INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

SPECIAL ISSUE FOR
INTERNATIONAL CONFERENCE ON
"INNOVATIONS IN SCIENCE & TECHNOLOGY: OPPORTUNITIES & CHALLENGES"

CLOUD ROBOTICS: AN OVERVIEW

K. S. GILDA, S. S. DIXIT

Assistant Professor, Department of Computer Science & Engineering, C.O.E. & T., Akola

Accepted Date: 07/09/2016; Published Date: 24/09/2016

Abstract: ***Cloud Robotics (CR)*** *is an emerging field within robotics, currently covering various application domains and robot network paradigms. Robots are limited in terms of computational capacity, memory and storage. Cloud provides unlimited computation power, memory, storage and especially collaboration opportunity. This paper describes the basic concepts and development process of cloud robotics and elaborates the overall architecture of these systems. ROS as robotic middleware, used to develop robotic application, is also discussed in brief. We discuss the technical challenges in computation, communications and security, and illustrate the potential benefits of cloud robotics in different applications. Further advancements in this field aim to establish a shared network resource for various robotics applications.*

Keywords: *Cloud Robotics (CR), Cloud Computing (CC), Robot Operating System (ROS).*

.

PAPER-QR CODE

Corresponding Author: MR. K. S. GILDA

Co Author: MR. S. S. DIXIT

Access Online On:

www.ijpret.com

How to Cite This Article:

K. S. Gilda, IJPRET, 2016; Volume 5 (2): 122-132

INTRODUCTION

Earlier (till 2007*) robots* were considered as industrial manipulators not allowing any humans to be in the vicinity of operating robots. The definition of robots has now changed allowing wide range of service robots to be called as robots or robotic devices. This evolution opened a field for new service-based models. But robots are limited in terms of computational capacity, memory and storage. Even though robots are connected via network (*network robotics*), to extend functional range, they face physical constraints as all computations are conducted onboard the robots, which have limited computing capabilities. To enhance the performance of robots and make them more capable of doing humanly tasks, offloading huge amount of computational tasks to some other medium will be the best policy. There lies the need for combining robotics with cloud computing. ***Cloud Robotics*** (CR) was born from the merger of cloud technologies and service robotics. The term "cloud-enabled robotics" was presented by James Kuffner for the first time at the *IEEE RAS Intl. Conference on Humanoid Robotics* in 2010.

Cloud computing is defined as "a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction." Cloud computing provides a natural venue to extend the capabilities of networked robotics. Computing clouds are classified into 3 types: public clouds, private cloud and hybrid cloud. *Cloud computing services* are divided into 3 types:

1. Software as a service (SaaS): software application is hosted as a service and end-users use the application on the browser.
2. Platform as a Service (PaaS): The end user creates, tests, and uploads applications using tools and libraries hosted by the service provider.
3. Infrastructure as a Service (IaaS): It involves hosting of hardware computing services like storage, hard drive, servers and network components.

The concept of *robot-as-a-service* (RaaS) refers to robots that can be dynamically combined to give support to the execution of specific applications. RaaS has three aspects of the system: structure, interface, and behavior. Cloud computing services hierarchy is topped by Cloud Robotics, a new paradigm and a *System of Systems* in itself.

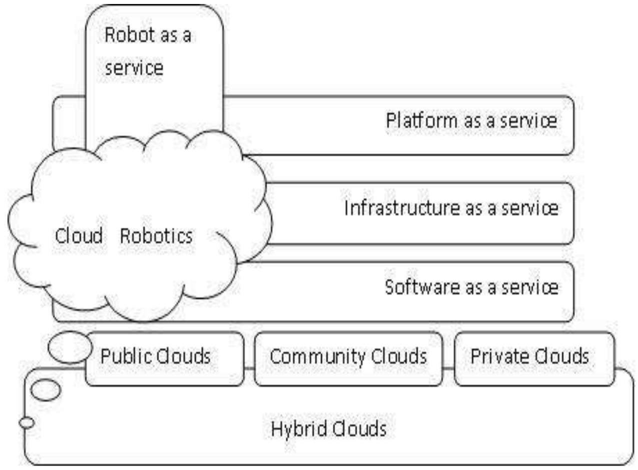

Fig 1: Cloud computing service models along with service layers.

There can be multiple ways of combining network robots and cloud computing. But significant *components* for a "cloud for robots" can be stated as follows:

1. A global library of images, maps, and object data, often with geometry and mechanical properties, expert system, knowledge base (i.e. semantic web, data centers)
2. Massively-parallel computation on demand for sample-based statistical modeling and motion planning, task planning, multi-robot collaboration, scheduling and coordination of system
3. Robot sharing of outcomes, trajectories, and dynamic control policies and robot learning support
4. Human sharing of "open-source" code, data, and designs for programming, experimentation, and hardware construction
5. On-demand human guidance and assistance for evaluation, learning, and error recovery
6. Augmented human–robot interaction through various ways.

The robotic actions associated with cloud robotics should have a specific sequence. The basic *sequence of robotic actions* is as follows:

1. Data capturing by the robot: In this step, there is a 2D camera and a 3D scanner which detects the object on-board of the robot. The output is a 2D image and a 3D point cloud, but only the 2D image is sent for processing to the cloud server.
2. Data Processing on cloud servers: The server returns a 3D CAD model of the object to the robot if it was able to identify the object based on the 2D image.
3. Processed data is applied: In this step, the robot determines the object's position and orientation based on the obtained 3D CAD model and the measured 3D point cloud. The robot, then, applies its functionality thereafter.

4. Sending feedback to the server: Finally, the robot sends the results of the functionality back to the server so that the server's database can be trained for further use.

Thus a cloud robotic module will be functional when all the necessary components are available within the combined framework and a proper sequence of actions is scheduled.

II. CR SYSTEM ARCHITECTURE: [2], [15]

Networked robotics is restricted by resource, information, and communication constraints. Effectiveness of the robotic network is limited by each robot's resources, including onboard computers, memories, and storage space. Physically, these onboard computing devices are restricted by the robots' size, shape, power supply, motion mode, and working environment. Next advancement is robots empowered with cloud technologies which is termed as Cloud Robotics. This section discusses the system architecture of Cloud Robotics.

Fig.2: Cloud Robotics- Robots are interconnected via M2M/M2C communications.

The architecture is organized into two complementary tiers: a machine-to-machine (M2M) level and a machine-to-cloud (M2C) level. On the M2M level, a group of robots communicate via wireless links to form a collaborative computing fabric (i.e., an ad-hoc cloud). On the M2C level, the infrastructure cloud provides a pool of shared computation and storage resources that can be allocated elastically for real-time demand. This elastic computing model allows the group of networked robots to offload computation-intensive tasks for remote execution.

M2M/M2C Communication Architecture

Robots in a network can communicate if they are within communication range of each other, and with the cloud servers if the robots are close to access points of the cloud infrastructure. A wireless M2M communication network can be formed by robots working cooperatively with each other to route and relay information. Standards like Zigbee, Bluetooth, and WiFi Direct have been developed for short range wireless communications between robots. For long range communications, radio frequency and microwave communication technologies may be used. A network of robots is dynamic and ad-hoc. There is no central controller to coordinate the communication flow in the network. Robots may leave and join the network, or may become unavailable because of unpredictable failures. The network becomes highly dynamic if robots are mobile.

All these considerations make the design of effective routing protocols difficult. Gossip algorithms are randomized methods designed to transmit a message from a source to a destination without any explicit route discovery mechanisms. If two nodes are within communication range, we say that they are neighbors. When a robot wants to send a message to a destination node, it randomly chooses one of its neighbors and transmits the message together with a header that contains the identifier of the destination and itself, and a time value indicating the validity period of the message. In another variant of the protocol, the message is simply broadcast to all neighbors, but depending on the application, this may incur high communication load in the network. Gossip protocols do not require route discoveries and maintenance, and are thus suited for highly dynamic mobile robotic networks. These protocols are also very simple to implement, and require minimal additional computation and memory resources. To overcome high message latency in Gossip protocol, a hybrid gossip algorithm can be used in which routes to frequently accessed nodescan be maintained.

III. ROS: [5], [16], [17], [18], [19], [20]

Since applications for Cloud Robotics have to be developed in a platform, majority of the cloud application developers choose ROS. Robot Operating System (ROS) is an open source middleware. There is an abstraction layer that resides between the operating system and the software application which is designed to manage the heterogeneity of the hardware, improve software application quality, simplify software design, and reduce development costs. This is known as a robotic *middleware*. ROS is one such middleware.

ROS is not an operating system but a *meta-operating system* for robot software which consists of a number of small tools to allow sharing of data between independent programs. ROS is an open source robot operating system that provides: hardware abstraction, low-level device control, message passing and package management.

ROS offers a graph-like pattern for networked robotics. It provides package tools and operating system-like tools. ROS allows independent, individual programs, referred to as nodes, to share data between themselves. ROS is based on nodes, messages, topics, and services. In ROS, Nodes are the software modules or processes in the control code. Nodes communicate with each other through 'publish/subscribe' message passing model. A special type of message, called service, consists of a pair of messages, one for request and the other for reply. Nodes can publish and subscribe to a single or multiple topics.

ROS can be used to create robots that can be used for our daily purposes and Cloud Robotics would help us augment the information contained in these robots. So, basically, we use ROS to develop robotic applications and CR to maintain them. The Personal Robot (PR2) is a portable manipulation platform built by Willow Garage. The entire (PR2) software structure is written with the help of ROS.

ROS is much easy to use and flexible as per the needs of the user. Since it is an open-source platform for robot software development, it is easy to maintain. The primary goal of ROS is to support code reuse in robotics research and development. Focusing on the primary goal of ROS, we can use it in cloud robotics so that code created for one robot can be easily reused in others too.

IV. CHALLENGES: [2], [6], [8], [21]

This section focuses on challenges in cloud robotics. Establishing a balance between real time demands and processing performance is a core issue of resource allocation. Cloud storage means the remote storage of data. Importance of data also imposes further requirements on cloud security. Also, interaction between cloud and robot needs data to be exchanged in a certain format. Technical challenges need to deal with network latency.

1. Allocating resources for computation

Given a computational task; the choices for resource allocation are- uploading task to cloud, self-processing or distributing task among group of networked robots. The choice selected has significant impact on overall performance.

As with inter-machine communications, the amount of computations on the cloud makes the emergence of delays. These delays includes pre-processing time for sending and receiving data, time consumed for sending and receiving data, time consumed in cloud for processing and lastly network delay. Although wireless technology has made significant progress, once connection problems between the robots and cloud services appear, serious delays are almost inevitable. Therefore, when designing a new algorithm, we need to design a load distribution algorithm with an ``any time'' characteristic. Once it is apparent that a task that cannot be properly uploaded to the cloud, a mechanism for dynamic allocation of computing tasks should be activated, thereby reducing the delay time.

2. Security

The connectivity inherent in the Cloud raises a privacy and security challenge. Data generated by Cloud-connected robots and sensors may include images or video or data from private homes or corporate trade secrets. Commercial science and technology solutions have suffered from serious data leakage incidents, especially during the upload of photos and video to the cloud. Current solutions proposed include a combination of multi-identity and personal identity authentication methods with layered encryption. Additionally, data security is at the core of cloud security, which is composed of isolation protection of static and dynamic data storage.

Cloud Robotics also introduces the potential of robots and systems to be attacked remotely: a hacker could take over a robot and use it to disrupt functionality or cause damage. These concerns raise new regulatory, accountability and legal issues related to safety, control, and transparency. The "We Robot" conference is an annual forum for ethics and policy research.

Another issue is trust and security in cloud robotics. We need the cloud environment to be trust-worthy. A malicious cloud can subtly sabotage an important task without the robot being aware of the damage. In military applications, the robot has to identify a trust-worthy cloud infrastructure to connect and to avoid malicious infrastructures (e.g., battlefield communication vehicles from an enemy).

3. Technical Challenges

On the technical front, new algorithms and methods are needed to cope with time-varying network latency and Quality-of-Service. Algorithms must be designed to degrade gracefully when the Cloud resources are very slow, noisy, or unavailable. When the Cloud is used for parallel-processing, the algorithms should also take into account that some remote processors may fail or experience long delays in returning results.

New algorithms are also needed that scale to the size of Big Data, which often contain dirty data. Big data maintains large datasets which are often mixed with unexpected data called *dirty* data. This requires new approaches to clean or sample effectively.

4. Data interaction between robot and cloud

Devices and sensors from different manufacturers may output data with different structures. Cloud devices can only handle and store data of specific structures. This means that the input interfaces need to transform the corresponding data structures into a unified format. At the output part, when the processing of uploaded data is complete, the ready-to-output data must also be transformed to specific formats. Output data formats need to be chosen according to the compatibility of the underlying devices and the real-time requirements of the entire data exchange.

V. LIMITATIONS: [1], [6], [8], [9]

Though robots can benefit from various advantages of cloud computing, cloud is not the solution to all of robotics.

- Controlling a robot's motion which relies heavily on sensors and feedback of controller won't benefit much from the cloud.
- Cloud-based applications can get slow or unavailable due to high-latency responses or network hitch. If a robot relies too much on the cloud, a fault in the network could leave it *brainless*.
- Tasks that involve real-time execution require on-board processing.
- Offloading large data to cloud is taking time, just as onboard computation.

VII. APPLICATIONS: [2], [3], [22], [23]

Cloud robotics has a lot of applications in data-intensive or computation intensive tasks in the areas of intelligent transportation, environment monitoring, health care, smart home, entertainment, education and defense. In this section, we focus on three robotic applications that gain potential benefit from cloud.

1. SLAM

SLAM (Simultaneous Localization and Mapping) refers to a technique for a robot to build a map of the environment without a priori knowledge, and to simultaneously localize itself in the unknown environment. SLAM is a data intensive and computation intensive. The steps such as map fusion and filtering for state estimation can be processed in a parallel fashion. Thus, these tasks can be offloaded to the cloud.

Further advancement is a cloud framework named as *Cloud framework for Cooperative Tracking and Mapping* (C2TAM). C2TAM is a visual SLAM system based a distributed framework where the CPU-intensive map optimization and storage is allocated as a service in the Cloud, while a light camera runs on robots for tracking. The robots need only internet connection for tracking and cooperative relocating. C2TAM provides a database consisting maps can be built and stored, stored maps can be reused by other robots. A robot can fuse its map online with a map already in the database, and several robots can estimate individual maps and fuse them together if an overlap is detected.

2. Grasping

Robotic grasping has been a research topic. If the object is precisely known, then various methods can be applied to synthesize the grasp. If the object is unknown or not precisely known, the problem is much more challenging, and involves the access and preprocessing of vast amounts of data and can be computationally intensive. By offloading this task to the cloud, grasping can be facilitated without requiring vast amounts of computing power, data, and storage space on the robotic platform. In addition, model knowledge of new objects learned by different robots can be shared in the cloud for future usage by other robots.

3. Navigation

Robotic navigation refers to a robot's activity to determine its own position with respect to a certain reference and then plan a path to reach a desired location. It can involve a combination of tasks such as localization, path planning, and mapping. Basically, there are two types of approaches: map-less approaches and map-based approaches. Map-less approaches rely on the observations of the perception sensors for navigation. Due to the limited onboard resources, these approaches usually suffer from reliability issues. Map-based robotic navigation is relatively reliable if a precise map is available.

VIII. FUTURE SCOPE: [1], [5], [8]

Cloud robotics is gaining popularity day by day. In the recent future we can expect wide range of applications through CR. The areas where CR is expected to advance in near future are as follows:

- Robots in the real-world need a lot of information, about because they are limited in capacity to explore the environment. The recognition of various objects, sounds, images, people can be done much more efficiently with the help of the cloud.
- Navigation of robots can efficient if Google Maps-like services are fast and reliable so that the robots could exceed the limit of its on-board capacity. Cloud services can revolutionize modeling and motion planning, by reducing costs and increasing the innovation component.
- The communication between robots is crucial component of a System of Robots. Agents could share their sensory information, calculation results, news or their experience (advancing to cognitive level), gained through their reasoning skills.
- We need to encourage people to generate open-source codes, share their plans, experience and results.
- In autonomous robots, human operator may be necessary who coordinates the robot's behavior in case of error.
- A smart city is proposed with cloud robotic services, cloud computing, big data, and other fields.
- Future Robots will have access to shared knowledge and new skills. A medium is provided by the cloud for the robots to share information, learn new skills, and acquire additional knowledge from each other. The cloud can host a database or library of skills or behaviors that map to different task requirements and environmental complexities.
- Future Robots will have access to large amounts of data. The robots can acquire knowledge to execute tasks through databases in the cloud. They do not have to deal with the creation and maintenance of data that comes in and goes out of the cloud.
- Currently CR uses ROS wherein the software runs locally. In future, Robotics and Automation as a service is proposed where the analysis and planning software runs in cloud.

IX. CONCLUSION:

Cloud Robotics is an evolutionary jump for robots, where communication between robot and humans has become essential characteristic of the system. Though there are few limitations, CR can be thought of the best possible solution for network robotic applications. Standalone robots can benefit cloud technologies and networked robots can perform collaborative works. Networked cloud-enabled robots can share computation resources, information and data with each other and can access new knowledge and skills. Cloud Robotics would bring standardization in the way robotics applications are developed today. The goal of Cloud

Robotics is to meet diverse requirements of end user in the domain of robotics without degrading the quality of service. Future works can focus on reliable connection, data offloading methods and ubiquitous networking among robots and cloud services.

REFERENCES

1. S. Jordan, T. Haidegger, L. Kovacs, I. Felde, I. Rudas, “The Rising Prospects of Cloud Robotic Applications”, IEEE 9th International Conference on Computational Cybernetics , Tihany, Hungary, pp 327-332, July 8-10, 2013.
2. Guoqiang Hu, Wee PengTay, and Yonggang Wen, “Cloud Robotics: Architecture, Challenges and Applications”, IEEE network, pp 21-28, May/June 2012.
3. BusraKoken, “Cloud Robotic Platforms”, Interdisciplinary Description of Complex Systems 13(1), 26-33, 2015.
4. Priyanki Jayantilal Vashi, “ Cloud Robotics: An Emerging Research Discipline”
5. AdarshaKharel , DorjeeBhutia , SunitaRai , DhrubaNingombam, “ Cloud Roboitcs using ROS”, International Journal of Computer Applications (IJCA) (0975 – 8887) National Conference cum Workshop on Bioinformatics and Computational Biology, NCWBCB- 2014.
6. J. Wan, S. Tang, H. Yan, Di Li, S. Wang, A. V. Vasilakos, “Cloud Robotics: Current Status and Open Issues”, IEEE Access, vol 4 , pp 2797-2807, June 2016.
7. Ken Goldberg,Ben Kehoe, “Cloud Robotics and Automation: A Survey of Related work”, Technical Report No. UCB/EECS-2013-5, http://www.eecs.berkeley.edu/Pubs/TechRpts/2013/EECS-2013-5.html.
8. Ben Kehoe, SachinPatil, Pieter Abbeel, Ken Goldberg,” A Survey of Research on Cloud Roboticsand Automation”, IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, VOL. 12, NO. 2, APRIL 2015.
9. Robotics http://en.wikipedia.org/wiki/Robotics.
10. Robot http:// en.wikipedia.org/wiki/Robot.
11. Cloud computing http:// en.wikipedia.org/wiki/cloud_computing.
12. Chibani, A. et al. : Ubiquitous Robotics: Recent Challenges and future trends.
13. J. Kuffner, Cloud-Enabled Robots. In proc. of the IEEE Intl. Conf.on Humanoid Robots, Nashville, 2010.
14. https://www.scribd.com/doc/47486324/Cloud-Enabled-Robots.
15. D. Shah, “Gossip Algorithms,” Foundations and Trends in Networking, 2008, vol. 3, no. 1.
16. http://www.ros.org.
17. http://www.ros.org/news/2010/03/robots-using-ros-aldebaran-nao.html.
18. Parker James Conroy, “The Development Of An Aerial Robotics Laboratory Highlighting The First Experimental Validation Of Optimal Reciprocal Collision Avoidance,” August 2013.
19. Willow Garage, “Robot operating system,” 2010.

20. Aaron Staranowicz, Gian Luca Mariottini, "A survey and comparison of commercial & open-source robotic simulator software".

21. We Robot, "We robot conference," 2012. [Online]. Available: http: //robots.law.miami.edu/

22. H. Durrant-Whyte and T. Bailey, "Simultaneous Localization and Mapping: Part I," IEEE Robotics & Automation Mag., vol. 13, 2006, pp. 99–110.\

23. F. Bonin-Font, A. Ortiz, and G. Oliver, "Visual Navigation for Mobile Robots: A Survey," J. Intelligent and Robotic Systems, vol. 53, 2008, pp. 263–96.