Article

Cluster of Single-Board Computers at the Edge for Smart Grids Applications

Safae Bourhnane ^{1,2,*}, Mohamed Riduan Abid ¹ , Khalid Zine-dine ³ , Najib Elkamoun ² and Driss Benhaddou ⁴

¹ School of Science and Engineering, Al Akhawayn University, Ifrane 53000, Morocco; r.abid@aui.ma

² Faculty of Sciences El Jadida, Chouaib Doukkali University, El Jadida 24000, Morocco; elkamoun@ucd.ac.ma

³ Faculty of Sciences, Mohammed V University, BP 8007, Rabat 11000, Morocco; khalid.zinedine@um5.ac.ma

⁴ School of Engineering and Technology, University of Houston, Houston, TX 77024, USA; dbenhadd@Central.UH.EDU

* Correspondence: s.bourhnane@aui.ma; Tel.: +212-61088-0040

Abstract: Increases in power demand and consumption are very noticeable. This increase presents a number of challenges to the traditional grid systems. Thus, there is the need to come up with a new solution that copes with the stringent demand on energy and provides better power quality, which gives a better experience to the end users. This is how the concept of smart grids (SG) came to light. SGs have been introduced to better monitor and control the power produced and consumed. In addition to this, SGs help with reducing the electricity bill through the integration of renewable energy sources. The underlying smartness of the SGs resides in the flow of information in addition to the flow of energy. Information/data flowing implies the use of smart sensors and smart meters that sense and send data about the power produced and consumed, and the data about the environment where they are deployed. This makes SGs a direct application of IoT. In this paper, we are implementing an edge platform that is based on single-board computers (SBCs) to process data stemming from SG. The use of SBCs is driven by the energy efficiency and cost effectiveness concepts that the SG is trying to apply. The platform in question is tested against a distributed job that averages random numbers using Hadoop's MapReduce programming model. The SBC that we are using in this implementation is the NVIDIA Jetson Developer Kit. The results of this work show that a cluster of SBCs is low-cost, easy to maintain, and simple to deploy, which makes it a great candidate for providing edge computing. Although it revealed a performance that beat the one of the remote cloud servers, it could not outperform the single-computer edge platform.

Keywords: single-board computers; edge computing; cloud computing; green computing; smart grids; Hadoop

Citation: Bourhnane, S.; Abid, M.R.; Zine-dine, K.; Elkamoun, N.; Benhaddou, D. Cluster of Single-Board Computers at the Edge for Smart Grids Applications. *Appl. Sci.* **2021**, *11*, 10981. <https://doi.org/10.3390/app112210981>

Academic Editor: Andreas Sumper

Received: 29 September 2021

Accepted: 15 November 2021

Published: 19 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The emergence of new technologies and applications implied a significant increase in the energy consumed. Figure 1 shows the electricity use in the US throughout the years [1]. We notice how the consumption has increased and how it is still expected to rise. Thus, there is an urgent need for a solution to cope with the stringent need for power.

That is when smart grids came to the surface. A smart grid can be defined as a traditional power grid with underlying intelligence. This intelligence resides in many features. In SGs, in addition to electricity, data also flows in the lines carrying information about the energy consumption and production by the users. The data flow and transmission are possible through the implementation of wireless sensor networks that are responsible for sensing and sending data regarding the consumption of different appliances in addition to the production of the renewable energy source. These data are later used to predict the consumption and eventually implement the energy efficiency. This makes SG an explicit IoT application.

Figure 1. Electricity use in the US. Adapted from [1].

The basic IoT architecture contains four main components: sensors, gateway, cloud, and end user/application. These components are connected as shown in Figure 2 below.

Figure 2. IoT application.

This architecture has been adopted for many years. However, researchers have been trying to improve it upon noticing that the processing at the cloud introduces significant latency, which is not acceptable in most of IoT applications. According to Keller et al. [2], parallel processing applications running on the cloud have an average response time of 1800 ms. Heavier applications can take up to 3000 ms before giving a response to the user. Moreover, in the SG context, many applications require a decision to be made after the processing. These decisions are essential for the functioning of the application and need to be made on the spot. Hence, there was a need for a new paradigm to reduce the latency for SG related IoT applications.

This is where edge computing (EC) comes in handy. Edge computing analyzes some of the data coming from IoT sensors at the edge of the internet to reduce the latency and allow for the decisions to be made in shorter periods of time. Therefore, the IoT architecture has changed to include edge platforms. The new architecture is depicted in Figure 3 below.

Edge platforms do not replace cloud servers. In the SG context, edge platforms are supposed to run control algorithms for the decision making, in addition to prediction algorithms for the forecast. Concerning cloud servers, they are supposed to permanently store the data and make data visible to the end user under readable formats.

Edge computing brings a number of benefits to SG. It guarantees low latency, which helps to monitor the frequency of the grid in real-time and eventually providing proactive actions. Moreover, data privacy is ensured by edge computing since data are being processed locally instead of at the cloud. Furthermore, since SGs are supposed to reduce the energy consumption and eventually save money, EC provides significant assistance with this area as it allows for the implementation of accurate forecasting and prediction algorithms.

Figure 3. IoT architecture including the edge platform.

Edge platforms are usually commodity computers running dedicated algorithms. Commodity computers can be a big source of energy consumption. A single computer consumes about 200 W/h when being used. If a computer is working for 24 h, it will be consuming around 4.8 kWh and emits about 525 kg of CO₂ per year. In stand-by mode, the power consumption falls to about the third which is still relatively high. Furthermore, datacenters that are based on commodity computers consume around 1% of the electricity utilized globally [3], which is considered a high percentage. Hence, and because of its high energy consumption, researchers and practitioners have been looking into replacing the commodity computer by single-board computers (SBCs) as they consume less energy, are smaller in size and less expensive. An SBC consumes about 2.7 W in an idle state and around 6.4 W in full load. This consumption is very low compared to the one of commodity computers.

In this paper, we are looking into the improvements brought by the implementation of an edge platform using a cluster of NVIDIA Jetson Developer kit. These improvements include the response time, the energy consumption of the cluster, and the accuracy of the prediction algorithms. Although SBCs have been trending and used in many applications and projects, their performance as computing units (or edge platforms) has not been widely tackled in the literature. Our work in this paper consists of using SBCs at the edge and comparing their performance to the one of a cloud and a single node edge platform based on a commodity computer. The aim behind this comparison is to decide whether SBCs can be used at the edge and will eventually be a good alternative to deliver green edge computing. To perform these comparisons, we are making use of Hadoop framework and we are benchmarking our platforms against one of the CPU-bound programs and that is the average of float numbers. Our results have shown that by using a local single machine cluster at the edge, we gained 142% of the execution time of the cloud while the SBC cluster allowed us to gain about 23%.

The rest of the paper is organized as follows: Section 2 presents an extensive literature review giving an idea about the work that was previously done in the same research area. In Section 3, we introduce the main concepts tackled in this paper. Then, Section 4 presents our main contribution by presenting the approach and the general architectures. Section 5 contains the details of the experimental setup and shows the hardware and software architectures. The results along with a detailed discussion are presented in Section 6. Section 7 contains the analysis of the results. Finally, we conclude and present future work in Section 8.

2. Literature Review

There has been a lot of ongoing research related to green computing (GC) where researchers have been trying to implement green data center using many different technologies. In this section of the paper, we are reviewing some of the work done by other scholars.

Authors in [4] are having a closer look at the reasons behind the growth of SBC clusters. Indeed, SBC clusters have reduced footprint, have less energy requirements, and

are cheaper which gives computational game changing results. According to the authors, SBC clusters are meant for educational purposes due to their light weight and low-cost. They also make good testbeds for outstanding technologies such as high-core density and ARM based architectures. The authors also identified a set of use cases where SBCs are seen as a strong future especially when more advanced and more powerful.

In [5], the authors analyze the performance, the efficiency, the value-for-money, and scalability of new SBC clusters based on Raspberry Pi 3 Model B, Raspberry Pi 3 Model B+, and Odroid C2, which is a competing platform. The authors mentioned that SBCs were not a practical option because of their low compute power. For the analysis purposes in their paper, the authors developed a new SBC clustering technique that is called the Pi Stack. This is a new construction that allows for an increased cluster density and improved power proportionality and control allowing SBC clusters to become useful compute infrastructures.

The work done in [6] evaluates the energy efficiency and the performance of single board computer based clusters. The authors carried out extensive testing to analyze the performance of these clusters using popular performance benchmarks. The layout consists of two clusters each containing 20 nodes. They used popular benchmarks for task execution time, memory/storage utilization, network throughput, and energy consumption. They found out that SBC clusters come with a lot of low-cost and low energy consumption advantages, however, the processing power remains significantly limited.

Edge computing has also been notably present in the literature recently. The authors in [7] discuss the emergence of the edge computing as a disruptive technology. It is supposed to deliver highly responsive cloud services for mobile computing. The concept of edge computing allows substantial computing and storage resources (cloudlets, micro datacenters, fog nodes) to be placed at the edge of the internet to be closer to mobile devices and sensors. Proximity is a real matter as it affects the end-to-end latency. This problem can be overcome if sufficient efforts are made in this sense. Proximity of cloudlets has a direct effect on delivering highly responsive cloud services, scalability via edge analytics, privacy-policy enforcement, and cloaking cloud outages. Although the edge brings many benefits, it still faces many technical and nontechnical challenges. The technical problems mainly reside in the algorithms needed for the collective control and sharing of cloudlets in distributed computing. As far as the nontechnical side is concerned, the biggest unknown is the possible business models for deploying cloudlets. Many industries, communities, and organizations will have to work hand-in-hand for the success of this deployment.

The work done in [8] describes mobile edge computing (MEC) as an emergent architecture consisting of having cloud computing services extended to the edge via exploiting mobile base stations. The authors present a comprehensive survey where they provide the definition of MEC along with relevant research and technological developments in the area. They also describe its advantages, architectures, and different application areas. In addition to that, they give an overview of security and privacy issues related to MEC and their solutions.

Many of the challenges and opportunities related to edge computing were mentioned in [9]. After stating many challenges and opportunities, the authors inferred that the edge computing paradigm is still in its infancy and has a lot of potential ahead to pave the way for many advancements in distributed computing. They also presented five research challenges: general purpose computing on edge nodes, discovering edge nodes, partitioning, and offloading tasks, uncompromising quality-of-service and experience, and using edge nodes publicly and securely. Furthermore, they introduced five major existing opportunities: standards, benchmarks and marketplace, framework and languages, lightweight libraries and algorithms, micro-operating systems and virtualization, and industry-academic collaborations.

Edge computing has also been used a lot in AI and introduces what is called edge AI. The paper in [10] presents an on-demand accelerating deep neural network interface through the edge computing. The research in the paper is driven by the limitation of mobile

devices' resources, which makes it hard to run deep neural networks. The authors propose a framework that they named Edgent which is a framework for edge computing for DNN. They implemented their prototype using a Raspberry Pi and a desktop computer.

Edge computing on SBCs was tackled in [11]. Since IoT devices are expanding, centralized cloud computing data centers are no longer suitable for IoT applications as they now present many challenges. These challenges are mainly related to the latency, high throughput, and bandwidth. This paper introduces edge computing as a solution to latency sensitive applications. Furthermore, the authors present an architecture that is based on low-cost single-board computers placed near data sources to deliver real-time responses to IoT applications while keeping scalability.

Edge devices are being interfaced with FPGAs to deliver low latency for IoT applications. In [12], the authors proposed the prototyping of an FPGA-based edge device designed for IoT. It mainly takes care of the connectivity space problem. The variety of IoT devices and the diversity of their communication mechanisms gave birth to a set of different communication standards and technologies. These technologies had to co-exist to provide interoperability among devices. That is why, the IoT-ARM specifies a full communication stack.

Green computing is also one of the main pillars in our work. It is also a trendy topic in the literature. The authors in [13] present green computing as a practice that uses computing resources in an eco-friendly way without impacting the overall performance of the system. The world is witnessing a huge need to adopt green computing practices due to many factors: rapid growth of the internet, increasing equipment power density, increasing cooling requirements, increasing energy costs, restrictions on energy supply and access, low server utilization rates, and the growing awareness of IT impact on the environment.

The use of SBCs can also be looked at as a measure to provide green computing via their use in green data centers. Many clusters have been conceived and developed based on Raspberry Pis. The authors in [14] present an HPC platform that is based on Raspberry Pi which makes it a cost effective and an energy efficient approach. The cluster studied in this work was part of a research project about smart grids. The nano/green data center consists of 5 Raspberry Pis connected in a cluster. The authors presented an analysis involving the performance of the cluster and the energy consumption. For this purpose, a comparison between the Raspberry Pi cluster and a single-node commodity hardware cluster was conducted. The authors used Hadoop as a big data analytics platform for both clusters and used the famous Terasort benchmark. The comparison results have shown that even though the single-node cluster outperforms the Raspberry Pi cluster, it consumes more energy. Thus, if energy efficiency and cost effectiveness are a matter but the performance is not, then the Raspberry Pi is the way to go.

The authors in [15] are presenting an architecture along with a technical framework that supports runtime verification and that is deployed on the edge. Their work targets IoT applications that require some validation along the process. As this operation can be challenging it thus requires the need for a new architecture. According to the authors, this architecture can be instantiated in a large number of settings. To prove that they presented the feasibility of the design and the details of the implementation though the evaluation of the performance.

SBCs have been used on the edge in a variety of domains. The work in [16] gathers the concepts of edge computing, machine learning, computer vision, and IoT applied to the monitoring of apple yield. The proposed architecture is based on a single board computer equipped with a camera for image acquisition. The results have shown that the performance regarding good apple detection was up to 99.89% which reveals the feasibility of this edge platform running artificial intelligence and IoT technologies.

3. Background

3.1. Single-Board Computers

SBCs are also called self-contained computers. They are based on a wide range of microprocessors and are designed on a high density for the integrated circuits used [17]. The use of SBCs comes with a variety of advantages. They have well integrated features as everything is native to the machine. Many types of slots are available which allows for advanced configuration. Furthermore, they can be easily produced which reduces their time-to-market in comparison to PCs and laptops. In addition to that, SBCs are considered more reliable and more energy efficient due to their light weight and compact size.

3.2. Edge Computing

“Edge” in our context literally means geographic distribution. Edge computing refers to the computing that is taking place near the data source [18]. Although computing is no longer done at the cloud, this does not exclude it from the architecture, it actually means that the cloud is coming closer to the users. One of the great advantages that the edge computing came up with is the speed and reduced latency.

3.3. NVIDIA Jetson Nano Developer Kit

One of the newly available SBCs on the market. It is considered a great alternative to Raspberry Pi as it comes with more functionalities and higher specifications. It is a small and powerful SBC that allows you to run multiple neural network algorithms at the same type. They work best for applications like image classification, object detection etc. [19].

3.4. Amdahl's Law

Amdahl's law is a mathematical method that allows for having an idea about the maximum improvement that can be achieved through improving a particular part of the system. It was named after Gene Amdahl and first introduced in 1967. Amdahl's law considers P the parallel portion of the system and $1-P$ the serial part of the system. The maximum speedup is then calculated using the following formula [20]: $S(N) = 1/(1-P) + (P/N)$, N being the number of processors. This formula implies that as we scale up in the number of processor, the speedup tends towards $1/(1-P)$.

4. Our Approach: General Architecture

The work done in this paper falls under the scope of a USAID sponsored research project called MiGrid. The main goal of this project is to develop a holistic approach that couples the integration and the storage of renewable energy sources into smart buildings providing a smart microgrid. The deployment of this project is taking place on a university campus. The general architecture of the project is shown in Figure 4 below [21].

From the architecture depicted above, we can notice six main components for this project:

1. Wireless sensor network
2. Wireless actuator network
3. Big data analytics platform
4. Renewable energy source
5. NI CompactRIO controller
6. Storage device

In this work, we are introducing a new component to the previously described architecture and that is an edge platform. The aim behind is to test and deploy a cost effective and energy efficient approach to deliver edge computing in an attempt to reduce the response time and improve the overall performance.

Figure 4. MiGrid general architecture.

The edge platform in question is based on SBCs, namely NVIDIA Jetson Nano Developer kit. It consists of three units networked together using ethernet cables. The edge platform sits between the WSN and the big data analytics platform, this will allow the data to be closer to the processing unit for time sensitive applications. The communication between the WSN, the edge platform, and the big data analytics platform is done wirelessly using Zigbee protocol through XBee communication module. The communication consists of sending and receiving data along with the processing results. Data refer to the values sensed by the WSN (temperature, humidity, and motion), in addition to data related to the consumption of the different appliances and the production of the PV panels.

The edge platform will take care of running time sensitive jobs mainly related to control tasks, while the big data analytics platform will be responsible for batch processing jobs, permanent storage of the data while making data visible to the user in readable and understandable formats.

The literature review conducted shows the existing gap in the deep and comprehensive analysis of the performance of SBC related to big data analytics jobs. Furthermore, while a large number of research articles are tackling edge computing as a new paradigm intending to reduce the latency and provide a better user experience, they do not emphasize on the platform delivering edge computing. Thus, in our work, we are not only studying the impact of edge integration on our overall architecture, but also looking into the possibility of building an edge platform on SBC. The idea behind this study is to implement an edge platform while respecting the energy efficiency and cost effectiveness pillars of smart grids.

5. Experimental Setup—Real-World Deployment

For the sake of this comparison, we made use of three different platforms:

1. Commercial cloud platform
2. Single-node Hadoop cluster
3. NVIDIA Jetson multi-node Hadoop cluster

We describe the implementation setup of each one of the platforms in the upcoming sections.

5.1. Cloud Platform

The first platform that we are deploying is based on a commercial cloud platform. We created three instances based on virtual machines that we are accessing through normal internet connection. Each instance has the following specifications:

- 1vCPU
- 1 GB RAM
- Ubuntu Linux 16.04

After launching the instances, we had access to them using PuTTY on a Windows computer. We established the communication between the virtual machines using SSH connection. The VMs are connected according to the architecture depicted in Figure 5.

Figure 5. Cloud instances architecture.

5.2. Hadoop Single-Node Cluster

The second platform that we are reviewing consists of having Hadoop installed on a single commodity computer.

We are using a Dell Precision Tower computer with the following specifications:

- CPU: Intel Xeon E5-1620V3 / 3.5 GHz
- Number of cores: Quad-core
- RAM: 16 GB
- Storage: 1 TB

We installed the same operating system that we installed on the VM instances and that is Ubuntu Linux 16.04 on top of which we have installed Hadoop. The Hadoop single-node cluster has the architecture shown in Figure 6.

Figure 6. Single machine cluster architecture.

5.3. NVIDIA Hadoop Multi-Node Cluster

The third and final platform we are investigating consists of a multi-node Hadoop cluster that is based on NVIDIA Jetson Developer Kit.

We are using three boards that we are working with have the following specifications:

- GPU: 128-core NVIDIA Maxwell
- CPU: Quad-core ARM
- Memory: 4GB
- Connectivity: Gigabit Ethernet

The three nodes are running Linux4Tegra OS that is based on Ubuntu 18.04. On top of the OS, we have installed Hadoop. The three nodes are connected according to the architecture in Figure 7.

Figure 7. Multi-node SBC cluster architecture.

The real-world deployment of the cluster is shown in Figure 8.

Figure 8. Real-world deployment of SBC cluster.

6. Results and Analysis

In order to test the performance of our proposed cluster and compare it to the ones of a remote cloud server and a local single-computer edge platform, we had to choose a bench-

mark that represents the main operations carried on in a real-world smart grid deployment. The edge platform we are conceiving is to be used to mainly run control/machine learning algorithms that are known to be CPU-bound, e.g., artificial neural networks, clustering, and aggregation. Hence, we opted for a benchmark that represents CPU-bound applications and that is the average of float numbers. We believe that the use of this benchmark will mimic the heaving application that are to be deployed on our SBC-based edge platform.

The three experiments that we conducted consist of having to average random float numbers using Hadoop on three different platforms. The execution steps of the experiments are depicted in Figure 9.

Figure 9. Experiment execution.

Each platform runs a Hadoop job using three different dataset sizes: 1GB, 5GB, 10GB. The job along with the dataset we sent from a local server that is running Windows. The outcome of the job execution is then returned along with the execution time of the job.

6.1. Hadoop in the AWS Cloud Platform

Figure 10 shows the execution time of the average.jar job on the cluster of virtual machines running on the commercial cloud cluster. We can infer from the graph that the execution time increases as we increase the size of the dataset. Moreover, the number of the nodes significantly affects the execution time.

Figure 10. Execution time of Hadoop on the cloud.

In the next section, we are looking into the execution time of the same job executed using Hadoop on a single machine.

6.2. Hadoop on a Single Machine Edge Platform

Similar to the previous graph, the dataset size impacts the performance of the cluster. The execution time increases as the dataset gets larger. The graph in Figure 11 summarizes the execution times.

Figure 11. Execution time of Hadoop on a single machine.

6.3. Hadoop on NVIDIA Cluster

Figure 12 shows the performance of the physical cluster that is based on NVIDIA Jetson SBCs. The figure looks similar to the previous ones, however, we can notice that the execution time (and eventually the response time) is significantly affected by the dataset size.

Figure 12. Execution time of Hadoop on the SBC cluster.

7. Analysis

The results of the experiments are presented in the previous section. The graph below summarizes the results obtained (Figure 13).

Figure 13. Comparison of the different execution times.

Figure 13 shows that the single node Hadoop cluster has the lowest response time which implies having the best performance, followed by the SBC cluster, and finally the cloud-based cluster.

From the results, we can notice that the single-node local Hadoop cluster improved the response time of the cloud by 142%, while the SBC-based multi-node cluster came with an improvement of about 23%.

One of the major advantages of using SBCs in a cluster is the ease of scalability. This implies the possibility to add as many nodes as needed easily and at a very affordable price. Next, we would like to have a look at the speed up of the NVIDIA Jetson Developer Kit when running the same job. For this, we have used Amdahl's law. We assumed that the parallelized part of the job consists 80% of the entire program. The speedup obtained is presented in Figure 14 below.

Figure 14. Speedup calculated using Amdahl's law.

The cluster speeds up quite fast as we add more node to it. This makes the scalability even easier and makes SBC a great alternative for commodity computers in addition to their lightweight, low electricity consumption, and low price. In a nutshell, SBCs are considered an energy efficient and cost-effective approach to deliver edge computing,

however, their performance is still extremely limited and cannot really compete with commodity computers. This leaves the choice to the user, if energy efficiency and cost effectiveness are of a high priority but the performance is not, then you can rely on SBC to deliver edge computing.

8. Conclusions and Future Work

The work conducted in this paper makes use of a cluster of single-board computers in order to build an edge platform. For its performance to be tested, we compared the SBC based platform to a cloud-based cluster and a single-node local machine. All platforms are running Hadoop as the underlying big data analytics framework and are benchmarked against a job that averages random float numbers. The results have shown that we get huge improvements in the response time by introducing edge platforms. From the results described previously, we were able to gain 142% of the execution time by using a single physical machine at the edge, and we gained 23% by using the cluster of SBCs as an edge platform.

As for future work, we are intending to build a larger SBC cluster and test it with AI and machine learning algorithms. Furthermore, we are planning on building different clusters based on different boards in order to get a clear idea about the performance of different boards and eventually decide on the best solution to deliver edge computing. In addition to that, we have the intention to expand the current testbed to include more buildings and eventually have enough data to implement more AI algorithms.

Author Contributions: Conceptualization, M.R.A. and D.B.; methodology, M.R.A. and D.B.; software, S.B.; validation, M.R.A., D.B., N.E. and K.Z.-d.; formal analysis, M.R.A. and D.B.; investigation, S.B., M.R.A. and D.B.; resources, M.R.A. and D.B.; data curation, S.B.; writing—original draft preparation, S.B.; writing—review and editing, M.R.A., D.B. and K.Z.-d.; visualization, M.R.A., D.B., K.Z.-d. and N.E.; supervision, M.R.A., D.B., K.Z.-d. and N.E.; project administration, M.R.A.; funding acquisition, M.R.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the United State Agency for International Development grant number 5-398.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to the confidentiality of the project.

Acknowledgments: This work is sponsored by US-NAS/USAID under the PEER Cycle 5 project grant# 5-398, entitled ‘Towards Smart Micro-grid: Renewable Energy Integration into Smart Buildings’.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. U.S. Electricity Consumption 2020 Statista. Available online: <https://www.statista.com/statistics/201794/us-electricity-consumption-since-1975/> (accessed on 7 October 2021).
2. Keller, M.; Karl, H. Response-Time-Optimized Distributed Cloud Resource Allocation. In Proceedings of the 2014 ACM SIGCOMM Workshop on Distributed Cloud Computing, Chicago, IL, USA, 17–22 August 2014; pp. 47–52.
3. Study: Data Centers Responsible for 1 Percent of World’s Energy Use. Available online: <https://www.datacenterknowledge.com/energy/study-data-centers-responsible-1-percent-all-electricity-consumed-worldwide> (accessed on 18 September 2021).
4. Johnston, S.J.; Basford, P.J.; Perkins, C.S.; Herry, H.; Tso, F.P.; Pazaros, D.; Mullins, R.D.; Yoneki, E.; Cox, S.J.; Singer, J. Commodity single board computer clusters and their applications. *Future Gener. Comput. Syst.* **2018**, *89*, 201–212. [CrossRef]
5. Basford, P.J.; Johnston, S.J.; Perkins, C.S.; Garnock-Jones, T.; Tso, F.P.; Pazaros, D.; Mullins, R.D.; Yoneki, E.; Singer, J.; Cox, S.J. Performance analysis of single board computer clusters. *Future Gener. Comput. Syst.* **2020**, *102*, 278–291. [CrossRef]
6. Qureshi, B.; Koubaa, A. On energy efficiency and performance evaluation of single board computer based clusters: A hadoop case study. *Electronics* **2019**, *8*, 182. [CrossRef]
7. Satyanarayanan, M. The emergence of edge computing. *Computer* **2017**, *50*, 30–39. [CrossRef]
8. Abbas, N.; Zhang, Y.; Taherkordi, A.; Skeie, T. Mobile Edge Computing: A Survey. *IEEE Internet Things J.* **2018**, *5*, 450–465. [CrossRef]

9. Varghese, B.; Wang, N.; Barbhuiya, S.; Kilpatrick, P.; Nikolopoulos, D.S. Challenges and Opportunities in Edge Computing. In Proceedings of the 2016 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, USA, 18–20 November 2016; pp. 20–26. [CrossRef]
10. Li, E.; Zeng, L.; Zhou, Z.; Chen, X. Edge AI: On-Demand Accelerating Deep Neural Network Inference via Edge Computing. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 447–457. [CrossRef]
11. Fernández-Cerero, D.; Fernández-Rodríguez, J.Y.; Álvarez-García, J.A.; Soria-Morillo, L.M.; Fernández-Montes, A. Single-board-computer clusters for cloudlet computing in internet of things. *Sensors* **2019**, *19*, 26. [CrossRef] [PubMed]
12. Gomes, T.; Pinto, S.; Gomes, T.; Tavares, A.; Cabral, J. Towards an FPGA-based edge device for the Internet of Things. In Proceedings of the 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA), Luxembourg, 8–11 September 2015. [CrossRef]
13. Saha, B. Green Computing. *Int. J. Comput. Trends Technol.* **2014**, *14*, 46–50. Available online: <http://www.ijcttjournal.org> (accessed on 9 June 2021). [CrossRef]
14. Bourhnane, S.; Abid, M.R.; Zine-Dine, K.; Elkamoun, N.; Benhaddou, D. High-Performance Computing: A Cost Effective and Energy Efficient Approach. *Adv. Sci. Technol. Eng. Syst. J.* **2020**, *5*, 1598–1608. [CrossRef]
15. Tsigkanos, C.; Bersani, M.M.; Frangoudis, P.A.; Dustdar, S. Edge-based Runtime Verification for the Internet of Things. *IEEE Trans. Serv. Comput.* **2021**, *1*, 1–14. [CrossRef]
16. Savvidis, P.; Papakostas, G.A. Remote Crop Sensing with IoT and AI on the Edge. In Proceedings of the 2021 IEEE World AI IoT Congress (AIIoT), Seattle, WA, USA, 10–13 May 2021; pp. 48–54. [CrossRef]
17. Techopedia. What Is a Single-Board Computer (SBC)? Available online: <https://www.techopedia.com/definition/9266/single-board-computer-sbc> (accessed on 30 June 2021).
18. The Verge. What Is Edge Computing? Available online: <https://www.theverge.com/circuitbreaker/2018/5/7/17327584/edge-computing-cloud-google-microsoft-apple-amazon> (accessed on 30 June 2021).
19. NVIDIA Jetson Nano Developer Kit NVIDIA Developer. Available online: <https://developer.nvidia.com/embedded/jetson-nano-developer-kit> (accessed on 13 June 2021).
20. Moore's and Amdahl's Law. Available online: http://www.umsl.edu/~jsiegelj/CS4740_5740/Overview/Amdahl\T1\textquoterights.html (accessed on 18 September 2021).
21. Abid, M.R.; Lghoul, R.; Benhaddou, D. ICT for renewable energy integration into smart buildings: IoT and big data approach. In Proceedings of the 2017 IEEE AFRICON, Cape Town, South Africa, 18–20 September 2017; pp. 856–861. [CrossRef]