ASTES

Advances in Science, Technology and Engineering Systems Journal Vol. 5, No. 6, 1598-1608 (2020)

www.astesj.com

ASTESJ

ISSN: 2415-6698

High-Performance Computing: A Cost Effective and Energy Efficient Approach

Safae Bourhnane[1,2,*], Mohamed Riduan Abid[2], Khalid Zine-Dine[3], Najib Elkamoun[1], Driss Benhaddou[4]

[1]*Chouaib Doukkali University, El Jadida, 24030, Morocco*

[2]*School of Science and Engineering, Al Akhawayn University, Ifrane, 53000, Morocco*

[3]*Mohammed V University in Rabat, Faculty of Sciences, Rabat, 10170, Morocco*

[4]*School of Engineering and Technology, University of Houston, Texas, TX 77004, USA*

ARTICLE INFO

Article history:
Received: 05 November, 2020
Accepted: 20 December, 2020
Online: 25 December, 2020

Keywords:
Big data
HPC
Energy efficiency
Green computing
Amdahl's law

ABSTRACT

The world is witnessing unprecedented advancements in ICT (Information & Communication Technology) related fields. These advancements are further boosted with the emergence of big data. It goes without saying that big data requires two major operations: storage and processing. The latter is usually provided through High-Performance Computing (HPC) which is delivered through two main venues: supercomputers or clustering. The second venue has been widely opted for as a cost-effective alternative when compared to supercomputers. However, with the widespread increase in deploying ICT-based applications, the parallel increase in energy consumption has become a real issue. Thus, researchers have been exploring approaches to conceive big data analytics platforms that are both cost-effective and energy-efficient. In this paper, we present a cost-effective and energy-efficient HPC clustering that is based on Raspberry PIs. Our approach leverages the concept of Green Computing. We evaluated our cluster performance and its energy consumption and compared it to a commodity server. We leveraged on the Amdahl's law to set the maximum speedup of the proposed approach. Our approach can be easily deployed for usage in different ICT-based applications that consider energy efficiency as a priority.

1. Introduction

Nowadays, ICT is interweaving into the fabrics of everyday life and touches all aspects of our lives: smart homes, smart cities, smart grids, smart agriculture, eHealth, autonomous vehicles, etc. All these aspects been further enhanced by the emergence of big data.

Indeed, big data is now becoming wide data. By looking at the amount of data generated each day from different sources (e.g. social networks, Internet of Things, business transactions, etc.), we can infer that these data will eventually need large and advanced platforms that can accommodate for their constantly increasing amount.

It is very common that Big Data faces two major challenges: storage and processing. The storage of the big data can be challenging as it needs to be correctly performed in order to allow for an appropriate and efficient processing later on.

As the data becomes more massive and the applications more demanding, the processing becomes very heavy. This implies the use of High-Performance Computing (HPC) as the ultimate solution for the increasing demand on the computing power. The term refers to combining processing powers to deliver a higher performance. Throughout the years, HPC has been provided via either supercomputers, or cluster of commodity computers. The first venue is no longer opted for mainly because of its high cost and not-so-easy maintenance which leaves the clustering as the perfect alternative.

Cluster Computing involves combining similar type of computing machines that work, transparently, like a single computing unit. These machines are connected using dedicated network protocols and usually Local-area networks (LAN). The aggregation of computing powers allows for a better efficiency to

*Corresponding Author: Safae Bourhnane, Al Akhawayn University in Ifrane, +212610880040, Email: s.bourhnane@aui.ma

solve complex operations using faster processing speed than a single machine.

Cluster Computing can be divided into three main categories: load balancing cluster, high availability cluster, and high-performance cluster. The latter is what we are selecting for the work presented in this paper. Clusters of computers represent the environment for different distributed platforms to store and process the big data. Hadoop is the most widely used case in point and it can be utilized for a wide range of applications thanks to the number of benefits it presents.

Hadoop is a famous open-source framework that is conceived for storing and processing data. It serves to run applications on clusters of computers which help it make use of the great processing power provided by the cluster. Hadoop has two main components: MapReduce and HDFS. MapReduce is the programming model brought by the framework while HDFS refers to Hadoop Distributed File System and hence represents the underlying file system.

Testing the cluster with a specific number of nodes can give an idea about the performance as the number of nodes increases. This concept is called "Speedup" and can be given using the renowned 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. In our case, we are making use of Amdahl's law to theoretically find the maximum speed up with multiple processors. Also, it will allow us to infer whether the Raspberry Pi cluster will reach the performance of a single commodity hardware and determine the number of nodes needed in the positive case.

Since we are concerned by the energy efficiency and the cost-effectiveness of the entire system, we are opting for an approach that respects both features. Our approach is based on Raspberry Pis as an alternative to commodity computers and that is because it is known for its low cost and low energy consumption.

Currently, the world is trying to implement a solution that copes with the increasing demand on processing power due to the expanding amount of the data produced, without consuming a large amount of energy. Eventually, researchers have been trying to implement HPC clusters based on low-cost and low-energy hardware. There is a significant amount of work that has been carried in this direction. Most of it used Raspberry Pi as a basis of the cluster and tried to prove that the hardware supports the installation of different big data analytics platforms, e.g. Hadoop. However, this does not give an idea about the real performance of the Raspberry Pi cluster.

For the Raspberry Pi cluster to pass the performance test, it needs to be compared to a performant machine/server. This comparison should be accomplished using the same big data analytics platform and by running the same jobs. This does not only imply the use of the same dataset, but also the same dataset size.

In this paper, we are testing our Raspberry Pi approach and comparing it to a commodity server through benchmarking both setups against the Terasort which is one of the mostly used Hadoop benchmarks. In addition to keeping track of the performance, we are measuring the energy consumed by both setups as energy efficiency is one of the main pillars of our study.

Moreover, we are looking at the speedup of the cluster in order to find the maximum value it can reach. This will allow us to have an idea about the size of the cluster corresponding to a higher performance. For this, we are making use of the Amdahl's Law.

The rest of the paper is organized as follows: Section II presents the literature review; we present the background in III and our proposed approach in section IV. Section V contains the results related to the single-node cluster while section VI presents the results of Raspberry Pi multi-node cluster. We compare both setups and present a discussion in section VII. Finally, we present the conclusion and future work in section VIII.

2. Literature Review

During the last few years, implementing Green Computing using Raspberry Pi has been trending among practitioners in the domain.

Raspberry Pi clusters have been the center of different research works that tackled it as a cost-effective and energy-efficient solution to deliver Green Computing. In this section, we are shedding the light on some of the work that has been previously carried out in that sense.

In [1] the authors implemented a Raspberry Pi cluster that consists of 300 nodes. Through their paper, they walked us through the steps of deployment and set up of the hardware and software plus the maintenance and the monitoring of the overall system. Moreover, the work presents some of the limitations and the challenges that would block the deployment of the cluster. These include the overall performance of the card that is considered relatively low due to the design of the flash memory, in addition to the low processor speed.

The next work described another low-cost cluster named Iridis-pi and that was conceived for the sake of demonstration [2]. The cluster contains 64 Raspberry Pis Model B with 700MHz processor and 256MB of RAM. The authors chose to host the cluster on a Lego chassis. The nodes are connected together using Ethernet cables. The entire system has an overall RAM of 16GB and 1TB of storage. In order to assess the numerical computer power, this research used the LINPACK benchmark for the single-node performance, and the High-Performance LINPACK benchmark in order to measure the overall throughput of the cluster. The results revealed a computational performance peak of around 65000kflops. More results showed that the cluster delivers a good scalability with large problems, and a less significant one with small problems where the network overhead becomes more noticeable.

Glasgow Raspberry Pi cluster [3] depicted the use of Raspberry Pi in data processing. The cluster consists of 56 Model B Raspberry Pi that are supposed to emulate the entire stack of the cloud. The devices are interconnected in a tree topology. Each node uses a 16GB SD card that hosts up to three co-located virtualized hosts provided via Linux containers. According to this research, the deployment of this cluster is still not mature. The authors are still investigating an easier and more secure approach

to manage the virtual resources. Also, their future work includes the implementation of live migration through the PiCloud.

The work in [4] introduced a study of the feasibility of implementing Raspberry Pi based data centers for Big Data processing. The paper examined further the advantages using Raspberry Pi for big data applications through a micro data center. The authors conducted a study that tackles the performance, the scalability, energy consumption, ease of management, etc. For testing purposes, they used Hadoop framework. This allowed the authors to discover that the cluster delivers a moderate performance.

Th authors in [5] explored the use of low-power and low-cost devices for pervasive computing (Cloud and Fog computing). The work tests the performance of the cluster using Apache Spark and HDFS. It also uses the same setup for a real and a virtualized environment. The virtualization has proved to be more significant with large amounts of data. Moreover, the virtualization also affects the energy consumed by the Raspberry Pi when the workload becomes higher.

The research in [6] presented the evaluation and comparison of 24 ARM board energy consumption and performance wise. Finally the Raspberry Pi was chosen as the basis for ARM HPC after looking at the tradeoffs. The results of this study have shown that the overall performance of the cluster remains less than that of the x86 server. However, the authors mentioned that this was not the main end goal of the project. The low-cost of the Raspberry Pi allows for affordable and more accessible cluster computing for the public and especially for students. Hence, the work presented in this research encourages the use of low-power parallel programming in education.

In [7], the authors explained the use of Image Processing algorithms to test the performance of the Raspberry Pi cluster. Image processing is known to require a lot of processing power that is why it is being used to assess the performance of the Raspberry Pi cluster. The authors developed an application that counted the wheat grains presented in a given picture. The Raspberry Pi cluster hosted a parallel computing application that was coded in Python in addition to the use of MPI. The application was tested on a single node first and then on the cluster of four nodes. The purpose was to test the speed of the application. The results have shown that the processing time using both single-node and multi-node Raspberry Pi clusters depends on the number of images processed.

The work in [8] presented the design and creation of a high performance Beowulf cluster based on 12 Raspberry Pis. In order to test the cluster, some mathematical benchmarks have been used to measure the performance of the cluster. The authors used a program to calculate the scalar multiplication of a matrix. The runtime and the GFLOPS were logged. The authors have mainly noticed the following: the peak performance of the system went up as the number of nodes increased, the speedup also increased as more nodes were added to the cluster and increasing the problem size increased the performance. The authors also mentioned a number of limitations related to memory. However, according to the study, the cluster can be used to automate some testing operations.

In [9], the research extended some previously done work by comparing a cluster of Raspberry Pi to multi core processors like i5 and i7 processor machines. The results of this work have shown that the Raspberry Pi cluster is ranked third performance wise after the i7 and the i5 processors. The Raspberry Pi is very limited in terms of resources. Even with 14 nodes it could not reach the performance of its competitors. Core i5 architecture has better resources and eventually performs better than the Raspberry Pi cluster. The core i7 architecture remains with the highest performance.

Our paper aims to extend the previously discussed work by conducting a set of experiments in order to compare our cluster to one single performant commodity machine. The work presented here does not provide an ultimate solution, it rather serves as a baseline for researchers to decide on the most suitable configuration with regards to their applications.

3. Background

3.1. High Performance Computing

High Performance Computing (HPC) deals with the implementation of algorithms or code taking into consideration the hardware used for each application [10]. It was conceived by scientists and engineers to cope with the problems that require more resources (CPU, memory, etc.) than what one single machine can provide.

There are two very known venues to implement HPC: either through owning supercomputers, or clustering commodity computers. The second option is what is generally opted for. HPC is usually provided via a set of computers that have almost the same characteristics, and that are connected to each other. Each one of these machines is referred to as a node [11].

HPC-based solutions usually have three main components: Compute, Network, and Storage. Compute servers are connected forming a cluster which is also connected to the storage component. These connections are not visible to the user who deals with the entire system as a single machine in a transparent manner

HPC has several use cases that can either be deployed on premises, at the edge, or on the cloud [12]:

- Research: HPC clusters help researchers find solutions to very complex problems, and hence bring contributions to the community.
- Entertainment: HPC is also used in media in order to edit films and come up with mind-blowing special effects.
- Artificial Intelligence: HPC implements machine learning algorithms that requires huge processing power.
- Finance: HPC clusters can be used to track stocks, design new products, simulate test scenarios, etc.

3.2. Green Computing

Currently, Green Computing (GC) is considered a shift that is supposed to deal with the constantly increasing energy consumption of the existing computing solutions. This is due to the increasing demand on computing power by different applications.

Green Computing consists a tailor-made solution for many applications as it is based on energy-efficiency as one of the pillars. GC as a concept, refers to the deployment of efficient and eco-friendly computing solutions. It is provided through four main technologies: green data centers, virtualization, cloud computing, grid computing [13].

3.3. Hadoop

It goes without saying that data have been generated in a massive manner throughout the years due to the advancements in all the fields and industries. Talking about huge datasets implies bringing up storage. Storage capacities of hardware systems have increased to keep up with the amount of streaming data. However, access speeds did not keep up. Thus, the traditional Database Management Systems (DBMS) are no longer considered an option to go for. Also, hardware failure is a problem that needs to be overcome.

Hadoop was introduced to solve the previously mentioned problems in addition to others. It is an opensource framework that deals with big data through providing a platform for distributed storage and processing. As Hadoop is used over a set of clustered computers, the possibility of a machine failing is relatively high. This may imply serious data loss if not dealt with correctly. Data replication is a way to tackle this issue effectively: having redundant data all over the system.

3.4. Raspberry Pi

The Raspberry Pi is a computer with the size of a credit-card that can be plugged to a monitor, a keyboard, a mouse, and other devices. It provides a low-cost option to explore computing and programming in different languages (e.g. Python). It is also capable of providing all the options of a normal computer: internet browsing, playing games, etc. [14].

Moreover, the Raspberry Pi has the ability to interact with the outer world through the connection with sensors and actuators. This is done through what is called General-Purpose Input Output (GPIO). This allows the computer to be the best option for robotics related projects and prototypes.

Since its first launching, RP has gone through a set of changes that resulted in different versions and models available in the market. Table 1 presents the models of Raspberry Pi and their characteristics.

Table 1: Raspberry Pi Models [15]

Model	Date	Price ($)	Core	Num. Cores	RAM
RP B	15/2/2012	35	ARM1176JZF-S	1	512M
RP 2	1/2/2015	35	Cortex-A7	4	1G
RP Zero	30/11/2015	5	ARM1176JZF-S	1	512M
RP 3	26/2/2016	35	Cortex-A53 64-bit	4	1G
RP A+	10/11/2014	35	ARM1176JZF-S	1	256M
RP Zero W	28/02/2017	10	ARM1176JZF-S	1	512M
RP WH	12/1/2018	14	ARM1176JZF-S	1	512M
RP 3 B+	14/3/2018	39	Cortex-A53 64-bit	4	1G
RP 3 A+	15/11/2018	25	Cortex-A53 64-bit	4	512M
RP 4 B	24/6/2019	37	Cortex-A72 (ARM v8) 64-bit	4	1/2/4

3.5. Project Scope

The work presented in this paper falls under the scope of a USAID sponsored research project named MiGrid. The project aims at developing a holistic approach that couples renewable energy storage and production in smart buildings.

The general architecture of the project is depicted in the Figure 1 below:

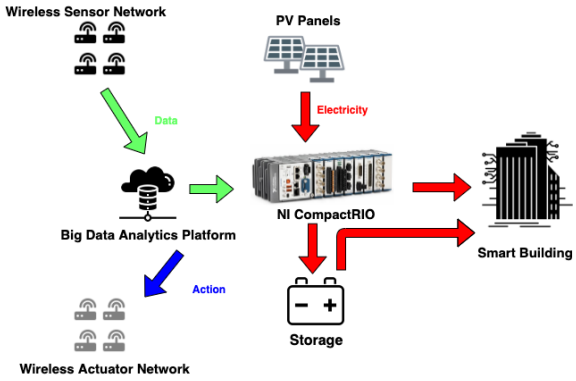

Figure 1: Smart Micro-Grid General Architecture

The architecture contains six main components [16]:

1. Wireless Sensor Network: (WSN) A number of wirelessly connected sensors that are supposed to sense data about the environment.
2. Wireless Actuator Network: Actuators that are connected and interfaced with the WSN. They are responsible for translating the digital signals into electrical ones.
3. Big Data Analytics Platform: A platform that implements a number of algorithms responsible for the processing and analysis of data provided by the WSN.
4. NI CompactRIO: the main controller of the system that decides whether the energy produced is to be stored, injected in the grid, or used to power appliances.
5. Storage: Batteries that store the excess of energy produced and make it available for later use.
6. Solar Panels: the main renewable energy source of the system.

The work tackled in this paper fits in the third (3) component as it presents and tests a potential implementation of the big data analytics platform that is based on Raspberry Pi as the main piece of hardware.

4. Proposed Approach

Our approach to the green cluster relies on using Raspberry Pis instead of commodity computers to deliver HPC. Our work consists of conducting a set of experiments that compare our green Raspberry Pi cluster to one single server. The comparison consists of looking at the performance in terms of CPU time, in addition to the energy cosumption of each solution. Furthermore, we are invistigating the impact of the number of CPUs and memory size on the performance of the single machine, and the impact of the number of nodes and the network performance on the Raspberry Pi cluster.

For the sake of this experiment we made use of two different architectures: a multinode Hadoop architecture, and a single node architecture.

4.1. Multi-node Hadoop Architecture

This setup consists of five Raspberry Pis. The hardware specifications are presented in Table 2:

Table 2 Raspberry Pi Specifications

SoC	Broadcom BCM2837
CPU	4xARM Cortex – A35, 1.2 GHz
GPU	Broadcom VideoCore IV
RAM	2GB
Networking	10/100 Ethernet, 2.4GHz 802.11n Wireless
GPIO	40-pin header, populated
Ports	HDMI, 3.5mm analogue audio-video jack, 4xUSB 2.0, Ethernet, Camera Serial
Price	$25-$195

The hardware architecture consists of the five Raspberry Pis connected together and to a switch through Ethernet connection. Figure 2 below describes the hardware architecture:

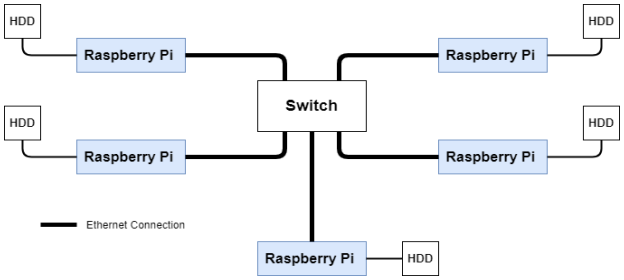

Figure 2: Multi-node Cluster Hardware Architecture

Each one of the Raspberry Pis has the software architecture presented in Figure 3:

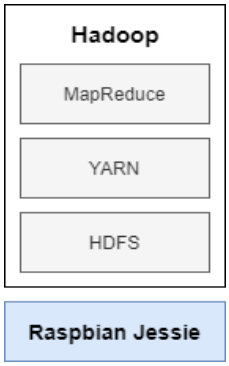

Figure 3: Multi-node Cluster Software Architecture

4.2. Single-node Cluster Architecture

The single node cluster consists of the following server with the specifications mentioned in Table 3 below. The corresponding software architecture is depicted in Figure 4:

Table 3: Single-Node Machine Specifications

Manufacturer	Dell, Inc
CPU	Intel Core i7 (6th Gen) 3.4 GHz
Number of cores	Octa-core
RAM	8GB
Hard Drive	SATA, HDD, 1TB
Networking	Ethernet, Fast Ethernet, Gigabit Ethernet
Graphic Controller	NVIDIA Quadro K620 2 GB
Price	$900

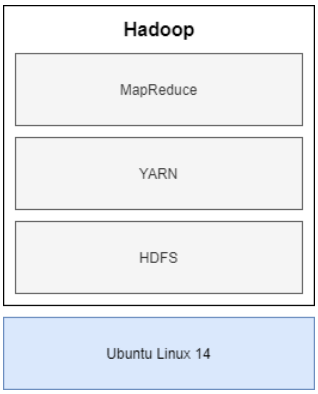

Figure 4: Single Node Machine Software Architecture

4.3. Benchmark

For this experiment, our setups were benchmarked against one of the most famous Hadoop benchmarks: Terasort.

The Terasort is a sorting algorithm implemented using MapReduce. There are two more functions related to Terasort that that are also implemented in Hadoop:

- Teragen: it generates the data to be sorted. We can specify the size of the dataset to be sorted.
- Terasort: the actual sorting jar that takes the result of the Teragen function.
- Teravalidate: it validates the output (the sorted result of the input data).

For the execution, we followed the steps presented in Figure 5:

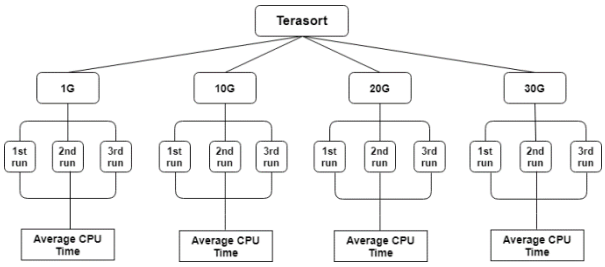

Figure 5: Terasort Execution Steps

We ran the experiment on four different dataset sizes: 1GB, 10GB, 20GB, and 30GB. For each run, and in order to avoid outliers in the results obtained, we ran each experiment many times and we kept the best (statistically) three results, and calculated the mean for both the energy consumption and the execution time.

4.4. Energy Consumption Measurement

In order to measure the energy consumed by each of the setups, we created a sensor node based on the Arduino Uno

microcontroller in addition to the SCT013 current sensor. Arduino is an open-source platform that is easy to use and mainly serves as a prototyping platform. The boards are able to read input form sensors and turn it into an output. We opted for Arduino due to its price affordability and accessibility user-experience wise [17].

The SCT013 sensor is a non-invasive current transformer that measures the intensity of the current in a conductor. These measurements are provided using the electromagnetic induction. They come in the form if clamps that can be wrapped around the equipment. The sensor is very accurate with an error rate of only 1-2% [18].

The Arduino circuit that we used is shown in Figure 6 below:

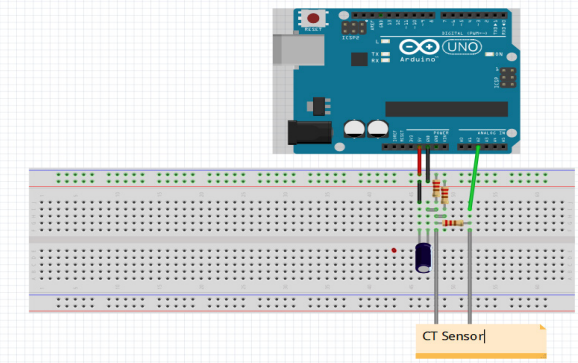

Figure 6: Current Sensor Arduino Circuit

5. Performance Evaluation

5.1. Single Node Cluster

a. Impact of the Number of CPUs

In this section, we are going to study the impact of the number of CPUs on the performance (i.e. the CPU time). The server we are working with is an octa-core machine. We ran the same job 8 times: limiting the number of CPUs from 1 to 8.

The results are given in Figure 7.

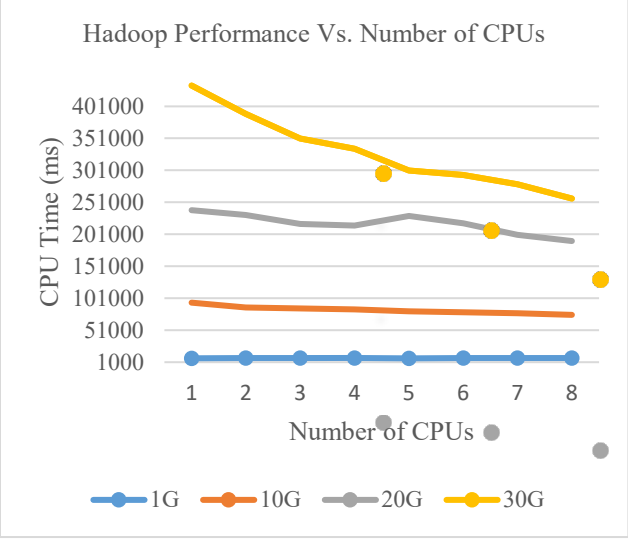

Figure 7: Hadoop Performance Vs. Number of CPUs

The job used in this comparison is the Terasort benchmark as mentioned previously in the paper. The sorting algorithm used as input files of different sizes.

As we can notice from the graph above, the difference in the CPU time when sorting 1Gb of data is not significant: it varies between 7350 ms and 7100 ms. For the CPU time corresponding to sorting 10Gb of data, it showed a slight decrease as the number of CPUs increased: it went from 94100 ms to 75005 ms. The most significant difference was shown in the performance of sorting 20Gb and 30Gb of data.

Concerning the CPU time resultant from sorting 20Gb of data, it scaled down from around 239000 ms to 191000 ms. Sorting 30Gb of data took eventually more time and went from 433590 ms using 1 CPU to about 251000 ms using 8 CPUs.

The number of CPUs used in the Dell Precision Tower machine has a significant impact on the performance of Hadoop jobs. This impact becomes more and more noticeable as the size of the job increases.

In the next section, we are going to have a closer look at the impact of memory size on the performance of Hadoop as a single node cluster.

b. Impact of Memory Size

In order to investigate the impact that the size of the memory has on the performance of Hadoop, we made use of the same benchmark (i.e. Terasort) and using the same input dataset sizes. The machine that we are working with in this experiment has a maximum of 8Gb of RAM. For each job, we had to limit the memory usage starting from 1Gb to 8Gb.

Figure 8 below shows the results of the experiment.

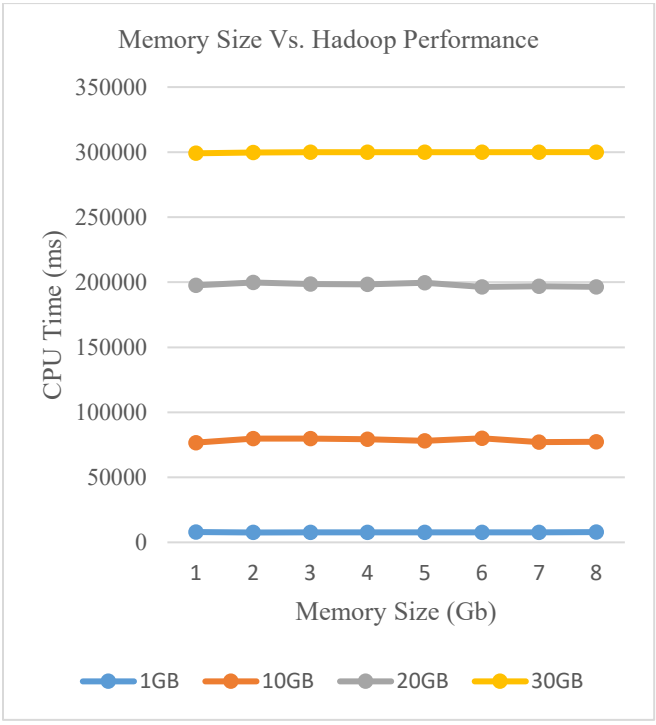

Figure 8: Memory Size Vs. Hadoop Performance

As shown in the figure above, scaling up in the size of the memory did not impact the performance of the cluster. The CPU

time remained the same throughout all the experiments. Sorting 1Gb of data took about 7500 ms with all the sizes of the memory. Concerning CPU time of sorting 10Gb of data, it was about 78000 ms and did not significantly change with all the memory sizes. Similarly, sorting 20Gb of data was stable at around 199000 ms. Also, the CPU time corresponding to sorting 30Gb of data was constant at around 300000 ms and was not impacted by the memory size. This mainly due to the nature of Hadoop and its technologies: they are disk-based and not memory-based. Unlike Spark, Hadoop does not work in memory, which somehow explains the delay that it presents compared to Spark platform.

c. Energy Consumption

Since we are concerned by the energy efficiency as a primary matter, we measured the energy consumed by the singe-node cluster for the same benchmark using 1Gb and 30Gb of input data. For the sake of this experiment, we used all 8 CPUs of the machine and unlimited memory resources.

The energy consumption measured is shown in Figure 9 and 10.

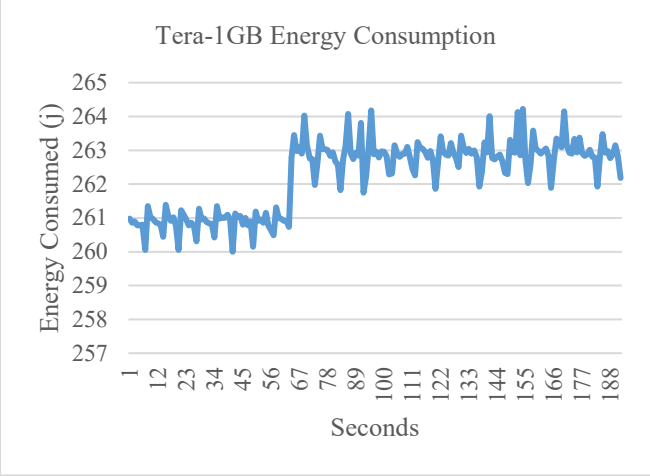

Figure 9: Tera-1GB Energy Consumption

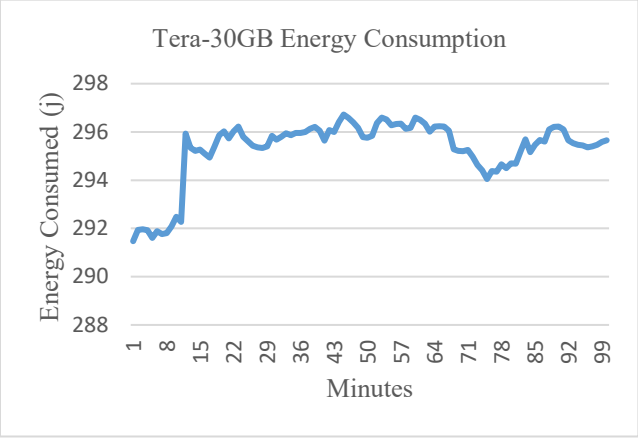

Figure 10: Tera-30Gb Energy Consumption

From the graphs above, we can notice that the energy consumption during the execution time of the Terasort benchmark with 1GB dataset went from 260 J/s to 290 J/s with an average of about 263 J/s. Running the same benchmark on a 30GB dataset size consumed an average of 295 J/s with the max being 296 J/s and the min 291 J/s.

5.2. Multi Node Raspberry Pi Cluster

a. Impact of the Number of Nodes

In this section, we are exploring the impact of the number of nodes in a multi-node cluster on the performance of Hadoop.

As described previously in the paper, we are dealing with a cluster of 5 Raspberry Pis that are connected together through ethernet. Each node has the same software architecture that is based on the same version of Hadoop.

For the sake of comparison, we are using the same benchmark as the previous experiment with the same dataset input sizes.

The result of running Terasort on Raspberry Pi is mentioned in the figure below.

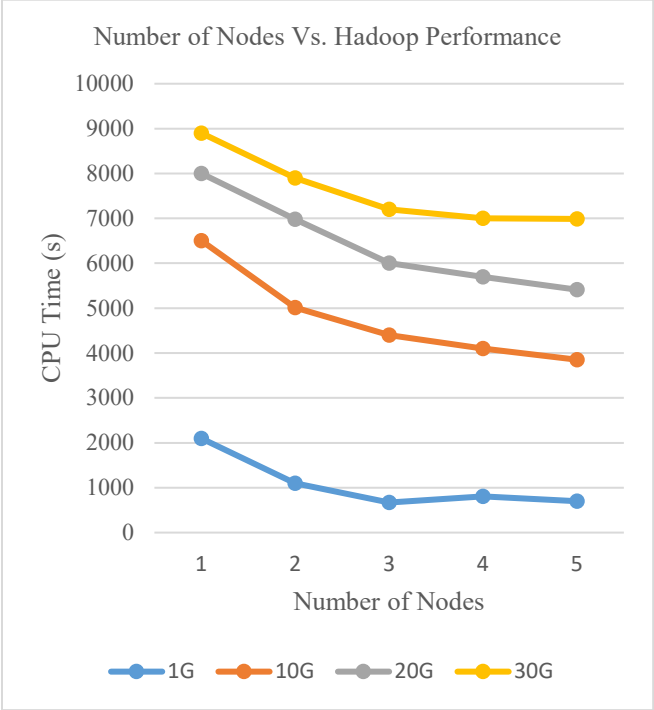

Figure 11: Number of Nodes Vs. Hadoop Performance

As we can infer from Figure 11, sorting all datasets using one node takes always longer than sorting them with more nodes. The time taken by the job using 1 node to sort 1Gb was about 2100s, and to sort 10Gb was 6500s, 8000s to sort 20Gb, and around 9000s to sort 30Gb of data. While using 5 nodes, the CPU time decreased significantly as it took only 700s to sort 1Gb of data, around 3800s to sort 10Gb of data, 5400s to sort 20Gb, and 6900s to sort 30Gb of data.

This means that adding Raspberry Pi nodes to the cluster increases the performance and reduces the CPU time taken by the jobs. Adding nodes means adding resources: memory size and CPUs. Knowing that each Raspberry Pi has 4 CPUs and 1Gb of RAM, working with 5 nodes implies working with 20 CPUs and 5Gb of RAM.

b. Impact of the Network

It goes without saying that any system that is distributed over a number of physical machines requires a network connection

between the nodes. This connection brings an overhead that affects the response time of the cluster.

In order to have a closer look at the impact of the network on the performance of Hadoop, we are connecting the 5 nodes of the cluster using two different types of switches with different throughputs: 100M and 1000M. Table 4 below describes both switches and presents their characteristics.

Table 4: NetGear Switch Vs. Fujitech Switch

	NetGear	**Fujitech**
Dimension	235.5x100.8x27	14.5x8.5x2.6
Network Characteristics	Gigabit Ethernet	Megabit Ethernet
Ports	8x10/100/1000	8x10/100
Energy Consumption	14W	-
Transfer Rate	1Gb/s	1Mbps
Weight	760g	-

Similar to the previous experiments, we are using the same benchmark and the same dataset sizes. The results of running Terasort using the two switches are shown in Figure 12.

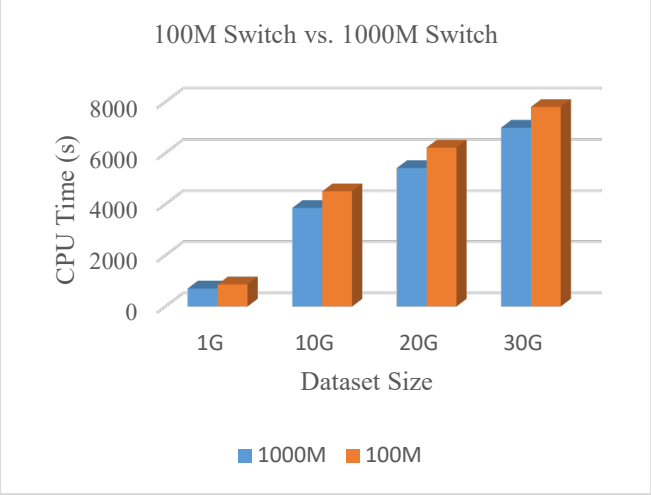

Figure 12: 100M Switch Vs. 1000M Switch

The results show the slight increase in the performance of the cluster as executing the same jobs using the 1000M switch takes less time than the other switch. The network clearly impacts the response time of the cluster. Hence, improving the network quality and equipment would result in a better performance.

c. Energy Consumption

Similar to the previous setup, we also looking at the energy consumption of the entire cluster.

Using the method described previously, we were able to measure the energy consumption of the cluster when sorting 1Gb of data and 30Gb of data. We used the same sensor for each of the slaves and we noticed that the power varies between 4W when jobs are being performed and 11W in an idle state. Regarding the master, the sensor showed a max of 6.7W and a min of 13W and that is because of the different external devices that are connected to it (i.e. mouse, keyboard, monitor, in addition to the external hard disk drive).

The sum of the values given by the sensors is shown in Figure 13 below.

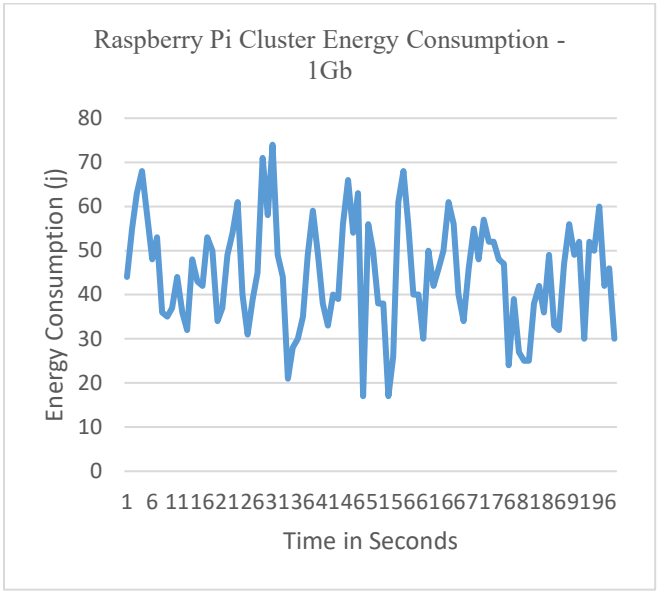

Figure 13: Raspberry Pi Cluster Energy Consumption - 1Gb

As we can notice, the entire 5-node cluster consumes between 17J/s and 70J/s. This is considered relatively low; we are looking into a detailed comparison later in this paper.

The energy consumed when sorting 30Gb of data using the same setup is shown in Figure 14.

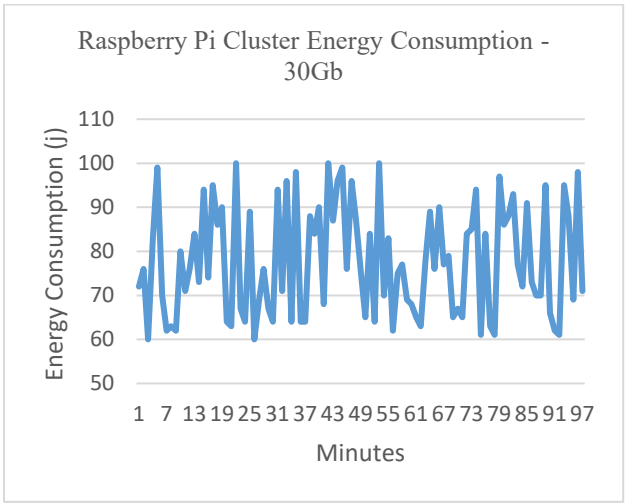

Figure 14: Raspberry Pi Cluster Energy Consumption - 30Gb

The energy consumed during the job is varying between 60J/s and 100J/s.

6. Comparison and Discussion

6.1. Performance Comparison: Single-node Vs. Multi-node Clusters

In order to have a general look over the difference between the two clusters, we are comparing the results given by all CPUs and full memory in the single-node cluster, and all nodes in the multi-node clusters.

The graph below shows the CPU time taken to sort all dataset sizes using both clusters.

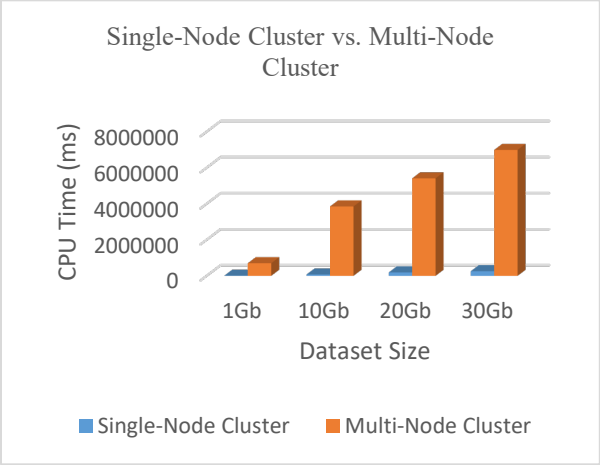

Figure 15: Single-Node vs. Multi-Node Cluster

Figure 15 above shows the clear difference between sorting the same dataset using the two different clusters. As the job becomes heavier (i.e. more data to be processed), the difference between the CPU times becomes more significant.

This implies that the 5-node cluster that we have is not enough to provide the same performance given by the single node cluster.

6.2. Energy Consumption Comparison: Single-node vs. Multi-node Clusters

As mentioned previously in this paper, our approach is supposed to be energy efficient as it tackles that concept of green computing. Thus, we are comparing the energy consumed by both clusters during the execution of the same job. We are comparing the first 100 seconds of sorting 1Gb of data using all CPUs and full memory for the single-node cluster, and all 5 nodes for the multi-node cluster. The result is shown in Figure 16.

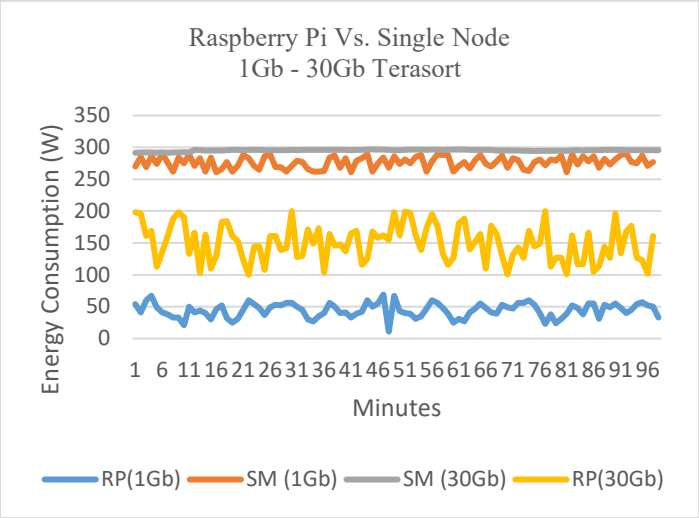

Figure 16: Raspberry Pi Cluster Vs. Single-Node Energy Consumption

The graph above shows the difference in the energy consumed by the two setups when sorting 1Gb and 30Gb of data using Hadoop. The results show that, not only the single machine consumes more energy, but also the RP cluster has a consumption that varies a lot compared to the single machine.

6.3. Discussion

The different experiments conducted in this paper were for the sake of determining how performant the Raspberry Pi green cluster can be.

According to the study conducted in [13] where the authors compared a commodity hardware cluster with a Raspberry Pi cluster using two different Hadoop benchmarks: Terasort and TestDFSIO. The results showed that the commodity hardware cluster outperformed the Raspberry Pi cluster. This was mainly due to the low computing power of the Raspberry Pi. The work performed in this paper sustains these results; In addition to that, since we are only using a single machine vs. a multi-node cluster, the network overhead is added and the performance is eventually decreased.

Another research has been done to test a Raspberry Pi Hadoop cluster against an image analysis in a cloud robotics environment [19]. Their research has proven that the Hadoop cluster lacks in performance compared to a Hadoop cluster that is based on virtual machines running on top of commodity computers.

Authors in [20] investigated the use of Raspberry Pi computers to implement an efficient solution for augmented computing performance. They used Hadoop along with benchmarks and compared the outcome to the one of a single commodity computer. The results showed that the single machine outperforms the clusters in almost all the operations. However, the low-cost and the light weight of the Raspberry Pi based solution makes it more suitable.

For the sake of our experiment, we are using the Amdahl's law to predict the performance of the Raspberry Pi cluster and decide on when the cluster will reach the performance of the single machine we are using.

According to the paralleled and serialized sections of the algorithm used, based on the Amdahl's law formula, the speedup is supposed to be as presented in Figure 17.

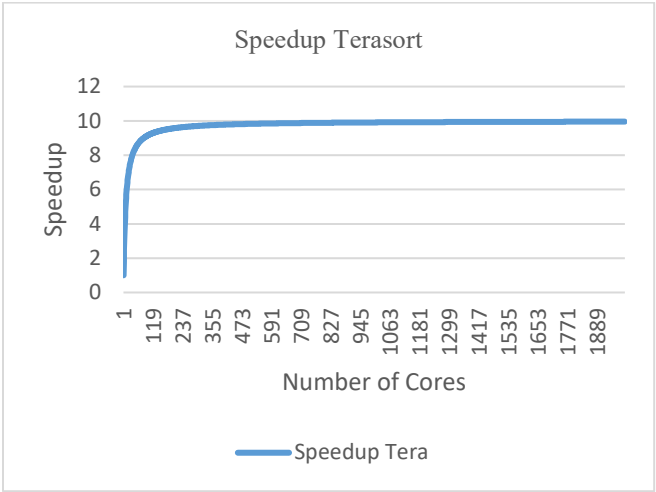

Figure 17: Speedup Terasort for Terasort Benchmark

We applied the formula for the number of cores from 1 to 2000 cores. Based on the speedup found, the performance is expected to increase according to the graph shown in Figure 18.

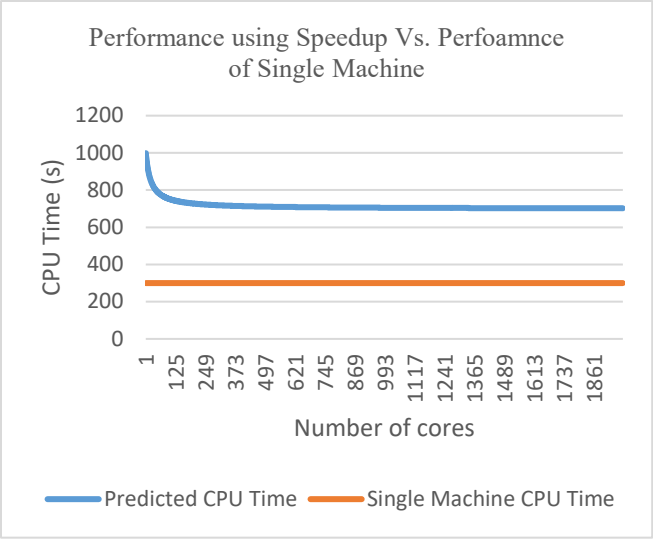

Figure 17: Performance of cluster using Speedup vs Performance of a Single Machine

Although we worked with 2000 cores (knowing that each RP has 4 cores, we are dealing with 500 Raspberry Pi nodes), we could not get the same performance as the single machine performance.

The best CPU time we achieved is 700 seconds using the Raspberry Pi cluster while the single machine performs the exact same job in 300 seconds.

At this level, we need to keep in mind that this study is theoretical, the real-world speedup can be given by the real measurements that we conducted and presented earlier in this paper. Also, the real-world measurements are affected by a number of other parameters and features: the network communication between the nodes that is, itself affected by the quality of the equipment used (i.e. switches as presented earlier).

The difference between the real-world and the theoretical results are shown in Figure 19.

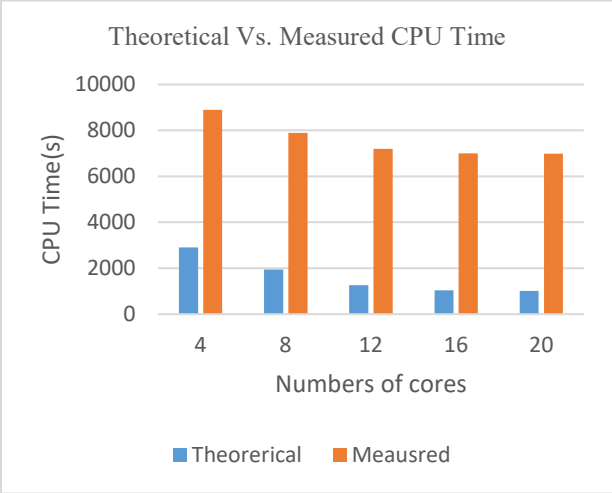

Figure 19: Theoretical vs. Measured CPU Time

Based on the theoretical approach that is built on Amdahl's law, the 500-node cluster will consume a minimum of 2000 W and 5000 W. In addition to that, the cluster will cost a minimum of $10000. Table 5 below summarizes the comparison between the single-node commodity computer and the Raspberry Pi 500-node cluster.

Table 5: Single-Node Vs. 500-Node RP Cluster

	Single-Node	**500-node RP cluster**
Price	$900	$10000
Max Energy Consumption	292W	Around 5000W
Min Energy Consumption	261W	Around 2000W
Scalability	Medium	Easy
Maintenance	Medium	Medium
Ease of use	Medium	Easy
Reliability	High	Medium

Oracle has been working on the world's largest Pi cluster where they used 1060 Raspberry Pi nodes to create a supercomputer.

According to [21], the cluster has 4240 cores for processing and it costs around $37100 without counting the external storage devices, the cables, 3D printed holders, etc.

7. Conclusion and Future Work

In this paper, we investigated the use of a Raspberry Pi clusters to provide High Performance Computing. The cluster performance was benchmarked against the Terasort algorithm and compared to a single legacy server machine. Both experimental setups run Hadoop. The experiments investigated the impact of the number of nodes and the network bandwidth on the performance of the Raspberry Pi cluster. Besides, we also tracked the impact of the number of cores and the RAM memory size on the performance of the single-node setup. We measured the energy consumed by both setups while performing the same operations.

The results of the experiments showed that the single-node cluster outperforms the Raspberry Pi cluster when sorting 1G, 10G, 20G, and 30G of data, but consumes less energy. However, and based on the Amdahl's law formula, the Raspberry Pi delivers a performance that is closer to the one of the single-node cluster when the number of cores reaches 2000. This implies having a cluster of 500 nodes that would cost around $10000 with a max energy consumption of 5000W. These results present a decent ground for researchers to base their choices on. If the performance is not a priority but the cost-effectiveness and the energy efficiency are, then the Raspberry Pi cluster is suitable. However, when the response time is essential, the most adequate solution is to opt for a more stable and performant solution.

As a future work, we are performing more projections on the number of CPUs used in a Raspberry Pi cluster, in order to determine (using prediction methods) the most suitable configuration according to the criteria/need of each application. Furthermore, we are examining more big data analytics solutions and investigating the possibility of putting in place a private cloud

that is based on Raspberry Pi. This will fall under the realm of green private clouds.

Conflict of Interest

The authors declare no conflict of interest

Acknowledgement

This work is sponsored by US-NAS/USAID under the PEER Cycle 5 project grant# 5-398, entitled 'Towards Smart Microgrid: Renewable Energy Integration into Smart Buildings".

References

[1] P. Abrahamsson, S. Helmer, N. Phaphoom, L. Nicolodi, N. Preda, L. Miori, M. Angriman, J. Rikkilä, X. Wang, K. Hamily, S. Bugoloni, "Affordable and energy-efficient cloud computing clusters: The Bolzano Raspberry Pi cloud cluster experiment," Proceedings of the International Conference on Cloud Computing Technology and Science, CloudCom, **2**, 170–175, 2013, doi:10.1109/CloudCom.2013.121.

[2] S.J. Cox, J.T. Cox, R.P. Boardman, S.J. Johnston, M. Scott, N.S. O'Brien, "Iridis-pi: A low-cost, compact demonstration cluster," Cluster Computing, **17**(2), 349–358, 2014, doi:10.1007/s10586-013-0282-7.

[3] F.P. Tso, D.R. White, S. Jouet, J. Singer, D.P. Pezaros, "The Glasgow raspberry Pi cloud: A scale model for cloud computing infrastructures," Proceedings - International Conference on Distributed Computing Systems, 108–112, 2013, doi:10.1109/ICDCSW.2013.25.

[4] N.J. Schot, "Feasibility of Raspberry Pi 2 based Micro Data Centers in Big Data Applications," 2015.

[5] W. Hajji, F.P. Tso, "Understanding the performance of low power raspberry pi cloud for big data," Electronics (Switzerland), **5**(2), 1–14, 2016, doi:10.3390/electronics5020029.

[6] M.F. Cloutier, C. Paradis, V.M. Weaver, "A raspberry Pi cluster instrumented for fine-grained power measurement," Electronics (Switzerland), **5**(4), 2016, doi:10.3390/electronics5040061.

[7] D. Marković, D. Vujičić, D. Mitrović, S. Ranđić, "Image Processing on Raspberry Pi Cluster," Ijeec - International Journal of Electrical Engineering and Computing, **2**(2), 2019, doi:10.7251/ijeec1802083m.

[8] D. Papakyriakou, D. Kottou, I. Kostouros, "Benchmarking Raspberry Pi 2 Beowulf Cluster," International Journal of Computer Applications, **179**(32), 21–27, 2018, doi:10.5120/ijca2018916728.

[9] A. Ashari, M. Riasetiawan, "High performance computing on cluster and multicore architecture," Telkomnika (Telecommunication Computing Electronics and Control), **13**(4), 1408–1413, 2015, doi:10.12928/TELKOMNIKA.v13i4.2156.

[10] M. Alexander, W. Gardner, B. Wilkinson, M.J. Sottile, T.G. Mattson, C.E. Rasmussen, Y. Robert, F. Vivien, Computational Science Series petascale computing: algorithms and applications edited by david a. bader process algebra for parallel and distributed processing.

[11] Introduction to HPC & What is an HPC Cluster? | WekaIO, Dec. 2020.

[12] High performance computing: Do you need it? | Network World, Dec. 2020.

[13] S. Bourhnane, M.R. Abid, R. Lghoul, K. Zine-Dine, N. Elkamoun, D. Benhaddou, "Towards Green Data Centers," in Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST, Springer: 291–307, 2020, doi:10.1007/978-3-030-45694-8_23.

[14] What is a Raspberry Pi?, Dec. 2020.

[15] Comparatif des modèles de Raspberry PI | Tableaux comparatifs - SocialCompare, Dec. 2020.

[16] M.R. Abid, R. Lghoul, D. Benhaddou, "ICT for renewable energy integration into smart buildings: IoT and big data approach," 2017 IEEE AFRICON: Science, Technology and Innovation for Africa, AFRICON 2017, 856–861, 2017, doi:10.1109/AFRCON.2017.8095594.

[17] Arduino - Introduction, Dec. 2020.

[18] Non-Invasive Sensor: YHDC SCT013-000 CT used with Arduino. (SCT-013) – PowerUC, Dec. 2020.

[19] B. Qureshi, Y. Javed, A. Koubâa, M.F. Sriti, M. Alajlan, "Performance of a Low Cost Hadoop Cluster for Image Analysis in Cloud Robotics Environment," Procedia Computer Science, **82**(March), 90–98, 2016, doi:10.1016/j.procs.2016.04.013.

[20] K. Srinivasan, C.Y. Chang, C.H. Huang, M.H. Chang, A. Sharma, A. Ankur, "An efficient implementation of mobile Raspberry Pi Hadoop clusters for Robust and Augmented computing performance," Journal of Information Processing Systems, **14**(4), 989–1009, 2018, doi:10.3745/JIPS.01.0031.

[21] Oracle: This 1,060 Raspberry Pi supercomputer is "world's largest Pi cluster" | ZDNet, Dec. 2020.