

Plataforma de HPC portable de bajo consumo energético para aplicaciones de minería de datos

Portable low-power High Performance Computing (HPC) Platform for data mining applications

Baixo consumo de energia portátil plataforma HPC para aplicações de exploração de dados

DOI: <http://dx.doi.org/10.23913/reci.v6i11.62>

Armando Saúl Carranza Sánchez

Instituto Tecnológico de Colima, México

G1546005@itcolima.edu.mx

Jesús Alberto Verduzco Ramírez

Instituto Tecnológico de Colima, México

averduzco@itcolima.edu.mx

Nicandro Farías Mendoza

Instituto Tecnológico de Colima

nfarias@itcolima.edu.mx

Francisco Cervantes Zambrano

Instituto Tecnológico de Colima, México

francisco.cervantes@itcolima.edu.mx

Fernando Rodríguez Haro

Universidad de Colima, México

ferharo@ucol.mx

Resumen

La necesidad de plataformas computacionales que proporcionen soporte a las aplicaciones denominadas intensivas ha estado incrementándose en muchas organizaciones debido al surgimiento de una serie de aplicaciones que requieren de manejo intensivo de datos y tiempos de respuesta cortos. Un ejemplo pueden ser las aplicaciones que utilizan la minería de datos. El hardware ha evolucionado de manera acelerada para satisfacer estas necesidades del cómputo intensivo. Por otro lado, pequeñas computadoras montadas en tarjetas denominadas SoC (System on Chip) (R., G., y M., 2013) han aparecido a partir de la miniaturización de componentes y la popularización de los sistemas embebidos, El ejemplo más significativo es Raspberry Pi (raspberrypi, 2016), una alternativa de bajo costo y consumo energético reducido que permite realizar tareas de cómputo. Debido a su popularidad, los fabricantes se han dedicado a incrementar su desempeño generando modelos con mayores prestaciones, los cuales se han vuelto una opción viable para el procesamiento intensivo de datos. Aquí analizamos dicho aspecto.

Para ello se describe la construcción de una plataforma de HPC basada en tarjetas SoC, lo que permite desarrollar y ejecutar aplicaciones de minería de datos. Los resultados obtenidos en las diferentes pruebas de operación y tolerancia a fallas muestran que dicha plataforma ofrece el rendimiento y la robustez necesarios para ser utilizada en el desarrollo de aplicaciones de minería de datos orientadas a la educación y también para la enseñanza de la disciplina antes mencionada, con una inversión que es posible alcanzar en instituciones académicas y pequeñas organizaciones.

Palabras clave: clúster, sistemas embebidos, computación de alto rendimiento, Sistema en Chip (SoC), minería de datos.

Abstract

The need for computing platforms that provide support to the so-called intensive applications has been increasing in many organizations because of the emergence of a series of applications that require intensive management of data and short response times. An example may be applications that use data mining. The hardware has evolved at an accelerated pace to meet these needs of data-intensive computing. On the other hand, small computers mounted on cards called SoC (System on Chip) (R., G., and M., 2013) have appeared from the miniaturization of components and the popularization of embedded systems, the most significant example is Raspberry Pi (raspberrypi, 2016), a low-cost alternative and reduced energy consumption that allows computing tasks. Because of its popularity, manufacturers have dedicated themselves to increase their performance by generating models with further benefits, which have become a viable option for data-intensive processing. Here we analyze that aspect.

Describes the construction of a High Performance Computing (HPC) platform based on SoC cards, allowing you to develop and run applications of data mining. The results obtained in the different operation and fault tolerance tests show that this platform offers the performance and robustness needed to be used in the development of applications of data mining oriented to education and also for the teaching of the above discipline, with an investment that is possible to achieve in academic institutions and small organizations.

Key words: cluster, embedded systems, High Performance Computing (HPC), System on Chip (SoC), Data mining.

Resumo

A necessidade de plataformas que fornecem suporte para aplicações intensivas chamada computação tem vindo a aumentar em muitas organizações, devido ao surgimento de uma série de aplicações que requerem gerenciamento de dados intensivos e tempos de resposta curtos. Um exemplo seria aplicações utilizando mineração de dados. O hardware tem evoluído a um ritmo acelerado para atender a essas necessidades de computação intensiva. Além disso, pequenos computadores montados em cartões de chamadas SoC (System on Chip) (R., G., M., 2013) têm aparecido desde a miniaturização de componentes e popularização de sistemas embarcados, o exemplo mais significativo framboesa Pi é (Pi framboesa, 2016), uma alternativa de baixo custo e de baixo consumo de energia que permite que as tarefas de computação. Devido à sua popularidade, os fabricantes têm dedicado-se a aumentar o seu desempenho gerando modelos de desempenho mais elevados, que se tornaram uma opção viável para o processamento de dados intensivos. Aqui analisamos este aspecto.

Para esta construção de uma plataforma HPC baseados cartões SoC descrito, permitindo desenvolver e executar aplicações de mineração de dados. Os resultados obtidos nos vários testes operacionais e tolerância a falhas mostram que esta plataforma fornece o desempenho ea robustez necessária para utilização no desenvolvimento de mineração aplicações orientada a dados educação e também para o ensino da disciplina acima com um investimento que pode ser conseguido em pequenas organizações e instituições acadêmicas.

Palavras-chave: clusters, sistemas embarcados, computação de alto desempenho, System on Chip (SoC), mineração de dados.

Fecha Recepción: Agosto 2016

Fecha Aceptación: Diciembre 2016

Introducción

La computación de alto rendimiento (HPC) es el uso de procesamiento paralelo para ejecutar aplicaciones de manera eficiente, confiable y rápida (García Nocetti, 2014). Los sistemas de cómputo de HPC han sido utilizados como herramientas para el desarrollo y ejecución de aplicaciones intensivas, tales como simulaciones computacionales y el cálculo de operaciones complejas, cuya solución implicaría un tiempo excesivo en equipos de cómputo convencionales. El HPC se basa en el uso de equipos de cómputo dotados con redundancia de hardware; ejemplos de estos equipos son clúster, supercomputadoras, entre otros.

Uno de los inconvenientes importantes del HPC es el alto costo, clasificado en el costo de adquisición y el costo de mantenimiento realizado durante la vida útil de estos equipos. Estos dos factores convierten al HPC en una tecnología reservada para organizaciones que disponen del fondo financiero suficiente. Rajovic et al. (2014) describen que los sistemas de HPC están presentes en el procesamiento de grandes cantidades de datos. Estos sistemas tienen, entre otras desventajas, el alto consumo energético necesario para su funcionamiento y además para operar el sistema de enfriamiento. En consecuencia, las organizaciones que manejan presupuestos reducidos, tales como universidades pequeñas, quedan al margen de poseer instalaciones de este tipo, lo que las sitúa en clara desventaja.

El proyecto Mont Blanc (Valero et al., 2013) plantea una alternativa enfocada en el uso de tecnologías de bajo consumo energético para HPC. Básicamente, la idea es utilizar SoCs, los cuales ofrecen ventajas con respecto a los equipos tradicionales de HPC al mitigar los aspectos mencionados (Rajovic et al., 2014).

La necesidad de los servicios proporcionados por el HPC continúa incrementándose en todo tipo de organizaciones. Un ejemplo es la minería de datos, la cual requiere de equipos de alto desempeño para su funcionamiento debido a la magnitud de los datos y procesos que genera. Actualmente se maneja el término HPDA (High Performance Data Analysis) para referirse al cómputo de alto rendimiento aplicado a la minería de datos.

Este proyecto busca crear una plataforma de HPC económica y de reducido consumo energético que sea una alternativa accesible para las organizaciones con reducido presupuesto, interesadas en el uso de la minería de datos para la mejora de sus procesos. El documento se desarrolla de acuerdo a las siguientes partes: propuesta de solución, descripción de la arquitectura, diseño, implementación, resultados y conclusiones.

Propuesta de solución

Con la finalidad de poner en práctica nuestra idea, las siguientes características técnicas han sido consideradas para el diseño e implementación de esta plataforma:

- **Bajo costo.** Uno de los principales objetivos de este proyecto es que el costo de esta plataforma se reduzca y sea así una opción rentable para instituciones académicas o pequeñas organizaciones que normalmente manejan presupuestos que impiden adquirir costosos equipos de supercómputo.
- **Reducido consumo energético.** La plataforma debe orientarse hacia el ahorro energético proporcionando servicios de alto rendimiento, pero con un consumo mínimo de energía. Con este objetivo, los componentes seleccionados para esta plataforma requieren de 5 volts y 2.5 amperes para funcionar, consumiendo 12.5 watts por hora. Esto implica un consumo energético bajo, comparado con el de una computadora personal que oscila entre 300 y 600 watts por hora.
- **Instalaciones y espacio.** Nuestra solución debe de ocupar para su instalación física un espacio reducido, en comparación al de las supercomputadoras que requieren grandes instalaciones para su operación. La plataforma estará integrada por elementos con dimensiones de 11 x 8 x 1.4 cm., que permiten cumplir con este requerimiento establecido al inicio del proyecto.
- **Potencia de procesamiento.** La plataforma construida debe proporcionar una capacidad de cómputo suficiente para poder procesar aplicaciones educativas y servir de soporte a la capacitación en las técnicas de minería de datos. Con el fin de obtener potencia de procesamiento necesaria, se integró un *clúster* con 24 tarjetas que en su conjunto proporcionan un rendimiento teórico de 768 Gflops y un estimado en Linpack (A. Petitet, 2016) de 614.40 Gflops y un consumo de 300 W/h.

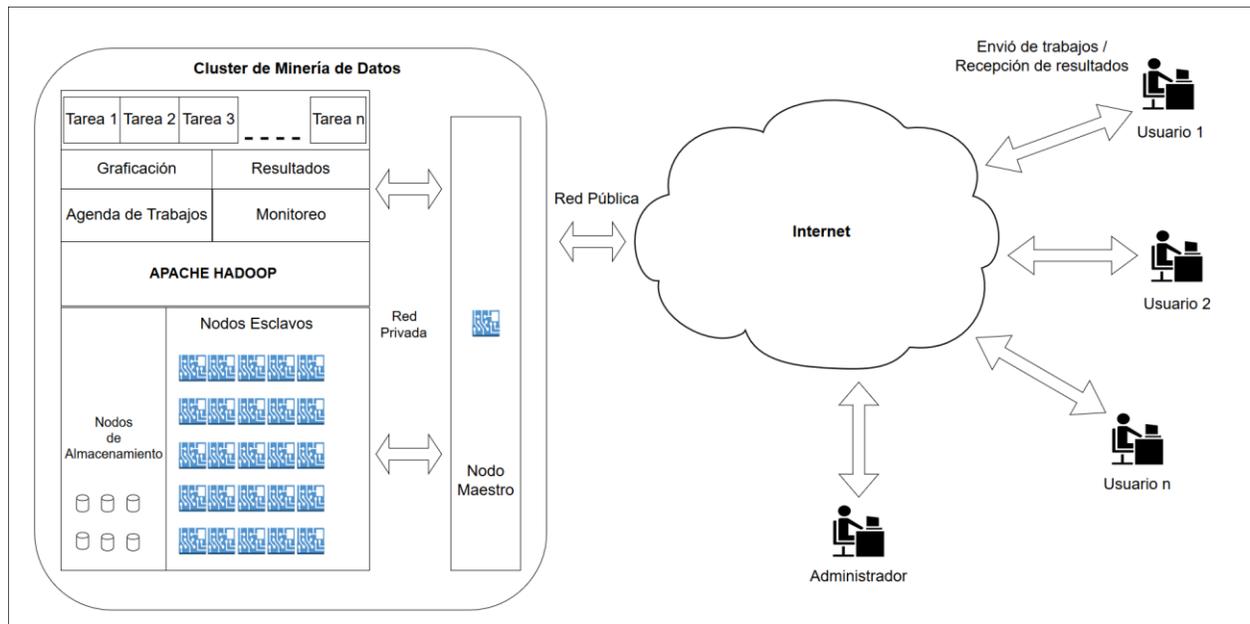
Diseño de la arquitectura

La plataforma está orientada a proveer servicios de infraestructura funcional de cómputo de alto rendimiento con el objetivo particular de usarse en aplicaciones de minería de datos. En la figura 1 se muestra el modelo conceptual de la plataforma.

La arquitectura consta de un conjunto de módulos funcionales, los cuales se describen a continuación:

Módulos de hardware

- **Nodo maestro.** Este nodo es el punto de entrada para los usuarios del *clúster* cuando desean procesar tareas. Asimismo, este nodo asigna tareas al resto de los nodos, controlando la entrada y salida de datos, y también proporciona soporte a la función de monitoreo de los nodos del *clúster*.
- **Nodos esclavos.** Realizan el procesamiento de las tareas designadas por el nodo maestro.
- **Nodos de almacenamiento.** Conjunto de nodos del *clúster* que cuentan con un disco duro para almacenar las tareas y el resultado de su procesamiento.
- **Red de interconexión privada.** Es la red de datos que interconecta los nodos maestros, esclavos y de almacenamiento.
- **Red de interconexión pública.** Esta conexión permite el acceso y uso de la arquitectura desde cualquier computadora conectada a internet.

Figura 1. Modelo Conceptual de la Plataforma.

Fuente propia: Verduzco et al., 2017

Módulos de software

- **Resultados.** En este módulo los datos obtenidos de una tarea realizada serán accesibles para el usuario; estos conjuntos de datos podrán ser consultados cuando el usuario lo requiera.
- **Graficación.** En este módulo los resultados obtenidos se muestran en un formato acompañado de gráficas que facilitan su análisis e interpretación.
- **Agenda de trabajo.** En este módulo se calendarizarán las tareas sometidas por los usuarios.
- **Monitoreo.** Este módulo permitirá dar seguimiento al estado operativo de la plataforma y la verificación del funcionamiento de cada nodo.
- **Apache Hadoop.** Es el módulo principal, el cual establece un entorno que permite administrar el procesamiento de tareas y el almacenamiento distribuido entre los nodos.

Implementación de la arquitectura

A continuación, se describen los detalles técnicos de la arquitectura y el software utilizado.

Configuración del clúster

Para la creación de la plataforma se utilizaron 24 nodos Cubieboard de los modelos A80 y Truck Plus (tabla 1). Los sistemas operativos soportados por las tarjetas son: Android (Google, 2016) y Lubuntu (Lubuntu, 2016). Para el proyecto se instaló el sistema operativo Lubuntu. El *clúster* fue configurado en la arquitectura maestro-esclavo, designando a un solo nodo como maestro y a los nodos restantes como esclavos.

Tabla 1. Tipos de SoC utilizadas en el proyecto.

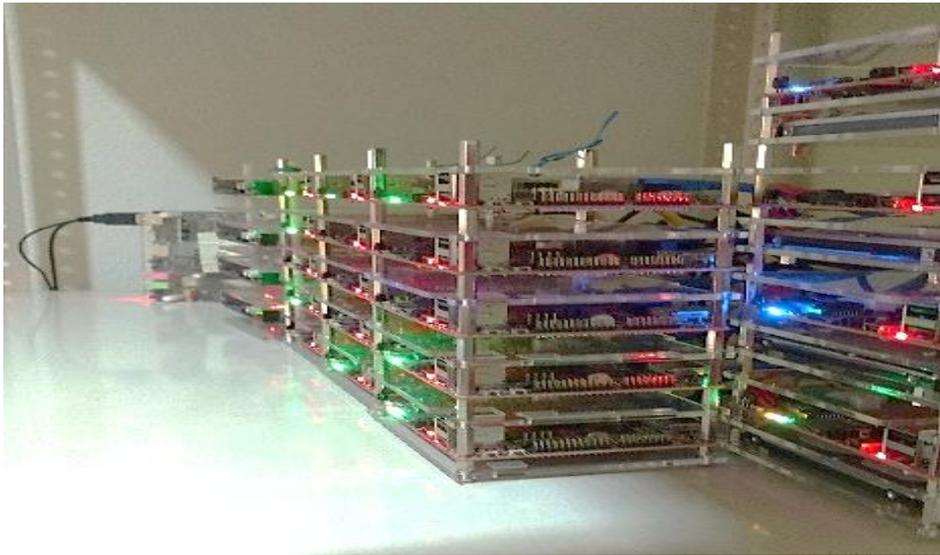
Tarjeta	Procesador	Almacenamiento	Red
Cubie board 4	Allwinner A80 Octa Core 4 x Cortex-A15 to 2016 MHz 4 x Cortex-A7 at 1320 MHz RAM: 2GB DDR3	8GB eMMC en memoria interna y 64 GB en SD	10M / 100M / 1G Gigabit Ethernet Wi-Fi with external antenna connection Bluetooth 4.0
Cubie Truck Plus	SoC A83T/H8 @ 2Ghz DRAM 2GiB DDR3 @ 672MHz (SK hynix H5TQ4G83AFR * 2)	NAND 8GB eMMC en memoria interna y en HDD hasta 2TB	10M / 100M / 1G Gigabit Ethernet

Fuente propia: Verduzco et al., 2017

Instalación de Apache Hadoop

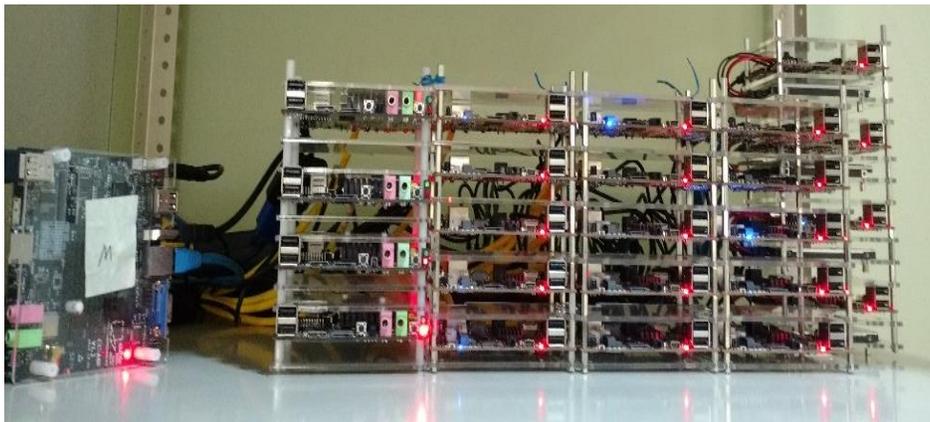
Se decidió instalar la plataforma Hadoop (Apache Hadoop, 2016), principalmente porque es muy utilizada tanto en instituciones académicas como en empresas para realizar análisis de datos, además de la variedad de *frameworks* que soporta. El método de instalación consistió en configurar en un solo nodo los programas Apache Hadoop y Apache Pig (Apache Pig, 2017), para posteriormente hacer replicaciones a los nodos restantes. Las figuras 2 y 3 muestran el clúster en operación.

Figura 2. Imagen del clúster de minería de datos en operación.



Fuente propia: Verduzco et al., 2017

Figura 3. Vista lateral del clúster de minería de datos en operación.



Fuente propia: Verduzco et al., 2017

Clúster de almacenamiento

El clúster requiere de almacenamiento dedicado para alojar las tareas y los datos generados. Para este fin, se configuró el clúster de almacenamiento integrado por cuatro nodos. A cada nodo se le instaló un disco duro de 1 TB.

Figura 4. Imagen del Clúster de almacenamiento.



Fuente propia: Verduzco et al., 2017

Resultados

Con el objetivo de determinar la funcionalidad de la plataforma, se decidió llevar a cabo diferentes pruebas clasificadas en: operación, tolerancia a fallos y estrés.

Entorno de realización de las pruebas

El entorno en el que se ejecutaron las pruebas es el siguiente:

- Un switch LINKSYS de 28 puertos Gigabit Ethernet.
- Cinco tarjetas cuebieboard a80 octa-core a 2.0 Ghz, 2 GB RAM, 8 GB en memoria NAND.
- Un nodo de las cinco fue establecido como nodo maestro, el cual asignará tareas de procesamiento a los nodos esclavos.
- 19 tarjetas cubie truck plus octa-core a 2.0 Ghz, 2 GB RAM, 8 GB en memoria NAND.
- A cuatro de estas tarjetas se añadió una unidad de disco duro para manejar el almacenamiento de archivos de entrada y salida.
- Sistema operativo Lubuntu versión 14.04 Kernel 3.4.39

- Apache Hadoop versión 2.7.2
- Software Apache Pig versión 0.16

Prueba de operación

Para realizar esta prueba fueron seleccionados un conjunto de datos resultantes de un estudio de calidad del aire realizado en diferentes provincias de España. Cada registro tiene lugar el primer día de cada mes durante el periodo del año 1997 hasta 2013, totalizando 153 121 registros. A continuación, en la Figura 5 se muestra el conjunto de instrucciones para procesar esta tarea y en la Tabla 2 los resultados obtenidos.

Figura 5. Algoritmo utilizado en la ejecución de la prueba de operación

1. `measure = load '/pruebas/calidad_del_aire_cyl_1997_2013.csv' using PigStorage(',') AS (date:chararray, co:float, no:float, no2:float, o3:float, pm10:float, sh2:float, pm25:float, pst:float, so2:float, province:chararray, station:chararray);`
2. `dump measure;`
3. `filter_measure = filter measure by date != 'dia';`
4. `measure_by_province = group filter_measure by province;`
5. `num_measures_by_province = foreach measure_by_province generate group, AVG(filter_measure.co) as measure;`
6. `DUMP num_measures_by_province.`

Fuente: Ramos, 2014

Tabla 2. Resultados prueba de operación.

Provincia	Media de Carbón en el Aire de 1997 – 2013
León	0.98
Soria	0.18
Burgos	0.86
Zamora	0.84
Ávila	0.96
Segovia	1.01
Palencia	1.17
Salamanca	1.38
Valladolid	0.68

Fuente propia: Verduzco et al., 2017

Prueba de tolerancia a fallos

El objetivo de esta prueba consiste en medir el impacto que tiene la falla de nodos del *clúster* al estar ejecutando un algoritmo. Para ello se tomó en cuenta la funcionalidad ofrecida por el sistema de archivos con el que trabaja Hadoop, el cual permite establecer el número de replicas de segmentos de datos distribuidos en los nodos del *clúster*. Para esta prueba, fueron consideradas las variables: número de nodos desactivados e índice de replicación, con el propósito de determinar el grado de éxito de que se completará la tarea y el consumo de memoria provocado por el índice de replicación seleccionado. La tarea seleccionada se procesó diez veces con el fin de obtener los indicadores de tolerancia a fallas antes señalados. La Tabla 3 muestra los resultados obtenidos.

Tabla 3. Resultados de la prueba de tolerancia a fallos.

Índice de replicación	Nodos desactivados	% Éxito al completar tarea	Consumo redundante de almacenamiento
1	2	10.0 %	0.11
3	4	20.0 %	0.24
6	8	40.0 %	0.61
9	4	20.0 %	0.24
12	4	20.0 %	0.24
15	4	20.0 %	0.24
18	4	20.0 %	0.24
21	4	20.0 %	0.24
23	18	90.0 %	4.14

Fuente propia: Verduzco et al., 2017

Asimismo, en la tabla 3 es visible que al configurar la plataforma con un índice de replicación igual a uno y con dos nodos fuera de operación, genera un éxito del 10 % en la terminación de la tarea, lo cual indica que la configuración del índice de replicación no es óptima para la plataforma. Con la configuración de replicación en seis y ocho nodos desactivados se obtuvo un éxito del 40 %, la cual es una candidata a la configuración óptima. Con el índice de replicación en 23 y 18 nodos desactivados se obtiene un éxito del 90 %, lo cual garantiza que se desarrolle la tarea si funcionan al menos seis nodos.

El inconveniente de las configuraciones del índice de replicación es el consumo redundante de almacenamiento ocasionado por las múltiples copias de datos residentes en los nodos.

Prueba de estrés

Este banco de pruebas tiene como propósito medir el tiempo de procesamiento del *clúster* de minería de datos. Para este fin, se tomó una colección de datos de 200 mil, 400 mil, 600 mil, 800 mil y 1 millón de registros, y con el objetivo de establecer tiempos de ejecución basada en la tendencia numérica de los datos se generaron las colecciones de datos antes mencionadas en las distribuciones numéricas siguientes, distribución Normal, Bernoulli, Chi-cuadrada, Hipergeométrica, Laplace, Lognormal, Poisson, Uniforme (Solano y Álvarez, 2005). El algoritmo que se aplicó a cada grupo de registros consiste en el procesamiento de lectura. A continuación, se presentan las instrucciones utilizadas:

Figura 6. Instrucciones ejecutadas para la prueba de estrés.

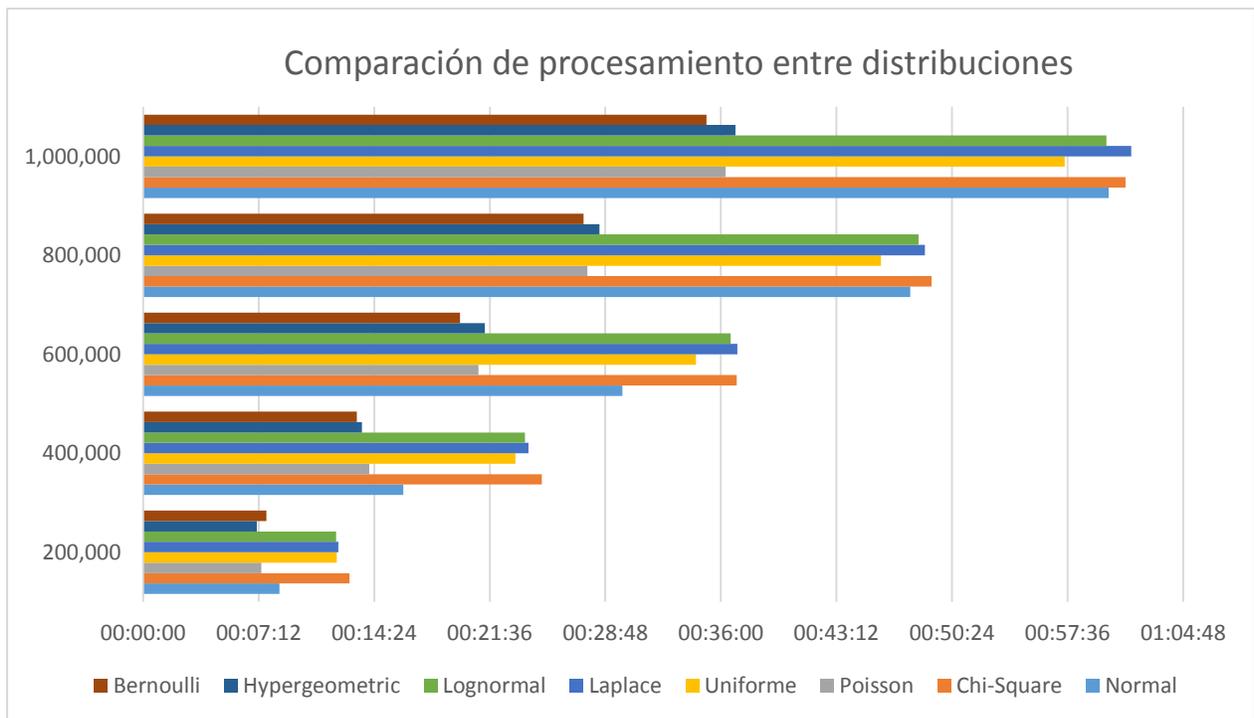
```

1. numeros = load '/numeroschiq/200milchiq.CSV' using PigStorage(',') AS (col1:float, col2:float,
col3:float, col4:float, col5:float, col6:float,col7:float, col8:float, col11:float, col10:float);
2. STORE numeros INTO 'pig_output_numeros1millonbernoulli' USING PigStorage('\t');
    
```

Fuente propia: Verduzco et al., 2017

Cada colección de datos se procesó tres veces y se obtuvo el promedio del tiempo requerido para completar la operación. La Figura 6 muestra los resultados obtenidos.

Figura 7. Comparación de procesamiento entre distribuciones.



Fuente propia: Verduzco et al., 2017

Como se puede apreciar en la Figura 7, los tiempos de ejecución varían dependiendo de la distribución numérica de donde provienen los datos. Las distribuciones con mayor tiempo de ejecución son Chi-cuadrada y Laplace.

Conclusiones

En este documento se muestra la implementación de una plataforma orientada al desarrollo y ejecución de aplicaciones de minería de datos. Lo notable de esta plataforma es que está constituida por tecnologías SoC, lo que implica un costo reducido. Otros aspectos a resaltar, son el reducido espacio que ocupan sus instalaciones y el bajo consumo energético asociado a su funcionamiento. Las distintas pruebas efectuadas nos permiten afirmar que esta plataforma proporciona la funcionalidad necesaria para ejecutar algoritmos orientados a la educación universitaria, investigación y capacitación. Todos los aspectos mencionados convierten a esta plataforma en una alternativa interesante para instituciones con bajo presupuesto que quieran contar con sistemas de HPC.

Trabajos futuros estarán orientados a mejorar la usabilidad de la plataforma mediante la configuración de distintos *frameworks* especializados en minería de datos.

Bibliografía

- A. Petitet, R. C. Whaley, J. Dongarra, A. Cleary (15 de Diciembre de 2016). *HPL - A Portable Implementation of the High-Performance Linpack Benchmark for Distributed-Memory Computers*. Obtenido de HPL - A Portable Implementation of the High-Performance Linpack Benchmark for Distributed-Memory Computers:
<http://www.netlib.org/benchmark/hpl/>
- Apache. (05 de Octubre de 2016). *Apache Hadoop*. Obtenido de Apache Hadoop:
<http://hadoop.apache.org/>
- Apache. (15 de 01 de 2017). *Apache Pig*. Obtenido de <https://pig.apache.org/>
- Barr, M., & Massa, A. (2006). *Programming Embedded Systems: With C and GNU Development Tools*. O'REILLY.
- Conaway, S. (03 de 07 de 2014). High Performance Data Analysis: Big Data Meets HPC. *High Performance Data Analysis: Big Data Meets HPC*. Recuperado el 14 de 02 de 2017, de <http://www.scientificcomputing.com/blog/2014/03/high-performance-data-analysis-big-data-meets-hpc>
- Cubieboard. (9 de Septiembre de 2016). Obtenido de cubieboard: <http://cubieboard.org/>
- Díaz, G. (31 de Mayo de 2016). Modelos de Programación Paralela. Merida, Venezuela.
- F. Cloutier, M., Paradis, C., & M. Weaver, V. (2014). Design and Analysis of a 32-bit Embedded High-Performance Cluster Optimized for Energy and Performance. *Hardware-Software Co-Design for High Performance Computing (Co-HPC), 2014*. doi:10.1109/Co-HPC.2014.7
- Garcia Nocetti, F. (Junio de 2014). *Cómputo de Alto Rendimiento (HPC) & Big Data*. Obtenido de *Cómputo de Alto Rendimiento (HPC) & Big Data*:
<http://www.inegi.org.mx/eventos/2014/big-data/doc/P-DemetrioGarcia.pdf>
- Google. (15 de 10 de 2016). <https://www.android.com/>. Obtenido de <https://www.android.com/>
- HADOOP. (2014). *HADOOP big data analysis framework*. tutorialspoint.
- J. Greaves, D. (2011). System on Chip Design and Modelling. *System on Chip Design and Modelling*. Cambridge, Inglaterra.
- Lubuntu. (05 de Octubre de 2016). Obtenido de Lubuntu: <http://lubuntu.net/>
- Pérez López, C. (2008). *Minería de Datos Técnicas y Herramientas*. Madrid: Thomson.

- R, R., G, M., & M, A. P. (2013). System on Chip (SoC) for Telecommand System Design. *International Journal of Advanced Research in Computer and Communication Engineering*, 1580-1585.
- Rajovic, N. R.-J.-F. (2016). The Mont-Blanc prototype: An Alternative Approach for HPC Systems.
- Rajovic, N., Rico, A., Puzovic, N., Adeniyi Jones, C., & Ramirez, A. (2014). Making the Case for an ARM-Based HPC System. *ELSEVIER*, 322-334.
- Ramos, J. A. (23 de Abril de 2014). <https://www.adictosaltrabajo.com>. Recuperado el 15 de Enero de 2017, de <https://www.adictosaltrabajo.com>:
<https://www.adictosaltrabajo.com/tutoriales/pig-first-steps/>
- Raspberrypi. (9 de Septiembre de 2016). Obtenido de raspberrypi: <https://www.raspberrypi.org/>
- Solano, H. L., & Álvarez, C. R. (2005). *Estadística descriptiva y distribuciones de probabilidad*. Barranquilla: Ediciones Uninorte.
- Srisuruk, W., & Kaewkasi, C. (s.f.). Low-Power Big Data Cluster. *Low-Power Big Data Cluster*. Suranaree, Tailandia. Obtenido de https://indico.cern.ch/event/311156/contributions/1684547/attachments/595776/819978/aiyara_cluster.pdf
- Valero, M., Rajovic, N., M. Carpenter, P., Gelado, I., Puzovic, N., & Ramirez, A. (2013, Noviembre 17-22). Supercomputing with Commodity CPUs: Are Mobile SoCs Ready for HPC? *2013 SC - International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*, 1-12. Denver,Co: IEEE.
doi:10.1145/2503210.2503281
- Wu, X., Zhu, X., Wu, G.-Q., & Ding, W. (2013). Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering*, 97 - 107. doi:10.1109/TKDE.2013.109