

Solución de Grandes Problemas aplicando HPC Multi-tecnología

Mercedes Barrionuevo ⁽¹⁾, Julian Escalante ⁽²⁾, Mariela Lopresti ⁽¹⁾, Maximiliano Lucero ⁽¹⁾,
Natalia Miranda ⁽¹⁾, M. Antonia Murazzo ⁽³⁾ y Fabiana Piccoli ^(1 y 2).

⁽¹⁾LIDIC- Universidad Nacional de San Luis, San Luis

⁽²⁾Universidad Autónoma de Entre Ríos, Concepción del Uruguay

⁽³⁾ Universidad Nacional de San Juan, San Juan

Argentina

{mbarrio, omlopres, mlucero, ncmiran, mpiccoli}@unsl.edu.ar

{escalantejulian18,maritemurazzo}@gmail.com

RESUMEN

Los problemas a resolver hoy en día mediante soluciones computacionales demandan muchos recursos, esto puede obedecer a dos factores: el tamaño del problema en relación a los datos a tratar, y la complejidad del problema. Puede ser uno o ambos factores, en cualquier caso, siempre los resultados se esperan en un tiempo razonable. Una de las soluciones propuestas es pensar en sistemas heterogéneos: una computadora formada por procesadores many y multicores con software capaz de tomar ventaja de cada una de las componentes subyacentes.

En este trabajo se exponen dos líneas de trabajo orientadas a distintos tipos de problemas, en las cuales se propone desarrollar técnicas de Computación de Alto Desempeño para el escenario descripto.

Palabras clave: Computación de Alto Desempeño. Modelos Híbridos. Problemas Complejos. Big Data.

CONTEXTO

Esta propuesta de trabajo se lleva a cabo dentro de los proyectos de investigación: “Tecnologías Avanzadas aplicadas al Procesamiento de Datos Masivos” (LIDIC, UNSL) y “Cómputo de Altas Prestaciones aplicado a la Solución de Grandes Problemas” (UADER); y del proyecto binacional CAPG-BA

66/13 entre la Universidad Nacional de San Luis y la Universidad de Pernambuco, Recife, Brasil. Además de docentes-investigadores de otras universidades, UNSJ.

1. INTRODUCCIÓN

El crecimiento de los problemas a resolver computacionalmente, respecto a la cantidad de datos y complejidad de las soluciones, hizo intensificar la búsqueda de nuevas opciones. Desde el punto de vista del hardware, los avances se enfocaron en el desarrollo de computadoras de propósito general con múltiples núcleos, dando origen a dos líneas de desarrollo: las arquitecturas multi-core (multi-núcleos) y las arquitecturas many-cores (muchos-núcleos o muchos-cores) [Vaj11]. En los multi-core, los avances se centraron en el desarrollo de mejoras para acelerar, generalmente, las aplicaciones secuenciales incorporando varios núcleos de procesamiento. Surgen así computadoras con al menos 2 procesadores por unidad central (multi). En el caso de los many-cores, los desarrollos se centraron en optimizar el desempeño de aplicaciones paralelas, en este tipo de arquitectura se encuentran las tarjetas gráficas o GPUs [Aam18].

Los multi-core y many-core tienen características propias, una de sus principales diferencias, como ya se enunció, es la de proveer mejor desempeño: soluciones secuenciales vs.

solución paralela. Para lograrlo, las primeras brindan lógica de control más compleja o memorias caché más rápidas y grandes por ejemplo. En el caso de las otras, la idea es optimizar el throughput de muchos threads ejecutando en paralelo, por ello si alguno de ellos está esperando por la finalización de una operación, otro es asignado al procesador para que no permanezca ocioso. En este tipo de arquitecturas, las áreas dedicadas a control y memoria caché son generalmente pequeñas [Hon10].

Ambos tipos de arquitecturas, si bien difieren en varios aspectos, ofrecen a todos los cores acceso a una única memoria compartida. Como la cantidad de memoria disponible es limitada, una alternativa son las arquitecturas con memoria distribuida, quienes permiten incrementar el espacio de almacenamiento incorporando el costo de las comunicaciones a través de la red [Bas16]. Las arquitecturas con memoria distribuida ponen a disposición un gran número de máquinas y sus memorias, eliminando el problema de la limitación de la RAM. Los clusters y cloud computing son ejemplos de ellas[Kau14, Mar17].

Una tercera alternativa son los sistemas híbridos, los cuales permiten combinar las características de los sistemas de memoria compartida y de memoria distribuida, incrementando la capacidad y poder de cómputo de la arquitectura resultante. Esto posibilita aprovechar las ventajas ofrecidas por ambas, múltiples procesos y threads con distintas administraciones de memoria coexistiendo en el sistema.

Desde el punto de vista del software, también existen diferentes clasificaciones, ellas dependen del objeto a considerar, por ejemplo si se considera qué se divide, los modelos de programación estándares, o paradigmas, son: paralelismo de datos y paralelismo de tareas [Pac11]; en cambio si se tiene en cuenta la arquitectura, es posible clasificar a los modelos

como de memoria compartida y de memoria distribuida o pasaje de mensajes. Para cada uno de ellos, existen diferentes herramientas, las cuales permiten llevarlos a cabo en las distintas arquitecturas, entre ellos OpenMP[Op19], MPI[Gro14] y CUDA[NVi19].

Teniendo en cuenta la demanda de cómputo de los problemas actuales y la posibilidad que nos brindan las arquitecturas híbridas: arquitecturas distribuidas con componentes multi y many-core, es posible combinar los paradigmas en un modelo de programación híbrido logrando aplicar una estrategia de paralelización más efectiva mediante múltiples niveles de paralelismo y reducción del overhead de comunicación. Existen muchas soluciones a problemas complejos de gran escala y para grandes volúmenes de datos, las cuales aplican técnicas HPC multi-tecnología, tanto de hardware como de software [Car18, Isu19].

Ante todo lo expuesto, nuestra motivación es investigar, verificar y poner en marcha nuevas técnicas y arquitecturas híbridas que ayuden a mejorar el procesamiento y sus tiempos de respuesta. Las técnicas de HPC serán nuestras herramientas para resolver con eficiencia cada uno de los objetivos: aplicar HPC multi-tecnología para resolver problemas complejos o de datos masivos en ambientes paralelos híbridos sobre arquitecturas many y multi-core. En la próxima sección presentamos diferentes casos de estudio para dos líneas de investigación.

2. LÍNEAS DE INVESTIGACIÓN, DESARROLLO E INNOVACIÓN

Los trabajos desarrollados se encuadran en dos líneas de investigación, una relacionada a la solución de problemas complejos con técnicas HPC y el otro a problemas Big Data. Para cada área nos enfocamos en:

- Problemas Complejos: Un Autómata Celular (AC) es un modelo matemático que

permite trabajar con sistemas dinámicos discretos de comportamiento complejo mediante el desarrollo de un conjunto de reglas simples. Estas reglas permiten especificar el nuevo estado de un componente en función de su estado actual y de su vecindario. Por sus características los AC son naturalmente paralelos, lo cual permite aplicar técnicas HPC en su solución computacional [Kau 84, Wol84].

- Uno de los usos más difundidos hoy en día de los AC, por su simplicidad y posibilidad de constante mejora, es en el ámbito de la simulación de procesos de difusión o dispersión, ya sea de rumores, enfermedades, etc. Analizar, prevenir y/o controlar la difusión y contagio de enfermedades, rumores, virus, etc. constituye una de las grandes preocupaciones del ser humano. Existen casos donde una enfermedad provocó la desaparición de una población entera o cambios demográficos importantes, por ejemplo la peste en (Europa, siglo XIV), fiebre amarilla (Buenos Aires, siglo XIX) o el cólera (Asia, siglo XIX), continuado la situación hasta hoy, por ejemplo con la Gripe A, el dengue, el ébola y el SIDA [Hup13, Wor14] o más recientes como el COVID-19. Ante estos problemas es importante y prioritario el estudio y control de estas enfermedades, como así también el análisis de su modo de transmisión o contagio. Una forma de atacar este problema es estudiando su comportamiento en un determinado medioambiente o realidad [Joh09, Mai13, Rey11, Yir12]. Los AC permiten expresar modelos realistas capaces de capturar con mayor fidelidad las características y el comportamiento del estado del sistema y su evolución, los cuales no pueden ser abordados analíticamente.
- En esta línea de trabajo, proponemos desarrollar un prototipo paralelo portable basado en AC para simular un proceso de difusión, tanto en arquitecturas multi-core, como many-core. En el caso de las primeras, hemos considerado tanto arquitecturas de memoria compartida y de memoria distribuida. Para los many-core, se tienen en cuenta las arquitecturas más conocidas y propuestas por Nvidia y AMD. Los problemas de difusión considerados son, en enfermedades: Gripe A y Covid-19; y, área social: noticias/rumores. Estas propuestas no permiten, por un lado estudiar en menos tiempo, el proceso de propagación teniendo en cuenta diferentes realidades: el tipo de población, su distribución y otras características; para luego tomar decisiones, como las campañas de vacunación, el aislamiento, cuarentena, mecanismo de contención, etc. Por el otro lado, evaluar el prototipo para las distintas arquitecturas y herramientas, analizando su portabilidad a plataformas heterogéneas.
- Grandes Volúmenes de Datos: En las últimas décadas el conjunto de datos a manipular es tan grande y complejo que los medios tradicionales de procesamiento son ineficaces. Existen diferentes áreas o aplicaciones donde se trabaja con gran cantidad de datos, en esta línea nos enfocamos en: seguridad en redes de datos, recuperación de la información y evolución y parentesco de las especies. El trabajo abordado en cada una de ellas es:
 - Seguridad en Redes de Datos: En el campo de la detección de anomalías en el tráfico en redes de datos, el problema consiste en la identificación de patrones no acordes al comportamiento normal del tráfico en la red [Bar16, Bar17]. Detectar un posible ataque requiere contar con tecnologías para la

clasificación del tráfico, asociando flujos de datos con las aplicaciones que los generan, lo que se denomina Sistema de Detección de Intrusos (IDS: Intrusion Detection System). Aquí proponemos la creación de un IDS híbrido, quien mediante técnicas de clasificación supervisada y no supervisada, visualización de datos, procesamiento de imágenes y HPC en GPU, detecta tanto ataques conocidos como potenciales en una red de área local. Para lograrlo, se busca identificar patrones que se desvían del comportamiento normal a partir del tráfico circulantes en la red (gran volumen de datos) en un tiempo cercano a su ocurrencia, a fin de poder tomar decisiones rápidas [Bar18].

- Recuperación de la Información: El incesante crecimiento de los datos, hace que en la actualidad las bases de datos no sólo son utilizadas para almacenar números o palabras, sino también documentos, archivos multimedia, muestras biológicas, huellas digitales entre otros tipos de datos, haciendo imposible realizar búsquedas de manera tradicional. Las consultas por similitud o búsqueda aproximada se presentan como una solución, por lo cual se necesitan métodos de acceso eficientes (índices) para permitir recuperar rápidamente los elementos que satisfacen los criterios de consulta, es decir los elementos son buscados considerando la cercanía de los mismos al elemento consultado. El modelo de espacio métrico [Cha01] resulta adecuado para ello. En él, las evaluaciones de la función de distancia requieren un alto costo computacional, por lo tanto es necesario la construcción de un índice para reducir la cantidad de cálculos. Uno de los índices está basado en permutantes y establece que dado un

conjunto de datos se pueden seleccionar elementos como puntos de referencia, donde el grado de cercanía de dos elementos está dado por la similitud de sus permutaciones [Lop13]. Hoy en día los grandes volúmenes de datos almacenados en bases de datos métricas hace que no sea suficiente procesar previamente el conjunto de datos mediante la creación de un índice, también es necesario acelerar las consultas mediante HPC. Aquí proponemos utilizar una arquitectura many-core para resolver consultas utilizando permutantes a fin de optimizar las búsquedas sobre grandes volúmenes de datos.

- Construcción de Árboles Filogenéticos: En el área de biología evolutiva [Ben09, Gar14, Yan14], la elaboración de un árbol filogenético en el cual se muestran las relaciones evolutivas de los seres vivos entre sí, reconocer los grados de cercanía de ancestros comunes, requiere contar con datos moleculares (fundamentalmente ADN) y/o morfológicos, los cuales identificarán la/s especie/s a incorporar en el árbol. El análisis filogenético demanda cálculos más complejos a medida que la cantidad de especies a catalogar crece, la cantidad de árboles a examinar en una búsqueda aumenta exponencialmente. Por esta razón es necesario contar con métodos, herramientas y técnicas HPC para datos masivos con el objeto de obtener información dentro de tiempos razonables. Para este caso, el objetivo es considerar ambientes HPC híbridos, a fin de lograr obtener buenos resultados, sin considerar el crecimiento exponencial de los datos a procesar.

Todas estas líneas de investigación tienen en cuenta la escalabilidad del problema y la

portabilidad de los desarrollos, aplicando técnicas de paralelismo híbrido y arquitecturas heterogéneas.

3. RESULTADOS OBTENIDOS/ESPERADOS

Como objetivos de las líneas de investigación nos planteamos facilitar el desarrollo de soluciones paralelas portables, de costo predecible, capaces de explotar las ventajas de modernos ambientes HPC a través de herramientas y “frameworks de computación” de alto nivel. Los resultados obtenidos hasta el momento son muy satisfactorios.

4. FORMACIÓN DE RECURSOS HUMANOS

Los resultados esperados respecto a la formación de recursos humanos son hasta el momento el desarrollo de 3 tesis de posgrado: 1 de doctor y 2 de maestría; y varias tesinas de grado en las distintas universidades intervinientes.

5. BIBLIOGRAFÍA

[Ben09] Benítez Burraco, A.. “*Genes y lenguaje: aspectos ontogenéticos, filogenéticos y cognitivos*”. ISBN 8429110046, 9788429110043. Reverte, 2009.

[Bas16] Basu, S. “*Parallel and Distributed Computing Architectures and Algorithms*”. ISBN: 8120352122, 9788120352124. PHI Learning Pvt. Ltd., 2016.

[Bar16] Barrionuevo, M. Lopresti, M., Miranda, N. Piccoli, F.. “*Un enfoque para la detección de anomalías en el tráfico de red usando imágenes y técnicas de Computación de Alto Desempeño*”. XXII Congreso Argentino De Ciencias de la Computación. CACIC 2016. Pp. 1166-1175. Octubre 2016, San Luis, Argentina.

[Bar17] Barrionuevo, M. Lopresti, M. Miranda, N. Piccoli, F. “*Un Modelo de Detección de Anomalías en una LAN usando K-NN y Técnicas de Computación de Alto*

Desempeño”. Communications in Computer and Information Science” (CCIS). ISBN: 978-3-319-75213-6. Editorial: Springer. Edición: Extranjera (inglés). Año: 2017. Volumen 790. Pp 219-228.

[Bar18] Barrionuevo, M., Lopresti, M., Miranda, N. and Piccoli, F. . “*Secure Computer Network: Strategies and Challenges in Big Data Era*”. Journal of Computer Science & Technologies. Editorial Red de Universidades Nacionales con Carreras de Informática (RedUNCI) Iberoamerican Science & Technology Education Consortium (ISTEC). Vol 3, pp 248-257. ISSN: 1666-6038 (electrónico), 1666-6046 (impreso). Diciembre 2018.

[Car18] CARLA 2018, *Selected Papers High of Performance Computing* 5th Latin American Conference. Communications in Computer and Information Science, Vol. 979. ISBN: 3030162052, 9783030162054. Colombia, 2018. Springer, 2019.

[Cha01] Chávez, E. Navarro, G., Baeza-Yates, R. and Marroquín, J.. “*Searching in metric spaces*”. ACM Comput. Surv. 33, 3 (September 2001), 273-321. 2001.

[Gar14] Garamszegi, L.. “*Modern Phylogenetic Comparative Methods and Their Application in Evolutionary Biology: Concepts and Practice*”. ISBN 3662435500, 9783662435502. Springer, 2014.

[Gro14] Gropp, W., Hoefler, T., Thakur, R. and Lusk, E. “*Using Advanced MPI: Modern Features of the Message-Passing Interface*”. Computer science & intelligent systems Scientific and Engineering Computation. ISBN: 0262527634 - 9780262527637. MIT Press, 2014.

[Hon10] Hong, S., Kim, H., Hong, S. and Kim, H.. “*An integrated GPU power and performance model*” in Proceedings of the 37th annual international symposium on Computer architecture - ISCA '10, Vol. 38, N° 3. Pp. 280-289. 2010.

[Hup13] Huppert, A. and Katriel, G.. “*Mathematical modelling and prediction*” in infectious disease epidemiology. Clinical Microbiology and Infection. Vol 19 N°11. Pp 999-1005, 2013.

- [Isu19] ISUM 2019, *Selected Papers of Supercomputing: 10th International Conference on Supercomputing. Communications in Computer and Information Science*, Vol. 1151. ISBN: 3030380432, 9783030380434. Springer Nature. México 2019.
- [Joh09] Johnson, T. and McQuarrie, B.. “*Mathematical modeling of diseases: Susceptible-infected-recovered (sir) model*”. In University of Minnesota, Morris, Math 4901 Senior Seminar, 2009.
- [Kau14] Kaur, K. and Rai, A.. “*A Comparative Analysis: Grid, Cluster and Cloud Computing*” *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 3, no. 3, pp. 2278–1021, 2014.
- [Kau84] Kaufman, S.. “*Emergent properties in random complex automata*”. *Physica D: Nonlinear Phenomena*. Vol 10 N°1 Pp 145-156, 1984.
- [Lop13] Lopresti, M., Miranda, N., Piccoli, F., Reyes, N. “*Solving Multiple Queries through the Permutation Index in GPU*”. 4th International supercomputing Conference in Mexico. Colima-México. 5-8 Marzo 2013.
- [Mai13] Maitanmi, O., Adekunle, Y. and Agbaje, M.. “*Model-Based Cellular Automata on Spread of Rumours*”. *Computer Science*, Babcock University. Nigeria. 2013.
- [Mar 17] Marinescu, D.. “*Cloud Computing: Theory and Practice*”. 2nd Ed. ISBN: 0128128119, 9780128128114. Morgan Kaufmann, 2017.
- [NVi19] Nvidia. “*CUDA C++ Programming Guide, Design Guide*”. https://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf. 2019.
- [Op19] OpenMP Architecture Review Board. “*OpenMP Application Programming Interface Specification Version 5.0*”. ISBN: 1795759887 - 978-1795759885. 2019.
- [Pac11] Pacheco, P. “*An Introduction to Parallel Programming*”, 1st ed., San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.
- [Rey11] Reyes, D.. “*Descripción y aplicaciones de los autómatas celulares*”, 2011. Technical Report, Autónoma de Puebla.
- [Vaj11] Vajda, A. “*Multi-core and Many-core Processor Architectures. In: Programming Many-Core Chips*”. Springer, Boston, MA. 2011.
- [Wol84] Wolfram, S.. “*Universality and complexity in cellular automata*”. *Physica D: Non-linear Phenomena*, Vol 10 N° 1. Pp 1-35, 1984.
- [Wor14] World Health Organization. *Influenza (seasonal)*. Fact sheet N211. 2014.
- [Yan14] Yang, Z.. “*Molecular Evolution: A Statistical Approach*”. OUP Oxford, 2014.
- [Yir12] Yiran, G. and Jinzhu, D. “*Research on Rumors Spread Based on Cellular Automata*”, in *Proceedings of the 2nd International Conference on Green Communications and Networks*. Vol 2. ISBN 978-3-642-35418-2. Pp 235-244 Londres, Inglaterra. Springer Heidelberg. 2012.