

Cloud Robotics: Sistema Multi-Robot conectado al Cloud público AWS

Rodriguez Ismael P.¹, Paniego Juan Manuel¹, Rodriguez Eguren Sebastian¹, Estrebou Cesar¹, De Giusti Armando E.^{1,2}

¹Instituto de Investigación en Informática LIDI (III-LIDI),

Facultad de Informática, Universidad Nacional de La Plata, Argentina

²CONICET, Ministerio de Ciencia, Tecnología e Innovación Productiva, Argentina.

{ismael, jmpaniego, seguren, cesarest, degiusti}@lidi.info.unlp.edu.ar

Abstract.

Este trabajo presenta el despliegue de un escenario de sistema multi-robot, integrado por robots conectados vía WiFi al Cloud público de Amazon Web Services, simulando un sistema de riego artificial de campo. Se detallan los prototipos desarrollados, como así también el protocolo de comunicación utilizado con el servicio de AWS IoT, y los algoritmos implementados que permiten procesar las imágenes adquiridas e identificar determinados patrones y objetos.

Keywords: *Cloud Robotics, Cloud Computing, Internet of Things, AWS, MQTT.*

1 Introducción

Los avances en el paradigma de Cloud Computing han provocado un factor disruptivo de las TI en la industria tecnológica. Según el Instituto Nacional de Estándares y Tecnologías del Departamento de Comercio de los EEUU (NIST), como en varias publicaciones de diversos autores, se ha definido a Cloud Computing como: “un paradigma informático de cómputo distribuido, que proporciona grandes conjuntos de recursos virtuales (como ser hardware, plataformas de desarrollo, almacenamiento y/o aplicaciones), fácilmente accesibles y utilizables por medio de una interfaz de administración web. Estos recursos son proporcionados como servicios (del inglés, “as a service”) y pueden ser dinámicamente reconfigurados para adaptarse a una carga de trabajo variable (escalabilidad), logrando una mejor utilización y evitando el sobre o sub dimensionamiento (elasticidad). El acceso a los recursos se realiza bajo la demanda de los usuarios, en base a un modelo de autoservicio” [1].

El modelo de Cloud Computing presenta las características y beneficios siguientes: recursos disponibles bajo demanda, escalabilidad y elasticidad, aprovisionamiento automático de recursos y autoservicio.

Dicho paradigma, brinda al menos tres modelos de despliegue: Cloud Público, Cloud Privado y Cloud Híbrido [2] [3].

Por otro lado, considerando la gran influencia de la robótica en la sociedad y la diversidad de servicios ofrecidos por robots, nos encontramos que los mismos presentan grandes limitaciones en consumo de energía, poder de cómputo, capacidad de almacenamiento, toma de decisiones, tareas cognitivas, etc.

En el año 2010, comenzaron a surgir proyectos de investigación (ej: RoboEarth [4]), que integraban las tecnologías de Cloud con los sistemas de robots. Es así, que James Kuffner propone el concepto de Cloud Robotics, basado en combinar las tecnologías de robots con el paradigma de Cloud Computing [5].

La idea de Cloud Robotics, permite por medio de aplicaciones tratar los datos de los componentes de hardware del robot (sensores, actuadores, cámaras, microcontroladores, memoria, etc.), sin importar las limitaciones de cómputo de las placas de desarrollo con microcontroladores y la capacidad de almacenamiento de las mismas [6]. En otras palabras, este concepto permite a los robots obtener resultados de tareas de cómputo intensivo, tales como: procesamiento de imágenes, reconocimiento de voz, determinación de rutas, confección de mapas, acciones cognitivas, etc., sin tratamiento local, sino en el Cloud.

Este paradigma brinda la capacidad de establecer escenarios para sistemas de multi-robot, donde cada robot se integra de un hardware mínimo; una placa microcontroladora con conectividad WiFi permite al robot comunicarse con el Cloud y otros robots. Los datos de sensores y/o sistemas de adquisición de imágenes se procesan en el Cloud y los actuadores de cada robot llevarán a cabo las operaciones necesarias [7].

El propósito de investigación del presente trabajo es el despliegue de un escenario de multi-robot, inicialmente con dos robots conectados al Cloud Público de Amazon Web Services con el fin de simular un sistema de riego artificial de campo. Para alcanzar tales objetivos, se determinó el método de comunicación con el Cloud; se desarrollaron algoritmos que permitan procesar imágenes adquiridas por medio de una cámara de video, identificando determinados patrones y objetos, como así también, se dotó a uno de los robots, de un sistema autónomo con capacidad de orientarse, trasladarse y detenerse.

La estructura de este trabajo es la siguiente: la Sección 2 introduce algunos conceptos elementales. A continuación, la Sección 3 presenta los prototipos confeccionados y sus componentes. La Sección 4 describe el trabajo experimental realizado y los resultados obtenidos. Por último, la Sección 5 expone las líneas de trabajo futuro en relación a esta investigación.

2 Algunos conceptos elementales

En esta Sección, mencionaremos algunos conceptos empleados en la investigación realizada.

- **Amazon Web Services**

Es una plataforma de Cloud Público, que provee “Infraestructura como servicio”; a través de la tecnología de virtualización, provee un gran conjunto de recursos de cómputo, como ser, almacenamiento y capacidad de procesamiento, que pueden ser

solicitados a demanda, como así también, se adecuan dinámicamente en tamaño conforme la necesidad del consumidor. Los servicios más conocidos de AWS son “Elastic Compute Cloud” (EC2), “Simple Storage Service” (S3) y en nuestro caso utilizaremos los servicios de “AWS Internet of Things” (AWS IoT) [8].

- Internet of Things (IoT)

Es un nuevo paradigma cuya definición deriva de considerar “objetos” o “cosas” conectadas a Internet por medio de redes Wireless, sean estas de tecnología WiFi o 4G LTE, utilizando protocolos de comunicación estándar. Dichas “cosas” pueden considerarse etiquetas de identificación por radio frecuencia (RFID), sensores, actuadores, teléfonos móviles, etc. [9].

- AWS IoT

Es un servicio que proporciona AWS con el fin de conectar, administrar y operar grandes conjuntos de dispositivos o cosas. Ofrece mecanismos de conectividad y seguridad para la transmisión de datos. También, permite que los datos, una vez enviados a la plataforma, puedan ser procesados por las aplicaciones de análisis masivo de datos (Elastic MapReduce), análisis predictivo para aprendizaje automático (Amazon Machine Learning), almacenamiento en bases de datos, etc. [10].

AWS IoT permite conectar fácilmente dispositivos al Cloud y a otros dispositivos. El servicio es compatible con HTTP, WebSockets y MQTT. Este último se ha convertido en el protocolo por excelencia. AWS IoT utiliza MQTT v.3.1.1 con QoS 0 y 1.

Cada dispositivo debe ser registrado como una “cosa” en AWS IoT, para lo cual se emitirá un certificado y un par de llaves privada-pública para el mismo. El certificado y la llave privada, junto con el certificado de la CA de Amazon deberán ser instalados en el dispositivo con el fin que este pueda conectarse al servicio de AWS IoT.

- MQTT

Message Queuing Telemetry Transport [11] es un protocolo de comunicación ligero, especialmente diseñado para tolerar conexiones intermitentes y reducir los requisitos de ancho de banda de la red. Desde finales del año 2014, fue presentado como un estándar abierto OASIS [12]. Soporta comunicación segura con TLS. Maneja tres niveles de calidad de servicio: QoS 0: A lo sumo una vez la entrega del mensaje. QoS 1: Al menos una vez la entrega del mensaje. QoS 2: Exactamente una vez la entrega del mensaje.

Este protocolo, implementa la comunicación de mensajes por medio de la publicación/suscripción sobre un tópico específico (canal de comunicación), como se puede observar en la figura siguiente:



Figura 1: MQTT: publicación/suscripción.

3 Prototipos y componentes

Con el fin de simular un escenario de riego artificial de campo, sectorizado en 4 zonas, se han implementado dos robots que interactúan con el Cloud vía conexión WiFi; de aquí en adelante se denomina al primer robot, como “Rover 4WD” y al segundo, “Gestor de Sensores de HS”. También, se ha desplegado una instancia de AWS EC2, para ejecutar los algoritmos de procesamiento de imágenes y toma de decisiones, la cual denominaremos aplicación “Master”.

3.1 Robot Rover 4WD

El robot Rover 4WD, emula un transporte cisterna, que recibe, a través del Cloud, la zona de riego a la que debe dirigirse. Para la confección de tal robot, se han utilizado los componentes siguientes:

- Chasis Rover de 4 ruedas
Consiste en dos placas de acrílico, con cuatro motores DC de 3v, cuatro ruedas y un codificador de velocidad.



Figura 2: Chasis Rover de 4 ruedas.

- Placa de desarrollo
Placa Raspberry Pi 3 con conexión WiFi, almacenamiento en memoria MicroSD de 8Gb y alimentación de energía de 5v.



Figura 3: Placa Raspberry Pi 3.

- Cámara de video
“Pi Camera Board”, de 5MP y calidad 1080p. Se integra a la placa Raspberry Pi 3.



Figura 4: Cámara de Video para Raspberry Pi.

- Puente H
Módulo L298 para controlar la dirección de los motores y por medio de PWM la velocidad de los mismos.



Figura 5: Puente H Módulo L298.

- Sensor Giroscopio y acelerómetro
El sistema MicroElectroMecánico MPU-6050, dispone de un sensor giroscopio de 3 ejes y un sensor acelerómetro de 3 ejes, integrados juntos por un procesador digital de movimiento (DMP); este módulo permite controlar el ángulo de giro del robot.



Figura 6: Sensor Giroscopio MPU-6050.

- Decodificador óptico
Módulo Optoacoplador de tipo “herradura” para codificar la rotación de la rueda.



Figura 7: Módulo Optoacoplador tipo “herradura”.

- Prototipo Rover 4WD ensamblado



Figura 8: Robot Rover 4WD.

3.2 Robot Gestor de Sensores HS

El robot Gestor de Sensores de Humedad de Suelo, adquiere datos de nivel de humedad de suelo, provenientes de los sensores. Para la confección de tal robot, se han utilizado los componentes siguientes:

- Placa de desarrollo

Placa Raspberry Pi 3 con conexión WiFi, almacenamiento en memoria MicroSD de 8Gb y alimentación de energía de 5v.



Figura 9: Placa Raspberry Pi 3.

- Sensores de Humedad de Suelo

Consisten en 4 sensores de humedad de suelo, módulos YL38 y YL69.



Figura 10: Sensor de Humedad de Suelo YL38 + YL69.

- Conversor ADC

Microchip MCP 3008, que convierte datos analógicos a digital. Posee 8 canales analógicos, de los cuales 4 se utilizan la conectar los sensores de humedad de suelo.



Figura 11: Conversor ADC MCP 3008.

- **Marcadores gráficos**

Consisten en 4 marcadores gráficos que identifican cada una de las zonas de riego, con un sensor de humedad de suelo en cada una de ellas.

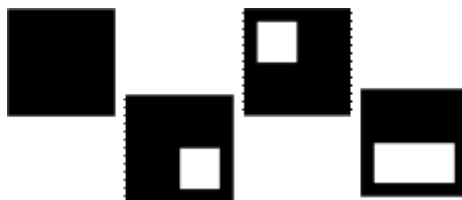


Figura 12: Marcadores Gráficos.

3.3 Instancia AWS EC2 – Aplicación “Master”

Consiste en el despliegue de una instancia de AWS EC2 “t2.large”, con 2 VCPU y 8GB de RAM; Sistema Operativo Linux distribución Ubuntu Server 14.04 LTS [13] [14]. Dicha instancia ejecuta los algoritmos de procesamiento de imágenes y la toma de decisiones sobre los robots Rover 4WD y Gestor de Sensores HS.

3.4 Escenario sistema de riego artificial de campo

Se puede observar, el sistema de riego de campo simulado, en la figura siguiente:



Figura 14: Escenario sistemas de riego artificial de campo.

4 Trabajo experimental

El trabajo de investigación consistió en desarrollar un escenario que simule un sistema de riego artificial de campo, emulando un robot cisterna de transporte y un gestor de sensores de humedad de suelo. Se puede observar la conexión de los componentes de los robots Rover 4WD y Gestor de Sensores HS, en las figuras 15 y 16, respectivamente.

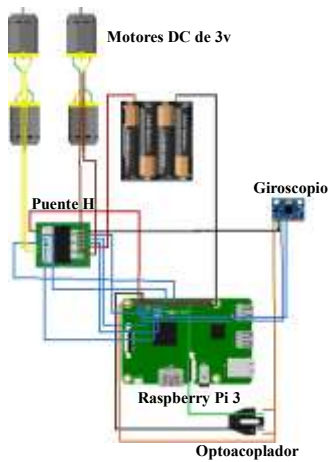


Figura 15: Rover 4WD.

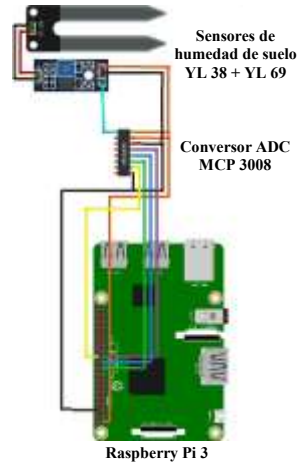


Figura 16: Gestor de Sensores HS.

Ambos robots, se conectan a internet vía tecnología WiFi, al Cloud Público de Amazon Web Services. Para tal fin se dispone de una cuenta en el proveedor de Cloud, y se utilizan los servicios AWS EC2 y AWS IoT.

Sobre la plataforma de AWS se despliega una instancia “t2.large”, con S.O. Linux Ubuntu Server 14.04 LTS, con Python v.3.4.2 y la librería “Paho Python Client” que dispone de una clase cliente que soporta MQTT v.3.1.1 sobre Python v.3.x. En esta instancia se instala el certificado de dispositivo de AWS IoT, el certificado de la CA de Amazon y la clave privada del dispositivo (la instancia es considerada un dispositivo o “cosa” conectada al servicio AWS IoT). Además, cuenta con la aplicación “Master”, escrita en lenguaje Python y tiene como objetivo procesar imágenes y tomar decisiones sobre los dispositivos robot Rover 4WD y Gestor de Sensores HS. En la figura 17 se puede observar el sistema multi-robot conectados a AWS IoT.

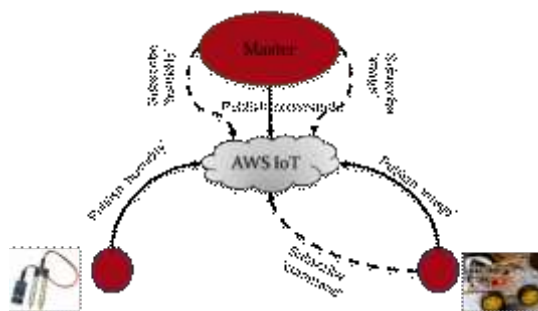


Figura 17: Cloud Robotics: Sistema multi-robot conectado a AWS IoT.

Los dispositivos se conectan a Internet y utilizan el protocolo de comunicación MQTT para interconectar los robots con el Cloud. Una vez efectuada negociación de

autenticación de seguridad, se establecen los canales de comunicación a utilizar (tópicos).

A continuación, sobre el robot Rover 4WD, se realiza una tarea de calibración al sensor giroscopio, como también, se ingresan los umbrales de nivel de humedad de suelo aceptables, en el robot Gestor de Sensores HS.

En esta instancia, los robots se encuentran preparados para comenzar a operar.

El robot de Sensores HS comienza a adquirir los niveles de humedad de suelo de cada una las zonas de riego, provistas de un sensor y un marcador gráfico que las distingue. Cada valor adquirido se compara con el valor del umbral, de la zona correspondiente, ingresado por el usuario al inicio. Esto permitirá, que solo aquellos que se encuentren dentro del umbral, sean enviados al Cloud.

Posteriormente, la aplicación “Master”, utilizando el protocolo MQTT (Figura 18), se suscribe al tópico “humidity” y recibe los valores de humedad, los cuales son procesados, asignándoles una prioridad, la cual servirá para decidir qué zona necesita de riego con mayor antelación.

El robot Rover 4WD, se suscribe al tópico “command”, por el cual recibe del Cloud el comando para iniciar su trabajo. En este instante, comienza la comunicación entre ambos. Principalmente, se realizan capturas de imágenes para ubicar la zona a ‘regar’. Esta captura, es realizada por el Rover y enviada al Cloud, publicando la misma en el tópico “image”, para iniciar la detección de la zona a través de los marcadores.

En el Cloud, la aplicación “Master” se suscribe al tópico “image” y por medio de JSON, regenera la imagen recibida del robot. Esta es procesada por el algoritmo de procesamiento de imágenes, utilizando la biblioteca OpenCV (Open Source Computer Vision) [15], por medio de la cual intenta identificar un cuadrilátero con contorno de color negro, para luego examinar su interior y obtener el ID del marcador de la imagen. En caso de no identificar el marcador, la aplicación “Master” publicará en el tópico “command” los comandos para girar a derecha o izquierda una cierta cantidad de grados; posteriormente, se publica sobre el mismo tópico, el comando para capturar una nueva imagen.

Este ciclo se repite hasta detectar en la imagen un marcador gráfico. Una vez detectado, se obtiene la distancia de separación del robot al marcador, y se publica en el tópico “command” el comando para avanzar una cierta distancia hasta detenerse a 20 cm del marcador. Ya posicionado frente al marcador, publica en el tópico “finish” para que el robot emita un sonido de finalización y realice un movimiento en reversa hasta su posición inicial, quedando disponible para realizar la siguiente búsqueda que especifique el Cloud.

Tópico	Mensaje	Descripción
humidity	Arreglo con el valor analógico de humedad y la 'zona' a la que hace referencia. Ej: [m':0, 'h':600]	Se envía desde el robot de sensor de HS. Se envía al detectar uno de los valores de humedad dentro del umbral definido.
command	Avanzar una distancia hacia adelante ('a x') o girar sobre el eje una determinada cantidad de grados ('[d] x'). Así como también el pedido de captura de imagen.	Los mensajes de comandos, se envían desde el Cloud al Rover 4WD. Según el contenido del 'payload' o 'mensaje', el rover actúa de manera distinta.
image	Imagen capturada, en base64	Se utiliza para transmitir las imágenes capturadas. Las mismas se pasan a base64 para poder ser transmitidas desde el rover al cloud.
finish	-	Su uso es solo para advertir desde el rover la finalización de las tareas al alcanzar el objetivo.

Figura 18: Tópicos de mensajes de MQTT.

5 Conclusiones y Líneas de Trabajo Futuro

En esta investigación se ha implementado un sistema multi-robot conectado al Cloud. Se han desarrollado algoritmos para el procesamiento de imágenes, utilizando las librerías de OpenCV, y toma de decisiones, para indicar las acciones que debe tomar cada robot. Podemos concluir que Cloud Robotics es una tecnología que favorecerá el avance de los sistemas multi-robot reduciendo las limitaciones actuales.

Las líneas de trabajo futuro relacionadas con este trabajo incluyen la integración de una cámara aérea, que permita procesar una imagen y determinar la ubicación del robot Rover 4WD y del Marcador de la zona de riego, para dirigir el robot hasta dicha zona. Además, es deseable incorporar cierto nivel de “inteligencia” para eludir obstáculos.

Por otro lado, adicionar uno o más robots Rover 4WD, adaptando la solución para atender la prioridad de la zona de riego por cercanía, estado de cisterna, etc. También es deseable, incorporar trabajos colaborativos de mayor complejidad a ser efectuados por un conjunto de robots.

Referencias

1. Mell P., Grance T.: “The NIST Definition of Cloud Computing”. In: Publicación Especial 800-145. Septiembre, 2011.
2. Pettoruti J., Rodriguez I., Chichizola F., De Giusti A.: “Análisis de la degradación de las comunicaciones en algoritmos de cómputo científico en un Cloud privado”. In: XII Workshop de Procesamiento Distribuido y Paralelo (WPDP) – XVIII Congreso Argentino de Ciencias de la Computación (CACIC2012). Bahía Blanca, Argentina, 2012.
3. Rodriguez, I., Pettoruti, J.E., Chichizola, F., De Giusti, A.: Despliegue de un Cloud Privado para entornos de cómputo científico. In: Proceedings del XI Workshop de Procesamiento Distribuido y Paralelo (WPDP) - XVII Congreso Argentino de Ciencias de la Computación (CACIC 2011). La Plata, Argentina. (2011).
4. RoboEarth. In: <http://www.roboearth.org>. Junio, 2016.
5. Kuffner J.: “Cloud-enabled robots”. In: IEEE-RAS International Conference on Humanoid Robot. Nashville, USA, 2010.
6. Wang L., Liu M., Meng M., Siegart R.: “Towards Real-Time Multi-Sensor Information Retrieval in Cloud Robotic System”. In: IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI). Hamburgo, Alemania, 2012.
7. Turnbull L., et al.: “Cloud Robotics: Formation Control of a Multi Robot System Utilizing Cloud Infrastructure”. In: Proceedings of IEEE – Southeastcon. Jacksonville, USA, 2013.
8. Bermudez I., Traverso S., Mellia M., Munafò M.: “Exploring the cloud from passive measurements: The Amazon AWS case”. In: Proceedings of IEEE – INFOCOM, 2013.
9. Atzori L., Iera A., Morabito G.: “The Internet of Things: A survey”. In: ELSEVIER Journal Computer Networks. Volume 54, Issue 15. 2010.
10. Informe técnico: AWS Whitepapers. “Core Tenets of IoT”. Abril 2016.
11. Protocolo MQTT: <http://mqtt.org>. Julio 2016.
12. OASIS MQTT: <https://www.oasis-open.org>. Julio 2016.
13. AWS Instance Types: <https://aws.amazon.com/es/ec2/instance-types>. Julio 2016.
14. AWS Marketplace: <https://aws.amazon.com/marketplace/pp/B00JV9JBDS>. Julio 2016.
15. OpenCV: <http://opencv.org>. Julio 2016