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

Walid A. Hanafy
University of Massachusetts Amherst
USA

Qianlin Liang
University of Massachusetts Amherst
USA

Noman Bashir
Massachusetts Institute of Technology
USA

Abel Souza
University of Massachusetts Amherst
USA

David Irwin
University of Massachusetts Amherst
USA

Prashant Shenoy
University of Massachusetts Amherst
USA

Abstract

The continued exponential growth of cloud datacenter capacity has increased awareness of the carbon emissions when executing large compute-intensive workloads. To reduce carbon emissions, cloud users often temporally shift their batch workloads to periods with low carbon intensity. While such time shifting can increase job completion times due to their delayed execution, the cost savings from cloud purchase options, such as reserved instances, also decrease when users operate in a carbon-aware manner. This happens because carbon-aware adjustments change the demand pattern by periodically leaving resources idle, which creates a trade-off between carbon emissions and cost. In this paper, we present `GAIA`, a carbon-aware scheduler that enables users to address the three-way trade-off between carbon, performance, and cost in cloud-based batch schedulers. Our results quantify the carbon-performance-cost trade-off in cloud platforms and show that compared to existing carbon-aware scheduling policies, our proposed policies can double the amount of carbon savings per percentage increase in cost, while decreasing the performance overhead by 26%.

CCS Concepts: • **Computer systems organization** → **Cloud computing**; • **Social and professional topics** → **Sustainability**.

Keywords: Sustainable Computing, Cloud Computing

ACM Reference Format:
Walid A. Hanafy, Qianlin Liang, Noman Bashir, Abel Souza, David Irwin, and Prashant Shenoy. 2024. Going Green for Less Green: Optimizing the Cost of Reducing Cloud Carbon Emissions. In *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3 (ASPLOS '24), April 27-May 1, 2024, La Jolla, CA, USA.* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3620666.3651374

ASPLOS '24, April 27-May 1, 2024, La Jolla, CA, USA

ACM ISBN 979-8-4007-0386-7/24/04.
https://doi.org/10.1145/3620666.3651374

1 Introduction

The increasing demand for computing and accelerating growth in cloud datacenter capacity have long raised concerns about their sustainability, environmental impact, and resulting carbon footprint [23]. Although cloud operators previously addressed such concerns by increasing datacenters' energy efficiency, or work done per unit of energy consumed, via software and hardware optimizations, recent work highlights that continuing improvements to energy efficiency are likely to see diminishing returns [12]. For example, large datacenters already operate near the optimal PUE value of 1.0, so there is little room to further improve PUE. Thus, to continue reducing their carbon emissions, cloud providers are increasingly employing *supply-side* optimizations, such as purchasing renewable energy from solar and wind farms to offset datacenter demand [22, 41]. However, eliminating all carbon emissions using supply-side techniques alone can be very expensive [6, 15].

Researchers have also proposed *demand-side* techniques to decrease computing's operational carbon emissions. These techniques utilize 1) visibility into grid energy's carbon intensity (in g·CO_2eq/kWh) and 2) application-level flexibility to *modulate* demand based on variations in energy's carbon intensity [6, 21, 31, 36, 44]. For example, prior work on Wait Awhile [44] and Ecovisor [35] utilize batch workloads' *temporal flexibility* to optimize carbon by executing jobs when energy's carbon intensity is low and pausing execution when carbon intensity is high. Importantly, while state-of-the-art techniques consider carbon-performance trade-offs, they ignore the cost implications of carbon-aware optimizations. Specifically, carbon-aware scheduling exploits batch jobs' temporal flexibility to delay their execution until energy's carbon intensity is low. However, this delay also increases the completion times of batch jobs. Thus, carbon-aware scheduling exhibits carbon-performance trade-off: decreasing carbon emissions generally results in longer completion times.

In addition to the performance trade-off above, there is also a cost trade-off when using temporal shifting to optimize carbon emissions. This cost trade-off manifests in