

# 1 References and Summaries from Cloud Robotics

**Cloud Networked Robotics.** In 2012, Kamei et. al. proposed establishing a field of *Cloud Networked Robotics*. While not directly concerned with using cloud computing in the context of autonomous systems, Kamei and colleagues present a case supporting swarms of robots with distinct functions that collaborate to effectively support individuals. They do however describe the need to provide distributed computation and data storage services to swarms of robots enabling information sharing and coordinated planning. Key needs they identify for distributed computing systems include providing abstractions for individual robots, so a swarm can be automatically coordinate actions toward a goal rather than requiring programming for single entities, and the ability to share different types of information freely among cooperating robots [1].

**Cloud Robotics: Current trends and possible use as a service.** Loren-cik in 2013 outlined the state of the art in cloud systems supporting robotics deployments, outlining advantages to using cloud computing with robots in cost and performance domains, at the expense of network connection and latency dependence [2].

**Cloud Robotics: Architecture, Challenges, and Applications** In 2012, Hu outlined both two primary patterns of communication within cloud robotics as well as rigorous guidelines. based on power demands, on when they should be used and why. He also presented communication protocols useful in both scenarios, and presented communication performance expectations[3].

Overall, this thorough evaluation characterizes computation based on power utilization well, but does not address other areas of concern, like urgency and result latency. Likewise, the structural communication patterns presented are well analyzed with respect to communication performance and information distribution, but those patterns are not yet addressed from other perspectives, like urgency, or data size, nor are specific uses motivated.

**Robot as a Service in Cloud Computing.** Chen proposed viewing robots as a service withing deployed constellations in 2011. In this model, which he did implement, he developed clear interfaces to robots using web-centric technologies in which robots could be controlled as a service rather than as a specific robotic entity. The implementation follows accepted service-oriented-architecture principles well, treating each robot as a service unit with specific services published within a larger service directory. These services can then be aggregated into specific applications within a given robotic constellation[4].

Chen developed and implemented the original robot as a service model, but as of yet has not provided analytical guidance on when this kind of model is appropriate, exactly why it should be used, and how it could be implemented at scale.

**Cloud Robotics: Towards Context Aware Robotic Networks** Quintas, in 2011, outlined and implemented a hybrid system of smart room sensors and robots integrated via message-based service oriented systems. This use of service oriented architecture (SOA) is closely related to cloud computing systems, and though it does not display the provisioning or scale of cloud computing in the outlined system, implementation of the services on top of a cloud provider's infrastructure would certainly address this gap[5].

Overall, the proposed system is both well designed and reflects common SOA services including service discovery, and demonstrates the applicability of this general class of architectures to cloud robotics. It does not however provide any guidance with regard to when this kind of system should be used, or how it should enhance the cost or performance of integrated robotic systems outside of implementing the Trinity Scenario.

**A Cloud Computing Environment for Supporting Network Robotics Applications.** In 2011, Agostinho et. al. established a cloud computing environment for controlling groups of robots in order to decrease command latencies between controllers and robots. Here, the robots hosted HTTP microframeworks, exposing interfaces with REST semantics to VMs installed on systems in close proximity. Those VMs were then controlled by remote users. Traffic between VMs and robots was over HTTP, with strict host-based firewalling protecting robots from spurious VM commands [6].

An efficient approach to bringing control elements closer to controlled robots certainly, though perhaps not very secure. Agostinho does not explicitly address security over the HTTP-centric control connections, though conceivably the connections could be secured by uni- or bi-directional authenticating HTTPS. This does however imply the existence of robust PKI, adding complexity to the system overall. The system as presented has no identifiable security mechanisms, making robot hijacking a straightforward exercise, though in all fairness this is a laboratory environment and the implications of such hijacking are trivial.

**DAvinCi: A Cloud Computing Framework for Service Robots.** Arumugam et. al. presented their work on DAvinCi, a cloud computing framework for robots using Hadoop, HDFS, and ROS to speed up parallelizable computation. The system as a whole used ROS over HTTP as well as Zigbee and IEEE 802.11 to pass data between the Hadoop cluster and the robots. The cluster itself had eight distinct nodes. Demonstrating the power of this approach, Arumugam and his team implemented a parallelized FastSLAM algorithm using Hadoop, realizing nearly an order of magnitude improvement in processing performance over a single computational node implementation [7].

Though an excellent example of using communication networks to increase overall system performance for robots, this is not yet a cloud solution by NIST's definition of cloud computing. This is a promising start however, and shows specifically how Hadoop and related technologies can fit into robotic control systems. This particular problem, mapping an area for a constellation of robots,

is a perfect example of a problem that only needs be solved once for a group of robots. Job management for individual robots in a larger constellation will be more complex, and will require more challenging job scheduling.

## References

- [1] K. Kamei, S. Nishio, N. Hagita, and M. Sato. Cloud networked robotics. *Network, IEEE*, 26(3):28–34, 2012.
- [2] D. Lorencik and P. Sincak. Cloud robotics: Current trends and possible use as a service. In *Applied Machine Intelligence and Informatics (SAMi), 2013 IEEE 11th International Symposium on*, pages 85–88, 2013.
- [3] Guoqiang Hu, Wee Peng Tay, and Yonggang Wen. Cloud robotics: architecture, challenges and applications. *Network, IEEE*, 26(3):21–28, 2012.
- [4] Yinong Chen, Zhihui Du, and M. Garcia-Acosta. Robot as a service in cloud computing. In *Service Oriented System Engineering (SOSE), 2010 Fifth IEEE International Symposium on*, pages 151–158, 2010.
- [5] J Quintas, P Menezes, and J Dias. Cloud robotics: Towards context aware robotic networks. In *Proc. of the 16th IASTED International Conference on Robotics, Pittsburgh, USA*, 2011.
- [6] L. Agostinho, L. Olivi, G. Feliciano, F. Paolieri, D. Rodrigues, E. Cardozo, and E. Guimaraes. A cloud computing environment for supporting networked robotics applications. In *Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on*, pages 1110–1116, 2011.
- [7] R. Arumugam, V.R. Enti, Liu Bingbing, Wu Xiaojun, K. Baskaran, Foong Foo Kong, A.S. Kumar, Kang Dee Meng, and Goh Wai Kit. Davinci: A cloud computing framework for service robots. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 3084–3089, 2010.