

Software y aplicaciones en Computación de Altas Prestaciones para el contexto de la UNdeC

Fernando Emmanuel FRATI, Jose TEXIER, Paula Cecilia RIVERA, Jonathan ALVAREZ, Fernanda CARMONA, Patricia FIGUEROLA, Francisco FRATI, Sebastián GUIDET, Roberto MILLON, Raul MORALEJO, Matías PEREZ, Emmanuel PORTUGAL, Donna RATTALINO, Alberto RIBA, Daniel ROBINS, Mara ROVERO, Javier RUITTI, Jorge TEJADA, Jusmeidy ZAMBRANO

Universidad Nacional de Chilecito

9 de julio 22, Chilecito, La Rioja, Argentina

{fefrati, jtexier, privera, jalvarez, fbcarmona, pfiguerola, ffrati, sguidet, rmillon, rmoralejo, mperez, eportugal, drattalino, ariba, drobins, mrovero, jruiitti, jtejada, jzambrano}@undec.edu.ar

RESUMEN

En la actualidad, para abordar problemas de mayor tamaño y complejidad estudios de ciencia básica y aplicada utilizan Computación de Altas Prestaciones (HPC - High Performance Computing). El HPC permite mejorar la capacidad, velocidad y precisión en el procesamiento de datos. Con el proyecto que da origen a este trabajo se propone abordar seis estudios desde la perspectiva del HPC, para explorar los aspectos centrales del paralelismo aplicado desde las Ciencias de la Computación en otras disciplinas.

Algunas de los estudios que se abordan ya vienen desarrollándose en la Universidad Nacional de Chilecito, otros se inician a partir de cooperaciones con otras instituciones o para formalizar trabajos finales de postgrado. En todos los casos, el HPC será abordado a través de un proceso metodológico organizado para:

- Consolidar una infraestructura de experimentación, desarrollo y producción de soluciones a problemas de HPC
- Desarrollar las capacidades científico-tecnológicas del equipo
- Fomentar la vinculación y transferencia con los sectores académico, social y productivo

Cada problema abordado reúne entre sus integrantes, investigadores especialistas en la disciplina del estudio, investigadores de Ciencias de la Computación y estudiantes en sus últimos años de formación de grado. Se

espera consolidar a corto plazo un grupo de investigación, desarrollo y transferencia que generará oportunidades de formación de recursos humanos, proveerá de servicios a la comunidad en el área de estudio y potenciará los vínculos de cooperación con otras instituciones.

Palabras clave: HPC, cómputo paralelo, aplicaciones, interdisciplinariedad, UNdeC.

CONTEXTO

La línea de investigación presentada es parte del proyecto “Software y aplicaciones en Computación de Altas Prestaciones” aprobado en el año 2018 por la Secretaría de Ciencia y Tecnología de la UNdeC para ser ejecutado desde marzo 2019 con una duración de 18 meses, convocatoria a proyectos de Investigación y Desarrollo. Además, en 2018 la UNdeC destinó fondos de PROMINF para la adquisición de equipamiento para el “laboratorio de sistemas paralelos”. También el “Plan de mejoramiento de la función de I+D+i” (Ministerio de Ciencia, Tecnología e Innovación Productiva de la Nación) prevé financiamiento para la adquisición de equipamiento durante 2019 a fin de desarrollar las capacidades en HPC de la UNdeC. Estas iniciativas permitirán consolidar de una infraestructura de experimentación, desarrollo y producción de soluciones a problemas de HPC.

1. INTRODUCCIÓN

La informática tiene su origen en la necesidad de los distintos sectores de la sociedad de conseguir mayor velocidad, confiabilidad y precisión para resolver sus problemas. Sin embargo, la capacidad de solución a un problema dado encuentra su límite en los tiempos requeridos por sus algoritmos. Superar ese límite requiere que el problema sea abordado mediante cómputo paralelo. Normalmente, esto implica estudiar tres aspectos clave: hardware, aplicaciones y software.

Durante décadas, la industria respondió a la creciente demanda de mayor poder computacional incrementando exponencialmente el rendimiento de los procesadores. Esto fue posible gracias a tres tecnologías clave (escalado de transistores, diseño de la microarquitectura y jerarquías de memoria)[1]. Sin embargo, esta forma de obtener mayor poder de cómputo encontró barreras físicas (disipación de calor y eficiencia energía fundamentalmente), limitando el rendimiento de los microprocesadores y los sistemas en general [2]. Desde el año 2005, el escalado tecnológico se ha venido aprovechando para aumentar el número de cores dentro del chip, dando lugar a una importante variedad de arquitecturas (multicores, commodity clusters, GPGPU y Cloud), que permiten alcanzar enormes picos de rendimiento [3], [4].

No obstante, reducir los tiempos de procesamiento y obtener la mayor eficiencia de ese hardware requiere el diseño y desarrollo de algoritmos paralelos [5], [6]. Transformar un algoritmo secuencial en uno paralelo no es trivial. Un multicore está compuesto por dos o más unidades de procesamiento donde cada una puede ejecutar un flujo secuencial de instrucciones. Si estos flujos secuenciales de instrucciones corresponden a tareas de un mismo programa concurrente, entonces se trata de una ejecución paralela. En general estos procesos necesitan algún mecanismo para comunicar resultados parciales entre sí.

Dependiendo de la arquitectura de cómputo, se consigue a través del uso de variables compartidas o del paso de mensajes entre procesos. Una transformación ‘implícita’ o transparente es deseable, pero el costo es una pérdida importante de rendimiento [7]. Por otro lado, debido a que las arquitecturas de cómputo cambian constantemente y a fin de reducir el esfuerzo necesario para migrar de una plataforma a otra, es deseable poder expresar los programas de manera independiente a la máquina. Por ello, el programador recurre a librerías estándares para expresar explícitamente el paralelismo: OpenMP, Pthreads, CUDA, OpenCL, MPI [8].

La secuencia cronológica en que se ejecutan las comunicaciones entre los procesos se conoce como historia del programa. Las suposiciones de orden de ejecución entre instrucciones heredadas del modelo de programación secuencial ya no son válidas, obligando al programador a utilizar algún mecanismo de sincronización para garantizar estados consistentes del programa. En este contexto, la correctitud de los algoritmos es más difícil de garantizar que en la computación serial. Frecuentemente el programador se equivoca al sincronizar los procesos, dando lugar a nuevos errores de programación: deadlocks, condiciones de carrera, violaciones de orden, violaciones de atomicidad simple y violaciones de atomicidad multivariable. Los métodos tradicionales de depuración de programas y las técnicas de sintonización de rendimiento existentes en la programación secuencial no se trasladan directamente a la programación paralela, requiriendo el uso de herramientas específicas [9]. Al diseñar una solución serial, existe un modelo teórico que permite estimar el desempeño de los programas antes de escribirlos. La evaluación del sistema paralelo (entendiendo por sistema al conjunto de software y hardware) se realiza a través de distintas métricas: tiempo de ejecución, speedup, eficiencia y overhead [10].

Las soluciones paralelas están tan estrechamente vinculadas con el hardware subyacente que dificultan enormemente

conseguir portabilidad de rendimiento: una solución pensada para una máquina paralela en memoria distribuida tendrá diferente eficiencia en una de memoria compartida, o incluso en otra máquina de memoria distribuida [11], [12].

Existen muchos aspectos que requieren ser tomados en cuenta al diseñar la solución paralela: tamaño del problema, división de datos o tareas, balance de carga, requerimientos de memoria, precisión de los cálculos, comunicaciones y sincronización entre procesos, errores de concurrencia, detección y tolerancia a fallos entre los más relevantes. La complejidad de los problemas requiere habilidades especiales de los desarrolladores: dominio de múltiples paradigmas de programación y frecuentemente múltiples lenguajes, conocimientos de redes y comprensión de la concurrencia y sus consecuencias. Por todo esto, se considera de gran interés el estudio de estos temas para el desarrollo de capacidades científico- tecnológicas en la UNdeC que favorezcan el trabajo interdisciplinario en la institución.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

Se propone abordar las siguientes líneas de I+D desde la perspectiva del HPC como eje central:

- Análisis de la diversidad molecular de microorganismos del suelo [13]–[18]. Estudio e implementación de algoritmos que contribuyan a reducir los tiempos de procesamiento y aumentar la capacidad de análisis referidos a este campo de la bioinformática, a fin de profundizar en el estudio de la diversidad molecular de microorganismos del suelo asociados a cultivos regionales.
- Detección y depuración de errores de concurrencia en programas paralelos [9], [19]–[22]. Estudio de técnicas y estrategias para implementar mecanismos de detección de errores de concurrencia robustos y confiables que contribuyan al

diseño de programas paralelos libres de errores.

- Identificación biométrica masiva mediante venas del dedo usando redes de aprendizaje extremo (ELM) [23]–[27]. Estudio de técnicas de computación paralela para mejorar la eficiencia de identificación biométrica masiva basada en venas de dedo, para la aceleración del preprocesamiento y extracción de características biométricas, y el diseño de algoritmos de ELM mejorados que manejen eficientemente lotes de datos de gran tamaño.
- Servicios basados en lingüística computacional para análisis de texto [28]–[34]. Estudio sobre modelos computacionales que reproduzcan aspectos del lenguaje humano, con el fin de realizar análisis lingüísticos como servicios para el Centro de Escritura en la UNdeC.
- Documentos inteligentes a través del Blockchain [35]–[38]. Estudio de la tecnología blockchain para garantizar la integridad de documentos universitarios.
- Nodo de información meteorológica [39], [40]. Estudio, diseño e implementación de algoritmos para reducir los tiempos de procesamiento, aumentar la capacidad de análisis y favorecer la escalabilidad de aplicaciones de análisis y proyección de datos climáticos.

3. RESULTADOS ESPERADOS

Se consolidará un grupo de investigación, desarrollo y transferencia a la comunidad con capacidad para abordar problemas de computación de altas prestaciones, que generará oportunidades de formación de recursos humanos, proveerá de servicios a la comunidad en el área de estudio y potenciará los vínculos de cooperación con otras instituciones. Con respecto a las líneas propuestas, se espera:

- Establecer las oportunidades de mejora a través del paralelismo de desempeño, tiempo de respuesta o capacidad de procesamiento de cada subproyecto abordado.

- Determinar la configuración de recursos de cómputo que mejor se ajuste a cada subproyecto.
- Conocer las propuestas de la comunidad académica y científica para resolver el problema.
- Definir los criterios de diseño de software que conduzcan a una mejora de la aplicación.
- Obtener productos de software que puedan ser ejecutados en la plataforma de HPC como servicios para las áreas disciplinares vinculadas.
- Mejorar la visibilidad del grupo de I+D a nivel regional, nacional e internacional.
- Aumentar la cantidad de trabajos de finalización de grado, tesis de postgrado y actividades específicas en temas de HPC en la UNdeC.
- Determinar el éxito del proceso de I+D.

4. FORMACIÓN DE RECURSOS HUMANOS

Cinco miembros del equipo poseen formación de postgrado a nivel de doctorado, uno de ellos es especialista en Cómputo de Altas Prestaciones. Cinco miembros se encuentran en su etapa final para obtener el grado de maestría en Informática, tres de los cuales desarrollan como tesis temas abordados por esta propuesta. Dos de estas tesis de maestría están siendo codirigidas por docentes de la Universidad Católica de Maule (Chile). Cada línea I+D propuesta integra al menos un docente investigador experto en el campo de cada estudio específico. Todos los temas propuestos se trabajan con estudiantes de grado de las carreras Ingeniería en Sistemas y Licenciatura en Sistemas de la UNdeC (ambas acreditadas por CONEAU). Los docentes forman parte de los equipos de diversas asignaturas de estas carreras, entre las que se encuentran programación, arquitecturas de computadoras y arquitecturas paralelas. Nueve docentes se encuentran categorizados en el programa de incentivos.

5. BIBLIOGRAFÍA

- [1] S. Borkar y A. A. Chien, «The Future of Microprocessors», *Commun ACM*, vol. 54, n.º 5, pp. 67–77, may 2011.
- [2] K. Ahmed y K. Schuegraf, «Transistor wars», *IEEE Spectr.*, vol. 48, n.º 11, pp. 50–66, nov. 2011.
- [3] V. V. Kindratenko *et al.*, «GPU clusters for high-performance computing», en *2009 IEEE International Conference on Cluster Computing and Workshops*, 2009, pp. 1–8.
- [4] J. Jeffers, J. Reinders, y A. Sodani, *Intel Xeon Phi Processor High Performance Programming: Knights Landing Edition*. Morgan Kaufmann, 2016.
- [5] G. R. Andrews, *Foundations of Multithreaded, Parallel, and Distributed Programming*. Addison Wesley, 2000.
- [6] A. Grama, A. Gupta, G. Karypis, y V. Kumar, *Introduction to Parallel Computing - Second Edition*. Pearson Education and Addison Wesley, 2003.
- [7] G. Hager y G. Wellein, *Introduction to high performance computing for scientists and engineers*. Boca Raton, FL: CRC Press, 2011.
- [8] J. Dongarra *et al.*, *Sourcebook of parallel computing*, vol. 3003. Morgan Kaufmann Publishers San Francisco, 2003.
- [9] F. E. Frati, «Software para arquitecturas basadas en procesadores de múltiples núcleos», Tesis, Facultad de Informática, 2015.
- [10] X.-H. Sun, «Scalability versus execution time in scalable systems», *J. Parallel Distrib. Comput.*, vol. 62, n.º 2, pp. 173–192, 2002.
- [11] C. Leopold, *Parallel and Distributed Computing: A survey of Models, Paradigms and approaches*. John Wiley & Sons, Inc., 2001.
- [12] S. Ghosh, *Distributed Systems: An Algorithmic Approach*, 1.ª ed. University of Iowa, Iowa City, USA: Chapman and Hall/CRC, 2006.
- [13] M. S. De, M. Prager, R. E. Naranjo, y O. E. Sanclemente, «El suelo, su metabolismo, ciclaje de nutrientes y prácticas agroecológicas», *Agroecología*, vol. 7, n.º 1, pp. 19–34, 2012.
- [14] R. F. Cerrato y A. Alarcón, «La microbiología del suelo en la agricultura sostenible», *Cienc. -Sum*, vol. 8, n.º 2, pp. 175–183, 2001.
- [15] J. P. Hulsenbeck y F. Ronquist, «MrBayes: Bayesian inference of phylogeny», *Bioinformatics*, vol. 17, pp. 754–755, 2001.
- [16] A. Stamatakis, «RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies», *Bioinformatics*, vol. 30, n.º 9, pp. 1312–1313, may 2014.
- [17] M. A. Suchard, P. Lemey, G. Baele, D. L. Ayres, A. J. Drummond, y A. Rambaut,

- «Bayesian phylogenetic and phylodynamic data integration using BEAST 1.10», *Virus Evol.*, vol. 4, n.º 1, ene. 2018.
- [18] D. L. Swofford, «PAUP*: phylogenetic analysis using parsimony (*and Other Methods), version 4», *Sinauer Assoc. Sunderland Mass.*, 2003.
- [19] S. V. Adve y H.-J. Boehm, «Memory models: a case for rethinking parallel languages and hardware», *Commun. ACM*, vol. 53, n.º 8, pp. 90–101, 2010.
- [20] M. Ronsse y K. De Bosschere, «RecPlay: a fully integrated practical record/replay system», *ACM Trans Comput Syst*, vol. 17, n.º 2, pp. 133–152, 1999.
- [21] S. Lu, S. Park, E. Seo, y Y. Zhou, «Learning from mistakes: a comprehensive study on real world concurrency bug characteristics», *SIGARCH Comput Arch. News*, vol. 36, n.º 1, pp. 329–339, 2008.
- [22] S. Lu, J. Tucek, F. Qin, y Y. Zhou, «AVIO: detecting atomicity violations via access interleaving invariants», *SIGPLAN Not*, vol. 41, n.º 11, pp. 37–48, 2006.
- [23] K. Wang, H. Ma, O. P. Popoola, y J. Liu, «Finger vein recognition», en *Biometrics, InTech*, 2011.
- [24] D. Ezhilmaran y P. R. B. Joseph, «A STUDY OF FEATURE EXTRACTION TECHNIQUES AND IMAGE ENHANCEMENT ALGORITHMS FOR FINGER VEIN RECOGNITION», p. 8, 2015.
- [25] G.-B. Huang, Q.-Y. Zhu, y C.-K. Siew, «Extreme learning machine: Theory and applications», *Neurocomputing*, vol. 70, n.º 1-3, pp. 489-501, dic. 2006.
- [26] G. Huang, G.-B. Huang, S. Song, y K. You, «Trends in extreme learning machines: A review», *Neural Netw.*, vol. 61, pp. 32-48, ene. 2015.
- [27] A. Akusok, K. Björk, Y. Miche, y A. Lendasse, «High-Performance Extreme Learning Machines: A Complete Toolbox for Big Data Applications», *IEEE Access*, vol. 3, pp. 1011-1025, 2015.
- [28] AMPLN, «Asociación Mexicana para el Procesamiento del Lenguaje Natural Main/Home Page». [En línea]. Disponible en: <https://www.ampln.org/>. [Accedido: 31-ago-2018].
- [29] M. Vallez y R. Pedraza, «El Procesamiento del Lenguaje Natural en la Recuperación de Información Textual y áreas afines», *Hipertext Net*, 2007.
- [30] J. Texier, F. E. Frati, F. B. Carmona, A. E. Riba, M. Pérez, y J. Zambrano, «La gestión de la información en abierto, vehículo importante para maximizar la visibilidad web», presentado en XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016, Entre Ríos, Argentina), 2016.
- [31] P. Gamallo Otero, J. C. Pichel Campos, M. García González, J. M. Abuín Mosquera, y T. Fernández Pena, «Análisis morfosintáctico y clasificación de entidades nombradas en un entorno Big Data», 2014.
- [32] S. W. D. Chien, C. P. Sishtla, S. Markidis, J. Zhang, I. B. Peng, y E. Laure, «An Evaluation of the TensorFlow Programming Model for Solving Traditional HPC Problems», en *International Conference on Exascale Applications and Software*, 2018, p. 34.
- [33] H. Guan, X. Shen, y H. Krim, «Egeria: A Framework for Automatic Synthesis of HPC Advising Tools Through Multi-layered Natural Language Processing», en *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, New York, NY, USA, 2017, pp. 10:1–10:14.
- [34] P. Suber, *Ensuring open access for publicly funded research*. British Medical Journal Publishing Group, 2012.
- [35] M. U. Wasim, A. A. Ibrahim, P. Bouvry, y T. Limba, «Law as a service (LaaS): Enabling legal protection over a blockchain network», en *Smart Cities: Improving Quality of Life Using ICT & IoT (HONET-ICT), 2017 14th International Conference on*, 2017, pp. 110–114.
- [36] A. Preukschat, *Blockchain: la revolución industrial de internet*. Gestión 2000, 2017.
- [37] L.-Y. Yeh, P. J. Lu, y J.-W. Hu, «NCHC blockchain construction platform (NBCLP): rapidly constructing blockchain nodes around Taiwan», en *Digital Libraries (JCDL), 2017 ACM/IEEE Joint Conference on*, 2017, pp. 1–2.
- [38] H. Dai *et al.*, «TrialChain: A Blockchain-Based Platform to Validate Data Integrity in Large, Biomedical Research Studies», *ArXiv Prepr. ArXiv180703662*, 2018.
- [39] A. Botta, W. De Donato, V. Persico, y A. Pescapé, «On the integration of cloud computing and internet of things», en *Future internet of things and cloud (FiCloud), 2014 international conference on*, 2014, pp. 23–30.
- [40] P. Yue, H. Zhou, J. Gong, y L. Hu, «Geoprocessing in cloud computing platforms—a comparative analysis», *Int. J. Digit. Earth*, vol. 6, n.º 4, pp. 404–425, 2013.