Dawarka, Viraj and Bekaroo, Girish ORCID: https://orcid.org/0000-0003-1753-4300 (2018) Cloud robotics platforms: review and comparative analysis. Proceedings 2018 International Conference on Intelligent and Innovative Computing Applications. In: ICONIC: MAURICON 2018 International Conference on Intelligent and Innovative Computing Applications, 06-07 Dec 2018, Plaine Magnien, Mauritius. e-ISBN 9781538664773, pbk-ISBN 9781538664780. (doi:10.1109/ICONIC.2018.8601220)

Final accepted version (with author's formatting)

This version is available at: http://eprints.mdx.ac.uk/29769/

Cloud Robotics Platforms: Review and Comparative Analysis

Viraj Dawarka
Faculty of IT, Design & Communication,
Curtin Mauritius, Telfair,
Moka, Mauritius
vdawarka@curtinmauritius.ac.mu

Girish Bekaroo
School of Science and Technology, Middlesex University Mauritius, Coastal Road, Uniciti, Flic-en-Flac, Mauritius g.bekaroo@mdx.ac.mu

***Abstract*—Due to the various advantages that the cloud can offer to robots, there has been the recent emergence of the cloud robotics paradigm. Cloud robotics permits robots to unload computing and storage related tasks into the cloud, and as such, robots can be built with smaller on-board computers. The use of cloud-robotics also allows robots to share knowledge within the community over a dedicated cloud space. In order to build-up robots that benefit from the cloud-robotics paradigm, different cloud-robotics platforms have been released during recent years. This paper critically reviews and compares existing cloud robotic platforms in order to provide recommendations on future use and gaps that still need to be addressed. To achieve this, 8 cloud robotic platforms were investigated. Key findings reveal varying underlying architectures and models adopted by these platforms, in addition to different features offered to end-users.**

Keywords—Cloud-Robotic Platforms, Robotics, Cloud, Knowledge Sharing Platform, Comparative Analysis.

I. INTRODUCTION

Recent advances in cloud computing technologies and robotics have led to the creation of the cloud-robotics paradigm, which relate to networked-connected robots that inherit from parallel computing and data sharing through the Internet. This association is expected to make robots lighter, cheaper and smarter since the utilization of cloud enables heavy processing to be done remotely such that smaller on-board computers are used within robots [1]. For instance, the process for robots to build the map of the environment for localization purposes is both computationally and data intensive and such tasks can be offloaded to the cloud [2]. Concerning data-intensive processes, robots can gain from massive amount of storage space when making use of clouds and also, knowledge acquired by robots can be shared to other such machines [3].

Due to the various benefits of the cloud-robotics paradigm, different platforms have emerged. Cloud-robotic platform enables the build-up of cloud-robotic based robots and some of them are open-source, whilst others are proprietary [4]. Some platforms are composed of a number of high-performance servers, databases, proxy-servers, amongst others [5]. In terms of research, despite the fact that various studies have been conducted regarding the aspects of cloud robotics in many fields, limited published literature is available on the comparison of existing cloud robotic platforms. As related work, one recent study conducted a survey of research on cloud robotics and automation by considering over 150 references in the area [6]. However, this study mentions about cloud robotics platform, limited comparative analysis has been conducted as this was not a major scope the paper. Another study described the development process of cloud robotics as well as the overall architecture of these systems [5]. The current problem that scientists, roboticists and researchers are facing is that an efficient cloud robotic platform is not available to meet the needs of developing an application using existing cloud robotic architectures. Thus, a mixture of platforms ranging from robotic open source software, web services, and cloud platforms are being used to cater the requirements from the chosen architectures which in turn makes it difficult to sustain. Through literatures, cases have been detected where cloud robotic architecture have not been implemented in real life and simulation was used to test the architecture's functionality and efficiency. Since multiple platforms are being used to develop applications using current cloud robotic architectures, the problem of platform dependencies occur and the application fail to function due to one platform not responding to another dependent platform when having flow of data communication. Within this study, cloud robotic platform are highlighted as a key component of the architecture of cloud robotics and although some of the platforms are described, limited comparison has been conducted. Cloud robotic platform in a cloud robotic framework or architecture is the main key component which allows the implementation of the application to further interact and communicate with other components such as robots, cloud or network but going through literature, there is no evidence of which platform is the best fit for this purpose. To address this gap, this paper critically examines and compares existing cloud robotic platforms in order to provide recommendations on future use when implementing cloud robotic architectures and gaps that still need to be addressed. Addressing this gap is expected to provide researchers and experts a comprehensive review of such platforms in addition to recommendations towards improvement of such platforms.

This paper is structured in the successive manner: In the first section, an introduction to the topic is produced then in second section, the methodology used to achieve the purpose is given, followed by a review of existing cloud robotic platform in Section 3. Then, a comparative analysis of the existing cloud-robotic platforms is provided in the fourth section, before making recommendations in Section 5. The work is concluded in Section 6 and avenues for future works are discussed.

II. Methodology

In order to achieve the purpose of this paper, an initial pool of cloud-robotic platforms was searched on Google and Google Scholar by using relevant keywords. The Google search engine was used as it prioritizes results based on relevancy while also employing a variety of techniques to improve search quality through page ranks [7, 8]. The initial search was conducted in March 2018, starting with Google Scholar where 74 conference and journal articles were thoroughly reviewed to assess relevancy, before complementing the search with a general Google search. Keywords such as Cloud robotic platform, cloud platform, robotic platform were used to search the database then the exact name of the cloud robotic platform was utilized to further narrow down the search. Finally, only 8 cloud-robotic platforms were identified, whilst the other sources were principally related to frameworks and state-of-the-art review of cloud-robotics. Once the platforms were identified, literature search pertaining to each platform was thoroughly conducted by going through relevant articles while also finding information on key websites. The information gathered was then analyzed and is presented in the next sections of this paper. Similar methodology was used in different studies performing comparative analysis [9, 10]. Using this methodology enables to review all the research work pertaining to this topic in the literature to find an accurate answer to the research problem.

III. Review of Cloud Robotic Platforms

Using the methodology defined in the previous section, different cloud-robotic platforms were identified. These are discussed as follows:

A. Rapyuta

Also known as the RoboEarth Cloud Engine, Rapyuta is an open-source cloud robotics platform [9]. It is based on an elastic computing model and active distributed secure computing environments in which robots are deeply associated, while allowing robots to contribute most of their services to other robots [10]. This platform is known to eliminate complexity, costs, possibility of deploying, interfacing and managing robotics systems so that more time is available to do other tasks [9]. By allowing access to the RobotEarth Knowledge Repository, Rapyuta allows robots to store and share information, offload computation and collaborate and achieve a common task [10]. Moreover, a range of capabilities is provided such as disk quota, I/O limits and memory limits configuration, among others. Furthermore, it also allows the outsourcing of around more than 3000 Robot Operating System (ROS) packages and is extensible to other robotic middleware [11]. Additionally, a recent work using this platform involves the pre-installation of Amazon Machine Image (AMI) which can launch the Rapyuta in one of the Amazon's data center in a short period of time and permits the robots to authenticate themselves, create one or more environments in the cloud and launch the process [11].

B. Robot Operating System (ROS)

The ROS platform enables the production of software modules in order to execute typical robot activities such as object recognition [12]. The concept of "not re-inventing the wheel" is the principal aim of the ROS platform, where it provides integrated libraries that are easy to use in addition to multitude facilities such as manipulation, navigation control, and hardware abstraction for sensors and actuators, among others [13]. ROS also gives the advantages of inter-platform operability between multiple programming languages such as C, Java and Python. As part of this platform, ROS processes or Nodes involve data processing in the platform and a message-passing distributed system derived on the publish/subscription paradigm is achieved by ROS where Nodes produce messages on Topics which other Nodes employ [14]. In terms of application, this platform has provided solution for real-time ball trajectory tracking for tennis and football events through the creation of an environment with the integration of Open Source Computer Vision libraries (OpenCV) for object detection and tracking [15]. At present, a newer version of ROS, namely ROS 2, is being actively developed to improve cross-platform support capability [16].

C. C2RO Cloud Robotics

Established in 2016 in Montreal, C2RO Cloud Robotics is a cloud-based software robotics platform for the global service robotics target market [17]. The C2RO platform connects robots using patent-pending technologies and augments the capabilities of robots through a fast and secure communication. It also provides a robot-agnostic software-as-a-service (SaaS) platform that utilizes an information processing technology, which functions in a real-time manner to grant robots an artificial intelligence solution in a secure, fast and inexpensive approach [18]. This platform was created in order to address the industrial automation demand comprising of problems such as lack of robots' connectivity that result in the inability for monitoring real time problem, the limitation of pre-programmed tasks due to limited onboard sensing and computing power as well as non-cooperation of robots [18]. In order to address these problems, the C2RO platform upgrades processing power via a hybrid solution of completing high-skilled tasks and a cloud-based robotics platform where multiple robots can share knowledge instantly across multiple sites and geographies [17].

D. Microsoft Robotics Developer Studio (MRDS)

Released by Microsoft in 2006, the MRDS platform enables programming robots in the Windows environment and it can interact with the circuits commonly known as microcontrollers on the robots to control actuators over a hardwired link or Bluetooth [19]. MRDS contains a .NET-based service-oriented runtime, comprising of components such as Concurrency and Coordination Runtime (CCR) and Decentralised software services (DSS) [20]. This platform concentrates in making robotics applications that allow either to be simulated by using Visual Simulated Environment (VSE) which is 3-D virtual simulator or through Visual Programming Language Environment (VPL), a programming interface that connects to real robots. MRDS is considered as a crucial product when bestowing Service-oriented architecture (SOA) framework for Robot as a Service (RaaS) in cloud computing to embedded systems [21].

E. REALabs

REALabs is a cloud robotic platform that enables the computer running the robotic application and mobile robot to interact with each other over the network [22]. This platform was built based on the Platform as a Service (PaaS) model [23]. REALabs is completely based on Web technologies and consists of four main software packages from its architecture, namely, the front-end package, the protocol handler, embedded package and the management package. The platform has been mainly used in Web Labs over public internet where the user implements the robotic application on personal computer so as to control the robot through the network [22]. Several updates have been conducted on the platform such as integration of Web services in the end of 1990s, HTTP/XML-based Remote Procedure Call and transition to the Representational State Transfer (REST) architecture [23].

F. Rospeex

Rospeex is a cloud robotic platform for multilingual spoken dialogues with robots for ROS developed by the National Institute of Information and Communications Technology (NICT) [24]. It is equipped with a straightforward interface for speech synthesis and speech recognition in different languages including Japanese, English [25]. This platform is free for use by roboticists and does not require authentication. Moreover, the platform comes with a bundle including a browser user interface, the Rospeex cloud services and the Rospeex modules comprising of voice activity detection, noise reduction and speech synthesis. The user interface of the rospeex platform has been developed in HTML5 and can operate on many platforms such as Linux, Windows and Android smart devices. Also, two types of users can adopt this platform, namely user and developer and with the use of Rospeex, large amount of robotics-related statements can be stored on the cloud server [24].

G. DAvinCi

DAvinCi stands for Distributed Agents with Collective Intelligence and was built in Singapore by the ASORO laboratory to produce 3D-models of environments for robots during simultaneous localization and mapping (SLAM) [21]. It supports an augmented architecture for large environment to allow group of robots to operate in large environments [26]. DAvinCi consists of three technologies, namely, the open source Hadoop Distributed File System (HDFS), ROS architecture and the Hadoop Map/Reduce Framework. The ROS architecture is used for sensor data collection and to communicate between clients and robots, the HDFS is used for data storage and Hadoop Map/ Reduce framework is used for batch processing of visual information and sensor data. An established way of communication and messaging between the robots and the DAvinCi server is provided by the ROS and its goal is to offload huge data workloads from the robots to a backend cluster system [26].

H. GostaiNet

GostaiNet was developed by the French robotic firm called Gostai and enables robots to perform face detection, speech recognition and other task remotely [27]. Seamless control of any robot from anywhere around the world is provided through a web browser and services are hosted by Gostai on the GostaiNet robotics cloud [28]. This architecture gives the opportunity to decentralize artificial intelligence in order to produce economic robots with complex behaviors and with autonomous capabilities [29]. The latest work that was made available is the Jazz robots that were implemented on Gostai's Urbi open-source ROS while also using the GostaiNet cloud-computing infrastructure to make cloud based video recording [30].

IV. Comparative Analysis

Review of the cloud-robotic platforms showed that most of them are open-source besides REALabs and DAvinCi platforms. While being open-source, different such platforms evolve continually as developers and other contributors keep on adding or updating features. This also promotes adoption of such solutions by developers, who can also obtain support from the community. Use of such platforms are also promoted by the fact that most platforms provide a user-friendly user interface where user can offload robotic data onto the cloud service. On the other hand, while being open-source, some platforms such as ROS do not have up-to-date documentation for its users due to associated maintenance costs. Similarly, most platforms enable sharing to other robots through the cloud environment, besides MRDS. When sharing data or knowledge to robots built in a different platform, a key challenge faced involves format for representing and exchanging data [6]. For instance, although sensor data in the form of images have a small number of popular formats, trajectory-related data have no standardized format [31, 32]. Moreover, Rapyuta and ROS were found to be highly compatible with other platforms as discussed earlier, whilst REALabs, showed to have the lowest compatibility. In TABLE I, Compatibility with other platforms has been discussed in terms of cross platform capabilities where a scale high means the platform can be used by a large number of other platforms, moderate means the platform can be utilized by an average number of platforms and low for a very minimal number of platforms. The comparative summary of the review conducted is given in TABLE I.

TABLE I. Comparative analysis

Cloud Robotic Platform	**Security**	**Open Source**	**Underlying Model or Architecture**	**Compatibility with other platforms**	**Sharing to other robots**
Rapyuta	Provides a secure customizable computing environment in the cloud to offload heavy computation.	Yes	Elastic computing model	High	Yes
ROS	A cryptographic method has been implemented in [35] to secure the ROS communication channels.	Yes	Publish/subscribe message passing architecture	High	Yes
C2RO	Virtual Barriers is used for data access control, Secure	No	Hybrid cloud robotics model	Moderate	Yes

	Sockets Layer and Transport Layer Security are used to avoid eavesdropping and "man in the middle" attacks.				
MRDS	The MRDS studio has a security manager page which manages authentication for users.	No	Service Oriented Architecture	Moderate	No
REALabs	A secure Single Sign On service based on SAML (Security Assertion Markup Language) for user authentication is provided [36].	Yes	Platform as a Service (PaaS) model	Low	Yes
Rospeex	A ROS node is implemented for security and network reasons so as not to conduct noise reduction and Voice Activity Detection.	No	Node structure model	Moderate	Yes
DAvinCi	A fail safe mechanisms is being worked on the communication between the DavinCi server and the robots during transfer of messages.	Yes	Combination of distributed ROS architecture, the open source Hadoop Distributed File System(HDFS) and the Hadoop Map/Reduce Framework	Moderate	Yes
GostaiNet	Image analysis within the system allows movement detection for the robots making it easy to detect intruders.	Yes	GostaiNet cloud computing architecture	Moderate	Yes

Furthermore as shown in Table I, different such platforms have varying architectures or models being utilized, while also having dissimilarity on focus of the platforms in terms of characteristics and abilities. For instance, MRDS is the only platform which provides two powerful engines for graphics and dynamics in order to allow the development of distributed and concurrent processes in an innovative way [20]. This fundamentally provides the advantage of developing service-oriented, concurrent and asynchronous applications with a myriad of programming languages, including the visual programming language. REALabs platform is based on the PaaS model and uses REST, which is an alternative to web services in order to allow interaction between virtual machines and mobile robots [22].

While operating PaaS, REALabs is deployed on a Virtual Machine where latency becomes a concern. Together with this issue, the mobile robots situated on the field do not allow the usage of cloud services in a variety of mobile robotics applications available [22]. As for ROS, it is based on the Publish/subscribe message passing architecture and the platform ensures a clean programming standards to the user by allowing threads in the application to publish and subscribe to messages only. ROS provides modularity when implementing robot applications where in case a component (e.g. a sensor or a motor) crashes, the entire application does not crash due to connection with a distributed message system. DAvinCi, as discussed earlier, uses a combination of architectures in order to provide improved performance of the system through the incorporation of several computing nodes. However, the main concern is that if one of the component does not work (e.g. DAvinCi server), problems may arise if improper backup mechanisms are unreliable. Another observed issue is that performance of the overall system varies based on the number of nodes inculcated in the system. Rapyuta, in turn, is based on the elastic computing model and provides a friendly interface when offloading robotic data while also providing access to a repository of shared knowledge amongst robots. Furthermore, a proper security aspect is imparted to each robot in this platform and provides a bidirectional communication with the robots. Finally, Rospeex is based on the node structure model and is dedicated only to speech capabilities of robots using browser user interface. Its modules are related to voice activity, speech synthesis and noise reduction together with cloud services. Since Rospeex uses its own cloud service, it becomes a disadvantage when it comes to multi-platform operation. Moreover, it was also highlighted in a previous study that the waiting time for speech recognition processing time in Rospeex is not appropriate as it can deteriorate the dialogues' interactive aspect [26].

V. RECOMMENDATIONS

The comparative analysis performed showed different limitations of existing cloud-robotic platforms and for each platform, recommendations are proposed as in TABLE II, for further research and improvement by research community.

TABLE II. RECOMMENDATIONS

Cloud Robotic Platform	Reported Issue	Recommendation
Rapyuta	High Computational latency	Colocation data centers can be integrated with the platform to ensure an exceptional network coverage.
ROS	Since ROS is an open-source framework, it is vulnerable to authentication, authorization and insecure communication issues. Lack of up-to-date documentation.	A cryptography algorithm such as Diffie-Hellman or Message Authentication Code (MAC) could be introduced in the framework for strengthen the security aspect of the framework. To provide up-to-date documentation for end-users.
C2RO	A large computational power is required for the robots when using visual simultaneous localization and mapping (SLAM) to locate themselves.	An Artificial Intelligence (AI) application can be implemented to increase the computation power of the robot so as it can be more autonomous when taking important decisions regarding localization and mapping.
MRDS	Incomplete models when simulating real time situation such as wheeled vehicles and modelling sonar.	More research could have been done in this domain where an accurate model can be proposed using MRDS platform to strengthen the capacity of real time processing in daily life.
REALabs	REALabs is based on RESTful interfaces, meaning it relies on HTTP for its security aspect making it vulnerable.	A more secure protocol can be used in the framework such as OAuth1.0a to provide secure web services using the RESTful interfaces.
Rospeex	Waiting time for speech recognition processing time is not appropriate Uses its own cloud service, which is a disadvantage when it comes to multi-platform operation.	Segmenting the Rospeex module that sends the speech file to the server into multiple fragments and sends them one by one for processing. Could integrate the use of several form of cloud service where it will be easier for a myriad of platform to operate when implementing applications.
DAvinCi	Owing to the utilization of Hadoop map reduce computing cluster, a high bandwidth usage is required.	A hosted filtering can be integrated in the framework between the DaVinCi server and public cloud to filter spam messages that are transmitted over HTTP.
GostaiNet	The provided services are proprietary as they are hosted by Gostai on the GostaiNet robotics cloud.	The services can be deployed in collaboration with a public cloud service to make it more accessible to many applications.

VI. Concluding Remarks

This paper examined and compared 8 cloud robotic platforms used by roboticists around the world when developing cloud robot applications. The review showed that most of the platforms are open-source, thereby inheriting some advantages and disadvantages of being open-source. Moreover, some platforms such as Rapyuta and ROS were found to be highly compatible with other cloud-robotic platforms while enabling integration of features. In addition, different such platforms were found to have varying underlying architectures or models, while also having dissimilarity on focus in terms of characteristics and abilities. Depending on the framework or architecture that will be used, each platform will act differently and will try to blend accordingly to give an efficient product. As future work, the same platforms will be further analyzed following application to develop cloud robotic applications. This will also help to increase the number of comparison criteria studied in the comparative analysis and an efficient platform can be provided for the implementation of cloud robotic application using the prescribed recommendation.

References

[1] E. Guizzo, “Robots with Their Heads in the Clouds,” IEEE Spectrum, 2011. [Online]. Available: http://spectrum.ieee.org/robotics/humanoids/robots-with-their-heads-in-the-clouds. [Accessed 5 Mar 2018].

[2] H. Durrant-Whyte and T. Bailey, “Simultaneous localization and mapping: part I,” IEEE robotics & automation magazine, vol. 13, no. 2, pp. 99-110, 2006.

[3] G. Hu, W. Tay and Y. Wen, “Cloud robotics: architecture, challenges and applications,” IEEE network, vol. 23, no. 6, 2012.

[4] K. Sugiura and K. Zettsu, “Rospeex: A cloud robotics platform for human-robot spoken dialogues,” in 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015.

[5] J. Wan, S. Tang, H. Yan, D. Li, S. Wang and A. Vasilakos, “Cloud robotics: Current status and open issues,” IEEE Access, vol. 4, pp. 2797-2807, 2016.

[6] B. Kehoe, S. Patil, P. Abbeel and K. Goldberg, “A survey of research on cloud robotics and automation,” IEEE Transactions on automation science and engineering, vol. 12, no. 2, pp. 398-409, 2015.

[7] S. Grin and L. Page, “The anatomy of a large-scale hypertextual Web search engine,” Computer networks and ISDN systems, vol. 30, no. 1-7, pp. 107-117, 1998.

[8] K. K. Lavania, S. Jain, M. K. Gupta and N. Sharma, “Google: A Case Study (Web Searching and Crawling),” International Journal of Computer Theory and Engineering, vol. 5, no. 2, pp. 337-340, 2013.

[9] A. Zakari, A. Lawan and G. Bekaroo, “Towards Improving the Security of Low-Interaction Honeypots: Insights from a Comparative Analysis,” in International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering, 2016.

[10] B. Kim and R. Neff, “Measurement and communication of greenhouse gas emissions from US food consumption via carbon calculators,” Ecological Economics, vol. 69, no. 1, pp. 186-196, 2009.

[11] Rapyuta, “A cloud Robotics Platform,” Rapyuta, 2018. [Online]. Available: http://www.rapyuta.org/. [Accessed 7 Apr 2018].

[12] D. Hunziker, M. Gajamohan, M. Waibel and R. D'Andrea, “Rapyuta: The roboearth cloud engine,” in 2013 IEEE International Conference on Robotics and Automation (ICRA);, 2013.

[13] G. Mohanarajah, D. Hunziker, R. D'Andrea and M. Waibel, “Rapyuta: A cloud robotics platform,” IEEE Transactions on Automation Science and Engineering, vol. 12, no. 2, pp. 481-493, 2015.

[14] L. Garber, “Robot OS: A new day for robot design,” Computer, vol. 46, no. 12, pp. 16-20, 2013.

[15] S. Chitta, “MoveIt: an introduction,” in Robot Operating System (ROS), 2016.

[16] ABI Research, “Robot Operating System Evolving to Foundational Software Platform,” ABI Research, 2014. [Online]. Available: https://www.abiresearch.com/press/robot-operating-system-evolving-to-foundational-so/. [Accessed 2 May 2018].

[17] H. Yadav, S. Srivastava, P. Mukherjee and B. Lall, “A real-time ball trajectory follower using Robot Operating System,” in IEEE 2015 Third International Conference on Image Information Processing (ICIIP), 2015.

[18] D. Hood and W. Woodall, “ROS 2 update – summary of alpha releases, architectural overview,” in ROSCon 2016, Seoul, Korea, 2016.

[19] J. Park, “What is Cloud Robotics?,” C2RO Cloud Robotics, 2018. [Online]. Available: http://c2ro.com/what-is-cloud-robotics/. [Accessed 7 May 2018].

[20] Robotics Business Review, “C2RO Cloud Robotics,” Robotics Business Review; 2018. [Online]. Available: https://www.roboticsbusinessreview.com/listing/c2ro-robotics/. [Accessed 7 May 2018].
[21] K. Johns and T. Taylor, Professional microsoft robotics developer studio, John Wiley & Sons, 2009.
[22] J. Jackson, “Microsoft robotics studio: A technical introduction,” IEEE robotics & automation magazine, vol. 14, no. 4, 2007.
[23] A. Koubaa and E. Shakshuki, Robots and Sensor Clouds, Springer International Publishing, 2016.
[24] A. do Nascimento, E. Cardozo, R. Souza and E. Guimarães, “A platform for cloud robotics,” IFAC-PapersOnLine, vol. 49, no. 30, pp. 48-53, 2016.
[25] E. Guizzo and T. Deyle, “Robotics trends for 2012,” IEEE Robotics & Automation Magazine, vol. 19, no. 1, pp. 119-123, 2012.
[26] K. Sugiura and K. Zettsu, “Rospeex: A cloud robotics platform for human-robot spoken dialogues,” in 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015.
[27] Rospeex, “What is rospeex?” Rospeex, 2018. [Online]. Available: http://rospeex.org/top/. [Accessed 6 May 2018].
[28] R. Arumugam, V. Enti, L. Bingbing, W. Xiaojun, K. Baskaran, F. Kong, A. Kumar, K. Meng and G. Kit, “DAvinCi: A cloud computing framework for service robots,” in 2010 IEEE International Conference on Robotics and Automation (ICRA), 2010.
[29] Sai Natkar (“Cloud Robotics: It’s time to offload their brain on Cloud, for better Robotic future!” ESDS, 2018. [Online]. Available: https://www.esds.co.in/blog/cloud-robotics-its-time-to-offload-their-brain-on-cloud-for-better-robotic-future/#sthash.lvdbLgds.dpbs. [Accessed 10 May 2018].
[30] G. Mester, “Cloud Robotics Model,” Interdisciplinary Description of Complex Systems: INDECS, vol. 13, no. 1, pp. 1-8, 2015.
[31] R. Heuillard, “Gostai: la robotique et la téléprésence abordables,” Clubic Tech, 2009. [Online]. Available: https://www.clubic.com/actualite-251396-gostainet-robotique-personnelle-telepresence.html. [Accessed 7 May 2018].
[32] H. Lichtman, “Gostai Jazz Telepresence Robot Unveiled,” Telepresence Options, 2010. [Online]. Available: http://www.telepresenceoptions.com/2010/12/gostai_jazz_telepresence_robot/. [Accessed 7 May 2018].
[33] E. Prestes, J. Carbonera, S. Fiorini, V. Jorge, M. Abel, R. Madhavan, A. Locoro, P. Goncalves, M. Barreto, M. Habib and A. Chibani, “Towards a core ontology for robotics and automation,” Robotics and Autonomous Systems, vol. 61, no. 11, pp. 1193-1204, 2013.
[34] M. Tenorth, A. Perzylo, R. Lafrenz and M. Beetz, “Representation and exchange of knowledge about actions, objects, and environments in the roboearth framework,” IEEE Transactions on Automation Science and Engineering, vol. 10, no. 3, pp. 643-651, 2013.
[35] B. D. S. Benjamin Breiling, "Secure Communication for the Robot Operating System," in 11th Annual IEEE International Systems Conference, Montreal, April 2017.
[36] E. G. L. R. R. S. F. P. a. F. P. Eleri Cardozo, "A Platform for Networked Robotics," in IEEE/RSJ International Conference on Intelligent Robots and Systems , Taipei, Taiwan, October 18-22, 2010.