




















Arquitecturas Multiprocesador en HPC: Software de Base, Modelos y Aplicaciones

Armando De Giusti ⁽¹⁾⁽²⁾ , Marcelo Naiouf⁽¹⁾ , Fernando G. Tinetti ⁽¹⁾⁽³⁾ , Horacio Villagarcía ⁽¹⁾⁽³⁾ , Franco Chichizola⁽¹⁾ , Laura De Giusti⁽¹⁾⁽³⁾ , Enzo Rucci⁽¹⁾⁽³⁾ , Adrián Pousa⁽¹⁾ , Victoria Sanz ⁽¹⁾⁽³⁾ , Diego Montezanti ⁽¹⁾ , Diego Encinas ⁽¹⁾ , Ismael Rodríguez⁽¹⁾ , Sebastián Rodríguez Eguren⁽¹⁾ , Leandro Libutti⁽¹⁾ , Manuel Costanzo ⁽¹⁾ , Juan Manuel Paniego⁽¹⁾ , Martín Pi Puig⁽¹⁾ , César Estrebow⁽¹⁾ , Javier Balladini⁽⁴⁾ 

¹Instituto de Investigación en Informática LIDI (III-LIDI),
Facultad de Informática, Universidad Nacional de La Plata – Comisión de Investigaciones Científicas de la
Provincia de Buenos Aires

²CONICET – Consejo Nacional de Investigaciones Científicas y Técnicas

³CICPBA – Comisión de Investigaciones Científicas de la Provincia de Buenos Aires

⁴Universidad Nacional del Comahue

{degiusti,mnaiouf,fernando,hvw,francoch,ldgiusti,erucci,apousa,vsanz,dmontezanti,dencinas,ismael,seguren,
llibutti,mcostanzo,jmpaniego,mpipuig,cesarest}@lidi.info.unlp.edu.ar; javier.balladini@gmail.com

Resumen

El eje de esta línea de I/D lo constituye el estudio de las arquitecturas multiprocesador que integran sistemas distribuidos y paralelos. Incluye como temas centrales:

- Arquitecturas many-core (GPU, procesadores MIC, TPUs), FPGAs, híbridas (diferentes combinaciones de multicores y aceleradores), y asimétricas.
- Desarrollo y evaluación de algoritmos paralelos sobre nuevas arquitecturas y su evaluación de rendimiento computacional y energético.
- Estudio y optimización de código heredado.
- Desarrollo y evaluación de estrategias de resiliencia.
- Modelado y simulación de E/S en HPC.

Palabras clave: *Sistemas Paralelos. Clusters. Arquitecturas asimétricas. GPU, FPGA, TPU. Eficiencia energética. Resiliencia. Portabilidad. Código heredado. E/S paralela.*

Contexto

La línea de investigación que se presenta es parte del proyecto “Computación de Alto Desempeño: Arquitecturas, Algoritmos, Métricas de Rendimiento y Aplicaciones en HPC, Big Data, Robótica, Señales y Tiempo Real.” del III-LIDI y de proyectos específicos apoyados por organismos nacionales e

internacionales. También es parte del proyecto “Procesamiento Eficiente de Grandes Datos mediante Cómputo de Altas Prestaciones, Fog y Edge” financiado por la Facultad de Informática de la UNLP y del proyecto “Unidad Inteligente para Control de Consumo Energético” financiado por la Secretaría de Políticas Universitarias y la UNLP.

En los temas relacionados existen cooperaciones con varias Universidades de Argentina, y se está trabajando con Universidades de América Latina y Europa en proyectos financiados por ERASMUS, CyTED y la OEI (Organización de Estados Iberoamericanos). En particular, se colabora con el proyecto “Computación de Altas Prestaciones Eficiente y Segura para Aplicaciones de Servicios de Salud Inteligentes”, de la Universidad Autónoma de Barcelona.

Por otra parte, se cuenta con financiamiento de diferentes empresas de Argentina, en particular para la formación de recursos humanos en las temáticas de Cloud, Fog y Edge Computing.

Se participa en iniciativas como el Consorcio de I+D+I en Cloud Computing, Big Data y Emergent Topics, conformado por Universidades de Argentina y España.

Asimismo, el III-LIDI forma parte del Sistema Nacional de Cómputo de Alto Desempeño (SNCAD) del Ministerio de Educación, Cultura, Ciencia y Tecnología de la Nación.

Introducción

Una de las áreas de creciente interés lo constituye el cómputo de altas prestaciones, en el cual el rendimiento está relacionado con dos aspectos: por un lado, las arquitecturas de soporte, y por otro, los algoritmos que hacen uso de éstas.

A la aparición de arquitecturas *many-core* (como las GPU o los procesadores MIC), se ha sumado el uso de FPGAs, debido a su potencia de cómputo y rendimiento energético. Su combinación en sistemas HPC da lugar a plataformas híbridas con diferentes características [22]. Lógicamente, esto trae aparejada una revisión de los conceptos del diseño de algoritmos paralelos (incluyendo los mismos lenguajes de programación y el software de base), así como la evaluación de las soluciones que éstos implementan.

Si bien la heterogeneidad del hardware permite maximizar las ganancias de rendimiento y eficiencia energética, implica el uso de lenguajes, librerías y herramientas específicas para cada una de ellas, lo que aumenta el costo de desarrollo y mantenimiento del software [21].

GPUs y Cluster de GPUs

Las GPUs son el tipo de acelerador dominante en la comunidad de HPC hoy en día debido a su alto rendimiento y bajo costo de adquisición. En la actualidad, tanto NVIDIA como AMD trabajan especialmente en mejorar la eficiencia energética de sus placas y disminuir el alto costo de programación.

La combinación de GPUs con otras plataformas paralelas, como clusters y multicores, brindan un vasto conjunto de posibilidades de investigación en arquitecturas híbridas, a partir de diferentes combinaciones como son:

- Máquinas multicore con más de una GPU, en las que se combinan herramientas de programación paralela como OpenMP/CUDA o Pthread/CUDA.
- Cluster de máquinas multicore, cada una con una o más placas de GPU, lo que permite

combinar OpenMP/MPI/CUDA o Pthread/MPI/CUDA.

Los desafíos que se plantean son múltiples, sobre todo en lo referido a distribución de datos y procesos en tales arquitecturas híbridas, a fin de optimizar el rendimiento de las soluciones.

FPGAs

Una FPGA (*Field Programmable Gate Array*) es una clase de acelerador basado en circuitos integrados reconfigurables. La capacidad de adaptar sus instrucciones de acuerdo con la aplicación objetivo le permite incrementar la productividad de un sistema y mejorar el rendimiento energético para ciertos tipos de aplicaciones. Tradicionalmente han sido utilizadas para el procesamiento digital de señales. Sin embargo, en los últimos años, existen tres tendencias claras para extender su uso a otros dominios. En primer lugar, el establecimiento de alianzas estratégicas entre fabricantes de procesadores y de FPGAs para integrar estos dispositivos en arquitecturas híbridas (Intel con Altera; IBM con Xilinx) [10][11]. En segundo lugar, el desarrollo de nuevas herramientas de programación para FPGAs empleando estándares familiares para HPC, con las cuales se espera reducir los tradicionales tiempos y costos de programación [27][31]. Por último, la incorporación de FPGAs a los servicios de Cloud abre nuevas oportunidades para la explotación de esta clase de aceleradores.

TPUs

Las unidades de procesamiento tensorial (TPU) son una clase de Circuitos Integrados de Aplicación Específica (ASIC) desarrolladas por Google con el propósito de acelerar las cargas de trabajo de aprendizaje automático que requieren las aplicaciones desarrolladas en su framework TensorFlow [29]. Su uso provee una alternativa a otras arquitecturas ya conocidas como CPUs, GPUs y MICs. En ese sentido, interesa analizar las tasas de aceleración y eficiencia energética provistas por esta nueva arquitectura, en comparación con el resto.

Eficiencia energética

La mejora de la eficiencia energética es una de las principales preocupaciones en la informática actual, principalmente a partir de las plataformas con gran cantidad de procesadores. Muchos esfuerzos están orientados a tratar la eficiencia energética y a las metodologías para medirla como ejes de I/D, como una métrica de evaluación relevante.

Entre los puntos de interés pueden mencionarse:

- Análisis de metodologías y herramientas para medir y optimizar el consumo energético.
- Estudio de técnicas para reducir el consumo energético en aplicaciones de HPC de acuerdo con las arquitecturas utilizadas.
- Evaluación de eficiencia energética de diferentes algoritmos y plataformas paralelas.
- Optimización de la eficiencia energética. A partir de los valores de energía que brindan los contadores hardware es posible definir estrategias de programación que lleven a reducir el consumo, manteniendo a su vez el rendimiento en valores aceptables [25].

Código heredado

La mayoría de los programas de simulación numérica que se emplean hoy en día fueron desarrolladas cuando las arquitecturas paralelas no existían. Por ello, este conjunto de aplicaciones abre oportunidades para desarrollar técnicas y herramientas que permitan optimizar el código, tanto desde el punto de vista computacional como desde la ingeniería de software [28].

Resiliencia

En la actualidad, lograr sistemas resilientes resulta un verdadero desafío considerando el creciente número de componentes, la cercanía a los límites físicos en las tecnologías de fabricación y la complejidad incremental del software. La corrección de las aplicaciones y la eficiencia en su ejecución se torna más importante en HPC debido a los extensos tiempos de ejecución. En ese sentido, resulta relevante desarrollar estrategias de detección y

recuperación de fallos, especialmente a través de librerías de software [15]. Por otro lado, resulta de interés la aplicación de mecanismos de resiliencia en aplicaciones distribuidas [34].

Simulación y Entrada/Salida paralela

A pesar de los avances tecnológicos, las operaciones de E/S en los centros de supercómputo siguen siendo un cuello de botella para determinadas aplicaciones HPC. El rendimiento de un sistema depende de la carga de trabajo (patrones de E/S de las aplicaciones) y de su configuración (hardware y software) [19]. Contar con herramientas que permitan modelar y predecir el comportamiento de este tipo de aplicaciones en HPC resulta fundamental para mejorar su rendimiento [5].

El análisis y diseño de modelos de simulación basados en la arquitectura de E/S paralela, permite disminuir la complejidad y cubrir las exigencias de las aplicaciones en HPC, al poder identificar y evaluar los factores que influyen en las prestaciones [6]. Además, considerando la relevancia que ha cobrado la simulación de transmisión de enfermedades intra-hospitalarias, resulta de interés utilizar la experiencia desarrollada para el modelado de estos sistemas [13] [32].

Dispositivos de bajo costo con capacidades para cómputo paralelo

En la actualidad se comercializan placas de bajo costo como Raspberry PI [20] u Odroid [33], que poseen múltiples núcleos simples. Asimismo, existen diversos dispositivos móviles con capacidades similares. Es de interés estudiar la manera de explotar el paralelismo en estos dispositivos para mejorar el rendimiento y/o consumo energético de las aplicaciones [1].

Líneas de Investigación, Desarrollo e Innovación

- Arquitecturas many-core (CPU, GPU y TPU) y FPGA. Análisis de este tipo de máquinas y de técnicas para desarrollar código optimizado.

- Arquitecturas híbridas (diferentes combinaciones de clusters, multicores, manycores y FPGAs). Diseño de algoritmos paralelos sobre las mismas. Técnicas de resiliencia.
- Exploración de nuevos lenguajes, modelos y estándares de programación para HPC.
- Consumo energético en las diferentes arquitecturas de alto desempeño, en particular en relación con los algoritmos paralelos y la configuración de la arquitectura. Análisis de metodologías y herramientas de medición. Modelado y estimación del consumo de potencia de arquitecturas HPC.
- Análisis y desarrollo de modelos e implementación de simuladores de la pila de software de E/S en HPC.

Resultados y Objetivos

Investigación experimental por realizar

- Desarrollar y evaluar algoritmos paralelos sobre nuevas arquitecturas paralelas. Analizar rendimiento, eficiencia energética y costo de programación.
- Analizar las capacidades de lenguajes no convencionales para procesamiento paralelo, considerando rendimiento y costo de programación.
- Realizar el desarrollo de nuevos planificadores de tareas para multicores asimétricos sobre diferentes sistemas operativos con el objetivo de maximizar el rendimiento y minimizar el consumo de energía [24][25].
- Analizar proceso de migración de algoritmos entre arquitecturas diferentes, considerando portabilidad, rendimiento y esfuerzo de programación.
- Calibrar y sintonizar el modelado y simulación de E/S en HPC conseguido para distintos escenarios e infraestructuras
- Desarrollar técnicas de tolerancia a fallas que permitan aumentar la resiliencia de sistemas paralelos y distribuidos.

Resultados obtenidos

- Se compararon soluciones paralelas para el método de cifrado AES en diversas arquitecturas de memoria compartida [26].

- Se evaluaron capacidades de paralelismo de lenguajes no convencionales en arquitecturas multicore, como Python [14][23] y Rust [3].
- Se realizó un estudio de migración de códigos CUDA a DPC++ usando el ecosistema oneAPI para el ámbito genómico y se evaluó el esfuerzo de programación requerido, la portabilidad y el overhead en la ejecución [2].
- Se realizaron y analizaron modificaciones al framework TensorFlow para permitir la maleabilidad de hilos [12].
- Se analizó la paralelización de un prototipo de simulador de transmisión de enfermedades intra-hospitalarias [13].
- Se exploró el uso de diferentes microcontroladores para aplicaciones de aprendizaje automático [7] [8].
- Se evaluó el impacto de las comunicaciones en un cluster heterogéneo de placas RPi [30]
- Se desarrolló y validó un modelo estadístico para consumo de potencia en placas RPi de diferentes generaciones [18].
- Se diseñó la herramienta SEDAR y evaluó su desempeño para detección y recuperación de fallos transitorios [15].
- Se desarrollaron técnicas de modelado y simulación de E/S en HPC que permiten predecir cómo los cambios realizados en los diferentes componentes afectan a la funcionalidad y al rendimiento del sistema [5][6].
- Se desarrolló una plataforma distribuida, escalable, híbrida y redundante que permite aprovechar recursos de cómputo basados en arquitecturas ARM a través de un servicio de cloud [9].
- Se analizó el impacto de la E/S en aplicaciones de aprendizaje automático profundo [19].
- Se desarrolló un modelo que permite predecir el consumo energético de un sistema al que, cuando ocurre una falla, se le aplican diferentes estrategias de ahorro de energía [16].

Organización de Eventos

En el año 2022 se han organizado las X Jornadas de Cloud Computing, Big Data & Emerging Topics (JCC-BD&ET 2022) en

Argentina, con participación de especialistas académicos del país y el exterior, además de empresas con experiencia en Cloud Computing [17] [4]. En junio de 2023 se tiene prevista la organización de las XI JCC-BD&ET.

Formación de Recursos Humanos

Dentro de la temática de la línea de I/D el último año se concluyeron 2 Tesis Doctorales. Al mismo tiempo se encuentran en curso 3 tesis de Doctorado en Ciencias Informáticas y 3 tesis de Maestría.

Además, se participa en el dictado de las carreras de Doctorado en Ciencias Informáticas, y Magíster y Especialización en Cómputo de Altas Prestaciones de la Facultad de Informática de la UNLP (acreditadas por la CONEAU con categoría A, B y A, respectivamente), por lo que potencialmente pueden generarse nuevas Tesis de Doctorado y Maestría, además de Trabajos Finales de Especialización.

Existe cooperación con grupos de otras Universidades del país y del exterior, y hay tesis de diferentes Universidades realizando su Tesis con el equipo del proyecto.

Respecto a las carreras de grado, se dictan por parte de integrantes de la línea de investigación tres materias directamente relacionadas con los temas de ésta: “Taller de Programación sobre GPUs”, “Cloud Computing y Cloud Robotics” y “Conceptos y Aplicaciones en Big Data”.

Referencias

[1] P. S. Rodríguez Eguren, F. Chichizola, and E. Rucci, “Análisis del uso de un cluster de Raspberry Pi para cómputo de alto rendimiento”. Actas del XXIV Congreso Argentino de Ciencias de la Computación (CACIC 2018), ISBN: 978-950-658-472-6, págs. 134-144, 2018.

[2] Manuel Costanzo, Enzo Rucci, Carlos García-Sánchez, Marcelo Naiouf, Manuel Prieto-Matías. “Migrating CUDA to oneAPI: A Smith-Waterman Case Study.” International Work-Conference on Bioinformatics and Biomedical Engineering 2022, ISBN 978-3-031-07802-6, págs. 103-116, Cham, doi: 10.1007/978-3-031-07802-6_9, junio de 2022.

[3] M. Costanzo, E. Rucci, M. Naiouf, and A. D. Giusti, “Performance vs Programming Effort between Rust and

C on Multicore Architectures: Case Study in N-Body”, Proceedings of 2021 XLVII Latin American Computing Conference (CLEI), ISBN: 978-1-66549-503-5, págs. 1-10, doi. 10.1109/CLEI53233.2021.9640225, 2021.

[4] Marcelo Naiouf, Armando Eduardo De Giusti, Franco Chichizola, Laura Cristina De Giusti, Enzo Rucci (editores). “Short papers of the 10th Conference on Cloud Computing, Big Data & Emerging Topics”. ISBN 978-950-34-2126-0, Facultad de Informática (UNLP), 106 páginas, doi: 10.35537/10915/139373, julio de 2022.

[5] D. Encinas, M. Naiouf, A. De Giusti, S. Méndez, D. Rexachs del Rosario, and E. Luque, “On the Calibration, Verification and Validation of an Agent-Based Model of the HPC Input/Output System” Proceedings of the The Eleventh International Conference on Advances in System Simulation (SIMUL 2019), ISBN: 978-1-61208-756-6, págs. 14-21, 2019.

[6] D. Encinas, S. Mendez, M. Naiouf, A. De Giusti, D. Rexachs del Rosario, and E. Luque, “An Agent-Based Model for Analyzing the HPC Input/Output System” International journal on advances in systems and measurements vol. 13, num. 3 & 4, págs. 192-202, 2020.

[7] César A. Estrebou, Martín Fleming, Marcos D. Saavedra, Federico Adra, Armando E. De Giusti. “Lightweight Convolutional Neural Networks Framework for Really Small TinyML Devices.”. Third International Conference on Smart technologies, Systems and Applications (SmartTech-IC), ISBN 978-3-030-99170-8, págs. 3-16, Cham, doi: 10.1007/978-3-030-99170-8_1, marzo de 2022.

[8] César Armando Estrebou, Marcos David Saavedra, Federico Adra, Martín Fleming. “TinyML for Small Microcontrollers.” X Jornadas de Cloud Computing, Big Data & Emerging Topics (La Plata, 2022), ISBN 978-950-34-2126-0, págs. 42-46, julio de 2022.

[9] David Petrocelli, Armando De Giusti, Marcelo Naiouf. “Collaborative, Distributed, Scalable and Low-Cost Platform Based on Microservices, Containers, Mobile Devices and Cloud Services to Solve Compute-Intensive Tasks.” European Conference on Parallel Processing 2021, ISBN 978-3-031-06156-1, págs. 545-548, Cham, doi: 10.1007/978-3-031-06156-1_47, junio de 2022.

[10] IBM. “IBM and Xilinx Announce Strategic Collaboration to Accelerate Data Center Applications”. Disponible en <https://www-03.ibm.com/press/us/en/pressrelease/48074.wss>

[11] Intel. “Intel Acquisition of Altera”. Disponible en <http://intelacquiresaltera.transactionannouncement.com>

[12] L. Libutti, L. De Giusti, and M. Naiouf. “PLANIFICACIÓN Y ACELERACIÓN DE ALGORITMOS DE MACHINE LEARNING”. Investigación joven (ISSN 2314-3991), vol. 7, num. 2, págs. 552-553, abril de 2021.

[13] Martin Paradiso, Lucas Maccallini, Agustina Vericat, Fernando Romero, Diego Encinas. An

- Approach to Parallelization of Respiratory Disease Spread Simulations in Emergency Rooms. 2022 International Conference on Computational Science and Computational Intelligence (CSCI'22). En prensa.
- [14] Milla, A., Rucci, E. (2022). Performance Comparison of Python Translators for a Multi-threaded CPU-Bound Application. In: Pesado, P., Gil, G. (eds) Computer Science – CACIC 2021. CACIC 2021. Communications in Computer and Information Science, vol 1584. Springer, Cham. https://doi.org/10.1007/978-3-031-05903-2_2
- [15] D. Montezanti, E. Rucci, A. D. De Giusti, M. Naiouf, D. Rexachs, and E. Luque, “Soft errors detection and automatic recovery based on replication combined with different levels of checkpointing”. *Future generation computer systems* (ISSN 0167-739X), vol. 113, págs. 240-254, doi. <https://doi.org/10.1016/j.future.2020.07.003>, 2020.
- [16] M. Morán, J. Balladini, D. Rexachs, and E. Rucci. “Some Issues to Consider in the Management of Energy Consumption in HPC Systems with Fault Tolerance”. Pertenece al libro: *Short papers of the 10th Conference on Cloud Computing, Big Data & Emerging Topics*, ISBN: 978-950-34-2126-0. Páginas: 17-22, 2022.
- [17] Enzo Rucci, Marcelo Naiouf, Franco Chichizola, Laura De Giusti, Armando De Giusti. “Cloud Computing, Big Data & Emerging Topics. 10th Conference, JCC-BD&ET 2022, La Plata, Argentina, June 28–30, 2022, Proceedings”. ISBN 978-3-031-14598-8. <https://doi.org/10.1007/978-3-031-14599-5>
- [18] J. M. Paniego, L. Libutti, M. P. Puig, F. Chichizola, L. De Giusti, M. Naiouf, and A. De Giusti, “Unified Power Modeling Design for Various Raspberry Pi Generations Analyzing Different Statistical Methods”. En: *Computer Science – CACIC 2019. communications in Computer and Information Science.*, ISBN: 978-3-030-48325-8, Springer International Publishing, págs. 53-65, 2020.
- [19] E. Párraga, B. León, R. Bond, D. Encinas, A. Bezerra, S. Mendez, D. Rexachs, and E. Luque. “Analyzing the I/O Patterns of Deep Learning Applications”. *Proceedings from the 9th Conference on Cloud Computing, Big Data & Emerging Topics (JCC-BD&ET 2021)*, vol. 1444, págs. 3-16, doi. 10.1007/978-3-030-84825-5_1, 2021.
- [20] Raspberry PI. <https://www.raspberrypi.org/>
- [21] Manuel Costanzo, Enzo Rucci, Carlos García-Sánchez, Marcelo Naiouf. “Early Experiences Migrating CUDA codes to oneAPI.” *IX Jornadas de Cloud Computing, Big Data & Emerging Topics (JCC-BD&ET 2021)*, ISBN 978-950-34-2016-4, págs. 14-18, julio de 2021.
- [22] Rucci, Enzo: “Evaluación de rendimiento y eficiencia energética en sistemas heterogéneos para bioinformática”. Tesis de Doctorado en Ciencias Informáticas (Facultad de Informática – UNLP). 2016.
- [23] Andrés Milla, Enzo Rucci. “Comparación de Rendimiento y Esfuerzo de Programación entre Numba y Cython para una Aplicación Multi-hilada de Alto Rendimiento”. Congreso Argentino de Ciencias de la Computación (CACIC 2022). En prensa.
- [24] Juan Carlos Saez, Adrian Pousa, Daniel Chaver, Fernando Castro, Manuel Prieto Matias: “ACFS: A Completely Fair Scheduler for Asymmetric Single-ISA Multicore Systems”. In: *ACM SAC 2015 (The 30TH ACM/SIGAPP Symposium on applied computing)*. 2015.
- [25] Saez, J.C., Pousa, A., Rodríguez-Rodríguez, R., Castro, F., Prieto-Matias, M. “PMCTrack: Delivering performance monitoring counter support to the OS scheduler”. *The computer journal* Volume 60, Issue 1 January 2017.
- [26] Victoria Sanz, Adrián Pousa, Marcelo Naiouf, Armando De Giusti. “Performance Analysis of AES on CPU-GPU Heterogeneous Systems”. En: *Cloud Computing, Big Data & Emerging Topics, 10th Conference, JCC-BD&ET 2022*. ISBN 978-3-031-14599-5, Springer International Publishing, pág. 31-42, agosto de 2022.
- [27] Sean Settle: “High-performance Dynamic Programming on FPGAs with OpenCL”. In: *IEEE High Performance Extreme Computing Conference*. 2013.
- [28] F. G. Tinetti, M. J. Perez, A. Fraidenraich, and A. E. Altenberg, “Legacy code and parallel computing: updating and parallelizing a numerical model”. *The journal of supercomputing* (ISSN 1573-0484), doi. 10.1007/s11227-020-03172-7, 2020.
- [29] Google Inc. “Cloud Tensor Processing Unit (TPU)” Disponible en <https://cloud.google.com/tpu/docs/tpus?hl=es-419>
- [30] O. C. Valderrama Riveros and F. G. Tinetti, “MPI Communication Performance in a Heterogeneous Environment with Raspberry Pi”, *Advances in Parallel & Distributed Processing, and Applications. Transactions on Computational Science and Computational Intelligence.*, ISBN: 978-3-030-69984-0, págs. 451-460, doi. 10.1007/978-3-030-69984-0_33, 2021.
- [31] Xilinx Inc. “SDAccel Development Environment”. [Online]. Disponible en <http://www.xilinx.com/products/design-tools/software-zone/sdaccel.html>
- [32] Maccallini, L., Encinas, D. O., & Romero, F. (2021). An Approach to the Modeling and Simulation of Intra-Hospital Diseases. *Journal of Computer Science and Technology*, 21(2), e14. <https://doi.org/10.24215/16666038.21.e14>
- [33] Odroid <http://www.hardkernel.com> Accedido 21 de marzo de 2016.
- [34] Suarez, Sergio Leonel. “Análisis de patrones de resiliencia en una arquitectura basada en microsistemas”. Tesis de Licenciatura en Sistemas. Universidad Nacional de La Plata. 2022. <http://sedici.unlp.edu.ar/handle/10915/149187>.