ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

A Study on Cloud Robotics: Ad-Hoc Cloud (Cloud Seeding)

Shepard Chifamba

PG Student, Dept. of I.T, Cloud Computing, SRM University, Chennai, India

ABSTRACT: Cloud seeding in cloud robotics is the concept of forming an adhoc cloud using the available robot resources. A team of robots working in the same field utilizing cloud robotics might experience a connection failure to the main node however this should not stop field work. The teamed robots surrender their resources to form a virtual adhoc cloud not only to load balance tasks but to share resources and information. In this paper the researcher explores further on how cloud seeding can best be done, the security implications as well as networking concerns involved. This however is not a permanent infrastructure but a way of circumventing the challenge of network failure between the main cloud infrastructure and the field robots in cloud robotics.

KEYWORDS*:* robotics; network; adhoc cloud; cloud infrastructure.

I. INTRODUCTION

Dating back to the medieval times, when the Greeks and the Romans are said to have attempted to automate their machinery as well as toys[1]; robotics has evolved over time as man endeavoured to make their chores lighter and easier. During the Victorian era as increase of focus on mathematics, engineering and science increased due to the industrial revolution in England; this added the need for automation as productivity became essential. Not only in manufacturing were robots used but also entertainment. According to NASA the term "robot" was first used in a play called "R.U.R." or "Rossum's Universal Robots" by the Czech writer Karel Capek.[2][18]. In the play a man would make a robot and the robot would kill the man. As fountains of computing were laid the automation of machines embraced programming and remote controlled machines were developed. These would work in conditions humans could not go in yet they were operated real time by a human operator.

According to Edwyn Gray remote operated vehicles were there as early as 1872 with John Erricson having invented a torpedo for the king of Sweeden[4]. The evolution enfolded mostly in the early twentieth century with major industries automated. The development of the microprocessor saw the robot having the ability to process and store its won data. Autonomous robots can be found in the entertainment industry with debates still hanging in the defence industry with concerns of conscience being questioned with David Hambling writing “the U.S. has been testing armed robots for decades. But while political and ethical caution has prevented the West from advancing with the concept”[5]

The concept of cloud robotics has roots in remote brains robots by Inaba as early as 1993. His effort was to minimise hardware resources on the robot. During that period the compute resources were bulk and heavy thus meant a robot would need even more power to carry the computer along. Inaba’s concept was to have the robot separate for the compute and with this approach they intended to utilise large-scale powerful parallel computers in processing. [6]

Around the early 1990s the internet was growing giving rise for the opportunity of interconnection as well as communication between computing entities. This later developed several approaches like service oriented architecture giving rise to cloud computing. Cloud computing which is defined as the provisioning of compute as well as storage resources over the internet on a utility basis whose characteristics among others include elasticity and self-provisioning. Cloud computing opened a gateway for efficient outsourcing of compute resources accessible any where in the world. As such the field of robotics joined the bandwagon and research is still going on in an attempt to have cloud based robotic systems which would ensure unlimited resources for field robots. These utilise a network infrastructure to communicate with the cloud and amongst them. A network of robots is any group of robotic devices connected to each other either via wire of wireless to achieve either a single or distributed tasks [3].

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

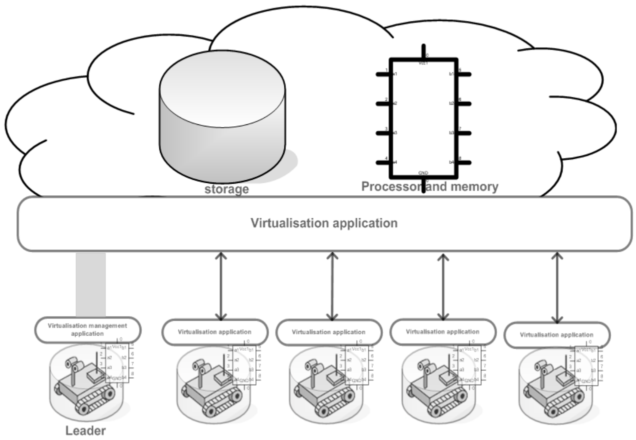


Fig. 1 Cloud seeding architecture

Networked robots can be classified as teleoperated or multi robotic systems according to Guoqiang Hu, et al [3]. Teleoperated robots work with a human operator controlling the robot(s) using a communication network. However in cloud robotics there is need for a gateway to the cloud and there a various approaches to the connection. Each robot independently connects directly to the cloud via wireless communication or in a group of robots a leader is elected to be the gateway in such a way that it routes all the traffic in the system.

In case of network failure or cloud failure, although there are various ways to ensure there is low failure rate in cloud computing, the robots should be able to quickly form their own adhoc cloud. Since they are networked and operating in the same network they would contribute their resources into an abstraction level. This approach would give a singular collective view the resources such that they can be share on-demand self service in a redundant way; by creating “cloud of dust” in an approach we propose to call cloud seeding.

This paper would be organised as follows, section II a detailed architecture of an adhoc cloud and the concept of cloud seeding, section III discusses the protocols and the networking construct in cloud seeding, it also details the security concerns in this approach. Section IV would critically analyse the challenges and opportunities in cloud seeding.

II. CLOUD SEEDING PROCESS AND CLOUD ARCHITECTURE

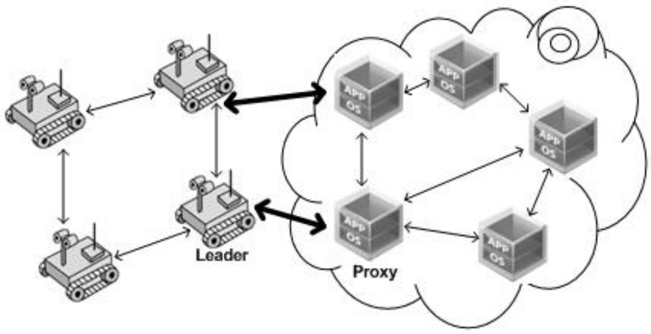
Individual field robots have their own resources, that is, compute, memory and storage, with them as they execute their tasks in the field. These can now be augmented by extending their capacity to the cloud through a process called offloading. Certain parameters can be set so as to offset the offloading otherwise if the task can be handled by the onboard resources then there would not be any need to offload. There are three architectures proposed by Guoqiang Hu, et al [3] for the construct of a cloud robotics system. These include Proxy-Based Model, Peer-Based Model, and Clone-Based Model.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

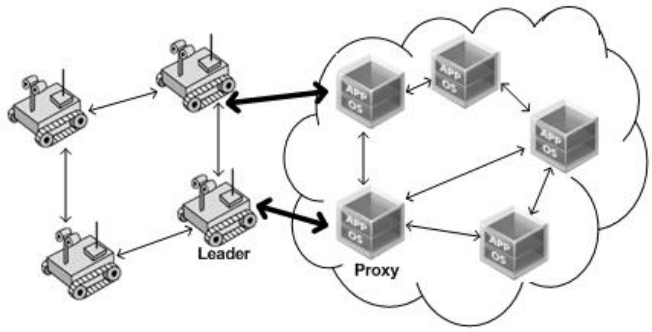
International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

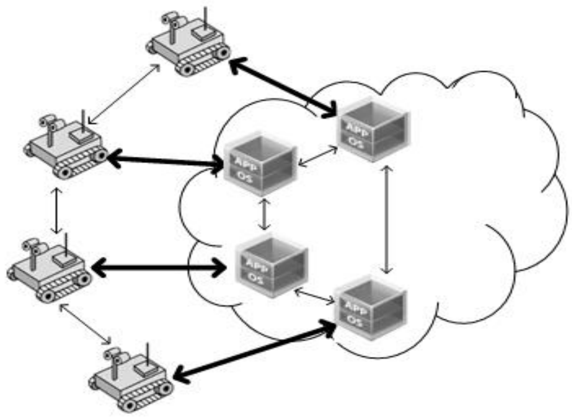
Vol. 3, Issue 4, April 2015



Fig, 3 Peer-Based cloud robotics model[3]



Fig,2 Proxy-Based cloud robotics model[3]



Fig,4 Clone-Based cloud robotics model[3]

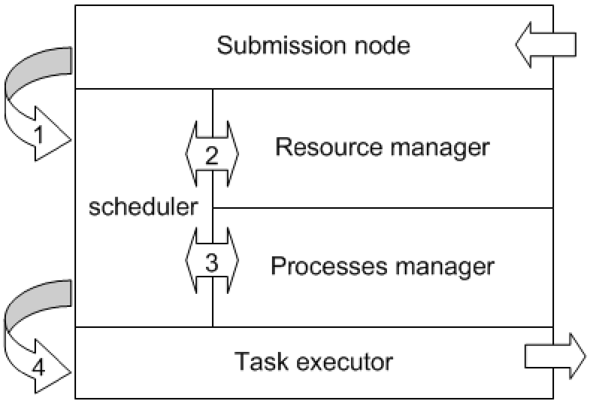


Fig 5 Seeder software structure

Proxy-Based Model: As shown in Fig. 2, A proxy VM is deployed which communicated directly with a robot leader to form a bridge between the Cloud and the robot network.

Peer-Based Model: From Fig. 3 the connection is clearly shown as a full mesh where the robots and the VMs make up a network, each of these (robot or VM in the cloud is taken as a computing unit.

Clone-Based Model: each robot has a corresponding "clone" VM in the cloud. There is a greater relationship between each pair such that a task can run in the robot or in its clone, Fig 4.

However this approach relies on an existing cloud infrastructure which is separate from the robots, in our approach we are looking at a scenario where the connection to infrastructure is lost and the robots have to form their own adhoc cloud by contributing their own resources for the formation of such a cloud. Any of the models discussed above can mutate into adhoc in-case of failure of connectivity, however with Peer-Based model and Clone-Based Model chances of total connection failure are very low compared to the Proxy-Based Model where there is centralised connectivity.

Each robot in the team should be aware of its own resources, capacity as well as utilisation at any given point. This would enable it to quickly surrender them for adhoc cloud formation. A team leader is elected; in Proxy-Based Model there is no need to elect a leader because it is already there. Election of a leader can be done based on four criterion, Capacity based, Last Connected and or proximity to the access point and position in relation to other robots.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

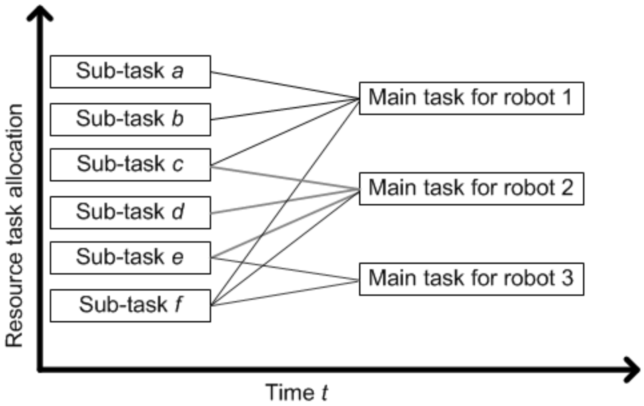


Fig 6 Task execution example using parallel model in task the executor

- Capacity based: The robot with the largest free capacity at the point of cloud formation would be the leader. This would ensure that some background processes can be accommodated unlike over loading a busy robot.
- Last connected: the robot that had last access to the cloud or that was last connected to access point is made leader. This is with the assumption that it ill be the first to be connected in case of connection reestablishment. This would ensure quicker offloading at the earliest possible time.
- Proximity to access point: this works with the pretext that when connectivity is re-established, it will have the strongest signal reception strength but however since the robots would be highly mobile this would mean changing of leadership rapidly.
- Position in relation to other robots: this model would elect the robot that is closer to every robot in the team as the leader. This would avoid a large number of hopes between the initiator robot and the leader, communication can be direct. However because of the mobility of the robots there can be need for periodic re-election of the leader.

The task of the leader is to coordinate resources usage and run an administrative role in the network. As per task priority as submitted by the robots, the power usage as well as remaining. A robot running a critical task would be allocated more resources and there one with least in priority can be put to rest and its resources used for other critical tasks. The leader is also responsible for robbing the main access to point at intervals to check is connection is up again.

The researcher proposes an application (Seeder software which is the virtualisation layer manager) that should run on every robot in the team. The Seeder software manages the adhoc-cloud's use of resources. It aids in the identification of the robots and authenticating them before the cloud is formed. Each robot should have an ID and a tag of its team so as to make authentication faster since they are all aware of each other in the team. After they all are authenticated, they choose a leader.

Once a robot is elected leader and administrative function should be automatically activated on it. This Seeder software would virtualise the physical resources, aggregate them and give them a logical view using the resources manager. For instance, storage is summed up and viewed as a single drive that can be partitioned or concatenated depending on requirements. The same for processor and memory. Allocation of these resources by leader would be done as per request and requirements, dynamically with a higher degree of elasticity.

Each robot should advertise its resource requirements, utilisation, capacity and the priority of its task; this is under the control of the process manager. If its tasks are of low priority that robot can be relegated to a data gatherer or its sensors are utilised by another robot while its resources are given up for use to the cloud. Cloud seeding would employ thin provisioning [8] in allocation of resources. No one robot should hold on to the resources it is not utilising. A task can be seen as a multiple of sub tasks that are executed remotely each executing on a robot in the team.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

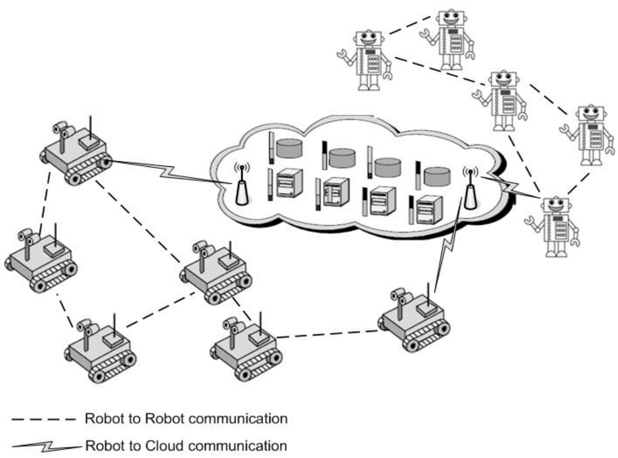


Fig 7 Robot-Cloud network

Due to capacity constrains tasks may not run as they would with a infrastructure based cloud, hence a criteria to give up tasks should be scheduled so as to ensure resources are available to critical activities. There are basically two determinants of which tasks to rest and which robot to continue.

- Power level and consumption rate: a robot with low power should be relieved of its task so as to keep power for resources to be used by other critical tasks. Also if the consumption rate of power a certain robot is high, that robot should recede into a low action state. This will help maintain power in all robots so as to prolong the availability of maximum capacity possible with the available resources.
- Task based: Robots running critical tasks should be kept active and those with least priority should be kept at minimal consumption so as to maximum resources are made available.

III. NETWORK CONSTRUCT FOR CLOUD SEEDING

Each robot has the capability of directly communicate with the other utilising standards [9] like WiFi Direct, Zigbee, and Bluetooth for short range and on a wider area they can make use of radio frequencies [10] and micro wave [11].

Due to the mobility of the robots there is need for protocols that supports and adapts fast in such an environment. As discussed earlier, there is a leader, and it is this leader that controls the network but however should not be the centre of the network since they would lead to a bottle neck of communication there by delaying data transfer.

High availability protocols with low latency as well as incurring low communication load in the network. There are various protocols for Mobile ad hoc networks (MANET)[13] and most of them are based on some variants of flooding. With this many routing messages are propagated through the network, unnecessarily increasing traffic as well as overheads despite various optimizations. However we propose a gossip routing protocol [14]in which each network node forwards a message to the nearest neighbour with some probability, to reduce the overhead of the routing protocols.

IV. HEDDLES AND CHALLENGES IN CLOUD SEEDING

In an attempt to device ways of forming an adhoc cloud for robots there are still many challenges that need to be addressed for the implementation of such an idea to be realised. This however being theoretical based there would be greater need for practical verifications.

A. Computation challenges

The greatest strength of an adhoc cloud is its ability to sustain critical task execution when the infrastructure based cloud is unavailable for any reason that can range from connection failure or choice of forming one. It pools

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

together a maximum possible capacity of the resources available so as to effectively and efficiently utilise them. In general the resources available would be as follows:

i. Processor capacity

$\pi = \sum Rcpu_i$

ii. Memory available

$\alpha = \sum Rmem_i$

The seeder software will logically concatenate the memory and manage the pages as if it was a single chip.

iii. Storage available

$\Omega = \sum dB(R_{dk})$: where dB is the storage capacity of each individual robot and R_{dk} denotes each individual robot's disk(s)

The logical storage can be concatenated or partitioned depending on need.

B. Communication challenges

In this proposal, time delays in communication between robots can not be tolerated since this would cause constrains on the resources seeded into this cloud. Thus a routing protocol, as discussed before, with low communication load on the network and with low latency would be the best for this scenario. Reducing overheads in communication would be the priority. However the proposed gossip protocols has high failure rate. But this trades off with ability to significantly lower overheads.

As discussed by Guoqiang Hu et al [3] when using gossip protocol, each robot chooses a neighbour randomly to send a message.

As supposed, if node i chooses node j with probability P_{ij} , where a zero probability implies that the two nodes can not communication due to communication range, and are therefore not neighbours. It can be shown[f] that the communication delay of disseminating a message from a single node in the M2M network to all N nodes in the network is O(logN/Φ), where Φ is the conductance of the network, given by

$$\Phi = \min_{S:|S|\le N/2} \frac{\sum_{i\in S, j\in S^c} P_{ij}}{|S|}.$$

The worst case communication delay would be O(N logN). But in an adhoc cloud a team would be in a shorter communication range thus with a team of robots with some maximum size M, the size M = 1. We see that the conductance of such a network is at least 1/M, so that the delay is bounded by O(M logN). I n our case M is kept constant as teams are kept fixed thus conductance can be maintained until such a point where one robot leaves the team due to power failure.

C. Security challenges

Security is the much talked challenge in every aspect of computing since data/information has been found to be the greatest business asset. Since the robot teams utilises a wireless network there unlimited points of break-ins. Any wireless receiver using the same standard as the team is using can get access. Attacks like robot-in-the-middle (which I analogous to man-in-the-middle attack[15]), information syphoning and network flooding [16] can be instigated.

However as teams are formed, each team member has a unique ID which it uses to authenticated itself when joining the team, the leader also tracks every member thus, besides ID based authentication, like colleagues the robots can also use trust based [18] authentication mechanism, knowing the behaviour of each other and their tasks so as to sideline activities which are out of the normal.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

D. Power challenges

These robots are mobile and have independent power supplies thus when they take the burden of processing upon themselves it implies that they would use more power thus their working time span is reduce exponentially. However as discussed earlier, those with low power can come to rest and only use their power for processing forsaking movements which consume more power.

E. Applications challenges

The other greatest challenge is on applications; there would be need for rigorous configuration and verification for the network to run smoothly as will the applications on it. The cloud-seeding software (hypervisor) might fail to be contained in one robot thus constraining further the resources that would be needed for task execution. However this is solved by load balancing with one robot (leader) acting as the controller while others execute subtasks in a parallel manner and their unification being a logical singular application running.

F. Operational challenges

A scenario might arise where two robotic teams might team up. Due to security restrictions and before said authentication mechanisms, it might be impossible for the teams to acknowledge each other since they have different team IDs and conflict on leadership might arise as well as trust issues since members of each team are only well known to their team. Analogous to human teams, the robotic teams can keep their leaders and teams and only exchange information. But however if full adhoc cloud is to be implemented the teams would be dissolved and a new leader election of all combined would be elected. Authentication would be on ID basis rather than trust basis.

V. DISCUSSION AND CONCLUSION

Having considered the constrains and the applicability which includes all the possibilities therein, the concept of an adhoc cloud –cloud seeding can be adopted. This is a way to ensure task continuation in the field; this also can be considered not as a backup to the infrastructure based cloud but to work on its own. Whenever need arises even without cloud infrastructure, robots can team up and make their own cloud to accomplish a certain task then move on to their respective obligations after finishing. Cloud computing and cloud robotics are still growing areas of research so as to power up processing and make resources available on a utility, self provisioning basis, as such much work is being done and also needs to be done for the realisation of such an initiative in the field of robotics.

For future work a more practical approach to this concept could be considered so as to verify they functional feasibility of this concept.

REFERENCES

1. http://www.thomasnet.com/articles/engineering-consulting/robotics-history
2. http://prime.jsc.nasa.gov/ROV/history.html viewwed 16/02/2015
3. Hu, Guoqiang, Wee Peng Tay, and Yonggang Wen. "Cloud robotics: architecture, challenges and applications." Network, IEEE 26.3: pp21-28. 2012
4. Gray, Edwyn. Nineteenth-century Torpedoes and their Inventors. Naval Institute Press: pp18, 2004.
5. http://www.popularmechanics.com/military/a10511/russia-wants-autonomous-fighting-robots-and-lots-of-them-16787165/
6. Inaba, Masayuki, Satoshi Kagami, Fumio Kanehiro, Yukiko Hoshino, and Hirochika Inoue. "A platform for robotics research based on the remote-brained robot approach." The International Journal of Robotics Research 19, no. 10: pp933-954 (2000).
7. IEEE Society of Robotics and Automation’s Technical Committee on Networked Robots. [Online]. Available: http://www-users.cs.umn.edu/isler/tc/
8. https://www.vmware.com/in/products/vsphere/features/storage-thin-provisioning
9. https://www.blackbox.nl/_AppData/cms/Default%20pages/Solutions/Resources/EN%20WP%20Wireless%20Standards.pdf
10. Martin D. Stoehr, RF Basics available online. http://pdfserv.maximintegrated.com/en/an/AN5300.pdf
11. LJ Create, An Introduction to Microwave Communications, 2007
12. Petteri Kuosmanen, Classification of Ad Hoc Routing Protocols, 2010 http://www.netlab.tkk.fi/opetus/s38030/k02/Papers/12-Petteri.pdf
13. Haas, Zygmunt J., Joseph Y. Halpern, and Li Li. "Gossip-based ad hoc routing." IEEE/ACM Transactions on Networking (ToN) 14.3: pp479-491 (2006).
14. D. Shah, “Gossip algorithms,” in Foundations and Trends in Networking,2008, vol. 3, no. 1.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 4, April 2015

15. Man in the middle attack, accessed 18/03/2015 https://www.owasp.org/index.php/Man-in-the-middle_attack, 2014
16. Afanasyev, Alexander, Priya Mahadevan, Ilya Moiseenko, Ersin Uzun, and Lixia Zhang. "Interest flooding attack and countermeasures in Named Data Networking." In IFIP Networking Conference, 2013, pp. 1-9. IEEE, 2013.
17. Hassan, Jahan, Harsha Sirisena, and Björn Landfeldt. "Trust-based fast authentication for multiowner wireless networks." Mobile Computing, IEEE Transactions on 7, no. 2: pp247-261 (2008).
18. Capek, Karel. "Rossum's universal robots." Prague, CZ (1920).

BIOGRAPHY

Shepard Chifamba is a Post Graduate Student in the Department of Information Technology at SRM University Chennai studying towards a Master of Technology Degree in Cloud Computing. He received a Bachelor of Science Honours Degree in Computer Science from Midlands State University (MSU), Zimbabwe in 2010 and is also studying towards a Master of Science Degree in Information Systems with National University of Science and Technology (NUST) Zimbabwe. His area of interest is Cloud Robotics.