

Análisis de desempeño de Serverless para problemas HPC

Joaquín Lebeti⁴ María Murazzo¹, Nelson Rodríguez¹, Ana Laura Molina¹

¹ Departamento de Informática - F.C.E.F. y N. - U.N.S.J.

⁴ Alumno Avanzado Licenciatura en Cs. de la Computación - F.C.E.F. y N. - U.N.S.J.
Complejo Islas Malvinas. Cereceto y Meglioli. 5400. Rivadavia. San Juan, 0264 4234129
lebejoaquin@gmail.com, marite@unsj-cuim.edu.ar, nelson@iinfo.unsj.edu.ar,
almm95@gmail.com

Resumen. Se ha demostrado que la ejecución de aplicaciones de HPC en el cloud es una opción viable a las arquitecturas paralelas o distribuidas convencionales, las cuales requieren un alto grado de administración, así como un pobre escalado de recursos. El enfoque tradicional para un usuario usualmente es utilizar al proveedor de Cloud para aprovisionar máquinas virtuales (VM) empleándolas de manera similar a una infraestructura local, con el consiguiente problema de la administración de recursos sumado a la degradación de la performance de las aplicaciones por la contextualización de los ambientes virtualizados. Serverless computing, permite a un usuario ejecutar código escrito en el lenguaje de programación de su elección, sin tener que aprovisionar primero una máquina virtual. Por otro lado, la elasticidad, disponibilidad, escalabilidad y la tolerancia a fallas son proporcionadas de manera transparente por el proveedor cloud. De esta manera es posible disminuir la complejidad de la administración de la infraestructura para el desarrollador, permitiéndole que se centre en la lógica de la aplicación. Y además surgen ventajas económicas, al pagar solo por el tiempo de uso. El trabajo se centra en el desafío de evaluar el costo, no solo monetario sino también de performance, de migrar aplicaciones de HPC a entornos serverless. Esta evaluación permitirá que se pueda tomar la decisión que infraestructura se usará con la finalidad que se obtenga el mejor beneficio de performance.

Keywords: Serverless Computing, HPC, Cloud Computing

1 Introducción

La popularización de IoT y la masificación de las infraestructuras cloud durante el último tiempo han abierto un mundo de posibilidades para las aplicaciones HPC. Esto se debe a que los dispositivos IoT generan una gran cantidad de datos, los cuales se hace impráctico tratarlos con paradigmas tradicionales. Para lograr el procesamiento adecuado de estos datos con características de velocidad y tamaño importantes, se hace necesario prescindir de los paradigmas de programación tradicionales [1]. Es por ello que es necesario aplicar algoritmos que permitan aprovechar la escalabilidad de recursos de cómputo y procesamiento de datos [2] [3]. En este sentido se plantea como solución al procesamiento de datos provenientes del IoT, técnicas de computación de alta prestaciones (HPC) con el fin de aumentar la performance de procesamiento.

Los entornos HPC son ideales para resolver aplicaciones científicas, computacionalmente costosas con manejo de grandes cantidades de datos, a fin de lograr resultados en menor tiempo. Dadas estas características, estas arquitecturas son las mejores candidatas para procesar datos provenientes del IoT. Si bien, las arquitecturas HPC han evolucionado en pos de obtener mejores tiempos de respuesta para las aplicaciones, presentan el inconveniente del escalado, de recursos de cómputo. Es por ello que una alternativa es migrar al cloud [4].

Cloud Computing se ha caracterizado por ser una tecnología centrada en ofrecer cómputo bajo demanda como cualquier otro servicio. Esto es una ventaja para montar aplicaciones donde es necesario el procesamiento intensivo, tales como aquellas aplicaciones que procesen y extraigan información de datos provenientes de dispositivos de IoT [5].

Los proveedores cloud alegan muchas ventajas en la migración de aplicaciones de HPC, como el acceso rápido a los recursos, costos más bajos y flexibilidad en la contratación y el aprovisionamiento de recursos. Un punto adicional es la seguridad, la cual, afirman es de alto nivel y en muchos casos muy difícil de implementar en la mayoría de los laboratorios, ya que tener personal de TI especializado en seguridad no es común [6].

Un aspecto más que lleva a la adopción del cloud como plataforma de despliegue de aplicaciones HPC es, mejorar la colaboración científica, es decir, facilitar la investigación colaborativa y la innovación; este aspecto ha sido el foco de este proyecto desde hace varios años al incorporar investigadores de otras universidades del país.

Otro tema es el potencial ahorro de tiempo que ofrece el cloud a los usuarios finales. Estos usuarios no necesitan preocuparse por actualizaciones de software, compatibilidad o parches de seguridad, pues todo esto es aprovisionado de forma transparente.

Sin embargo, el cloud tiene dos grandes desventajas, la primera es la degradación de la performance de las aplicaciones al montarlas sobre arquitectura virtualizada, debido a que genera overhead en la contextualización de las máquinas virtuales; la segunda desventaja cuando se despliegan aplicaciones en el cloud, es que es responsabilidad de la organización mantener funcionando de forma correcta la infraestructura que se necesite para el despliegue de las aplicaciones, lo cual lleva a cargar costos sobre el presupuesto para su mantenimiento y soporte [7].

En este sentido, la aparición del Serverless Computing [8] logra que los desarrolladores no tengan que preocupar por el aprovisionamiento y escalado de la infraestructura, por lo que se pueden centrar en la lógica de sus aplicaciones. De esta forma es posible lograr la abstracción de la gestión de servidores (aprovisionamiento, configuración, escalado, etc.) para que los usuarios, en este caso desarrolladores, puedan enfocarse en la lógica de sus aplicaciones.

2 Trabajos relacionados

Se ha mencionado en párrafos anteriores las ventajas y desventaja de migrar aplicaciones HPC al cloud. Sin embargo, el tiempo y el esfuerzo necesarios para configurar los recursos virtuales pueden ser mayores que el tiempo y el esfuerzo reales dedicados a hacer los cálculos. En contraposición, si se usa el paradigma serverless,

será posible tener control más granular sobre el servicio prestado al dejar en manos del proveedor cloud la administración de la infraestructura.

En [9], se ha realizado un mapeo sistemático de 89 casos de uso donde se aplicó el paradigma serverless, para resolver problemas en su mayoría que se encuadran en HPC. Pero hay escasa información sobre una comparativa de performance entre las aplicaciones ejecutándose sobre paradigma serverless frente a las mismas aplicaciones ejecutándose sobre una infraestructura cloud tradicional, en la cual se pueda hacer un análisis de comportamiento para posteriormente decidir cuál es la mejor solución para ejecutar aplicaciones HPC.

En este trabajo se usa como punto de partida [10] y profundiza las tareas de investigación en base a [11], [12], [13], entre otras de los últimos años, en las cuales se han explorado y evaluado el rendimiento del uso de serverless en aplicaciones HPC. Si bien estos estudios demuestran que serverless es fácil de usar y económico, no se ha cuantificado su efectividad sobre el enfoque convencional de aplicaciones corriendo en cloud.

3 Definición de la infraestructura

Para definir la infraestructura cloud que se utilizará, se han analizado los tres principales proveedores cloud públicos del mercado: AWS, Microsoft Azure y Google Cloud Platform y sus respectivas opciones para serverless: AWS Lambda, Azure Functions y Cloud Functions. Además se ha usado el estudio “Performance evaluation of heterogeneous cloud functions” (Figiela et al., 2018) donde se evalúa el desempeño del recurso FaaS que provee cada uno de los proveedores basándose en 7 características: rendimiento computacional, rendimiento de red, transferencia de datos entre la función y el storage del proveedor, sobrecargas de la función, duración de instancias, costos, heterogeneidad de la infraestructura. En este estudio se evalúan estas características en base a 5 hipótesis: el rendimiento de una función es proporcional al tamaño, el rendimiento de la red es proporcional al tamaño de la función, las sobrecargas no son relativas al tamaño de la función y son consistentes, las instancias se reutilizan entre llamadas y se reciclan a intervalos regulares, las funciones se ejecutan en hardware heterogéneo. Del análisis realizado, se concluye que el funcionamiento de AWS Lambda y Cloud Functions es consistente con la hipótesis de que el rendimiento de una función está relacionado con el tamaño de la misma. En el caso de Lambda, esta relación es más fuerte que en Cloud Functions ya que ésta, en más de un aspecto estudiado, presenta una mejora en el rendimiento en un 5% del total de ejecuciones realizadas. Por otro lado, Azure Functions presenta diferencias en el rendimiento con las dos primeras ya que el tamaño de la función no es configurable. En este trabajo se usa como infraestructura cloud Google Cloud Platform.

4 Escenarios de Trabajo

Se plantean tres escenarios en los cuales se evaluarán: tiempo de ejecución, tiempo de respuesta, performance, precio, resultados, facilidad de uso, tiempo de contextualización, aspectos particulares de cada escenario que lo destacan.

El primer escenario de evaluación es cloud “tradicional”. Este escenario se hace sobre un recurso que funciona como PaaS para el cual se utilizará Dataproc, el cual permite poner a funcionar un cluster con Spark de n nodos, n-1 workers y 1 master.

El segundo escenario es serverless y el tercer escenario, se trata de una extensión del primero y la filosofía de serverless del segundo escenario, busca combinar ambos componentes. Para estos dos escenarios se usará Cloud Functions. La configuración de este escenario consta de n nodos, n-1 workers y 1 master para hacer una equivalencia con un cluster de n-1 workers y 1 master de la configuración de Dataproc. El nodo maestro será el encargado de desencadenar la ejecución de la función worker n-1 veces para arrancar n-1 instancias en “simultáneo” y luego recolectar los resultados que obtenga cada una de esas instancias para resolver un resultado único que será el resultado final del problema a resolver. Para resolver cada una de las partes del problema, en la función worker se utilizará pandas para trabajar el archivo de datos como un dataframe.

5 Resultados obtenidos

El objetivo del estudio es evaluar el comportamiento de FaaS para problemas de HPC con datos provenientes de IoT, y como el rendimiento de las funciones es variable respecto del tamaño de la misma, se evaluará el mismo problema con diferentes tamaños de función.

El problema a resolver es determinar, a partir de datos históricos obtenidos por sensores desde 2009 hasta 2022, la hora del día en que la cantidad de CO (monóxido de carbono) es más alta. El dataset en cuestión cuenta inicialmente con una cantidad aproximada de 110 mil registros, es decir, un registro por cada hora del día desde 2009 hasta el 2022 en las ciudades de Centenario, Córdoba y La Boca. Con el objetivo de dar resultados precisos, como el dataset contaba con alrededor del 40% de registros sin datos, lo que se hizo en un comienzo es una limpieza de los mismos. Tras esto se obtuvo un dataset con 70 mil registros que será utilizado para comprobar el comportamiento de las dos configuraciones anteriormente descritas.

Para hacer una comparativa válida, se resolvió el problema con dos configuraciones equivalentes. La configuración de recursos mínima para cada nodo en Dataproc es de 1 vCPU y 3.5GB de RAM.

Si bien es interesante ver el comportamiento de la memoria, la comparativa principal es sobre el procesador. Por ello, para hacer una comparativa equivalente, se comenzará evaluando con 1 vCPU (3 workers, 1 master)

Para la configuración PaaS con Dataproc se configuran los recursos del cluster desde una interfaz gráfica y se trabaja sobre de Jupyter

Actualmente se está trabajando en la ejecución de los escenarios para FaaS con el objeto de poder realizar el análisis de performance de ambas soluciones. También se pretende lograr recabar información sobre los aspectos positivos y negativos del uso de FaaS para resolver problemas de HPC, así como la viabilidad de esta implementación.

6 Conclusiones y Futuros trabajos

Aun no se cuenta con información acabada sobre el comportamiento de Serverless para resolver problemas HPC. Solamente se han realizado las pruebas sobre una infraestructura tradicional. Las configuraciones de FaaS ya están listas, pero aún falta la ejecución de los escenarios con lo que es aun imposible realizar el análisis de performance que se fijó como objetivo.

Referencias

- [1] Farhan, L., Kharel, R., Kaiwartya, O., Quiroz-Castellanos, M., Alissa, A., & Abdulsalam, M. (2018, July). A concise review on Internet of Things (IoT)-problems, challenges and opportunities. In 2018 11Th International Symposium On Communication Systems, Networks & Digital Signal Processing (CSNDSP) (pp. 1-6). IEEE.
- [2] Medel, D., Murazzo, M. A., Molina, A. L., Sánchez, F., Cornejo, M., Rodríguez, N. R., ... & Piccoli, M. F. (2019). La Computación de Alta Performance como soporte a los sistemas altamente distribuidos. In XXI Workshop de Investigadores en Ciencias de la Computación (WICC 2019, Universidad Nacional de San Juan).
- [3] Barrionuevo, M., Escalante, J., Lopresti, M., Lucero, M., Miranda, N. C., Murazzo, M. A., & Piccoli, M. F. (2020). Solución de grandes problemas aplicando HPC multitecnología. In XXII Workshop de Investigadores en Ciencias de la Computación (WICC 2020, El Calafate, Santa Cruz).
- [4] de Souza Cimino, L., de Resende, J. E. E., Silva, L. H. M., Rocha, S. Q. S., de Oliveira Correia, M., Monteiro, G. S., ... & de Castro Lima, J. (2017, November). IoT and HPC integration: revision and perspectives. In 2017 VII Brazilian Symposium on Computing Systems Engineering (SBESC) (pp. 132-139). IEEE.
- [5] Biswas, A. R., & Giaffreda, R. (2014, March). IoT and cloud convergence: Opportunities and challenges. In 2014 IEEE World Forum on Internet of Things (WF-IoT) (pp. 375-376). IEEE.
- [6] Añel, J. A., Añel, J. A., Montes, D. P., Iglesias, J. R., & Romano. (2020). Cloud and Serverless Computing for Scientists. Springer International Publishing.
- [7] Malla, S., & Christensen, K. (2020). HPC in the cloud: Performance comparison of function as a service (FaaS) vs infrastructure as a service (IaaS). Internet Technology Letters, 3(1), e137.
- [8] Baldini, I., Castro, P., Chang, K., Cheng, P., Fink, S., Ishakian, V., ... & Suter, P. (2017). Serverless computing: Current trends and open problems. In Research advances in cloud computing (pp. 1-20). Springer, Singapore.
- [9] Eismann, S., Scheuner, J., Van Eyk, E., Schwinger, M., Grohmann, J., Herbst, N., ... & Iosup, A. (2020). A review of serverless use cases and their characteristics. arXiv preprint arXiv:2008.11110.
- [10] Rodríguez, N. R., Murazzo, M. A., Medel, D., Parra, L., Molina, A. L., Sánchez, F., ... & Vargas, L. (2021). Procesamiento paralelo sobre arquitecturas serverless para tratamiento de datos provenientes del IoT. In XXIII Workshop de Investigadores en Ciencias de la Computación (WICC 2021, Chilecito, La Rioja).
- [11] Niu, X., Kumanov, D., Hung, L. H., Lloyd, W., & Yeung, K. Y. (2019, September). Leveraging serverless computing to improve performance for sequence comparison.

In Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (pp. 683-687).

[12] Spillner, J., Mateos, C., & Monge, D. A. (2017, September). Faaster, better, cheaper: The prospect of serverless scientific computing and hpc. In Latin American High Performance Computing Conference (pp. 154-168). Springer, Cham.

[13] Chard, R., Skluzacek, T. J., Li, Z., Babuji, Y., Woodard, A., Blaiszik, B., ... & Chard, K. (2019). Serverless supercomputing: High performance function as a service for science. arXiv preprint arXiv:1908.04907.