment. Moving the entire microservice application to a greener location, although significantly reduces the overall carbon footprint, comes with substantial operational and migration costs. Our preliminary results show that the application performance and resource needs directly influence the amount of resource costs of infrastructure expansion and the level of carbon emission reduction. Therefore, there needs to be a more clever deployment of *part* of the microservices in greener locations, in a way that strikes a balance across carbon and resource efficiency, while minimizing infrastructure expansion cost ($).

Future work will design and build CaREful (Carbon and Resource Efficient) resource orchestration, that minimizes the additional costs associated with expanding operations to greener locations. This novel resource orchestrator for the cloud-edge continuum will selectively and dynamically migrate subgroups of microservices across applications to locations with lower carbon intensity, leveraging new carbon-aware solutions on spatial and temporal shifting and novel resource-aware solutions that accurately predict future resource usage and detect anomalous workload behaviors. This is a particularly challenging problem given the heterogeneity of the cloud-edge resources, the diversity of the microservice applications and the multi-objective problem of optimizing alongside both carbon, resource and cost efficiency.

References

[1] K. Fu, W. Zhang, Q. Chen, D. Zeng, X. Peng, W. Zheng, and M. Guo, "Qos-aware and resource efficient microservice deployment in cloud-edge continuum," in *2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 2021, pp. 932–941.

[2] L. Gu, D. Zeng, J. Hu, B. Li, and H. Jin, "Layer aware microservice placement and request scheduling at the edge," 05 2021, pp. 1–9.

[3] D. Kimovski, R. Mathá, J. Hammer, N. Mehran, H. Hellwagner, and R. Prodan, "Cloud, fog, or edge: Where to compute?" *IEEE Internet Computing*, vol. 25, no. 4, pp. 30–36, 2021.

[4] N. Filinis, I. Tzanettis, D. Spatharakis, E. Fotopoulou, I. Dimolitsas, A. Zafeiropoulos, C. Vassilakis, and S. Papavassiliou, "Intent-driven orchestration of serverless applications in the computing continuum," *Future Generation Computer Systems*, vol. 154, pp. 72–86, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X23004910

[5] K. Ray and A. Banerjee, "Horizontal auto-scaling for multi-access edge computing using safe reinforcement learning," *ACM Trans. Embed. Comput. Syst.*, vol. 20, no. 6, Oct. 2021. [Online]. Available: https://doi.org/10.1145/3475991

[6] G. Bartolomeo, M. Yosofie, S. Bäurle, O. Haluszczynski, N. Mohan, and J. Ott, "Oakestra: A lightweight hierarchical orchestration framework for edge computing," in *2023 USENIX Annual Technical Conference (USENIX ATC 23)*. Boston, MA: USENIX Association, Jul. 2023, pp. 215–231. [Online]. Available: https://www.usenix.org/conference/atc23/presentation/bartolomeo

[7] P. Parastar, G. Caso, J. A. O. Iglesias, A. Lutu, and O. Alay, "Rethinking the mobile edge for vehicular services," *Computer Networks*, vol. 253, p. 110687, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S138912862400519X

[8] E. Saurez, H. Gupta, A. Daglis, and U. Ramachandran, "Oneedge: An efficient control plane for geo-distributed infrastructures," in *Proceedings of the ACM Symposium on Cloud Computing*, ser. SoCC '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 182–196. [Online]. Available: https://doi.org/10.1145/3472883.3487008

[9] S. Nastic, T. Pusztai, A. Morichetta, V. C. Pujol, S. Dustdar, D. Vii, and Y. Xiong, "Polaris scheduler: Edge sensitive and slo aware workload scheduling in cloud-edge-iot clusters," in *2021 IEEE 14th International Conference on Cloud Computing (CLOUD)*, 2021, pp. 206–216.

[10] G. Christofidi, K. Papaioannou, and T. D. Doudali, "Is machine learning necessary for cloud resource usage forecasting?" in *Proceedings of the 2023 ACM Symposium on Cloud Computing*, ser. SoCC '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 544–554. [Online]. Available: https://doi.org/10.1145/3620678.3624790

[11] G. Christofidi and T. D. Doudali, "Do predictors for resource overcommitment even predict?" in *Proceedings of the 4th Workshop on Machine Learning and Systems*, ser. EuroMLSys '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 153–160. [Online]. Available: https://doi.org/10.1145/3642970.3655838

[12] G. Christofidi, K. Papaioannou, and T. D. Doudali, "Toward pattern-based model selection for cloud resource forecasting," in *Proceedings of the 3rd Workshop on Machine Learning and Systems*, ser. EuroMLSys '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 115–122. [Online]. Available: https://doi.org/10.1145/3578356.3592588

[13] A. S. Prasad, D. Koll, J. O. Iglesias, J. A. Aroca, V. Hilt, and X. Fu, "Rconf(pd): Automated resource configuration of complex services in the cloud," *Future Generation Computer Systems*, vol. 87, pp. 639–650, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X17314231

[14] Y. Guo, J. Ge, P. Guo, Y. Chai, T. Li, M. Shi, Y. Tu, and J. Ouyang, "Pass: Predictive auto-scaling system for large-scale enterprise web applications," in *Proceedings of the ACM Web Conference 2024*, ser. WWW '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 2747–2758. [Online]. Available: https://doi.org/10.1145/3589334.3645330

[15] B. Wang, D. Irwin, P. Shenoy, and D. Towsley, "Invar: Inversion aware resource provisioning and workload scheduling for edge computing," in *IEEE INFOCOM 2024 - IEEE Conference on Computer Communications*, 2024, pp. 1511–1520.

[16] B. Shayesteh, C. Fu, A. Ebrahimzadeh, and R. Glitho, "Adaptive feature selection for predicting application performance degradation in edge cloud environments," *IEEE Transactions on Network and Service Management*, vol. PP, pp. 1–1, 01 2024.

[17] D. Ohana, B. Wassermann, N. Dupuis, E. Kolodner, E. Raichstein, and M. Malka, "Hybrid anomaly detection and prioritization for network logs at cloud scale," in *Proceedings of the Seventeenth European Conference on Computer Systems*, ser. EuroSys '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 236–250. [Online]. Available: https://doi.org/10.1145/3492321.3519566

[18] L. Li, X. Zhang, X. Zhao, H. Zhang, Y. Kang, P. Zhao, B. Qiao, S. He, P. Lee, J. Sun, F. Gao, L. Yang, Q. Lin, S. Rajmohan, Z. Xu, and D. Zhang, "Fighting the fog of war: Automated incident detection for cloud systems," in *2021 USENIX Annual Technical Conference (USENIX ATC 21)*. USENIX Association, Jul. 2021, pp. 131–146. [Online]. Available: https://www.usenix.org/conference/atc21/presentation/li-liqun

[19] H. Nizam, D.-S. Zafar, Z. Lv, F. Wang, and X. Hu, "Real-time deep anomaly detection framework for multivariate time-series data in industrial iot," *IEEE Sensors Journal*, vol. PP, pp. 1–1, 12 2022.

[20] A. Lechowicz, N. Christianson, J. Zuo, N. Bashir, M. Hajiesmaili, A. Wierman, and P. Shenoy, "The online pause and resume problem: Optimal algorithms and an application to carbon-aware load shifting," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 7, no. 3, Dec. 2023. [Online]. Available: https://doi.org/10.1145/3626776

[21] W. A. Hanafy, Q. Liang, N. Bashir, A. Souza, D. Irwin, and P. Shenoy, "Going green for less green: Optimizing the cost of reducing cloud carbon emissions," in *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3*, ser. ASPLOS '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 479–496. [Online]. Available: https://doi.org/10.1145/3620666.3651374

[22] T. Sukprasert, A. Souza, N. Bashir, D. Irwin, and P. Shenoy, "On the limitations of carbon-aware temporal and spatial workload shifting in the cloud," in *Proceedings of the Nineteenth European Conference on Computer Systems*, ser. EuroSys '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 924–941. [Online]. Available: https://doi.org/10.1145/3627703.3650079

[23] W. A. Hanafy, Q. Liang, N. Bashir, D. Irwin, and P. Shenoy, "Carbonscaler: Leveraging cloud workload elasticity for optimizing carbon-efficiency," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 7, no. 3, Dec. 2023. [Online]. Available: https://doi.org/10.1145/3626788

[24] V. Gsteiger, P. H. D. Long, Y. J. Sun, P. Javanrood, and M. Shahrad, "Caribou: Fine-grained geospatial shifting of serverless applications for sustainability," in *The 30th ACM Symposium on Operating Systems Principles (SOSP'24). ACM*, 2024.

[25] "Electricity maps - spain," https://app.electricitymaps.com/zone/ES.

[26] "Electricity maps - sweden," https://app.electricitymaps.com/zone/SE.

[27] Y. Gan, Y. Zhang, D. Cheng, A. Shetty, P. Rathi, N. Katarki, A. Bruno, J. Hu, B. Ritchken, B. Jackson, K. Hu, M. Pancholi, Y. He, B. Clancy, C. Colen, F. Wen, C. Leung, S. Wang, L. Zaruvinsky, M. Espinosa, R. Lin, Z. Liu, J. Padilla, and C. Delimitrou, "An open-source benchmark suite for microservices and their hardware-software implications for cloud & edge systems," in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, ser. ASPLOS '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 3–18. [Online]. Available: https://doi.org/10.1145/3297858.3304013

[28] Y. Zhang, W. Hua, Z. Zhou, G. E. Suh, and C. Delimitrou, "Sinan: Ml-based and qos-aware resource management for cloud microservices," in *Proceedings of the 26th ACM international conference on architectural support for programming languages and operating systems*, 2021, pp. 167–181.

[29] "Jaeger tracing," https://www.jaegertracing.io.

[30] L. Lannelongue, J. Grealey, and M. Inouye, "Green algorithms: quantifying the carbon footprint of computation," *Advanced science*, vol. 8, no. 12, p. 2100707, 2021.

[31] "Aws pricing on-demand," https://aws.amazon.com/ec2/pricing/on-demand/.