

**CARRERA DEL INVESTIGADOR CIENTÍFICO Y  
TECNOLÓGICO**  
**Informe Científico<sup>1</sup>**

**PERIODO <sup>2</sup>: 2012-2013**

Legajo N°:

**1. DATOS PERSONALES**

*APELLIDO: Tinetti*

*NOMBRES: Fernando Gustavo*

*Dirección Particular: Calle: N°:*

*Localidad: Tolosa, La Plata CP: 1900 Tel:*

*Dirección electrónica (donde desea recibir información): fernando@info.unlp.edu.ar*

**2. TEMA DE INVESTIGACION**

*Cómputo de Alto Rendimiento*

**3. DATOS RELATIVOS A INGRESO Y PROMOCIONES EN LA CARRERA**

*INGRESO: Categoría: Inv. Asistente Fecha: Oct. 2005*

*ACTUAL: Categoría: Inv. Adj. sin Director desde fecha: Dic. 2009*

**4. INSTITUCION DONDE DESARROLLA LA TAREA**

*Universidad y/o Centro: UNLP*

*Facultad: Informática*

*Departamento: -*

*Cátedra: -*

*Otros: Instituto de Inv. en Informática-LIDI*

*Dirección: Calle: 50 y 120 N°: -*

*Localidad: La Plata CP: 1900 Tel: 4227707*

*Cargo que ocupa: Prof. Asociado*

**5. DIRECTOR DE TRABAJOS. (En el caso que corresponda)**

*Apellido y Nombres: --*

*Dirección Particular: Calle: -- N°: --*

*Localidad: -- CP: -- Tel: --*

*Dirección electrónica: --*

.....  
Firma del Director (si corresponde)

.....  
Firma del Investigador

<sup>1</sup> Art. 11; Inc. "e" ; Ley 9688 (Carrera del Investigador Científico y Tecnológico).

<sup>2</sup> El informe deberá referenciar a años calendarios completos. Ej.: en el año 2008 deberá informar sobre la actividad del período 1°-01-2006 al 31-12-2007, para las presentaciones bianuales.

## 6. EXPOSICION SINTETICA DE LA LABOR DESARROLLADA EN EL PERIODO.

*Debe exponerse, en no más de una página, la orientación impuesta a los trabajos, técnicas y métodos empleados, principales resultados obtenidos y dificultades encontradas en el plano científico y material. Si corresponde, explicita la importancia de sus trabajos con relación a los intereses de la Provincia.*

El trabajo durante el período informado se ha continuado en la dirección de atacar la problemática alrededor de Cómputo de Alto Rendimiento (o HPC, de su sigla en inglés: High Performance Computing), incluyendo varios puntos de vista. Estos puntos de vista incluyen desde el problema de las aplicaciones heredadas hasta la optimización y paralelización en las arquitecturas de procesamiento actuales. El objetivo del trabajo realizado y a continuar es analizar y resolver varios aspectos que normalmente imponen restricciones y/o penalizaciones para lograr el máximo rendimiento posible de las aplicaciones que normalmente tienen mayores requerimientos de procesamiento. Estos aspectos se pueden analizar en varios contextos y es lo que se ha venido realizando desde hace varios años y durante el período informado en particular:

1) Desde la perspectiva de las aplicaciones ya desarrolladas y/o incluidas las aplicaciones heredadas (o *legacy*, tal su término técnico usual en inglés). En este sentido, se ha avanzado en varias direcciones específicas:

1.1) Análisis de las características más problemáticas de los programas científicos. Se han analizado y se considera importante seguir analizando por un lado la estructura general/amplia de un proyecto de software en producción y por el otro detalles específicos del software que son particularmente complejos de tratar en cuanto a su optimización y mantenimiento al menor costo posible. Al menos parte de varias de las publicaciones realizadas en el período tratan esta problemática, junto con la que se describe a continuación

1.2) Transformación de código. Una vez identificadas las características del software científico actualmente en producción (y en algunos casos, el que se está produciendo), se deben proveer formas al menos metodológicas para cambiar/mejorar aquellas que atentan contra la optimización de rendimiento.

1.3) Tratamiento integral de optimización y paralelización. Se propone y en parte se ha implementado un método completo de manejo de software que incluye la transformación tanto para optimización como específicamente para paralelización de procesamiento.

En todos los casos, el estudio se hace utilizando programas/software heredado que es ampliamente aceptado en el ambiente de HPC/científico, como los asociados a modelos climáticos y de ingeniería mecánica, por ejemplo.

2) En el nivel más bajo de abstracción del software se continúa con el estudio y el desarrollo de optimizaciones específicas tanto para procesamiento secuencial como paralelo en multiprocesadores, ambientes híbridos incluyendo GPU (Graphic Processing Units), como multicomputadoras en general homogéneas y heterogéneas. En todos los casos el estudio utilizando BLAS (Basic Linear Algebra Subprograms) provee un contexto acotado para el estudio caso por caso.

3) Análisis de rendimiento: se ha avanzado en el contexto de evaluación en base a un sistema de relojes sincronizados que a su vez se puede tratar de manera similar o considerar en relación con un sistema de tiempo real. Por otro lado, y también relacionado con esta problemática se está considerando aprovechar la información directamente proveniente del hardware no solamente en cuanto a tiempo (algo que se está haciendo) sino en cuanto a eventos específicos que se suceden en el hardware de procesamiento actual y que el propio hardware es capaz de monitorizar e informar para optimización.

4) Directamente desde la perspectiva del propio hardware disponible y actual de procesamiento, se considera la paralelización en general en el contexto de los multiprocesadores, las GPUs (Graphic Processing Units) y los clusters que, de hecho, podrían ser construidos en base a nodos con multiprocesamiento y/o procesamiento en GPUs.

## **7. TRABAJOS DE INVESTIGACION REALIZADOS O PUBLICADOS EN ESTE PERIODO.**

**7.1 PUBLICACIONES.** *Debe hacer referencia exclusivamente a aquellas publicaciones en las que haya hecho explícita mención de su calidad de Investigador de la CIC (Ver instructivo para la publicación de trabajos, comunicaciones, tesis, etc.). Toda publicación donde no figure dicha mención no debe ser adjuntada porque no será tomada en consideración. A cada publicación, asignarle un número e indicar el nombre de los autores en el mismo orden que figuran en ella, lugar donde fue publicada, volumen, página y año. A continuación, transcribir el resumen (abstract) tal como aparece en la publicación. La copia en papel de cada publicación se presentará por separado. Para cada publicación, el investigador deberá, además, aclarar el tipo o grado de participación que le cupo en el desarrollo del trabajo y, para aquellas en las que considere que ha hecho una contribución de importancia, deberá escribir una breve justificación.*

Nota: si bien se menciona en cada publicación la tarea de quien suscribe como investigador, director de tesis o proyectos de investigación, todas las publicaciones tienen relación con el lugar de trabajo y el proyecto que codirige quien suscribe. Más específicamente, el contexto general de todos los trabajos es el del Proyecto de Incentivos “Arquitecturas Multiprocesador Distribuidas: Modelos, Software de Base y Aplicaciones”, desarrollado en el III-LIDI bajo la dirección del Ing. Armando E. De Giusti y codirección de quien suscribe en desarrollo desde Enero de 2010 hasta Diciembre de 2013.

Nota: se numera cada artículo para simplificar el seguimiento de la documentación adjunta (artículos completos).

Congreso/Publicación: ACM SIGPLAN Fortran Forum, Volume 31 Issue 1, April 2012, ACM New York, NY, USA, ISSN: 1061-7264, pp. 5-22:

**1)** Título: “Fortran Legacy software: source code update and possible parallelisation issues”, Fernando G. Tinetti, Mariano Méndez.

### **Abstract**

We are working on a process for carrying out a set of transformations on Fortran legacy projects. We started our work for parallelization and reduction of runtime at least on multiprocessing systems, but we found necessary to update several old Fortran features and/or legacy software issues as a previous task. We present how to define and implement several source code transformations in order to enhance readability and, also, provide a source code that we think is more likely to be parallelized in subsequent work. Furthermore, we propose that some parallelization (e.g. for shared memory parallel computers) can be made at least as a tool-guided process, i.e. as other source code transformations. In the most simple cases, some transformations from sequential to parallel processing could be made automatically by a source code software analysis and transformation tool, which could provide the user (programmer/developer) a suggested way of Parallelisation. Source code transformations are initially

approached as restructurings, and implemented by changing the abstract syntax tree (AST) program representation. We provide some comments on our preliminary work on the source code transformations directly focussed on parallelization that we expect to implement almost automatically.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación: WICC 2012 - XIV Workshop de Investigadores en Ciencias de la Computación, 25-27 de Abril de 2012, Universidad Nacional de Misiones, Posadas, Argentina, ISBN 978-950-766-082-5:

2) Título: “Sistema de Virtualización con Recursos Distribuidos”, Pablo Pessolani, Toni Cortes, Silvio Gonnet, Fernando G. Tinetti, pp. 49-53.

#### RESUMEN

Si bien los Sistemas Operativos disponen de características de seguridad, protección, gestión de recursos, etc. éstas parecen ser insuficientes para satisfacer los requerimientos de los sistemas informáticos actuales. Las tecnologías de virtualización existentes han sido y continúan siendo masivamente adoptadas para cubrir esas necesidades de los sistemas y aplicaciones por sus características de particionado de recursos, aislamiento, capacidad de consolidación, seguridad, soporte de aplicaciones heredadas, facilidades de administración, etc. Una limitante de las tecnologías de virtualización tradicionales es que el poder de cómputo de una Máquina Virtual está limitado al poder de cómputo de la máquina física que la contiene. Por otro lado, los Sistemas Operativos Distribuidos de Imagen Unica ofrecen un poder de cómputo que integra en forma transparente los recursos computacionales de los nodos que constituyen un cluster. Sus características de transparencia evita la reprogramación de las aplicaciones heredadas que fueron desarrolladas para sistemas centralizados. La línea de I/D aquí presentada se orienta hacia un Sistema de Virtualización con Recursos Distribuidos que permita construir Máquinas Virtuales con mayor poder de cómputo que una máquina física combinando e integrando las tecnologías de virtualización y Sistemas Operativos Distribuidos de Imagen Unica.

Tipo de Participación: dirección de la idea básica del trabajo y redacción del artículo.

3) Título: “Aplicaciones Científicas Numéricas: El (Ciclo de Vida del) Software Heredado”, Mariano Méndez, Fernando G. Tinetti, pp. 523-527.

#### RESUMEN

El objetivo de esta línea de investigación es el estudio de las aplicaciones científicas de cómputo numérico desde el punto de vista de la Ingeniería de Software y del Procesamiento Paralelo. Además, se orienta a desarrollar técnicas y herramientas destinadas a reestructurar dichas aplicaciones permitiendo que éstas evolucionen a través del tiempo mejorando sus prestaciones y su estructura.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**4) Título:** “Técnicas de Recuperación de Información en Grandes Volúmenes de Datos Heterogéneos con Bases de Datos NOSQL”, Damián P. Barry, Carlos E. Buckle, Renato Mazzanti, Gustavo Samec, Cristian Pacheco, Rodrigo Jaramillo, Ignacio Real, Ignacio Aita, Juan Manuel Cortez, Fernando G. Tinetti, pp. 722-725.

**Resumen**

El presente proyecto se enfoca en la evaluación de las técnicas existentes para recuperación eficiente de información sobre grandes volúmenes de datos heterogéneos. Dichas técnicas permitirán establecer las capacidades necesarias con las que debería contar una base de datos de información masiva, tanto desde la perspectiva de almacenamiento y técnicas de indexación, como de distribución de las consultas, escalabilidad y rendimiento en ambientes heterogéneos. Para ello se diseñarán arquitecturas tanto centralizadas como distribuidas, y se realizarán las correspondientes verificaciones, estableciendo los porcentajes de mejora de rendimiento para cada arquitectura.

**Tipo de Participación:** Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**5) Título:** “Arquitecturas Multiprocesador Distribuidas: Cluster, Grid y Cloud Computing”, Armando De Giusti, Fernando G. Tinetti, Marcelo Naiouf, Horacio Villagarcía, Franco Chichizola, Laura De Giusti, Mónica Denham, Ismael Rodriguez, Diego Montezanti, Victoria Sanz, Fabiana Leibovich, Emmanuel Frati, José Pettorutti, Adrián Pousa, Enzo Rucci, Silvana Gallo, Luciano Iglesias, Javier Ballardini, pp. 778-782.

**Resumen**

Caracterizar las arquitecturas multiprocesador distribuidas enfocadas a cluster, grid y cloud computing, con énfasis en las que utilizan procesadores de múltiples núcleos (“multicores”), con el objetivo de modelizarlas, estudiar su escalabilidad, analizar y predecir performance de aplicaciones paralelas y desarrollar esquemas de tolerancia a fallas en las mismas. Analizar y desarrollar software de base para clusters de multicores, tratando de optimizar el rendimiento de tales arquitecturas para diferentes modelos de programación paralela y diferentes paradigmas de resolución de aplicaciones. En el año 2011 se han agregado dos líneas de interés:

El estudio de arquitecturas basadas en GPGPU y su comparación con clusters de multicores, así como el empleo combinado de GPUs y multicores en computadoras de alta performance.

El análisis de la eficiencia energética, considerando el impacto de la arquitectura, el sistema operativo, el modelo de programación y el algoritmo específico.

Es de hacer notar que este proyecto se coordina con otros dos proyectos en curso en el IILIDI, relacionados con Algoritmos Distribuidos/Paralelos y Sistemas de Software Distribuido.

**Tipo de Participación:** dirección de la idea básica del trabajo y redacción del artículo.

**6) Título:** “Planificación de Sistemas de Tiempo Real”, Fernando Romero, Fernando G. Tinetti, pp. 817-820.



## RESUMEN

La presente línea de investigación se inscribe en la temática de Sistemas de Tiempo Real, profundizando sobre dos aspectos claves: planificación de tareas y latencia de los sistemas. Se utilizarán herramientas de simulación y sistemas reales. Una vez obtenidas posibles mejoras, se implementarán en algún Sistema Operativo de Tiempo Real (SOTR). Se debe tener en cuenta que la planificación impacta en todo el sistema con lo que surgen problemas tales como inversión de prioridades[18], lo cual también es tema de este trabajo.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación: Proceedings of the 2012 International Conference on Parallel and Distributed Processing Techniques and Applications, Vol. 1, Hamid R. Arabnia (Editor), Hiroshi Ishii, Minoru Ito, Kazuki Joe, Hiroaki Nishikawa (Associate Editors), CSREA Press, July 16-19, 2012, Las Vegas, Nevada, USA, ISBN: 1-60132-227-5:

7) Título: "Hybrid Algorithms for Matrix Multiplication on Multicore Clusters", Fabiana Leibovich, Marcelo Naiouf, Laura De Giusti, Fernando G. Tinetti, Armando De Giusti, pp. 913-918.

### Abstract

Hybrid programming (through messages and shared memory) has gained importance since the appearance of multicore cluster architectures, fruit of the technological advance of processors and the physical limitations imposed by traditional architectures. This new programming paradigm allows exploiting the new memory hierarchy offered by the architecture. The purpose of this work is to carry out a comparative analysis between two hybrid algorithms that use two different parallel programming libraries for shared memory (PThreads and OpenMP). Both algorithms use MPI as message passing mechanism. For the experiments, the classic matrix multiplication problem is used, which is the basis for numerous applications. The test architecture used for the experimental analysis is a multicore cluster. The performance obtained and programming tools are compared.

Tipo de Participación: dirección de la idea básica del trabajo y redacción del artículo.

Congreso/Publicación: CACIC 2012 - XVIII Congreso Argentino de Ciencias de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina, Octubre 8-12 de 2012, ISBN 978-987-1648-34-4.

8) Título: "Legacy Fortran Software: Applying Syntactic Metrics to Global Climate Models", Mariano Méndez, Jeffrey Overbey, Fernando G. Tinetti, pp. 847-856.

### Abstract

It is difficult to maintain legacy Fortran programs that use outdated programming constructs, especially when this maintenance requires a detailed understanding of the code (e.g., for parallelization). Initially, we want to gauge the prevalence of such constructs by applying straightforward syntactic metrics to some well-known global climate models. Detailed information regarding files, subroutines, and loops has been collected from each model by applying a lightweight source

code static analysis based on ASTs (Abstract Syntax Tree) for a posterior analysis. Modernizing Fortran Legacy programs is still a challenge. Our objective has been to collect relevant information on these programs to help us approach parallelizing legacy scientific programs in a shared memory environment (e.g. using multi-core processors). The data we collected indicate that old Fortran features are still being used on these models in these days. Furthermore, we propose some metrics to be used as a guide to determine how many changes a program needs in order to be modernized, optimized, and eventually, parallelized.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**9)** Título: "Trabajo Preliminar para la Obtención de Tiempos Sincronizados en Clusters con Nodos de Múltiples Núcleos", Fernando Romero, Horacio Villagarcía, Fernando G. Tinetti, pp. 1471-1477.

#### Resumen

En este artículo se presenta una extensión de la metodología y herramienta para instrumentación de programas paralelos en plataformas de cómputo distribuidas que se venía elaborando en trabajos anteriores. Específicamente, la extensión contempla el hardware de múltiples núcleos que actualmente se utiliza en los nodos individuales de los clusters. En el contexto de instrumentación de tiempo en ambientes distribuidos es fundamental la sincronización de los relojes que intervienen. Se mantiene el algoritmo básico de sincronización, usando las estrategias clásicas que se utilizan en ambientes distribuidos para entornos de cluster, con una red de interconexión sobre la que se tiene acceso exclusivo (o controlado) para todas las comunicaciones entre las computadoras que se sincronizan. Este ambiente es específicamente el de los entornos de cómputo paralelo en clusters.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación: Proceedings 24th IASTED International Conference on Parallel and Distributed Computing and Systems, PDCS 2012, November 12 – 14, 2012, Las Vegas, USA, ISBN 978-0-88986-940-0, pp. 222-229:

**10)** Título: "Sequential and Shared and Distributed Memory Parallelization in Clusters: N-Body/Particle Simulation", Fernando G. Tinetti, Sergio M. Martin.

#### ABSTRACT

The particle-particle method for N-Body problems is one of the most commonly used methods in computer driven physics simulation. These algorithms are, in general, very simple to design and code, and highly parallelizable. In this article, we present the most important approaches for the application of the three performance improvement areas on these algorithms when executed on high performance computing (HPC) clusters: 1) sequential optimization (a single core in a node of the cluster), 2) shared memory parallelism (in a single node with multiple CPUs available, just like a multiprocessor), and 3) distributed memory parallelism (in the whole cluster). For each one of the improvement areas we

present the employed techniques and the obtained performance gain. Also, we will show how some (sequential/classical) code optimizations are almost essential for obtaining at least acceptable parallel performance and scalability.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación: WICC 2013 - XV Workshop de Investigadores en Ciencias de la Computación, 18-19 de Abril de 2013, Universidad Autónoma de Entre Ríos, Paraná, Argentina, ISBN 9789872817961.

**11)** Título: "Un mecanismo de IPC de microkernel embebido en el kernel de Linux", Pablo Pessolani, Toni Cortes, Silvio Gonnet, Fernando G. Tinetti, pp. 11-15.

#### Resumen

Existe una marcada tendencia en la industria de comercializar procesadores con múltiples núcleos, conocidos como multi-cores. Se prevé que en el mediano plazo la cantidad de núcleos aumente significativamente hasta miles de núcleos por procesador [1] denominados many-cores. Esta tendencia requiere de sistemas operativos (OS) que puedan aprovechar éstas tecnologías y que adapten su funcionamiento de tal forma de que no se transformen en el cuello de botella de las aplicaciones que ejecutan sobre ellos como consecuencia de la contención de los recursos que comparten [2].

Los OS basados en microkernel, los OS multi-kernel, algunos exokernels, y ciertas tecnologías de virtualización ofrecen ventajas significativas en su diseño para su adaptación a sistemas many-cores. Estas arquitecturas requieren de un mecanismo de transferencia de mensajes para comunicar entidades tales como procesos, hilos, kernels o máquinas virtuales.

En este artículo se presentan el trabajo de investigación y desarrollo de un mecanismo de IPC basado en Minix 3 (denominado M3-IPC) embebido dentro del kernel de Linux. M3-IPC permite incorporar servicios de un OS basado en microkernel (Minix) dentro de un OS monolítico (Linux) conformando un sistema híbrido donde pueden convivir aplicaciones y servicios de ambas arquitecturas y adaptarlos de manera no abrupta a los sistemas many-cores.

Tipo de Participación: Dirección y desarrollo de la idea principal/básica del trabajo, elaboración de ideas y experimentos asociados al trabajo y redacción del artículo.

**12)** Título: "Ingeniería de Software en el desarrollo de Aplicaciones para Dispositivos Móviles", Pablo Thomas, Nicolás Galdamez, Lisandro Delia, Federico Cristina, Sebastián Dapoto, Fernando Tinetti, Patricia Pesado, Armando De Giusti, pp. 586-589.

#### Resumen

Se presenta una línea de investigación y desarrollo que tiene por objeto estudiar temas relacionados con aplicaciones para dispositivos móviles, fundamentalmente aspectos de Ingeniería de Software orientados al desarrollo e implementación de aplicaciones móviles, sobre diversos entornos operativos y el estudio y desarrollo de aspectos de conectividad entre dispositivos móviles.



Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**13)** Título: "Escalabilidad y Paralelización mediante el uso de Hadoop Distributed File System", Damián P. Barry, Carlos E. Buckle, Rodrigo Jaramillo, Ignacio Real, Fernando G. Tinetti, pp. 677-680.

#### Resumen

En el marco del proyecto de investigación TECNICAS DE RECUPERACION DE INFORMACION EN GRANDES VOLUMENES DE DATOS HETEROGENEOS CON BASES DE DATOS NOSQL, el presente trabajo se orienta a evaluar configuraciones de clusters utilizando Hadoop Distributed File System (HDFS) para comprobar las capacidades de disponibilidad, escalabilidad y paralelización en la recuperación de información.

Dicha evaluación permitirá establecer las capacidades necesarias con las que debería contar un File System Distribuido, tanto desde la perspectiva de almacenamiento y técnicas de indexación, como de distribución de las consultas, paralelización, escalabilidad y rendimiento en ambientes heterogéneos.

Para ello se diseñarán arquitecturas tanto centralizadas como distribuidas, y se realizarán las correspondientes verificaciones, estableciendo los porcentajes de mejora de rendimiento para cada arquitectura.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**14)** Título: "Arquitecturas Multiprocesador en HPC: Software de Base, Métricas y Aplicaciones", De Giusti Armando E., Tinetti Fernando G., Naiouf R. Marcelo, Chichizola Franco, De Giusti Laura C., Villagarcía Horacio, Montezanti Diego, Frati F. Emmanuel, Pousa Adrián, Rodríguez Ismael P., Denham Mónica M., Iglesias Luciano, pp. 686-690.

#### Resumen

Esta línea de Investigación está dentro del proyecto "Arquitecturas Multiprocesador Distribuidas. Modelos, Software de Base y Aplicaciones", acreditado por el Ministerio de Educación y de proyectos específicos apoyados por organismos nacionales e internacionales.

Asimismo los proyectos "Eficiencia energética en Sistemas Paralelos" y "Algoritmos Paralelos utilizando GPGPUs. Análisis de rendimiento" financiados por la Facultad de Informática de la UNLP.

En el tema hay cooperación con varias Universidades de Argentina y se está trabajando con Universidades de América Latina y Europa en proyectos financiados por CyTED, AECID y la OEI (Organización de Estados Iberoamericanos).

Se participa en iniciativas como el Programa IberoTIC de intercambio de Profesores y Alumnos de Doctorado en el área de Informática.

Por otra parte, se tiene financiamiento de Telefónica de Argentina en Becas de grado y posgrado. Asimismo el III-LIDI forma parte del Sistema Nacional de Cómputo de Alto Desempeño (SNCAD) del MINCYT.

Tipo de Participación: Dirección y desarrollo de la idea principal/básica del trabajo, elaboración de ideas y experimentos asociados al trabajo y redacción del artículo.

**15)** Título: "Tópicos de Diseño de Aplicaciones de Tiempo Real", Fernando Romero, Diego Encinas, Luciano Iglesias, Armando De Giusti, Fernando G. Tinetti, pp. 740-743..

#### Resumen

El objetivo de esta línea de investigación es el estudio y desarrollo de sistemas de software que poseen restricciones temporales, como son los Sistemas de Tiempo Real (STR) [4] [6] [15] haciendo hincapié en los aspectos de simulación, planificadores y comunicaciones.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación: The Journal of Supercomputing, May 2013, Volume 64, Issue 2, ISSN 0920-8542, DOI 10.1007/s11227-012-0863-x, pp 638-659.

**16)** Título: "Restructuring Fortran legacy applications for parallel computing in multiprocessors", Fernando G. Tinetti, Mariano Méndez, Armando De Giusti.

#### ABSTRACT

As it is widely known, multi-core computers are broadly used these days, and automatic parallelization of sequential programs is still a challenge. In this context, we propose a set of code transformations to be applied automatically by a tool in order to transform sequential legacy systems into their parallel version. We implement these transformations by applying a lightweight source code analysis based on rewritable AST (Abstract Syntax Tree). Since it is not always possible to automatically parallelize the code, we also implemented some specific analyses in order to report possible changes that would allow specific parallelization. Additionally, we present some examples in which these transformations were conducted and the corresponding performance experiments.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación: Congreso Español de Informática, Jornadas Sarteco 2013, Madrid, España, 17 al 20 de Septiembre 2013, ISBN: 978-84-695-8330-2, pp. 306-311.

**17)** Título: "Análisis del Impacto de la Jerarquía de Memoria en Clusters de Multicores Utilizando Contadores de Hardware", Fabiana Leibovich, Laura De Giusti, Marcelo Naiouf, Franco Chichizola, Fernando G. Tinetti, Armando De Giusti.

#### Resumen

El objetivo principal de la programación híbrida es aprovechar la jerarquía de memoria que las arquitecturas cluster de multicore proveen. En este sentido, partiendo de un mismo caso de estudio (multiplicación de matrices), este

trabajo se enfoca en la comparación de dos soluciones híbridas (donde se combina pasaje de mensajes y memoria compartida) que utilizan diferentes estrategias de paralelización. Las mismas utilizan de distinta manera la jerarquía de memoria presente en la arquitectura de experimentación (cluster de multicore), en particular haciendo un uso diferente del nivel L1 de cache. En función de los tiempos de ejecución obtenidos y utilizando contadores de hardware se busca verificar que el factor de mayor incidencia sobre el tiempo de ejecución es la optimización del acceso a cache L1.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación: CACIC 2013 - XIX Congreso Argentino de Ciencias de la Computación, Universidad CAECE, Mar del Plata, Argentina, Octubre 21-25 de 2013, ISBN 978-987-23963-1-2.

**18)** Título: "NetworkDCQ: A Multi-platform Networking Framework For Mobile Applications", Federico Cristina, Sebastián Dapoto, Fernando G. Tinetti, Pablo Thomas, Patricia Pesado, pp. 1075-1086.

#### ABSTRACT

Currently, the number of mobile applications that require (wireless) connectivity is constantly increasing. The need for sharing information among mobile devices exists in many applications, and almost every data exchange between these devices involve the same requirements: a means for discovering other mobile devices in a wireless network, establishing logical connections, communicating application data, and gathering information related to the physical connection. This paper proposes an open source developer-oriented framework that acts as a network support layer for host discovery, data communication among devices, and quality of service characterization, which can be used for developing several types of applications and is proposed for different platforms, such as Android Java, J2SE, and J2ME.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**19)** Título: "Inversión de prioridades: prueba de concepto y análisis de soluciones", Raúl Benencia, Luciano Iglesias, Fernando Romero, Fernando G. Tinetti, pp. 1395-1404.

#### Resumen

La planificación de tareas es el punto crucial de un sistema de tiempo real. Dicha función es llevada a cabo por el planificador del sistema operativo, diseñado para poder cumplir con las restricciones temporales de dichas tareas, teniendo en cuenta sus valores de prioridad. Al haber recursos compartidos por estas tareas, se produce el efecto llamado inversión de prioridades. En este trabajo se analiza dicho efecto y se evalúan las soluciones implementadas para este problema en el Sistema Operativo de Tiempo Real GNU/Linux con parche RT-PREEMPT.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**20)** Título: "N-Body Simulation Using GP-GPU: Evaluating Host/Device Memory Transference Overhead", Sergio M. Martin, Fernando G. Tinetti, Nicanor B. Casas, Graciela E. De Luca, Daniel A. Giulianelli, pp. 164-173.

#### Abstract

N-Body simulation algorithms are amongst the most commonly used within the field of scientific computing. Especially in computational astrophysics, they are used to simulate gravitational scenarios for solar systems or galactic collisions. Parallel versions of such N-Body algorithms have been extensively designed and optimized for multicore and distributed computing schemes. However, N-Body algorithms are still a novelty in the field of GP-GPU computing. Although several N-body algorithms have been proved to harness the potential of a modern GPU processor, there are additional complexities that this architecture presents that could be analyzed for possible optimizations. In this article, we introduce the problem of host to device (GPU) – and vice versa – data transferring overhead and analyze a way to estimate its impact in the performance of simulations.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**21)** Título: "Procesamiento de Señales SAR: Algoritmo RDA para GPGPU", Mónica Denham, Javier Areta, Isidoro Vaquila, Fernando G. Tinetti. pp. 174-183.

#### Resumen

En este trabajo se presenta una solución secuencial y una paralela del algoritmo RDA (Range Doppler Algorithm) para el procesamiento de señales de radares SAR (Synthetic Aperture Radar). La solución paralela se desarrolló en C CUDA para GP-GPU (General Purpose Graphic Processing Units). Se describe la solución desarrollada, se muestran los primeros resultados y se describen las futuras optimizaciones para dicho algoritmo.

Tipo de Participación: Dirección y desarrollo de la idea principal/básica del trabajo, elaboración de ideas y experimentos asociados al trabajo y redacción del artículo.

**22)** Título: "Análisis de rendimiento del algoritmo SGP4/SDP4 para predicción de posición orbital de satélites artificiales utilizando contadores de hardware", Federico J. Díaz, Fernando G. Tinetti, Nicanor B. Casas, Graciela E. De Luca, Sergio M. Martin, Daniel A. Giulianelli", pp. 940-949.

#### Resumen

Durante los últimos 25 años, la predicción de posición orbital de los cuerpos en orbitas cercanas y medianas a la tierra, se calculo mediante el conjunto de algoritmos de la familia SGP (acrónimo de "Simplified General Perturbations"). En la última década, se produjo un incremento considerable en la cantidad de satélites artificiales, aumentando también el número de objetos inutilizados en órbita (comúnmente llamados "Basura Espacial"). Este incremento requiere un

mayor esfuerzo computacional de los algoritmos utilizados. Para aprovechar en forma más eficiente los recursos computacionales actuales, puede ser necesario optimizar los algoritmos mencionados, e incluso plantear una solución paralela. El análisis aquí propuesto pretende determinar el rendimiento del algoritmo, identificando las zonas de cálculo intensivo, utilizando contadores de hardware para medir transferencias de memoria cache y rendimiento.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación (Capítulo de Libro): A. De Giusti, G. Simari, P. Pesado (Eds.), XVIII Argentine Congress of Computer Science. Selected Papers, Editorial de la Universidad Nacional de La Plata, ISBN 978-987-1985-20-3, Oct. 2013, Cap 23:

**23)** "Preliminary Work to Get Synchronized Times in Multicore Node Clusters", F. Romero, H. Villagarcía, Fernando G. Tinetti, pp. 257-264.

#### ABSTRACT

This paper presents an expansion of the methodology and tool for instrumenting parallel programs in distributed computer platforms that has been described in previous works. Specifically, this expansion considers multicore hardware that is currently used in the individual nodes of clusters. In the context of instrumentation in distributed environments, it is essential that all involved clocks are synchronized. The basic synchronization algorithm remains almost unchanged, using the classic strategies that are commonly used in distributed environments for cluster configurations, with an interconnection network with exclusive (or controlled) access to all communications among the synchronized computers. This environment is specifically that of parallel computing environments in clusters.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación (Capítulo de Libro): Moisés Torres Martínez (Editor), Supercomputing in México: A Navigation through Science and Technology, Volume 4, 1st Edition, ISBN (printed version): 978-607-450-918-2, ISBN (electronic version): 978-607-450-919-9, Dec. 2013 (artículos seleccionados del 4th International Supercomputing Conference in México 2013, Manzanillo, Colima, México, March 2013):

**24)** "Optimization and Parallelization Experiences Using Hardware Performance Counters", Fernando G. Tinetti, Sergio M. Martín, Fernando E. Frati, Mariano Méndez, pp. 157-166.

#### ABSTRACT

Current hardware for compute intensive tasks includes a large amount of processing facilities, which are sometimes difficult to use in an optimized way. High performance computing (HPC) is always focused in solving challenging (or, at least, compute intensive) problems for which the response time is the priority. We have been working from two different but usually complementary research problems: a) updating and parallelizing legacy (HPC/numerical) software, and b) analyzing different problems and approaches to optimization and parallel processing in clusters. We have found that raw hardware event



counters do not always directly provide useful information. We also found some guidelines for evaluating performance using those counters in the context of optimization and parallelization. In this article, we present those guidelines along with the performance evaluation tools that we used to determine objectively what parts of the algorithm offered better chances of improvement.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**7.2 TRABAJOS EN PRENSA Y/O ACEPTADOS PARA SU PUBLICACIÓN.** *Debe hacer referencia exclusivamente a aquellos trabajos en los que haya hecho explícita mención de su calidad de Investigador de la CIC (Ver instructivo para la publicación de trabajos, comunicaciones, tesis, etc.). Todo trabajo donde no figure dicha mención no debe ser adjuntado porque no será tomado en consideración. A cada trabajo, asignarle un número e indicar el nombre de los autores en el mismo orden en que figurarán en la publicación y el lugar donde será publicado. A continuación, transcribir el resumen (abstract) tal como aparecerá en la publicación. La versión completa de cada trabajo se presentará en papel, por separado, juntamente con la constancia de aceptación. En cada trabajo, el investigador deberá aclarar el tipo o grado de participación que le cupo en el desarrollo del mismo y, para aquellos en los que considere que ha hecho una contribución de importancia, deberá escribir una breve justificación.*

Nota: Los artículos continúan la numeración anterior para simplificar el seguimiento de la documentación adjunta (artículos completos y notificación de la aceptación)

Congreso/Publicación: The 2014 International Conference on Computational Science and Computational Intelligence (CSCI'14) March 10-13, 2014, Las Vegas, USA:

**25) "An Automated Approach to Hardware Performance Monitoring Counters",  
Fernando G. Tinetti, Mariano Méndez**

#### Abstract

Program performance optimization could be a very complex process, even with current software development facilities/tools. An Integrated Development Environment (IDE) usually does not include many aids for optimization and/or performance evaluation. We propose to include performance evaluation through hardware monitoring counters into IDE software. Currently, it is possible to reach hardware monitoring counters via many libraries, and we have also seen that many of those libraries are approximately at the same abstraction level (including the way at which they allow access to the hardware counters). Thus, it is not only possible to include some performance evaluation library into the development process but, also, including specific aids to use some library via configurable/adjustable code snippets. We show, as a proof of concept, an Eclipse plug-in to help High Performance Computing (HPC) programmers to access hardware monitoring event counters using PAPI (Performance API). The plug-in is able to automatically include source code to count specific events available via PAPI in sections of source code defined by the programmer. Also, given that the code is automatically included, it would

be also possible to remove that code from the release version (for the production environment).

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**26)** "Optimizing a GPU Algorithm Through Hardware Profiling Analysis", Fernando G. Tinetti, Sergio M. Martin

#### Abstract

Usage of GPU-based architectures for scientific computing has been steadily increasing in the last years. This new paradigm for both programming and execution has been applied to solve several classic problems much faster than using the conventional multiprocessor and/or multicomputer approach. These architectures allow an increase in performance –compared to conventional CPU processors – for specific types of algorithms that are particularly suitable for its greater number of simpler cores which execute one single instruction at a time, each one for different sets of data. Since this is still a relative new technology, GPU device manufacturers as well as independent researchers have published several experiences (success stories), best practices, and optimization guides to aid developers for obtaining the maximum program performance. However, there is still little information about the possible optimizations that can only be harnessed by analyzing the specific device's hardware performance counters. In this paper, we discuss several optimizations based on hardware profiling and share our learned lessons about how such data can be used to optimize a scientific algorithm on a GPU using CUDA.

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

Congreso/Publicación: Capítulo del libro a publicarse con los mejores artículos seleccionados del Congreso Argentino de Ciencias de la Computación, A. De Giusti, G. Simari, P. Pesado (Eds.), Editorial de la Universidad Nacional de La Plata:

**27)** "N-Body Simulation Using GP-GPU: Evaluating Host/Device Memory Transference Overhead", Sergio M. Martin, Fernando G. Tinetti, Nicanor B. Casas, Graciela E. De Luca, Daniel A. Giulianelli

#### Abstract

N-Body simulation algorithms are amongst the most commonly used within the field of scientific computing. Especially in computational astrophysics, they are used to simulate gravitational scenarios for solar systems or galactic collisions. Parallel versions of such N-Body algorithms have been extensively designed and optimized for multicore and distributed computing schemes. However, N-Body algorithms are still a novelty in the field of GP-GPU computing. Although several N-body algorithms have been proved to harness the potential of a modern GPU processor, there are additional complexities that this architecture presents that could be analyzed for possible optimizations. In this article, we introduce the problem of host to device (GPU) – and vice versa – data transferring overhead and analyze a way to estimate its impact in the performance of simulations

Tipo de Participación: Desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

**7.3 TRABAJOS ENVIADOS Y AUN NO ACEPTADOS PARA SU PUBLICACION.**

*Incluir un resumen de no más de 200 palabras de cada trabajo, indicando el lugar al que han sido enviados. Adjuntar copia de los manuscritos.*

N/A

**7.4 TRABAJOS TERMINADOS Y AUN NO ENVIADOS PARA SU PUBLICACION.**

*Incluir un resumen de no más de 200 palabras de cada trabajo.*

N/A

**7.5 COMUNICACIONES.** *Incluir únicamente un listado y acompañar copia en papel de cada una. (No consignar los trabajos anotados en los subtítulos anteriores).*

“B D y CAR/D”, en las Jornadas de Definición Estratégica en Big Data, 27 de Agosto de 2013, Estudio de Prospectiva TIC, Secretaría de Planeamiento y Políticas del Ministerio de Ciencia, Tecnología e Innovación Productiva y Fundación Sadosky

(<http://www.fundacionsadosky.org.ar/jornadas-de-definicion-estrategica-en-big-data-2>)

“Procesamiento Distribuido y Procesamiento Paralelo: Similitudes, Diferencias y Areas de Investigación”, Instituto de Investigación y Transferencia en Tecnología (ITT), Sec. de Investigación, Desarrollo y Transferencia de Tecnología, UNNOBA, 19 de Septiembre de 2013.

**7.6 INFORMES Y MEMORIAS TECNICAS.** *Incluir un listado y acompañar copia en papel de cada uno o referencia de la labor y del lugar de consulta cuando corresponda.*

Lugar y Fecha: III-LIDI, Facultad de Informática, UNLP, Agosto 2012, Reporte Técnico UNLP-PD-2012-1. Disponible en <http://ftinetti.zxq.net>.

Título: “Android: Definiciones Básicas y Desarrollo de Aplicaciones”, Federico Cristina, Sebastián Dapoto, Fernando G. Tinetti.

Lugar y Fecha: III-LIDI, Facultad de Informática, UNLP, Octubre 2012, Reporte Técnico PLA-001-2012. Disponible en <http://ftinetti.zxq.net/FortranLegacy/index.html>

Título: “A GPU Approach to Fortran Legacy Systems”, Mariano Méndez, Fernando G. Tinetti.

Lugar y Fecha: III-LIDI, Facultad de Informática, UNLP, Diciembre 2012, Reporte Técnico UNLP-PD-2012-2. Disponible en <http://ftinetti.zxq.net>.

Título: “NetworkDCQ: Multiplatform Network Support for Discovery, Communication, and QoS in Mobile Devices”, Fernando G. Tinetti, Federico Cristina, Sebastián Dapoto.

**8. TRABAJOS DE DESARROLLO DE TECNOLOGÍAS.**

**8.1 DESARROLLOS TECNOLÓGICOS.** *Describir la naturaleza de la innovación o mejora alcanzada, si se trata de una innovación a nivel regional, nacional o internacional, con qué financiamiento se ha realizado, su utilización potencial o actual por parte de empresas u otras entidades, incidencia en el mercado y niveles de facturación del respectivo producto o servicio y toda otra información conducente a demostrar la relevancia de la tecnología desarrollada.*

N/A

**8.2 PATENTES O EQUIVALENTES.** Indicar los datos del registro, si han sido vendidos o licenciados los derechos y todo otro dato que permita evaluar su relevancia.

N/A

**8.3 PROYECTOS POTENCIALMENTE TRASNFERIBLES, NO CONCLUIDOS Y QUE ESTAN EN DESARROLLO.** Describir objetivos perseguidos, breve reseña de la labor realizada y grado de avance. Detallar instituciones, empresas y/o organismos solicitantes.

N/A

**8.4 OTRAS ACTIVIDADES TECNOLÓGICAS CUYOS RESULTADOS NO SEAN PUBLICABLES** (desarrollo de equipamientos, montajes de laboratorios, etc.).

N/A

**8.5 Sugiera nombres (e informe las direcciones) de las personas de la actividad privada y/o pública que conocen su trabajo y que pueden opinar sobre la relevancia y el impacto económico y/o social de la/s tecnología/s desarrollada/s.**

N/A

**9. SERVICIOS TECNOLÓGICOS.** Indicar qué tipo de servicios ha realizado, el grado de complejidad de los mismos, qué porcentaje aproximado de su tiempo le demandan y los montos de facturación.

N/A

**10. PUBLICACIONES Y DESARROLLOS EN:**

**10.1 DOCENCIA**

Página web y apuntes o comentarios de los temas de la asignatura Programación Distribuida, Cursada 2012, Licenciatura en Informática, Fac. de Informática, UNLP, <http://pd2012.zxq.net/>.

Página web y apuntes o comentarios de los temas de la asignatura Computabilidad y Complejidad, Cursada 2012, Licenciatura en Informática, Fac. de Informática, <http://cyc2012.zxq.net/>.

Página web y apuntes o comentarios de los temas de la asignatura Programación Distribuida, Cursada 2013, Licenciatura en Informática, Fac. de Informática, UNLP, <http://pd2013.zxq.net/>.

Página web y apuntes o comentarios de los temas de la asignatura Computabilidad y Complejidad, Cursada 2013, Licenciatura en Informática, Fac. de Informática, <http://cyc2013.zxq.net/>.

**10.2 DIVULGACIÓN**

Página web de divulgación sobre publicaciones y detalles importantes a tener en cuenta en código heredado en HPC,

<http://ftinetti.zxq.net/FortranLegacy/index.html>

Es de hacer notar que esta página también se ha utilizado y pretende ser utilizada para recolección de información de un sector muy relevante en el área de HPC, que es el de los desarrolladores y usuarios de programas científicos. De hecho, contiene una encuesta y los resultados actualizados de la misma que se ponen a disposición en general.

**11. DIRECCION DE BECARIOS Y/O INVESTIGADORES.** *Indicar nombres de los dirigidos, Instituciones de dependencia, temas de investigación y períodos.*

Co-Director del Proyecto de Incentivos 11/F010 “Arquitecturas Multiprocesador Distribuidas: Modelos, Software de Base y Aplicaciones”. Director: Prof. Armando E. De Giusti.

Institución de Dependencia: III-LIDI, Fac. de Informática, UNLP

Nombres de los Dirigidos: Horacio Villagarcía, Oscar Bria, María José Abásolo, Laura De Giusti, Franco Chichizola, Javier Giacomantone, Fernando Romero, Mónica Denham, Emilio Luque, Dolores Isabel Rexachs del Rosario, Francisco Tirado, Luciano Iglesias, Ismael Rodriguez, Diego Montezanti, Diego Encinas, Victoria Sanz, Fabiana Leibovich, Emanuel Frati.

Período: Enero 2010 - Diciembre 2013

Director del Proyecto “Técnicas de Recuperación de Información en Grandes Volúmenes de Datos Heterogéneos con Bases de Datos NoSQL”. Co-Director: Damián Pablo Barry.

Institución de Dependencia: UNPSJB, Fac. de Ing., Sede Pto. Madryn.

Nombres de los Dirigidos: Carlos Ezequiel Eduardo Buckle, Renato Mazanti, Gustavo Samec, Rodrigo Jaramillo, Ignacio Real, Cristian Pacheco, Ignacio Aita, Juan Manuel Cortez.

Período: Abril 2012 - Marzo 2014.

Director de la Investigadora Mónica Malén Denham. Institución: CONICET (Consejo Nacional de Investigaciones Científicas y Técnicas). Cargo: Investigador Asistente. Resolución: Resolución D N° 3965, 16/11/2012.

Director del Proyecto “Procesamiento digital de Señales Eléctricas, Protocolos y Aplicaciones WEB”. Co-Director: Ricardo A. López.

Institución de Dependencia: UNPSJB, Fac. de Ing., Sede Trelew.

Nombres de los Dirigidos: Ricardo Antonio López, Sonia Mabel Soulier, Sebastián Pablo Wahler, Marcelo Eleazar Gómez, Nahuel Defossé, Diego Marcos Van Haaster, Emilio Pincioli, Pedro Abel Kostantinoff, Marcos Alfredo Paredes, Carina Edith Pires.

Período: Enero 2013 - Diciembre 2015.

**12. DIRECCION DE TESIS.** *Indicar nombres de los dirigidos y temas desarrollados y aclarar si las tesis son de maestría o de doctorado y si están en ejecución o han sido defendidas; en este último caso citar fecha.*

Postgraduado: Jorge P. Dignani

Postgrado: Especialista en Redes y Seguridad, Fac. De Informática, UNLP.

Defensa: Junio 2012. Calificación: Aprobado – Distinguido.

Postgraduado: Roberto J. R. Paz

Postgrado: Maestría en Redes de Datos y Comunicaciones, Fac. De Informática, UNLP

En curso, no finalizada.

(Co-director) Postgraduado: Nicolás Macía



Postgrado: Maestría en Redes de Datos y Comunicaciones, Fac. De Informática, UNLP

Dirección: Javier Diaz

En curso, no finalizada.

**13. PARTICIPACION EN REUNIONES CIENTIFICAS.** *Indicar la denominación, lugar y fecha de realización, tipo de participación que le cupo, títulos de los trabajos o comunicaciones presentadas y autores de los mismos.*

The 24th IASTED International Conference on Parallel and Distributed Computing and Systems, PDCS 2012, November 12 – 14, 2012, Las Vegas, USA.

Tipo de participación: Special Session Organizer: "High Performance Processing (HPC) in Clusters". Más información en <http://www.iasted.org/conferences/home-789.html> y en <http://www.iasted.org/conferences/session1-789.html>

The 24th IASTED International Conference on Parallel and Distributed Computing and Systems ~PDCS 2012~, November 12 – 14, 2012, Las Vegas, USA.

Tipo de participación: integrante de comité de programa y evaluación de artículos.

AST 2012 - Simposio Argentino de Tecnología, 41 JAIIO, 27-31 de Agosto de 2012, Universidad Nacional de La Plata, La Plata, Argentina.

Tipo de participación: integrante de comité de programa y evaluación de artículos.

XII Workshop Procesamiento Distribuido y Paralelo (WPDP), XVIII Congreso Argentino de Ciencias de la Computación, 8-12 de Octubre 2012, Bahía Blanca, Argentina.

Tipo de participación: integrante de comité de programa y evaluación de artículos.

XII Workshop Procesamiento Distribuido y Paralelo (WPDP), XVIII Congreso Argentino de Ciencias de la Computación, 8-12 de Octubre 2012, Bahía Blanca, Argentina.

Tipo de participación: presentación de artículo.

The 11th IASTED International Conference on Parallel and Distributed Computing and Networks ~PDCN 2013~ February 11-13, 2013, Innsbruck, Austria.

Tipo de participación: integrante de comité de programa y evaluación de artículos.

AST 2013 - Simposio Argentino de Tecnología, 42 JAIIO, 16-20 de Septiembre de 2013, Universidad Nacional de Córdoba, Córdoba, Argentina.

Tipo de participación: integrante de comité de programa y evaluación de artículos.

The 4th International Conference Confluence 2013, The Next Generation Information Technology Summit, 26th - 27th Sept. 2013, Amity University Uttar Pradesh, India.

Tipo de participación: integrante de comité de programa y evaluación de artículos.

XIII Workshop Procesamiento Distribuido y Paralelo, XIX Congreso Argentino de Ciencias de la Computación, CACIC 2013, Universidad CAECE Mar del Plata, 21-25 de Octubre de 2013.

Tipo de participación: integrante de comité de programa y evaluación de artículos.

**14. CURSOS DE PERFECCIONAMIENTO, VIAJES DE ESTUDIO, ETC.** *Señalar características del curso o motivo del viaje, período, instituciones visitadas, etc.*  
N/A

**15. SUBSIDIOS RECIBIDOS EN EL PERIODO.** *Indicar institución otorgante, fines de los mismos y montos recibidos.*

Subsidio Institucional para Investigadores de la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires, otorgado por Resolución N° 2410/12, por la suma de \$5.600.

Subsidio Institucional para Investigadores de la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires, otorgado Resolución N° 243/13, por la suma de \$6.000.

**16. OTRAS FUENTES DE FINANCIAMIENTO.** *Describir la naturaleza de los contratos con empresas y/o organismos públicos.*

N/A

**17. DISTINCIONES O PREMIOS OBTENIDOS EN EL PERIODO.**

“Premio a la Labor Científica, Tecnológica y Artística”, categoría Investigador Formado, de la Universidad Nacional de La Plata, Dic. 2012.

**18. ACTUACION EN ORGANISMOS DE PLANEAMIENTO, PROMOCION O EJECUCION CIENTIFICA Y TECNOLÓGICA.** *Indicar las principales gestiones realizadas durante el período y porcentaje aproximado de su tiempo que ha utilizado.*

Organismo: Facultad de Informática, UNLP.

Tarea/Gestión: Integrante de la Comisión Asesora de Investigaciones y Postgrado del Honorable Consejo Académico.

Porcentaje aproximado de tiempo: 1%.

Organismo: International Association of Science and Technology for Development (IASTED).

Tarea/Gestión: Integrante del Technical Committee on Parallel & Distributed Computing & Systems (básicamente integrante de comités de programa y evaluador de artículos de investigación, tareas incluidas en otros items de este informe).

Porcentaje aproximado de tiempo: menos del 1%.

Organismo: Facultad de Informática, UNLP.

Tarea/Gestión: Integrante del Comité Académico del Magíster en Redes de Datos.

Porcentaje aproximado de tiempo: menos del 1%.

Organismo: Universidad Nacional de La Plata, Sec. de Ciencia y Técnica.

Tarea/Gestión: Integrante de las Comisiones Asesoras Técnicas (CAT) Exactas para Becas de la UNLP, 2012-2013, por Resolución Nro. 1340/12, 13 de Dic. de 2012.

Porcentaje aproximado de tiempo: menos del 1%.

**19. TAREAS DOCENTES DESARROLLADAS EN EL PERIODO.** *Indicar el porcentaje aproximado de su tiempo que le han demandado.*

Profesor de la Asignatura: Computabilidad y Complejidad, Licenciatura en Informática y Licenciatura en Sistemas de la Facultad de Informática de la UNLP.

Porcentaje aproximado de tiempo: 10%.

Profesor de la Asignatura: Programación Distribuida, Licenciatura en Informática y Licenciatura en Sistemas de la Facultad de Informática de la UNLP.

Porcentaje aproximado de tiempo: 10%.

Profesor del Curso de Postgrado 2012: Clusters. Programación en Clusters. Curso válido para Doctorado en Ciencias Informáticas de la Facultad de Informática de la UNLP, otorgando 5 créditos. Curso regular del postgrado de Especialista en Cómputo de Altas Prestaciones y Tecnología GRID, de la Facultad de de Informática de la UNLP.

Porcentaje aproximado de tiempo: 1%.

Profesor del Curso de Postgrado 2013: Evaluación de rendimiento en HPC usando contadores de hardware. Curso válido para Doctorado en Ciencias Informáticas de la Facultad de Informática de la UNLP, otorgando 5 créditos. Curso regular del postgrado de Especialista en Cómputo de Altas Prestaciones y Tecnología GRID, de la Facultad de de Informática de la UNLP.

Porcentaje aproximado de tiempo: 1%.

**20. OTROS ELEMENTOS DE JUICIO NO CONTEMPLADOS EN LOS TITULOS ANTERIORES.** *Bajo este punto se indicará todo lo que se considere de interés para la evaluación de la tarea cumplida en el periodo.*

Editor del Journal of Computer Science and Technology (JCS&T). Edición electrónica en <http://journal.info.unlp.edu.ar/journal> con ISSN: 1666-6038 y edición impresa con ISSN: 1666-6046.

Evaluador del The Journal of Supercomputing, Springer Science+Business Media B.V., ISSN: 0920-8542 (Paper) 1573-0484 (Online).

Editorial Board Member del Journal Parallel and Distributed Computing and Networks, ISSN (Online): 1925-5543, ISSN (Hardcopy): 1925-5535, desde Abril de 2011.

Evaluador de Parallel & Cloud Computing, American V-King Scientific Publishing, LTD, ISSN (Online): 2304-9456, ISSN (Hardcopy): 2304-9464, desde Octubre de 2012.

Dirección del Pasante Mg. Mariano Méndez. Período: Agosto 2013 – Diciembre 2013. Asignatura Taller de Proyecto de Ing. en Computación, Fac. de Ingeniería -Fac. de Informática, UNLP.

**21. TITULO Y PLAN DE TRABAJO A REALIZAR EN EL PROXIMO PERIODO.** *Desarrollar en no más de 3 páginas. Si corresponde, explicita la importancia de sus trabajos con relación a los intereses de la Provincia.*

Cómputo de Alto Rendimiento: Optimización y Paralelización

Tal como se describe en la sección "6. Exposición Sintética de la Labor Desarrollada en el Período", se considera muy importante continuar el trabajo de investigación en

múltiples niveles de abstracción relacionados con la mejora u optimización de Cómputo de Alto Rendimiento (o HPC, de su sigla en inglés: High Performance Computing): aplicaciones heredadas, operaciones básicas de cómputo intensivo, análisis y monitorización de rendimiento, optimización de los recursos disponibles en las arquitecturas de cómputo paralelo más usadas actualmente: múltiples núcleos (*cores*), múltiples procesadores, GPUs, y clusters homogéneos y heterogéneos.

Las aplicaciones heredadas tienen múltiples características representativas en el campo de cómputo científico, tales como:

- a) Grandes requerimientos de cómputo numérico.
- b) Implementación de modelos matemáticos de procesos físicos.
- c) Utilizadas en producción para resolver problemas de la importancia de estudios climáticos de escala global y de ingeniería mecánica.
- d) Implementados originalmente por científicos no directamente relacionados con las técnicas, metodología y herramientas de cómputo y software disponibles actualmente.
- e) No necesariamente han evolucionado satisfactoriamente en cuanto a la utilización optimizada y/o aprovechamiento del hardware de procesamiento actual.

Teniendo en cuenta estas características, se continuará trabajando en un proceso de transformación de aplicaciones heredadas con el fin de optimizarlas y paralelizarlas, para ser adaptadas a las plataformas de cálculo actuales [17] [12]. El proceso que se propone es completo en cuanto a que se encarga de todo el manejo del proyecto de software como tal, teniendo en cuenta que tradicionalmente el costo de mantenimiento del software ha sido excesivamente alto, por encima del 50% del costo total [18] [5] [6]. Sin embargo, dentro del proceso completo de manejo del software la actividad que se considera de fundamental importancia es la de reestructuración (*restructuring*), dado que es la que efectivamente permite evolucionar el software y para la que no siempre se tienen las herramientas ni la metodología necesarias para transformar un programa disponible y en producción. Más específicamente, se están incluyendo y se continuarán estudiando diferentes tipos de análisis del código para, por un lado, mejorar las características del propio software como tal, para hacerlo más propenso a los cambios y a un mantenimiento más simple (menos costoso) y, por el otro, directamente para mejorar el rendimiento, haciendo posible la optimización y la paralelización del mismo.

Es importante recalcar que "paralelización" tiene actualmente muchas variantes: desde la llamada a nivel de hardware como ILP (*Instruction Level Parallelism*) hasta la distribución del procesamiento en plataformas muy débilmente acopladas y heterogéneas, aún con múltiples niveles de heterogeneidad como pueden encontrarse en un cluster heterogéneo, con diferentes multiprocesadores y GPUs en los nodos del cluster. Como parte del estudio y manejo de software heredado, es necesario estudiar y producir herramientas o al menos una metodología de trabajo sobre el software que proporcione documentación del software, que es normalmente inexistente o muy pobre o muy desactualizada en el caso de las aplicaciones heredadas.

Las operaciones básicas de cómputo intensivo, y más específicamente las operaciones relacionadas con álgebra lineal (BLAS) sobre matrices densas o ralas, si bien son de muy bajo nivel de abstracción siempre son importantes desde dos puntos de vista: por un lado, para estudiar el máximo aprovechamiento de los recursos de cómputo disponibles y por el otro porque muchas aplicaciones dependen casi exclusivamente de ellas en cuanto a rendimiento [7]. Para estas operaciones, que se pueden equiparar en cuanto a simplicidad o bajo nivel de abstracción con otras de ámbitos no numéricos como las búsquedas en bases de datos no SQL, las opciones de paralelización siempre son muy amplias y dependientes de la arquitectura que actualmente es muy variada. Esta amplitud de opciones combina dos realidades muy importantes en este contexto: a)

la simplicidad de las propias operaciones, y b) las múltiples arquitecturas de cómputo subyacentes. En parte relacionado con este tema, pero que incluso es más amplio, se tiene el desarrollo actual del área denominada "Big Data". En este sentido, se continuará hacia el avance siempre desde la perspectiva de cómputo de alto rendimiento. Si bien es un área sobre la que recién se está comenzando a poner atención tanto en nuestro país como en nuestra provincia, el desarrollo de conocimiento y herramientas de cómputo y evaluación asociadas se considera de vital importancia en el actual contexto científico y tecnológico.

Si bien la simplicidad de las propias operaciones puede ser vista como una restricción, dado que las definiciones de cada una de ellas es casi el propio algoritmo, se ha demostrado que esta simplicidad de todas maneras deja espacio a diferentes implementaciones optimizadas que se puede utilizar dependiendo de la arquitectura subyacentes. Quizás el ejemplo más representativo en este sentido haya sido y es la operación de multiplicación de matrices, en cuyo caso se debe recordar que en la actualidad es ampliamente utilizada la implementación por bloques, que además se adapta (vía el tamaño de los bloques, justamente) a los tamaños de cache disponible en cada arquitectura. Como se comentó antes, la variedad en las arquitecturas de cómputo ha crecido notoriamente en los últimos años, y se deben tener en cuenta a la hora de optimizar y paralelizar el cómputo, sea éste del tipo numérico o no numérico, o específicamente para implementar operaciones de álgebra lineal o u otro tipo de operaciones, o sea sobre matrices densas o ralas.

En términos más específicos de arquitecturas, se puede mencionar lo que en la literatura se conoce como multiprocesadores, que aunque homogéneos en construcción, son heterogéneos para la caracterización de rendimiento. Incluso actualmente se está tratando de incorporar el control de "afinidad" (*affinity*) en especificaciones tales como la de OpenMP [14] dado que se reconoce que las computadoras con múltiples procesadores de múltiples núcleos por construcción son multiprocesadores pero en términos de rendimiento y por la ubicación de datos en caches, la administración inadecuada de los hilos de ejecución (threads) pueden generar penalizaciones importantes de rendimiento. Quizás éste sea el ejemplo más sencillo, y justamente esto hace que sea necesario estudiar con atención otras combinaciones de arquitecturas de cómputo como la que incluye GPUs para optimizar y utilizar apropiadamente el o los modelos de cómputo/programación de las arquitecturas.

Las operaciones y el procesamiento en el caso de las operaciones no numéricas como las de recuperación de información en bases de datos no SQL quizás está más "restringido" al ambiente estrictamente distribuido por el uso de modelos de cómputo como map-reduce [1] [4]. Sin embargo, los problemas pueden llegar a ser similares a los del caso numérico cuando se avance en la investigación de la optimización de rendimiento es estas operaciones específicas sobre bases de datos no SQL [3].

El área de evaluación de rendimiento se ha visto enriquecida con la inclusión de monitorización de eventos específicos que el hardware mismo de los procesadores es capaz de reportar a las aplicaciones [9] [8] [10]. Aunque este tipo de monitorización y el análisis resultante puede ser considerado de muy bajo nivel de abstracción, es muy importante para la optimización tanto del cómputo secuencial como paralelo en un procesador y aún en multiprocesadores donde cada procesador tiene parte de la jerarquía de memoria integrada junto con varios núcleos (*cores*) de procesamiento. Además, se puede integrar sin problemas a una visión más global de evaluación de rendimiento utilizando la sincronización de relojes de las computadoras de un cluster, por ejemplo [16].



Se pretende avanzar en la inclusión de monitorización por hardware al menos porque es la más confiable, dado que es el propio hardware el que la implementa (no dejando lugar a especulaciones), y también porque se puede integrar no solamente de manera metodológica a la evaluación global de rendimiento, como se ha mencionado antes, sino que también se puede integrar ya como herramienta más automatizada en el proceso completo de manejo de proyectos de software de HPC. En este sentido, se estudiará la utilización de herramientas como perf [15] y bibliotecas más orientadas a la instrumentación, como PAPI (Performance Application Programming Interface) [13] [11].

El caso de herramientas de monitorización como perf es interesante porque sigue la línea de trabajo en cierto grado tradicional de otras herramientas muy utilizadas y ampliamente aceptadas en HPC como los *profilers*. El caso de las bibliotecas de instrumentación sigue una línea de un poco menor nivel de abstracción pero a la vez es más versátil en cuanto a qué monitorizar y específicamente en qué lugar de la aplicación. Teniendo en cuenta que la mayor cantidad de tiempo de ejecución suele darse en una parte reducida del código (normalmente menor al 20% o aún menor al 10%) [2], las bibliotecas de instrumentación son muy atractivas para mirar en detalle el código realmente relevante desde el punto de vista del rendimiento. Sin embargo, la versatilidad y el bajo nivel de abstracción atentan, en cierto modo, contra su utilización, o al menos contra la utilización sencilla en los proyectos de software de HPC. Es por eso que se estudiará la posibilidad de manejo automatizado de estas bibliotecas, identificando las tareas que siempre se tienen que hacer para utilizarlas. A priori, se puede decir que lo importante es identificar: a) qué eventos monitorizar, b) en qué parte del código, c) qué mostrar y cuándo. Se procurará que las tareas asociadas a estos detalles se implementen con código que podrá ser insertado automáticamente o de manera asistida en función de las selecciones o necesidades de quien tenga que llevar a cabo la evaluación de rendimiento. Se pretende avanzar, de hecho, en la inclusión de estas operaciones en el contexto de un ambiente de desarrollo integrado (IDE: Integrated Development Environment).

#### Referencias

[1] A. Abouzeid, K. Bajda-Pawlikowski, D. J. Abadi, A. Rasin, A. Silberschatz, "HadoopDB: An Architectural Hybrid of MapReduce and DBMS Technologies for Analytical Workloads", PVLDB 2(1): 922-933 (2009)

[2] A. V. Aho, J. D. Ullman, Foundations of Computer Science: C Edition (Principles of Computer Science Series), W. H. Freeman, 1994, ISBN 0716782847.

[3] D. P. Barry, C. E. Buckle, R. Mazzanti, G. Samec, C. Pacheco, R. Jaramillo, I. Real, I. Aíta, J. M. Cortez, F. G. Tinetti, "Técnicas de Recuperación de Información en Grandes Volúmenes de Datos Heterogéneos con Bases de Datos NOSQL", WICC 2012 - XIV Workshop de Investigadores en Ciencias de la Computación, 25-27 de Abril de 2012, Universidad Nacional de Misiones, Posadas, Argentina, ISBN 978-950-766-082-5, pp. 722-725.

[4] J. Dean, S. Ghemawat, "MapReduce: simplified data processing on large cluster", Communications of the ACM - 50th anniversary issue: 1958 - 2008 CACM, Volume 51 Issue 1, January 2008.

[5] A. Eastwood, "Firm Fires Shots at Legacy Systems", Computing Canada, Nol. 19, No. 2, 1993.

- [6] L. Erlikh, "Leveraging Legacy System Dollars for E-Business", IT Professional, Vol. 2, No. 3, 2000, IEEE Computer Society.
- [7] J. J. Dongarra, Performance of Various Computers Using Standard Linear Equations Software, Electrical Engineering and Computer Science Department, University of Tennessee, Knoxville, Oak Ridge National Laboratory, University of Manchester, CS - 89 - 85, February 5, 2013.
- [8] Intel Programming Guide, 2010, "Intel Microarchitecture Codename Nehalem Performance Monitoring Unit Programming Guide (Nehalem Core PMU)", <http://software.intel.com/file/30320>, (30320-Nehalem-PMU-Programming-Guide-Core.pdf).
- [9] P. Irelan, S. Kuo, "Performance Monitoring Unit Sharing Guide", Intel White Paper, <http://software.intel.com/file/30388>, (30388-PMU-Sharing-Guidelines.pdf)
- [10] D. Levinthal, Performance Analysis Guide for Intel Core i7 Processor and Intel Xeon 5500 processors, 2009.  
[http://software.intel.com/sites/products/collateral/hpc/vtune/performance\\_analysis\\_guide.pdf](http://software.intel.com/sites/products/collateral/hpc/vtune/performance_analysis_guide.pdf) (performance analysis guide.pdf)
- [11] A. Malony, S. Biersdorff, S. Shende, H. Jagode, S. Tomov, G. Juckeland, R. Dietrich, P. Duncan Poole, C. Lamb, "Parallel Performance Measurement of Heterogeneous Parallel Systems with GPUs," International Conference on Parallel Processing (ICPP'11), Taipei, Taiwan, September 13-16, 2011.
- [12] M. Méndez, J. Overbey, F. G. Tinetti, "Legacy Fortran Software: Applying Syntactic Metrics to Global Climate Models", CACIC 2012 - XVIII Congreso Argentino de Ciencias de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina, Octubre 8-12 de 2012, ISBN 978-987-1648-34-4, pp. 847-856.
- [13] S. Moore, J. Ralph, "User-defined Events for Hardware Performance Monitoring," ICCS 2011 Workshop: Tools for Program Development and Analysis in Computational Science, [www.sciencedirect.com](http://www.sciencedirect.com), Singapore, June 1, 2011.
- [14] OpenMP Architecture Review Board, OpenMP Application Program Interface, Version 4.0 - RC 2 - March 2013, Public Review Release Candidate 2.
- [15] perf: Linux profiling with performance counters, [https://perf.wiki.kernel.org/index.php/Main\\_Page](https://perf.wiki.kernel.org/index.php/Main_Page)
- [16] F. Romero, H. Villagarcía, F. G. Tinetti, "Trabajo Preliminar para la Obtención de Tiempos Sincronizados en Clusters con Nodos de Múltiples Núcleos", CACIC 2012 - XVIII Congreso Argentino de Ciencias de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina, Octubre 8-12 de 2012, ISBN 978-987-1648-34-4, pp. 1471-1477.
- [17] F. G. Tinetti, M. Méndez, "Fortran Legacy software: source code update and possible parallelisation issues", ACM SIGPLAN Fortran Forum, Volume 31 Issue 1, April 2012, ACM New York, NY, USA, ISSN: 1061-7264, pp. 5-22.
- [18] M. V. Zelkowitz, A. C. Shaw, J. D. Gannon, Principles of Software Engineering and Design, 1979, Prentice Hall Professional Technical Reference.

---

**Condiciones de la presentación:**

- A. El Informe Científico deberá presentarse dentro de una carpeta, con la documentación abrochada y en cuyo rótulo figure el Apellido y Nombre del Investigador, la que deberá incluir:
- Una copia en papel A-4 (puntos 1 al 21).
  - Las copias de publicaciones y toda otra documentación respaldatoria, en otra carpeta o caja, en cuyo rótulo se consignará el apellido y nombres del investigador y la leyenda "Informe Científico Período .....".
  - Informe del Director de tareas (en los casos que corresponda), en sobre cerrado.
- B. Envío por correo electrónico:
- Se deberá remitir por correo electrónico a la siguiente dirección: [ininvest@cic.qba.gov.ar](mailto:ininvest@cic.qba.gov.ar) (puntos 1 al 21), en formato .doc zipeado, configurado para papel A-4 y libre de virus.
  - En el mismo correo electrónico referido en el punto a), se deberá incluir como un segundo documento un currículum resumido (no más de dos páginas A4), consignando apellido y nombres, disciplina de investigación, trabajos publicados en el período informado (con las direcciones de Internet de las respectivas revistas) y un resumen del proyecto de investigación en no más de 250 palabras, incluyendo palabras clave.

---

**Nota:** El Investigador que desee ser considerado a los fines de una promoción, deberá solicitarlo en el formulario correspondiente, en los períodos que se establezcan en los cronogramas anuales.