

Parallel and Distributed Computing Using MPI on Raspberry Pi Cluster

Khin Nyein Myint, Myo Hein Zaw, and Win Thanda Aung

Abstract—Parallel and distributed computing is a complex and has been become essential research topic area in computer science. The birth of high performance computing can be traced back to the start of commercial computing in the 1950's. In this paper, an approach of parallel and distributed computing using raspberry pi cluster had been proposed. Odd even transaction sorting algorithm on Rpi cluster had been implemented and then compare our performance with node 2 to node 10, totally forty cores of computer cluster. The MPI library is selected to establish the communication and synchronization between the processors. It has been also compared upon the sorting method with one to forty processes on Rpi for various processes. According to the test results, the cluster can give high performance speed for above five millions integers with forty cores. Conveniently, we hoped that this paper may provide an opportunity for computer science students to learn and understand about HPC using super cheap computer and then how to apply the parallel and distributed computing using MPI.

Index Terms—Cluster computing, distributed computing, HPC, MPI, parallel computing.

I. INTRODUCTION

Distributed computing is the process of combining the power of several computing tasks, which are logically distributed and may even be geologically distributed, to collaboratively run a single computational task in a transparent and coherent way, so that they appear as a single, centralized system. In distributed computing, a problem is divided into many tasks, each of which is solved by one or more computers or cluster of nodes which communicate with each other via message passing mechanisms. The message parsing interface (MPI) is meant to provide essential virtual topology, synchronization and communication functionality between a set of processes. Typically, for maximum performance each CPU (or core in a multi-core machine) will be assigned just a single process.

Message passing is an activity where the processors coordinate their activities by explicitly sending and receiving messages. MPI is a de facto standard for parallel programming on distributed memory systems. The most

important advantages of this model are twofold: achievable performance and portability. Performance is a direct result of available optimized MPI libraries and full user control in the program development cycle. Portability arises from the standard API and the existence of MPI libraries on a wide range of machines. It is the most common method of programming parallel and distributed system. MPI is considered today's standard in message passing library. MPI can be used with C/C++, Python and many other languages for parallel computers, clusters, and heterogeneous networks

This paper represents the outcome of a hands-on opportunity to better understand distributed computing, parallel performance using MPI libraries, how to work odd even transition sorting algorithm on pi cluster and its potential benefits to higher education.

II. THEORETICAL BACKGROUND

A. The Art of High Performance Computing

The first type of high performance computing was mainframe computing. One of the main tasks required was billing, a task that almost every type of business needs to perform and is conveniently run as a batch process. Batch processing allows a sequence of several programs or "jobs" to be run without manual intervention. Batch processing saves processing time that is normally wasted with human interaction and jobs can be processed in shifts, allowing the more interactive or urgent processes to run during the day shift and billing or non-interactive jobs to be run during the night shift.

In the 1970's, the manufacturers of supercomputers shifted computer models into personal computing, increasing the performance of personal computers. After the advent of the CRAY-1 super computer in 1976, vector computing took over the high performance marketplace for 15 years. CRAY-1 used RISC (Reduced Instruction Set) processors and vector registers to perform vector computing. In the late 1980's, IBM connected RISC microprocessors by using the butterfly interconnection network. This allowed developers to create systems with consistent shared memory caches for both processing and data storage DASH (Dual Access Storage Handling) was proposed by Stanford University in the beginning of the 1990's. DASH achieved consistency of distributed shared memory cache, by maintaining a directory structure for data in each cache location. Today, more and more parallel computer systems use commercial microprocessors and the interconnection network structure. This distributed memory parallel computer system is known as clustering. Parallel computers have entered a new era

Manuscript received October 19, 2019; revised January 20, 2020. This research was primarily conducted as an independent research project by Parallel and Distributed Computing Using MPI on Raspberry Pi Cluster for University of Computer Studies, Monywa (UCS-Monywa).

Khin Nyein Myint is with the Faculty of Computer Science, University of Computer Studies, Monywa, Myanmar (e-mail: khinnyeinyint.knm@gmail.com).

Myo Hein Zaw is with University of Computer Studies, Monywa, Myanmar (e-mail: mheinzaw@gmail.com).

Win Thanda Aung is with the Faculty of Information Science, University of Computer Studies, Pinlong, Shan State, Myanmar (e-mail: winthanda.monywa@gmail.com).

where there is currently unprecedented development [1].

High performance computing (HPC) refers to the computing system, including several processors as part of a single machine or a cluster of several computers as an individual resource. High performance computing owes its feature of high speed computing to its great ability to process information. Therefore the main methodology that is currently applied to high performance computing is parallel computing .

B. Power of Cluster Computing Requirements

Clusters, built using commodity-off-the-shelf (COTS) hardware components and free, software, are playing a major role in solving large-scale science, engineering, and commercial applications. Cluster computing has emerged as a result of the convergence of several trends, including the availability of inexpensive high performance microprocessors and high speed networks, the development of standard software tools for high performance distributed computing , and the increasing need of computing power for computational science and commercial applications [2],[3].

Clusters have evolved to support applications ranging from supercomputing and mission-critical software, through web server and e-commerce, to high performance database applications. Cluster computing provides an inexpensive computing resource to educational institutions. Colleges and universities need not invest millions of dollars to buy parallel computers for the purpose of teaching "parallel computing". A single faculty member can build a small cluster from student lab computers, obtain free software from the web, and use the cluster to teach parallel computing. Many universities all over the world, including those in developing countries, have used clusters as a platform for high performance computing.

C. The Most Popular Computational Power (Super Cheap Cluster)

One of the challenges in the use of a computer cluster is the cost of administrating it which can at times be as high as the cost of administrating N independent machines, if the cluster has N nodes. In some cases this provides an advantage to shared memory architectures with lower administration costs [4]. This has also made super cheap clusters popular, due to the ease of administration. The Raspberry Pi is an inexpensive computer that rivals the size of a credit card. Linking a series of Raspberry Pi into a distributed computing system would just as well create a very small sized cluster. Those performing research into the field of distributed computing could potentially use these small scale clusters as a personal research tool [2].

Next trend, the Green computing technology or initiative is currently becoming popular in various corporate organizations. However, there is actually nothing new about such innovative concept. The concept of green computing has been around 1992, when the Environmental Protection Agency(EPA) in US initially begun the Energy Star project, which is a voluntary labeling program established to promoting energy efficiency among computer hardware of corporate organizations. This cluster is a green computing device because this can save energy, heat and money.

D. Speedup for Parallel Computing

Gustafson's law (or Gustafson–Barsis's law [5]) gives the theoretical speedup in latency of the execution of a task *at fixed execution time* that can be expected of a system whose resources are improved. Speedup is a first-hand performance evaluation in parallel computing. Total amount of work to be done in parallel varies linearly with the number of processors. However, it is a controversial concept, which can be defined in a variety of ways. The Gustafson–Barsis's Law describes any sufficiently large problem can be efficiently parallelized with a speedup,

$$S = p - \alpha (p-1) \quad (1)$$

where S is the speed up, p is the number of processors, and α is the serial portion of the problem. Gustafson proposed a fixed time concept which leads to scaled speedup for larger problem sizes.

III. RELATED RESEARCH WORKS

At the Midwest Instruction and Computing Symposium 2017 (MICS 2017), researchers from Department of Computer Science Augsburg College presented a cluster based on eight Raspberry Pi 3 Model B modules [6]. The realized cluster was tested using the Monte Carlo method for calculating the value of Pi. Calculation was performed on eight nodes with one to four processes per node.

Similarly to our work, d'Amore et al [7], proposed the use of Raspberry Pi boards. They built a cluster of six Raspberry Pi boards to evaluate the performance of Big Data applications. The authors concluded that the cluster is an affordable solution to their problem. It is just a dawn of Big Data application.

Alternatively, in this research odd even sorting algorithm is tested 10 nodes with 40 cores RPi cluster. Five millions integers are transported in even and odd. After that, even and odd are sorting in ascending order. The contribution of current research is mathematical model of performance using empirical methodology.

IV. SYSTEM DESIGN AND IMPLEMENTATION

An educational RaspberryPi Cluster composed of ten computational nodes has been built to compute parallel programs. Raspberry Pi 3model B has the following features:

- CPU: Quad-core 64-bit ARM Cortex A53, 1.2GHz
- GPU: 400MHz Video Core IV multimedia
- Memory: 1GB LPDDR2-900 MHz SDRAM
- USB: 4 ports
- Video outputs: HDMI, composite video (PAL and NTSC)
- Network: 10/100Mbps Ethernet and 802.11n Wireless LAN
- Peripherals:17 GPIO, HAT ID Bus
- Bluetooth: 4.1
- Power source: 5V via Micro USB or General Purpose Input Output (GPIO) header.

The appearance of the Raspberry Pi module is as shown in Fig. 1

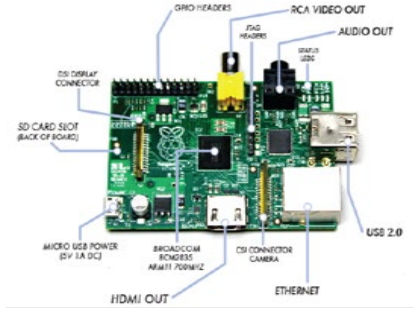


Fig. 1. Raspberry Pi 3 Model B.

Total ten RPi boards are used to build super cheap cluster for this research. One single board processor consists of 4 cores. So, our cluster has 40 cores to compute any parallel execution.

Hardware Requirements for implementing this cluster and the main total costs in Myanmar kyats for build this cluster is shown in Table I.

TABLE I: CLUSTER PART LIST

No	Item	Price (Kyats)	Qty	Cost
1	Rpi 3 Model B	67,000	10	670,000
2	Dlink 24ports switch	80000	1	80000
3	USB power adapter	2500	10	25000
4	8G SDcard	8000	10	80000
5	60 meter network cable	3000	10	30000
6	Cluster frame	20000	2	40000
Total				925000

The total costs for implementing this 40 cores cluster is not more than 10 lakhs in Myanmar kyats. So without question, it is a really super cheap cluster machine.

Software requirements of this implementation are as follow,

- Raspbian (a free operating system based on Debian GNU/Linux 9)
- Python 3.6 programming tools
- MPICH3 and MPI4Py library

Architecture design for current research is shown in Fig. 2.

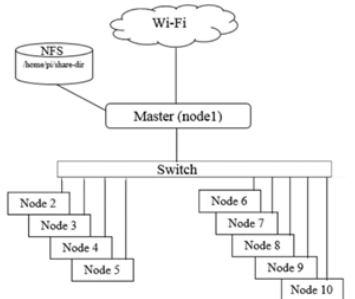


Fig. 2. High performance computing system architecture.

The fast Resbery pi moedel B is assign as a Head node and the other 9 components are assigned as slave nodes. Network File System (NFS) is used for this cluster configuration. For current research, 24 ports switch is used for connecting device. For next step, this connecting device can be replaced with a router which can assign public IP address for

communication via internet. This concept is used to geographically distributed clusters can be collaborated for high performance computing.

According to the logical modal and network diagram, bench mark Rpi cluster is implemented. The appearance 10 nodes and 40 cores Rpi rack cluster of UCS Monywa is as shown in Fig. 3.



Fig. 3. An appearance of implemented cluster in UCS (Monywa).

Consequently evaluation of each node is tuned by well know MPI library program. The performance of each node is shown in Fig 4. The evaluation of each node is 2.942×10^{-01} Gflop. This result is really good more than expectation and feasible to use in computer engineering simulation and HPC.

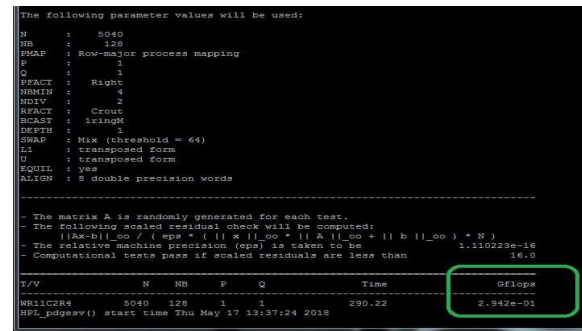


Fig. 4. Evaluation of each node.

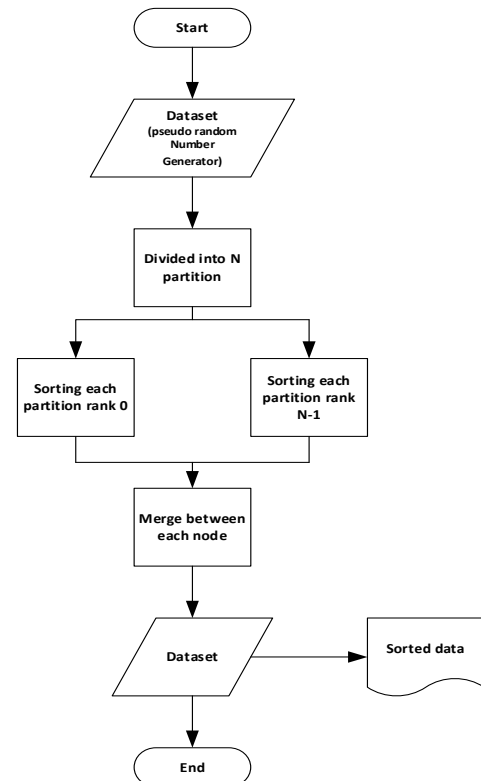


Fig. 5. Flow chart for odd even transposition sort program.

As a consequently 40 cores clusters is used to sort odd even transposition sorting algorithm. The required integer numbers are generated from pseudo random number generator. According to the mathematical formulation, these millions integer are gotten and then categorized by odd even phase. And then sorts this data in ascending order with 40 cores of processors. Fig. 5 illustrates the odd even transposition sorting program design.

V. EXPERIMENTAL RESULTS

These five millions integers are handled by selected cores of processor depend on various thread from 1 to 40. The results of the research are compare with Gustafson's law. After that, mathematical model is driven based on the results of the research using empirical method.

The bench mark cluster is built using proposed logical model and then performance of N processors is denoted. Follow by the Gustafson's Law, to get the peak performance of cluster that the tasks is divided into 40 processes for Rpi 40 cores cluster. The performance results depend upon the number of processors and threads are shown in Table II.

TABLE II: PERFORMANCE RESULT (IN SECONDS) TABLE FOR NUMBER OF PROCESSES VS PROCESSORS

Number of processes	10	20	30	40
Number of processor				
2	15.58	20.54	48.34	67
3	11.30	22.25	27.84	43
4	8.60	15.81	21.98	31
5	8.48	13.62	18.12	25
6	8.11	14.44	16.15	22
7	7.73	13.76	15.05	19
8	7.71	10.18	12.63	17
9	7.63	10.18	12.55	15
10	7.29	10.01	11.21	13

According to the tested results, performance versus number of processors and thread graph for 40 processes on 40 cores clusters is shown in Fig. 6.

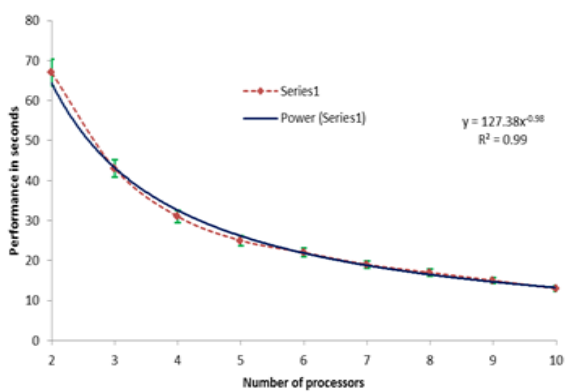


Fig. 6. Performance analysis of Rpi cluster for odd-even transposition sort.

Series1-performance result of the research

Power- estimated power trend line for equation

Applying empirical research methodology, mathematical trend line is driven as follow,

$$T \propto x^* e^{-z} \quad (2)$$

where T – performance of N cores cluster; x – number of processors.

Rpi clusters performance is depend on the number of processor in exponentially.

Hence, Performance of N cores cluster for N process can be predicted as follow:

$$y = 127.38x^{-0.98}$$

$$R^2 = 0.99$$

The proposed mathematical model is the accuracy of 99% and tolerance is 1% in empirical methodology. Fig. 7 illustrates speed up for Rpi cluster. This figure describes the speedup as shown in graph, according to the number of processors. This trend line shows nearly as linearly speedup.

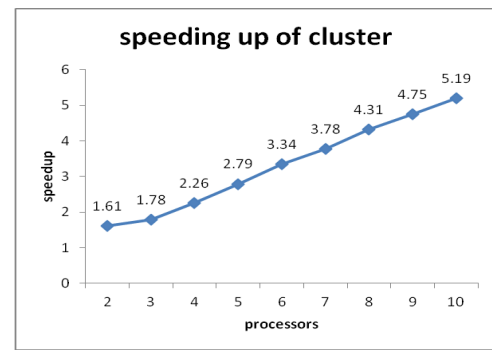


Fig. 7. Speedup of the odd-even transposition sort algorithm using Rpi.

VI. CONCLUSION AND FURTHER EXTENSION

Overall, the Rpi Cluster has proved quite successful. The result performances have been shown perfectly acceptable for research simulation needs. This super cheap cluster has great potential to test and research different and new methods in the field of parallel and distributed computing. Moreover FPGA's can be connected to each raspberry pi node as accelerators. Tradeoffs between General purpose parallel programming and FPGA accelerators can be analyzed. Exploring different input data for example, Voice commands, Sensor Data. A real time IOT application can be chosen and parallelized to analyze the performance.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Myo Hein Zaw guide Khin Nyein Myint to implement this research using Raspberry Pi cluster and testing the result for parallel and distributed computing. Win Thanda Aung is recorded the result running time and drawing the data flow diagram for algorithm. Khin Nyein Myint is writing this research paper and testing this device using sorting algorithm. To publish this research the authors are working all together.

ACKNOWLEDGMENT

Firstly, I would like to acknowledge to my supervisor Dr.Myo Hein Zaw, Professor, University of Computer

Studies, Monywa for his encouragement, valuable comments and close guidance for the implementation of this research. Secondly, I would like to express my sincere gratitude to my colleagues for their invaluable support and encouragement for this research.

REFERENCES

- [1] L. Li, "High performance computing applied to cloud computing," thesis, Technology, Communication and Transport Degree Programme in Information Technology, Lapland University of Applied Science, 2015.
- [2] R. Buyya, *High Performance Cluster Computing: Systems and Architectures*, Prentice Hall, 1999.
- [3] R. Buyya, *High Performance Cluster Computing: Programming and Applications*, Prentice Hall, 1999.
- [4] D. Mitrović, D. Marković, and S. Randić, "Raspberry Pi module clustering and cluster application capabilities," in *Proc. 7th International Scientific Conference Techniques and Informatics in Education Faculty of Technical Sciences*, Čačak, Serbia, 25 -27th May 2018.
- [5] M. McCool, A. D. Robison, and J. Reinders, *Structured Parallel Programming: Patterns for Efficient Computation*, 2012.
- [6] G. Dorr, D. Hagen, B. Laskowski, E. Steinmetz, and D. Vo, "Introduction to parallel processing with eight node raspberry Pi cluster," in *Proc. Midwest Instruction and Computing Symposium (MICS2017)*, 2017.
- [7] M. d'Amore, R. Baggio, and E. Valdani, "A practical approach to big data in tourism: A low cost raspberry Pi cluster," *Information and Communication Technologies in Tourism 2015: Proceedings of the International Conference in Lugano*, Springer, 2015, pp. 169-181.

Copyright © 2020 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).



Khin Nyein Myint was born in 1981. She lives in Monywa. She is a master degree holder. So, she got her Bachelor degree specializing in computer science, at university of computer studies, Mandalay (UCSM) in 2001 and master degree in 2003. She did her research thesis on analysis system for traditional medicine and their usage, UCSM in 2003. She has nearly 16 year experience on computer science teaching. First, she was as a demonstrator for 1 year at UCSM. Then, she had transferred to the government computer collage (Kalay) which is situated near Chin state that is the lowest development state of Myanmar. After her experience had 7 years, she got a promotion as an assistant lecturer at computer university Monywa for her valuable usefulness of previous research work. Then she has transferred to computer university (Kalay) again for her previous performance at that university in 2014. After that, she had transferred to her native town university of computer studies, Monywa (UCS-Monywa) in 2017 with lecturer promotion from department of higher education of Myanmar. Now she is working as a lecturer at faculty of computer science of UCS-Monywa. Furthermore, she is trying a popular research Parallel and Distributed Computing Using MPI on Raspberry Pi Cluster in UCS-Monywa.



Myo Hein Zaw received the BE (IT) in mandalay technological university in 2002. He received the ME (IT) in 2005 and Ph.D. in 2008 from moscow institute of electronic technology (MIET). He is a professor in university of computer studies (Monywa). He is a professional member of IEEE association. His current research interests include distributed computing, grid computing and internet of things technology.



Win Thanda Aung received the Bachelor and Master degrees in the computer science at university of computer studies, Yangon (UCSY), in 2001 and 2003, respectively, and the PhD (IT) degree in UCSY in 2010. She is an associate professor in faculty of information science in university of computer studies (pinlon). Her research interests include data mining and web mining.