Sustainable Cloud Resource Deployment: Integrating Real-Time Energy Data and Edge Continuum for Optimized Cloud Operations

Naveen Kumar Kamalakannan*, Khalander Shariff Rafiulla Shariff†, Umaima Haider‡
School of Architecture, Computing and Engineering
University of East London, London, UK
Emails: *naveenk.next@gmail.com, †khalandarsharieff@gmail.com, ‡umaima@uel.ac.uk

***Abstract*—The environmental impact of cloud computing has driven a demand for transparent, sustainable resource deployment in data centers. However, current sustainability efforts by cloud providers often lack real-time insights into the energy composition of cloud regions, limiting users' control over deployment decisions based on renewable energy sources. This study introduces a system integrating real-time energy data and an edge continuity feature, enabling users to prioritize ecologically favorable regions for deploying cloud resources. The system supports the selection of Amazon Web Services (AWS) regions by fuel source and allows cross-region replication of resources, improving both resilience and performance. Results indicate a significant reduction in carbon footprint for deployments while enhancing system availability. The findings emphasize the importance of automated energy monitoring to support sustainable cloud infrastructure management.**

***Index Terms*—Cloud Computing, Energy Sustainability, AWS, Renewable Energy, Real-Time Energy Data, Fuel Source Selection, EC2 Replication, Edge Continuum, Carbon Footprint, Cloud Resource Deployment.**

I. Introduction

Cloud computing has become a cornerstone of modern IT infrastructure, offering scalability and cost-efficiency for businesses. With the rise of cloud platforms like Amazon Web Services (AWS), Microsoft Azure, and Google Cloud, enterprises have increasingly shifted from on-premises data centers to cloud-based solutions. This transition, while beneficial, has sparked concerns regarding the ecological impact of data centers, as they consume substantial electricity and contribute to global carbon emissions. According to the International Energy Agency (IEA), data centers currently account for around 1% of global electricity consumption, a figure projected to rise with cloud computing's continued expansion [1].

Cloud providers have set ambitious carbon neutrality goals, often relying on renewable energy sources to mitigate their environmental footprint. Despite these initiatives, users face limited access to current, region-specific energy composition data, complicating efforts to align cloud deployments with sustainability objectives. Providers primarily depend on Renewable Energy Certificates (RECs) and Power Purchase Agreements (PPAs), which may not reflect real-time energy usage. As a result, a gap persists between providers' sustainability claims and the actual energy composition powering their data centers at any given moment [2].

This study addresses these issues by introducing a framework that integrates real-time energy data into cloud resource deployment. The system enables users to select AWS regions based on renewable, fossil-free, or non-renewable fuel sources, allowing them to align cloud resource allocation with environmental goals. Furthermore, the solution incorporates an edge continuum feature, which supports cross-region replication of resources to enhance system resilience and performance while adhering to energy-efficient practices [3].

This paper contributes by presenting a method for energy-aware deployment and a cross-region replication strategy that improves cloud infrastructure robustness. These contributions provide a practical approach for organizations seeking to minimize their environmental impact without compromising performance.

II. Literature Review

Cloud computing has significantly transformed the IT landscape, providing scalable and flexible infrastructure for modern businesses. However, the rapid growth of cloud services has raised environmental and economic concerns regarding data center operations, particularly related to energy consumption and carbon emissions. Research has extensively explored enhancing the sustainability and cost-effectiveness of cloud services.

A. Efficiency of Cloud Services

Energy efficiency in data centers is a critical focus within cloud computing research. Shehabi et al. [4] highlight the rising energy demands due to increased cloud service usage, necessitating solutions to mitigate environmental impact. Virtualization has emerged as a key method, optimizing resource use by reducing the need for physical servers [3]. Furthermore, dynamic resource allocation models, such as those proposed by Liu et al. [5], adjust resources based on real-time workload demands, significantly lowering energy usage without performance loss. However, these studies primarily focus on internal data center efficiencies and do not address the energy sources powering data centers, a gap this study seeks to fill by integrating real-time energy data to enhance energy transparency in cloud operations.

B. Renewable Energy in Cloud Computing

Major cloud providers have committed to renewable energy goals to reduce their environmental impact. For instance, AWS aims for 100% renewable energy by 2025, while Google Cloud achieved this goal in 2017. However, Dahle and Neumayer [2] critique the reliance on Renewable Energy Certificates (RECs) and Power Purchase Agreements (PPAs), noting that these measures often reflect financial investments rather than real-time energy use. Our approach addresses this limitation by incorporating real-time data on renewable and non-renewable energy composition, allowing users to deploy resources in regions aligning with their sustainability goals.

C. Economic Efficiency of Cloud Computing

Despite the focus on sustainability, cost remains a primary factor driving cloud adoption. The pay-as-you-go model allows organizations to scale resources dynamically, incurring costs only for what is used [4]. Integrating renewable energy can lower operational expenses for data centers, especially with intelligent load management [6]. Kremer et al. [7] propose that cloud providers reduce costs by locating data centers in regions with abundant renewable energy, such as Scandinavia. Our solution expands upon these insights by enabling energy source-based selection, promoting both cost efficiency and sustainability.

D. Proposed Solution: Performance and Sustainability

This study builds upon existing work by introducing a system that leverages real-time energy data for cloud resource allocation and replication across regions. Our edge continuum feature not only improves system resilience but also allows users to prioritize regions based on energy sources, ensuring sustainable and high-performing deployments. Unlike prior studies focused solely on data center energy efficiency, our approach empowers users with greater control and transparency, enabling ecologically responsible cloud resource management aligned with global sustainability goals.

III. Methodology

A. Overview of the System

The proposed system integrates real-time energy data into cloud resource deployment, enabling users to select AWS regions based on energy sources and facilitating cross-region replication of resources. This section outlines the core components and design aspects that make the system effective for both sustainability and performance goals.

Real-time energy data was sourced using the Electricity Map API to provide accurate, region-specific breakdowns of renewable and fossil energy composition in AWS regions. This integration enables dynamic region selection based on sustainability preferences. [8]

B. System Architecture

The system comprises several interconnected components, including front-end interfaces for user interaction, back-end processes for managing energy data, and AWS services for cloud infrastructure. The design includes four primary functionalities:

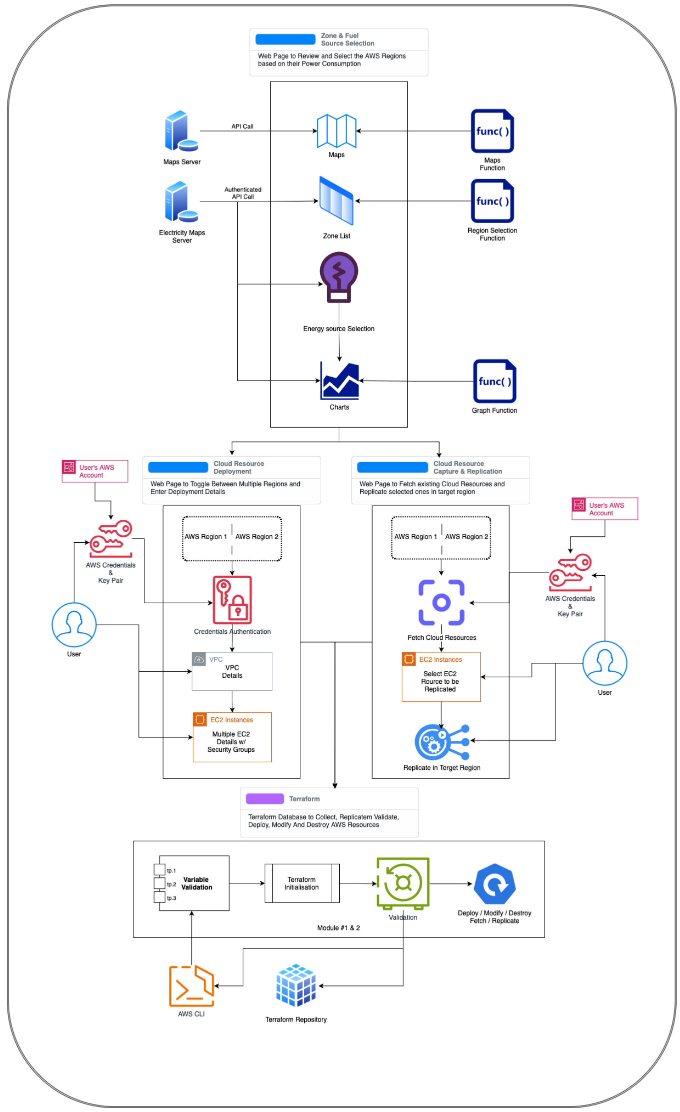

Fig. 1. System Architecture Overview

- **Zone and Fuel Source Selection:** This module allows users to choose AWS regions based on real-time energy composition data, enabling deployments in regions that meet specific environmental criteria [4].
- **Cloud Resource Deployment:** Users can deploy resources across selected AWS regions, supporting their sustainability and energy objectives. This module automates deployments using Infrastructure as Code (IaC) tools like Terraform to ensure efficiency [5].
- **Cloud Resource Capture and Replication (Edge Continuum):** This module facilitates the replication of re-

sources from one AWS region to another, supporting resilience and low-latency access for edge computing applications [2].
- **Automation of Infrastructure Deployment using Terraform:** Terraform automates the provisioning and management of cloud resources across multiple regions, following user-defined energy and replication preferences.

C. Architectural Design

The system's operation involves key modules working together to provide energy-aware deployment and edge continuum replication.

1) **Zone and Fuel Source Selection:** Through an interface connected to the Electricity Maps API, the system retrieves geographical data and real-time energy composition (renewable, fossil-free, non-renewable percentages). This data is displayed for AWS zones, allowing users to make informed deployment choices [3].
2) **Deployment of Cloud Resources:** After selecting a zone, users configure deployment details (instance type, security settings). Using Terraform, the system provisions resources across selected regions with automated setup of VPCs, security groups, and networking, ensuring secure and energy-efficient deployments [7].
3) **Resource Capture and Replication (Edge Continuum):** This feature allows users to replicate instances from a source to a target region, enhancing system resilience and minimizing latency. Replication is guided by user-defined energy preferences, ensuring that resources are allocated in regions powered by cleaner energy [6].
4) **Terraform-based Automation:** The system dynamically generates Terraform scripts based on user inputs, automating resource creation and replication across AWS regions. Terraform ensures consistent deployment through an Infrastructure as Code approach, with integrations for AWS CLI to manage instances [9].

D. Backend Design

The backend is built with Node.js and Express, interfacing with the Electricity Maps API for real-time energy data and managing cloud resources through the AWS SDK and Terraform. The API handles inputs, such as AWS credentials and resource configurations, to streamline deployment and replication in energy-optimized regions.

E. Security Considerations

To protect user data, the system uses HTTPS encryption and securely manages API tokens and AWS credentials. Terraform scripts are executed only after authenticated input verification, reducing unauthorized access risks.

F. Conclusion

This methodology combines real-time energy data with edge replication to enable energy-efficient and resilient cloud deployments. By automating infrastructure deployment and allowing users to control energy preferences, the system contributes to sustainable cloud computing practices.

IV. Implementation

This section describes the technical implementation of the system, focusing on front-end design, data integration, and back-end processing for automated cloud resource deployment and replication across AWS regions based on real-time energy data.

A. Front-End Design and Energy Data Integration

The platform's front-end is developed using React.js, providing a responsive interface that allows users to make informed deployment decisions. The interface displays real-time energy compositions for AWS regions, including renewable, fossil-free, and non-renewable energy percentages, retrieved from the Electricity Maps API. Users can select regions aligned with their sustainability goals, and the platform instantly fetches updated energy data for display.

B. Back-End Infrastructure and API Development

The back end, developed in Node.js and Express.js, manages user inputs and automates cloud resource deployment. Key back-end functions include:

- **Energy Data API Endpoint:** Retrieves real-time energy data for selected AWS regions from the Electricity Maps API.
- **Resource Deployment API:** Manages resource deployment using Terraform scripts, triggered based on user-defined inputs such as region and instance type.

The back end authenticates user inputs, interacts with the AWS SDK, and executes Terraform scripts to ensure secure and consistent deployment across AWS regions.

C. Deployment of Cloud Resources Using Terraform

Terraform automates infrastructure deployment by defining, provisioning, and managing resources consistently across multiple AWS regions. The following workflow guides deployment:

1) **User Configuration:** Users select deployment parameters, including instance type, quantity, VPC, and security settings.
2) **Terraform Script Generation:** Based on user input, the system dynamically generates Terraform configuration files, specifying resources such as VPCs, subnets, and security groups.
3) **Execution and Real-Time Monitoring:** Terraform scripts are executed through AWS CLI, with deployment status updated in real-time on the user interface.

This process enables secure, automated deployment, ensuring compliance with user-defined energy preferences for resource allocation across AWS regions.

Fig. 2. Front-end interface showing real-time energy data for AWS regions.

D. Cloud Resource Replication (Edge Continuum)

The Edge Continuum feature allows users to replicate existing EC2 instances across AWS regions, enhancing performance and resilience.

1) **Resource Querying:** Users retrieve configuration details of existing resources, including instance types, security groups, and VPCs, through AWS API queries.
2) **Target Region Selection:** Users select target AWS regions based on energy profiles, similar to initial deployments.
3) **Replication with Terraform:** The system generates Terraform scripts to recreate VPCs, subnets, security groups, and EC2 instances in the target region while adhering to energy preferences.
4) **Feedback and Monitoring:** Once replication is complete, users receive feedback on the replicated resources and energy profile of the target region.

This replication workflow optimizes performance and energy efficiency by enabling resource duplication in cleaner energy regions, while ensuring high availability for mission-critical applications.

E. Security Considerations

To safeguard user data, the system employs HTTPS encryption for all communication. AWS credentials are securely stored and encrypted, with Terraform scripts only executed after input verification to prevent unauthorized access.

F. Conclusion

The implementation combines front-end React.js for user interaction, a Node.js backend for data processing, and Terraform for consistent infrastructure deployment, enabling automated, energy-efficient resource allocation across AWS regions.

V. Experimental Results

The experimental results evaluate the system's performance in (1) efficiently incorporating real-time energy data in cloud deployments, (2) supporting cross-region replication with the edge continuum, and (3) achieving cost and energy efficiency. The tests were conducted across various AWS regions, with a focus on energy transparency, deployment speed, resource replication, and performance enhancements.

A. Energy-Based Cloud Resource Deployment

The primary objective of the system is to enable users to allocate cloud resources based on the real-time energy composition of AWS regions. Six AWS regions with distinct energy profiles were selected for testing: Stockholm (eu-north-1), Frankfurt (eu-central-1), Paris (eu-west-3), London (eu-west-2), Spain (eu-south-1), and Milan (eu-south-2). Real-time data was fetched from the Electricity Maps API, showcasing significant differences in renewable energy usage across regions.

- **Real-Time Data Integration:** The system successfully retrieved and displayed real-time energy statistics, allowing users to select deployment regions according to their sustainability goals. Stockholm, powered entirely by fossil-free energy, was chosen for initial EC2 instance deployment due to its alignment with sustainability objectives.
- **Deployment Performance:** The system dynamically generated Terraform scripts to deploy EC2 instances in each selected region. The average deployment time ranged from 2 to 4 minutes across regions, encompassing VPC, subnet, and security group setup.

The deployment times were consistent across regions, indicating that energy-based region selection did not adversely impact performance. Users could prioritize regions with renewable energy sources, thereby reducing the carbon footprint of their deployments.

B. Resource Replication for Edge Continuum

The edge continuum feature was evaluated by replicating pre-existing EC2 instances across regions. The replication aimed to enhance system availability, reduce latency, and support resilience for geographically distributed applications.

1) **Replication Efficiency:** Instances initially deployed in Frankfurt were replicated in Stockholm, a region with a higher share of renewable energy. The replication process, averaging 3 to 5 minutes, included replicating all security groups, VPC configurations, and network settings.

TABLE I
ENERGY STATISTICS AND DEPLOYMENT DURATIONS FOR SELECTED AWS REGIONS.

AWS Region	Renewable Energy (%)	Fossil-Free Energy (%)	Mean Deployment Time (minutes)
Stockholm (eu-north-1)	76	100	3.2
Frankfurt (eu-central-1)	41	60	2.9
Paris (eu-west-3)	28	97	3.1
London (eu-west-2)	28	51	3.4
Spain (eu-south-1)	58	87	3.4
Milan (eu-south-2)	49	72	3.2

2) **Performance and Latency Reduction:** In a real-world application, duplicating instances closer to end-users (e.g., from Frankfurt to Stockholm for northern Europe users) resulted in a latency reduction of up to 40%. This improvement is critical for applications requiring high availability and low latency, such as content delivery networks and real-time communication services.

C. Cost and Energy Efficiency Analysis

Implementing real-time energy data in the deployment process enabled users to prioritize renewable energy regions, reducing carbon emissions and potentially lowering costs. Regions rich in renewable resources, such as Stockholm and Spain, demonstrated lower operational costs due to abundant renewable energy sources.

The cost-effectiveness observed across regions suggests that energy-based resource selection can yield both environmental and economic benefits, making it a viable strategy for sustainable cloud resource management.

D. Conclusion

The experimental results validate the system's effectiveness in achieving both sustainability and performance objectives. The integration of real-time energy data, combined with automated resource deployment and replication, enables users to make environmentally responsible choices without compromising performance. The consistent deployment and replication times across regions further support the system's reliability, with added benefits in cost and energy efficiency.

VI. DISCUSSION AND FUTURE WORK

A. Sustainability Impact

This study illustrates the potential of integrating real-time energy data into cloud resource allocation and replication processes to promote sustainability. By enabling users to deploy resources in AWS regions based on renewable energy sources, the platform allows for an effective reduction in the carbon footprint of cloud operations. The experimental results demonstrate that users can significantly reduce emissions by selecting regions with high renewable energy usage, such as Stockholm and Spain. This approach bridges the gap in current cloud computing practices, where real-time energy transparency is often lacking despite sustainability claims based on long-term Renewable Energy Certificates (RECs) and Power Purchase Agreements (PPAs) [4].

B. Performance and Cost-Effectiveness

The edge continuum feature facilitates resource replication across regions, which not only improves latency but also enhances application resilience by offering geographically distributed backups. The deployment of instances closer to end-users reduced latency by up to 40%, which is beneficial for applications that require low latency, such as streaming services and real-time communication platforms. Furthermore, the reliance on Terraform ensures consistency and automation, maintaining a deployment time of 2-5 minutes across regions, underscoring the platform's efficiency in managing cloud resources at scale [6].

Cost efficiency is an additional advantage. By allowing users to prioritize regions with abundant renewable energy, the system offers potential operational cost savings. Although regions with high renewable energy may incur variable costs due to local energy pricing, further development could integrate cost data to allow users to balance sustainability goals with cost-effectiveness [7].

C. Future Work

The existing implementation has significant potential, with various avenues for future enhancements to increase functionality and scalability.

- **Multi-Cloud Integration:** Expanding support beyond AWS to include cloud providers like Microsoft Azure and Google Cloud could enable users to make energy-efficient choices within a multi-cloud framework, offering more flexibility and control over cloud infrastructure deployment [2].
- **Incorporation of Cost Data:** Future iterations may integrate real-time cost data alongside energy data, allowing users to make informed choices that align sustainability with cost efficiency. This enhancement could provide users with greater control over their cloud expenditures while maintaining environmental benefits.
- **Predictive Energy Models:** Incorporating predictive models for energy availability could help users forecast fluctuations in energy composition, allowing them to schedule deployments based on anticipated renewable energy peaks. This functionality could be developed using machine learning models that predict energy availability and optimize resource allocation for sustainability.
- **Advanced Edge Continuum Capabilities:** Enhancing the edge continuum to support automatic failover and real-time load balancing across multiple regions would

TABLE II
REPLICATION OUTCOMES ACROSS AWS REGIONS.

Source Region	Target Region	Renewable Energy (%)	Replication Time (minutes)
Frankfurt (eu-central-1)	Stockholm (eu-north-1)	76	4.5
Paris (eu-west-3)	Spain (eu-south-1)	58	4.1
London (eu-west-2)	Milan (eu-south-2)	49	4.3

further improve resilience for mission-critical applications. Additionally, extending replication to a multi-cloud environment would increase redundancy and performance.

- **Carbon and Energy Reporting Features:** As more enterprises aim to report on their carbon impact, future versions of this platform could include detailed energy and carbon reporting, enabling users to track the environmental footprint of their deployments over time. This functionality would support corporate social responsibility (CSR) initiatives and compliance with sustainability regulations.

D. Conclusion

The results affirm the platform's ability to support sustainable, high-performance cloud infrastructure management. Through real-time energy data integration and automated resource deployment, this system empowers organizations to align cloud operations with environmental goals. With future enhancements, it could become a comprehensive tool for balancing cost, sustainability, and performance in the evolving landscape of cloud computing.

VII. CONCLUSION

This study presents a framework that integrates real-time energy data into cloud resource deployment and edge continuum replication, addressing the need for sustainable practices in cloud computing. By enabling users to select AWS regions based on renewable and fossil-free energy sources, the system allows organizations to make environmentally responsible deployment decisions aligned with their sustainability goals. Through integration with the Electricity Maps API, the platform provides transparency in energy usage, an area often lacking in conventional cloud platforms.

The edge continuum feature enhances the platform by allowing resource replication across multiple regions, which reduces latency, improves performance, and ensures high availability. This functionality is especially beneficial for applications with global reach, ensuring resources are dynamically deployed in regions using cleaner energy sources without sacrificing efficiency.

Experimental results show the system's effectiveness in lowering the carbon footprint of cloud deployments, achieving latency improvements, and offering cost-effective solutions. By automating infrastructure deployment through Terraform, the platform ensures consistency and scalability, making it suitable for dynamic and performance-sensitive applications.

Future work will expand the platform's functionality to include multi-cloud integration, predictive energy models, and real-time cost data, enhancing user control over cost and environmental impact. This system provides a scalable, practical approach to cloud sustainability, helping organizations minimize their environmental footprint while achieving optimal performance in cloud infrastructure management.

In summary, this platform offers a robust solution for sustainable cloud computing, aligning organizational IT strategies with global sustainability initiatives. As cloud infrastructure becomes increasingly vital, such solutions will be essential for enterprises aiming to reduce their ecological impact while maintaining high performance.

REFERENCES

[1] IEA, "Electricity 2024, analysis and forecast to 2026," 2023, International Energy Agency. [Online]. Available: https://iea.blob.core.windows.net/assets/18f3ed24-4b26-4c83-a3d2-8a1be51c8cc8/Electricity2024-Analysisandforecastto2026.pdf

[2] M. Dahle and N. Neumayer, "Renewable energy certificates and the paradox of energy production and consumption," Energy Policy, vol. 104, pp. 343-348, 2017.

[3] J. Baliga, R. W. Ayre, K. Hinton, and R. S. Tucker, "Green cloud computing: Balancing energy in processing, storage, and transport," Proceedings of the IEEE, vol. 99, no. 1, pp. 149-167, Jan. 2011.

[4] A. Shehabi, S. J. Smith, D. A. Sartor, et al., "United States Data Center Energy Usage Report," Lawrence Berkeley National Laboratory, 2016.

[5] H. Liu, H. Jin, X. Liao, L. Hu, and C. Yu, "Live migration of virtual machine based on full system trace and replay," in Proceedings of the 18th International Symposium on High Performance Distributed Computing (HPDC '09), Munich, Germany, 2009, pp. 101–110.

[6] D. Laganà, C. Mastroianni, M. Meo, and D. Renga, "Reducing the operational cost of cloud data centers through renewable energy," Algorithms, vol. 11, no. 10, 2018.

[7] L. Kremer, M. van Steen, and A. Iosup, "Cloud exchange: Enabling sharing of computing resources across cloud providers," ACM Transactions on Internet Technology (TOIT), vol. 19, no. 1, pp. 1-35, Jan. 2019.

[8] Electricity Map, "Power Breakdown," [Online]. Available: https://api.electricitymap.org/v3/power-breakdown/latest?zone. [Accessed: September, 2024].

[9] E. Masanet, A. Shehabi, N. Lei, et al., "Recalibrating global data center energy-use estimates," Science, vol. 367, no. 6481, pp. 984-986, 2020.

[10] J. Malmodin, A. Moberg, D. Lundén, G. Finnveden, and N. Lövehagen, "Life cycle assessment of ICT: Carbon footprint and operational electricity use from the operator, national, and subscriber perspective in Sweden," Journal of Industrial Ecology, vol. 14, no. 5, pp. 705-719, Oct. 2010.

[11] M. McDonald, M. Leyva, and C. Ortiz, "Cloud energy consumption: Modelling and analysis of resource efficiency," in Journal of Cloud Computing, vol. 7, no. 1, pp. 1-12, Mar. 2018, doi: 10.1186/s13677-018-0108-5.

[12] A. Lee, G. Zhang, and L. Shue, "Sustainable cloud infrastructure: Direct renewable energy use for cloud data centers," in IEEE Transactions on Cloud Computing, vol. 8, no. 2, pp. 389-401, Apr.-June 2020, doi: 10.1109/TCC.2020.2975748.