

CARRERA DEL INVESTIGADOR CIENTÍFICO Y TECNOLÓGICO

Informe Científico¹

PERIODO ²: 2014-2015

1. DATOS PERSONALES

APELLIDO: Tinetti

NOMBRES: Fernando Gustavo

Dirección Particular: Calle: Nº:

Localidad: Tolosa - La Plata CP: 1900 Tel:

Dirección electrónica (donde desea recibir información, que no sea "Hotmail"):

fernando@info.unlp.edu.ar

2. TEMA DE INVESTIGACION

Cómputo Paralelo y Distribuido

3. DATOS RELATIVOS A INGRESO Y PROMOCIONES EN LA CARRERA

INGRESO: Categoría: Inv Asistente Fecha: 2005

ACTUAL: Categoría: Inv. Independiente desde fecha: 2015

4. INSTITUCION DONDE DESARROLLA LA TAREA

Universidad y/o Centro: UNLP

Facultad: Informática

Departamento: --

Cátedra: --

Otros: --

Dirección: Calle: 50 y 120 Nº:

Localidad: La Plata CP: 1900 Tel: 4227707

Cargo que ocupa: Prof. Titular

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

Apellido y Nombres: --

Dirección Particular: Calle: -- Nº: --

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

Dirección electrónica: --

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

.....
Firma del Investigador

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

² El informe deberá referenciar a años calendarios completos. Ej.: en el año 2014 deberá informar sobre la actividad del período 1°-01-2012 al 31-12-2013, para las presentaciones bianuales.

6. RESUMEN DE LA LABOR QUE DESARROLLA

Descripción para el repositorio institucional. Máximo 150 palabras.

Estudio de alternativas de cómputo paralelo y distribuido para resolución de problemas de cómputo de alto rendimiento (HPC: High Performance Computing) y posiblemente de tiempo real. Análisis de factibilidad y aplicación de optimización de cómputo secuencial, tanto en procesadores escalares como superescalares. Análisis de factibilidad y aplicación de cómputo paralelo en computadoras paralelas con arquitectura de memoria compartida (ej: "multicore" y "manycore"). Análisis de factibilidad y aplicación de cómputo paralelo en arquitecturas de cómputo con memoria distribuida (ej: clusters). Análisis y utilización de la relación entre optimización y paralelización (en sus múltiples formas) con análisis de rendimiento y tiempo de respuesta (posiblemente con restricciones de tiempo real). Análisis de costos relacionados con HPC y tiempo real, con aplicaciones en robótica. Utilización de optimización y paralelización en simulación numérica (ej: modelos climáticos y astrofísicos). Optimización y paralelización de software de HPC heredado.

7. 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.

Se ha avanzado en las siguientes líneas de trabajo, todas relacionadas con Cómputo de Alto Rendimiento o HPC: High Performance Computing: a) Numérico, mayormente relacionado con sistemas heredados y su optimización y paralelización, b) Orientado al procesamiento de Big Data, sobre todo en la nube o cloud, y c) Tiempo Real, sobre todo lo relacionado con sensado y procesamiento, en algunos casos relacionado con robótica. Se puede decir que aunque en todos los casos lo importante es la rapidez con la que se obtiene un resultado de cómputo, los escenarios de aplicación son claramente diferentes. Además de la particularidad de HPC, una característica común a varias de las aplicaciones es la de procesamiento distribuido para el aprovechamiento de recursos de cómputo. En el caso del procesamiento más tradicional de HPC, que está estrechamente ligado con el cómputo numérico, la necesidad de utilizar plataformas distribuidas se da por las limitaciones de las arquitecturas de cómputo de memoria compartida, y en este caso los clusters son la alternativa más sencilla, de menor costo y de mayor disponibilidad desde hace varios años. En el caso del cómputo orientado al procesamiento de grandes volúmenes de datos, actualmente denominado Big Data el cómputo en la nube es esencialmente distribuido, dado que tanto los diferentes orígenes de los datos como las plataformas de cómputo de esos datos son distribuidos y sin posibilidad de ser pensados o "adaptados" a ambientes centralizados o de memoria compartida. El procesamiento de Tiempo Real y principalmente en el área de aplicación de la robótica es muy usual encontrar y tener que resolver problemas del ámbito de los sistemas distribuidos tradicionales, que involucran diferentes lugares físicos (distribución física), comunicaciones y distribución de cómputo para paralelizar y poder responder a eventos en tiempo y forma, tal como lo indica la problemática de tiempo real clásica. De las tres líneas de investigación, la más avanzada en cuanto a que tiene mayor tiempo de trabajo en el ámbito de la investigación es la de HPC tradicional, con desarrollos y resultados específicamente orientados a cómputo numérico y sistemas de HPC heredados. Uno de los resultados más relevantes ha sido el desarrollo de una metodología completa de manejo de sistemas heredados, sobre todo considerando la problemática no solo de optimización y paralelización sino también de versionado y documentación. Se pretende que la resolución o manejo de estas tres problemáticas: cómputo, versionado y documentación, hará posible que los sistemas

heredados dejen de tener las grandes desventajas de inmovilidad de cambio que actualmente tienen. Esto es más importante aún en los sistemas de simulación complejos que actualmente están en producción y que no se pueden adaptar a las nuevas arquitecturas de cómputo, que son paralelas desde el hardware mismo hacia los mayores niveles de abstracción de procesamiento. La problemática de big data no tiene alternativa en cuanto a procesamiento paralelo y distribuido, y se está intentando el avance en cuanto al aprovechamiento de las técnicas más clásicas y estándares de cómputo paralelo y distribuido. Las aplicaciones de robótica plantean, por su lado, algunos problemas relativamente diferentes al de HPC, pero combinado con éste por sus requerimientos de tiempo de respuesta limitados (con características algunas veces catastróficas en caso de no cumplimiento de plazos) y procesamiento distribuido.

8. TRABAJOS DE INVESTIGACION REALIZADOS O PUBLICADOS EN ESTE PERIODO.

8.1 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.*

1) "Optimizing a GPU Algorithm Through Hardware Profiling Analysis", Fernando G. Tinetti, Sergio M. Martin, CSCI, The 2014 International Conference on Computational Science and Computational Intelligence (CSCI'14) March 10-13, 2014, Las Vegas, USA, ISBN 978-1-4799-3010-4/14, pp. 45-51.

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: Dirección, desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

2) "An Automated Approach to Hardware Performance Monitoring Counters", Fernando G. Tinetti, Mariano Méndez, CSCI, The 2014 International Conference on

Computational Science and Computational Intelligence (CSCI'14) March 10-13, 2014, Las Vegas, USA, ISBN 978-1-4799-3010-4/14, pp. 71-76.

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: Dirección, desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

3) "Arquitecturas Multiprocesador en Cómputo de Altas Prestaciones: Software de Base, Métricas y Aplicaciones", A. De Giusti, F. G. Tinetti, M. Naiouf, F. Chichizola, L. De Giusti, H. Villagarcía, D. Montezanti, E. Frati, A. Pousa, I. Rodriguez, S. Eguren, M. Denham, L. Iglesias, M. Mendez, XVI Workshop de Investigadores en Ciencias de la Computación, 7 y 8 de mayo de 2014, Universidad Nacional de Tierra del Fuego, Ushuaia, Tierra del Fuego, Argentina, ISBN 978-950-34-1084-4, pp. 706-711.

Abstract

Caracterizar las arquitecturas multiprocesador distribuidas enfocadas especialmente a cluster y cloud computing, con énfasis en las que utilizan procesadores de múltiples núcleos (multicores y GPUs), 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. Profundizar el estudio de arquitecturas basadas en GPUs y su comparación con clusters de multicores, así como el empleo combinado de GPUs y multicores en computadoras de alta performance. En particular estudiar performance en Clusters "híbridos". Analizar la eficiencia energética en estas arquitecturas paralelas, considerando el impacto de la arquitectura, el sistema operativo, el modelo de programación y el algoritmo específico. Analizar y desarrollar software de base para clusters de multicores y GPUs, tratando de optimizar el rendimiento.

En el año 2013 se han incorporado nuevas líneas de interés:

- El desarrollo de aplicaciones sobre Cloud y en particular las aplicaciones de Big Data en Cloud.
- La utilización de los registros de hardware de los procesadores para la toma de diferentes decisiones en tiempo de ejecución.
- El desarrollo de herramientas para la transformación de código heredado, buscando su optimización sobre arquitecturas paralelas.

Es de hacer notar que este proyecto se coordina con otros proyectos en curso en el III-LIDI, relacionados con Algoritmos Paralelos, Sistemas Distribuidos y Sistemas de Tiempo Real.

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

4) "Planificación de Sistemas de Tiempo Real: Diseños y Problemas", Fernando Romero, Diego Encinas, Luciano Iglesias, Armando De Giusti, Fernando G. Tinetti, XVI Workshop de Investigadores en Ciencias de la Computación, 7 y 8 de mayo de 2014, Universidad Nacional de Tierra del Fuego, Ushuaia, Tierra del Fuego, Argentina, ISBN 978-950-34-1084-4, pp. 785-788.

Abstract

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) haciendo hincapié en los aspectos de simulación, planificadores y comunicaciones.

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

5) "Microcontroladores Asociados a Medición y Comunicaciones en Sistemas SCADA de Energía", Ricardo A. López, Emilio Pinciroli, Fernando G. Tinetti, XVI Workshop de Investigadores en Ciencias de la Computación, 7 y 8 de mayo de 2014, Universidad Nacional de Tierra del Fuego, Ushuaia, Tierra del Fuego, Argentina, ISBN 978-950-34-1084-4, pp. 767-773.

Abstract

Los Sistemas de Adquisición de Datos y Control (SCADA Supervisory Control And Data Acquisition), toman de distintas zonas y dispositivos y del campo de control, un conjunto variado de señales y datos de medida. Constituyendo un subsistema independiente, en las Estaciones Transformadoras Eléctricas (EETT), existe un conjunto de aparatos de medición de la energía, cada uno de ellos asociado a un campo de suministro, que recolectan la información. Existe la factibilidad de asociar estos medidores a microcontroladores, efectuando así un aprovechamiento más eficiente y de menor costo de la medición, con algunas aracterísticas interesantes para las empresas. Consecuentemente, es posible construir con aquellos, una verdadera red de comunicaciones estandarizada que permite además del aprovechamiento por parte del Scada, aportar otras capacidades de supervisión, resguardo y auditoría.

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

6) "Implementación de Mensajes Rápidos y Valores de Muestreo IEC61850 sobre Ethernet con Microcontroladores", Marcelo E. Gómez, Sebastián P. Wahler, Fernando G. Tinetti, Ricardo A. López, XVI Workshop de Investigadores en Ciencias de la Computación, 7 y 8 de mayo de 2014, Universidad Nacional de Tierra del Fuego, Ushuaia, Tierra del Fuego, Argentina, ISBN 978-950-34-1084-4, pp. 774-778.

Abstract

La IEC61850 es una norma que establece un estándar para automatizar las tareas habituales dentro de una Estación Eléctrica (ET), tanto en distribución como transmisión de energía. La norma posee complejidad y gran parte de ella se dedica a las comunicaciones necesarias entre los diferentes componentes dentro de la ET. Estos elementos reciben el nombre de Dispositivos Electrónicos Inteligentes (IEDs: Intelligent Electronic Devices), existiendo gran cantidad y variedad de ellos dentro de la ET, intercambiando comandos y datos para llevar adelante la operación eléctrica. La norma define el uso de la pila TCP/IP para el intercambio de todos los mensajes (comandos y datos) y en la capa de aplicación, el uso de la gran variedad de protocolos preexistentes en la industria. Se definen diferentes tipos de mensajes en concordancia con diferentes necesidades dentro de la subestación y algunos de estos tipos no usan la pila completa TCP/IP.

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

7) "Implementación de Sistemas SCADA Utilizando Lenguajes de Alto Nivel", Nahuel Defossé, Diego Van Haaster, Lautaro Pecile, Fernando G. Tinetti, Ricardo A. López, XVI Workshop de Investigadores en Ciencias de la Computación, 7 y 8 de mayo de 2014, Universidad Nacional de Tierra del Fuego, Ushuaia, Tierra del Fuego, Argentina, ISBN 978-950-34-1084-4, pp. 779-784.

Abstract

Los Sistemas de Adquisición de Datos y Control (SCADA: Supervisory Control And Data Acquisition), toman del campo variables de estado provenientes de diferentes dispositivos. A fin de presentar al usuario una idea cabal del estado del proceso supervisado, el sistema debe ser capaz de trabajar con la totalidad de los datos obtenidos de los dispositivos y posteriormente ofrecer una representación amigable al humano. En este documento, se describe un proyecto para la implementación del Núcleo de Adquisición de manera acorde a los requerimientos específicos que se plantean.

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

8) "Synthetic Aperture Radar Signal Processing using GPGPU", Mónica Denham, Javier Areta, Isidoro Vaquila, Fernando G. Tinetti, CARLA 2014, First HPCLATAM-CLCAR Joint Confer-ence, Latin American High Performance Computing Conference, Universidad Técnica Federico Santa María, Valparaíso, Chile, 20 - 22 October, 2014.

Abstract

In this work an efficient parallel implementation of the Chirp Scaling Algorithm (CSA) for Synthetic Aperture Radar (SAR) processing is presented. The architecture selected for the implementation is General Purpose Graphic Processing Unit (GPGPU), as it is well suited for scientific applications and real time implementation of algorithms. The analysis of a first implementation led to several improvements which resulted in an important final speedup. Details of the issues found are explained, and the performance improvement of their correction explicitly shown.

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

9) "Integrating Software Metrics for Fortran Legacy into an IDE", Mariano Mendez, Fernando G. Tinetti, CACIC 2014, XX Congreso Argentino de Ciencias de la Computación, Universidad Nacional de La Matanza, San Justo, Octubre 20-24, 2014, ISBN 978-987-3806-05-6, pp. 771-780.

Abstract

Software Metrics have been used since the 70s, their purpose is to measure different software attributes, such as complexity and maintainability, to name a few. Software Metrics help programmers obtain valuable information about programs. That information is essential when working with legacy systems. Scientists have been producing Fortran programs for the last six decades, and some of those programs became legacy years ago. We have implemented a set of well known software metrics for Fortran into a widely used IDE (Integrated Development Environment) by means of AST (Abstract Syntax Tree). This integration allows developers to obtain software metrics from their source code while they are programming.

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

10) "Inversión de prioridades: análisis de desempeño", Santiago Alessandri, Luciano Iglesias, Fernando Romero, Fernando G. Tinetti, CACIC 2014, XX Congreso Argentino de Ciencias de la Computación, Universidad Nacional de La Matanza, San Justo, Octubre 20-24, 2014, ISBN 978-987-3806-05-6, pp. 334-341.

Abstract

La planificación de tareas es el punto crucial de un sistema de tiempo real. Esta función es llevada a cabo por el planificador del sistema operativo, diseñado para poder cumplir con las restricciones temporales de las tareas a realizar, teniendo en cuenta sus valores de prioridad. Cuando hay recursos compartidos por estas tareas, se puede producir el efecto llamado inversión de prioridades. En este trabajo se analiza este efecto y se evalúan las soluciones implementadas para este problema en el Sistema Operativo de Tiempo Real GNU/Linux con parche RT-PREEMPT [1]. Además, se evalúa otro mecanismo, conocido como restauración de recursos, que no está implementado en GNU/Linux.

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

11) "Climate Models: Challenges for Fortran Development Tools", Mariano Méndez, Fernando G. Tinetti, Jeffrey L. Overbey, Second International Workshop on Software Engineering for High Performance Computing in Computational Science and Engineering, November 21, 2014, IEEE Computer Society, New Orleans, USA, pp. 6-12.

Abstract

Climate simulation and weather forecasting codes are among the most complex examples of scientific software. Moreover, many of them are written in Fortran, making them some of the largest and most complex Fortran codes ever developed. For companies and researchers creating Fortran development tools—IDEs, static analyzers, refactoring tools, etc.— it is helpful to study these codes to understand the unique challenges they pose. In this paper, we analyze 16 well-known global climate models and collect several syntactic metrics, including lines of code, McCabe cyclomatic complexity, presence of preprocessor directives, and numbers of obsolescent Fortran language constructs. Based on these results, we provide

some guidelines for people wishing to develop software development tools for Fortran. Notably, such tools must scale to million-line code bases, they must handle constructs that the ISO Fortran standard has deemed obsolescent, and they must work fluently in the presence of C preprocessor directives.

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

12) "Sistemas de Tiempo Real Distribuidos Usando Microcontroladores", Fernando Romero, Mariano Mendez, Luciano Iglesias, Diego Encinas, Armando E. De Giusti, Fernando G. Tinetti, XVII Workshop de Investigadores en Ciencias de la Computación, 16 y 17 de Abril de 2015, Universidad Nacional de Salta, Salta, Argentina, ISBN 978-987-633-134-0.

Abstract

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) haciendo hincapié en los aspectos de simulación, planificadores y comunicaciones, en el contexto de implementación con microcontroladores.

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

13) "Infraestructura de Cloud Computing", María Antonia Murazzo, Fernando G. Tinetti, Nelson R. Rodriguez, Miguel Guevara, XVII Workshop de Investigadores en Ciencias de la Computación, 16 y 17 de Abril de 2015, Universidad Nacional de Salta, Salta, Argentina, ISBN 978-987-633-134-0.

Abstract

El avance notable de tecnologías como la computación distribuida, Internet y grid computing, han posibilitado que Cloud Computing forme parte de un nuevo modelo de computación y de negocios. Cloud Computing está transformando los modos tradicionales de cómo las organizaciones utilizan y adquieren los recursos de Information Technology (IT). Representa un nuevo tipo del valor de la computación en red, gracias a la entrega de mayor eficiencia, escalabilidad masiva y más rápido y fácil desarrollo de software.

En la actualidad las organizaciones han comenzado a montar sus propias infraestructuras de cloud con el objeto de contar con las ventajas de Cloud Computing sumado a la privacidad de los datos que consideren estratégicos y adecuada a las necesidades del área.

Contar con una infraestructura de cloud asegura una infraestructura escalable y personalizada que permita a los usuarios explotar el potencial de la virtualización en función a las necesidades de las aplicaciones que se van a desarrollar.

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

14) "Software de Base, Métricas y Aplicaciones en Arquitecturas Multiprocesador orientadas a Cómputo de Altas Prestaciones", Armando E. De Giusti, Fernando G. Tinetti, Marcelo Naiouf, Franco Chichizola, Laura De Giusti, Horacio Villagarcía Wanza, Diego Montezanti, Fernando Emmanuel Frati, Adrián Pousa, Ismael Pablo Rodríguez, Sebastian Eguren, Mónica Denham, Luciano Iglesias, Mariano Mendez,

XVII Workshop de Investigadores en Ciencias de la Computación, 16 y 17 de Abril de 2015, Universidad Nacional de Salta, Salta, Argentina, ISBN 978-987-633-134-0.

Abstract

Esta línea de Investigación está dentro del proyecto 11/F018: "Arquitecturas multiprocesador en HPC: Software de base, Métricas y Aplicaciones acreditado por el Ministerio de Educación y de proyectos específicos apoyados por organismos nacionales e internacionales.

El III-LIDI forma parte del Sistema Nacional de Cómputo de Alto Desempeño (SNCAD) del MINCYT y en esta línea de I/D 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).

En la Facultad de Informática de la UNLP (a partir del equipo del proyecto) se han incorporado asignaturas optativas en la currícula de grado de las carreras de Licenciatura en Informática, Licenciatura en Sistemas e Ingeniería en Computación relacionadas con Cloud Computing, Big Data y Programación sobre GPGPUs. Además, la Facultad aprobó y financia el proyecto "HPC y Cloud Computing. Aplicaciones".

Se participa en iniciativas como el Programa IberoTIC de intercambio de Profesores y Alumnos de Doctorado en el área de Informática, así como el desarrollo de la Maestría y Especialización en Computación de Altas Prestaciones, acreditadas por CONEAU.

Por último, se tiene financiamiento de Telefónica de Argentina en Becas de grado y posgrado y se ha tenido el apoyo de diferentes empresas (IBM, Microsoft, Telecom, INTEL, AMAZON AWS) en las temáticas de Cloud Computing y Big Data.

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

15) "Hotspot: a Framework to Support Performance Optimization on Multiprocessors", Fernando G. Tinetti, Andres More, Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'15), Volume I/II, Editors: Hamid R. Arabnia, Hiroshi Ishii, Kazuki Joe, Hiroaki Nishikawa, Hayaru Shouno, Lou D'Alotto, George A. Gravvanis, George Jandieri, Georgios Sirakoulis, Ashu M. G. Solo, William Spataro, Fernando G. Tinetti, Giuseppe A. Trunfio, CSREA Press, July 27-30, 2015, Las Vegas Nevada, USA, ISBN Vol I: 1-60132-400-6, ISBN Vol II: 1-60132-401-4, ISBN Set: 1-60132-402-2, pp. 171-176.

Abstract

High Performance Computing (HPC) programs are usually developed by domain specialists, and they are not always experts on performance optimization. Application domain specialists rely on a systematic and unattended (and/or partially guided) method to support performance analysis and optimization. This paper presents a framework that simplifies the job, including several case studies as a validation step and as a proof of concept. The main goal is to make the data recollection task straight-forward, allowing to focus on experimentation and optimization. The framework gathers context information about the program and the underlying system, detailing scaling behavior, execution profile, resource utilization, and bottlenecks. In a nutshell, this work contributes with a performance report generator for OpenMP programs.

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

16) "Hadoop Scalability and Performance Testing in Heterogeneous Clusters", Fernando G. Tinetti, Ignacio Real, Rodrigo Jaramillo, Damian Barry, Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'15), Volume I/II, Editors: Hamid R. Arabnia, Hiroshi Ishii, Kazuki Joe, Hiroaki Nishikawa, Hayaru Shouno, Lou D'Alotto, George A. Gravvanis, George Jandieri, Georgios Sirakoulis, Ashu M. G. Solo, William Spataro, Fernando G. Tinetti, Giuseppe A. Trunfio, CSREA Press, July 27-30, 2015, Las Vegas Nevada, USA, ISBN Vol I: 1-60132-400-6, ISBN Vol II: 1-60132-401-4, ISBN Set: 1-60132-402-2, pp. 441-446.

Abstract

This paper aims to evaluate cluster configurations using Hadoop in order to check parallelization performance and scalability in information retrieval. This evaluation will establish the necessary capabilities that should be taken into account specifically on a Distributed File System (HDFS: Hadoop Distributed File System), from the perspective of storage and indexing techniques, and query distribution, parallelization, scalability, and performance in heterogeneous environments. The software architecture will be designed and evaluated as either centralized or distributed, and the relevant experiments will be carried out establishing the performance improvement for each architecture.

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

17) "Evaluación de Sistemas Operativos de Tiempo Real sobre Microcontroladores", Santiago Medina, Martín Pi Puig, Juan Manuel Paniego, Matías Dell'Oso, Fernando Romero, Armando De Giusti, Fernando G. Tinetti, CACIC 2015, XXI Congreso Argentino de Ciencias de la Computación, Universidad Nacional del Nordeste de la Prov. de Buenos Aires, Junín, Octubre 5-9, 2015, ISBN 978-987-3806-05-6, pp. 971-979.

Abstract

Se presentan mediciones sobre diferentes sistemas operativos que soportan las características de tiempo real, instalados sobre un sistema de microcontrolador. Dichas evaluaciones caracterizan los posibles tiempos de respuesta, es decir, un límite que determina el tipo de aplicación en la cual pueden ser utilizados, dentro del marco del desarrollo de aplicaciones de Sistemas Embebidos de Tiempo Real. Una de las principales métricas es el tiempo de latencia, el cual representa el tiempo que transcurre desde que se produce una entrada hasta que se emite la salida de respuesta. Luego, no solo se evalúa este tiempo, sino la variabilidad del mismo, ya que en tiempo real es fundamental el determinismo. Si bien no es la única condición que debe cumplir un sistema operativo para ser considerado de tiempo real, es un requisito necesario. Los tiempos de latencia y su variabilidad, determinan las cotas de requerimientos temporales que brinda el sistema.

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

18) Capítulo de Libro

Jorge R. Finocchietto, Patricia M. Pesado, (Eds.), Computer Science & Technology Series. XIX Argentine Congress of Computer Science. Selected Papers, Editorial de la Universidad de La Plata (EduLP), ISBN 978-987-1985-49-4. Octubre 2014, "N-

Body Simulation Using GP-GPU: Evaluating Host/Device Memory Transference Overhead", Sergio Martin, Fernando G. Tinetti, Nicanor, Casas, Graciela De Luca, Daniel Giulianelli, pp. 43-54.

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: Dirección, desarrollo de la idea básica del trabajo, implementación de algoritmos y experimentos asociados al trabajo y redacción del artículo.

19) Capítulo de Libro

Guillermo E. Feierherd, Patricia Pesado, Osvaldo M. Sposito, Computer Science & Technology Series. XX Argentine Congress of Computer Science. Selected papers, Editorial de la Universidad Nacional de La Plata (EDULP), ISBN: 978-987-1985-71-5, Octubre 2015, "Integrating Software Metrics for Fortran Legacy into an IDE", Mariano Mendez, Fernando G. Tinetti, pp. 125-134.

Abstract

Software Metrics have being used since the 70s, their purpose is to measure different software attributes, such as complexity and maintainability, to name a few. Software Metrics help programmers obtain valuable information about programs. That information is essential when working with legacy systems. Scientists have been producing Fortran programs for the last six decades, and some of those programs became legacy years ago. We have implemented a set of well known software metrics for Fortran into a widely used IDE (Integrated Development Environment) by means of AST (Abstract Syntax Tree). This integration allows developers to obtain software metrics from their source code while they are programming.

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

8.2 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.*

Synthetic aperture radar signal processing in parallel using GPGPU", Mónica Denham, Javier Areta, Fernando G. Tinetti, aceptado para su publicación en The Journal of Supercomputing, Springer International Publishing AG.

Abstract

In this work an efficient parallel implementation of the Chirp Scaling Algorithm for Synthetic Aperture Radar processing is presented. The architecture selected for the implementation is the general purpose graphic processing unit, as it is well suited for scientific applications and real-time implementation of algorithms. The analysis of a first implementation led to several improvements which resulted in an important speed-up. Details of the issues found are explained, and the performance improvement of their correction explicitly shown.

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

8.3 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

8.4 TRABAJOS TERMINADOS Y AUN NO ENVIADOS PARA SU PUBLICACION.

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

N/A

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

"Procesamiento Distribuido y Procesamiento Paralelo: Similitudes, Diferencias y Areas de Investigación", Invitado por la Facultad de Ciencias Exactas y Naturales y Agrimensura, UNNE y el Instituto de Modelado e Innovación Tecnológica (IMIT), UNNE. Fecha: 7/3/2014. Lugar: Facultad de Ciencias Exactas y Naturales y Agrimensura, UNNE, Corrientes, Argentina.

"HPC: Aplicaciones y Software Heredado - HPC: Aplicaciones y Software", Invitado por la Subsecretaría de Coordinación Institucional Ministerio de Ciencia, Tecnología e Innovación Productiva. Fecha: 5-6 Agosto de 2014. Lugar: Centro de Computación de Alto Desempeño de la Universidad Nacional de Córdoba, Argentina.

"¿Grandes Datos en la Nube?", Facultad de Informática de la UNLP, en el marco de las II Jornadas de Cloud Computing, realizadas entre el 30 de junio y el 4 de Julio de 2014.

"Cómputo de Alto Rendimiento: Computación Distribuida y Paralela", Invitado por la Facultad de Ciencias Exactas y Naturales y Agrimensura de la UNNE, Corrientes, para dictar el curso "Intel Galileo", 20 de Noviembre de 2015.

8.6 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, Julio 2014, Technical Report LS-01-2014, July 2014. Disponible en <http://fernando.usr.me>.

Título: "Climate Models: A Software Engineering Approach", Fernando G. Tinetti, Mariano Méndez.

9. TRABAJOS DE DESARROLLO DE TECNOLOGÍAS.

9.1 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

9.2 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

9.3 PROYECTOS POTENCIALMENTE TRANSFERIBLES, 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

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

N/A

9.5 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

10. 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

11. PUBLICACIONES Y DESARROLLOS EN:

11.1 DOCENCIA

Página web y apuntes o comentarios de los temas de la asignatura Programación Distribuida, Cursadas 2014 y 2015, Licenciatura en Informática, Fac. de Informática, UNLP, que se pueden acceder desde

<https://lidi.info.unlp.edu.ar/~fernando/clases/distr/distr.html>

Página web y apuntes o comentarios de los temas de la asignatura Computabilidad y Complejidad, Cursadas 2014 y 2015, Licenciatura en Informática, Fac. de Informática, que se pueden acceder desde

<https://lidi.info.unlp.edu.ar/~fernando/clases/cyc/cyc.html>

Página web y apuntes o comentarios de los temas de la asignatura Bases de Datos distribuidas, Cursadas 2014 y 2015, Licenciatura en Informática, Fac. de Informática, que se pueden acceder desde

<https://lidi.info.unlp.edu.ar/~fernando/clases/bdd/bdd.html>

Página web y apuntes o comentarios de los temas de la asignatura Taller de Proyecto II, Cursadas 2014 y 2015, Ingeniería en Computación, Fac. de Informática-Fac. de Ingeniería, que se pueden acceder desde

<https://lidi.info.unlp.edu.ar/~fernando/clases/tpII/tpII.html>

11.2 DIVULGACIÓN

N/A

12. 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 "Arquitecturas Multiprocesador en HPC: Software de Base, Métricas y Aplicaciones" (Dirección: Armando 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 2014 – Diciembre 2018.

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.

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.

13. 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: Maria Antonia Murazzo.

Postgrado: Especialización en Redes de Datos.

En curso, no finalizada.

Postgraduado: Leopoldo José Rios

Postgrado: Especialización en Ingeniería de Software.

En curso, no finalizada.

Postgraduado: Mariano Méndez

Postgrado: Doctorado en Informática

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.

14. 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.*

Coordinador del área "Procesamiento de señales y Sistemas de Tiempo Real" del WICC 2015 - Workshop de Investigadores en Ciencias de la Computación

(<http://www.unsa.edu.ar/wicc2015/?Coordinadores>) realizado en Salta, el 15 y 16 de Abril de 2015.

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

- 16. 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° 833/14, por la suma de \$7000.

Subsidio Institucional para Investigadores de la Comisión de Investigaciones Científicas de la Provincia de Buenos Aires, otorgado Resolución N° 1266/14, por la suma de \$8750.

- 17. OTRAS FUENTES DE FINANCIAMIENTO.** *Describir la naturaleza de los contratos con empresas y/o organismos públicos.*
N/A

- 18. DISTINCIONES O PREMIOS OBTENIDOS EN EL PERIODO.**
N/A

- 19. 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: Facultad de Informática, UNLP.

Tarea/Gestión: Director de Convenios de Cooperación Científica y de Postgrado.

Porcentaje aproximado de tiempo: 1%.

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

Profesor de la Asignatura: Taller de Proyecto II, Ing. en Computación, Facultad de Informática y Facultad de Ingeniería de la UNLP.

Porcentaje aproximado de tiempo: 5%.

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: 5%.

Profesor del Curso de Postgrado 2015: 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%.

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

N/A

22. TÍTULO 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.*

Título: Procesamiento Paralelo y Distribuido

Se propone la continuidad de las tareas desarrolladas hasta el presente informe, de hecho continuando la labor en las tres líneas de trabajo mencionadas anteriormente y dentro de las cuales ya se han desarrollado trabajos que dieron lugar a sus correspondientes publicaciones (también enumeradas anteriormente):

a) HPC (High Performance Computing) aplicado a cómputo numérico/simulaciones numéricas. Se propone continuar con la línea de trabajo de optimización y paralelización de modelos numéricos ya desarrollados y posiblemente en producción. Esta es una de las áreas de mayores requerimientos en cuanto a la adaptación (optimización) a las nuevas arquitecturas de cómputo. El software numérico (también denominado de cómputo científico) desarrollado y en producción en la actualidad no se pensó para las arquitecturas de cómputo actuales. Por un lado, el objetivo inicial fue la implementación de procesos físicos o representaciones de procesos que debían y deben ser simulados. Estas implementaciones, por lo tanto, no se pensaron "para" las arquitecturas de cómputo sino "para" los sistemas que se debían simular. Por otro lado, los modelos que inicialmente se optimizaron (o aún paralelizaron, que son muy pocos en proporción) difícilmente podían tener en cuenta hace diez o más años las arquitecturas de cómputo actuales, desde la paralelización a nivel de instrucciones (ILP: Instruction Level Parallelism) hasta los denominados "multicores" y "manycores" y los clusters de multi y many cores. Como parte del problema general de optimización y paralelización de programas de cómputo científico, se pretende avanzar en varios aspectos:

a.1) El aspecto fundamental es efectivamente en el de obtener mejoras (reducciones) en los tiempos de cómputo propiamente dichos. Como parte de este avance, si bien es estándar que sea llevado a cabo "caso por caso" (programa por programa), se pretende avanzar en los patrones o las formas de optimización y paralelización de cómputo de acuerdo al aprovechamiento del hardware de procesamiento en un core de ejecución, normamente por medio de ILP y optimización de uso de la jerarquía de memoria. Avanzando en la posibilidad de mejora, se pretende seguir avanzando en la paralelización de cómputo con memoria compartida, tanto en lo referido a multicores como many cores como las GPU (Graphic Processing Units) utilizadas para HPC. Y en el modelo de cómputo paralelo más distribuido, también se pretende avanzar con el cómputo en clusters, sean de multi o many cores.

a.2) Caracterización/documentación de software científico heredado (legacy). Además de la propia complejidad del proceso y del cómputo involucrado, uno de los mayores inconvenientes para la optimización y paralelización de código heredado es la falta de documentación o la documentación obsoleta (que no refleja el estado actual del software). En este sentido ya se han hecho muchos avances y se pretende continuar, dado que, por ejemplo, se han implementado múltiples métricas (como por ejemplo las detalladas en el capítulo de libro 15 de la enumeración de publicaciones dada en uno de

los puntos anteriores de este informe), pero no se han llegado a evaluar la importancia relativa de cada una de ellas (en qué "orden" deberían ser tenidas en cuenta) ni cómo resolver los casos en los cuales las métricas exponen problemas importantes del software.

a.3) Aunque se ha avanzado en la evaluación de rendimiento, aún no se ha llegado a una propuesta mínima de optimizaciones o paralelizaciones posibles a partir de evaluación de rendimiento por perfilado (profiling) clásico o por la instrumentación con bibliotecas como PAPI (Performance Application Programming Interface).

b) Procesamiento de/relacionado con Big Data. A diferencia de los programas científicos clásicos, este tipo de procesamiento se presenta como completamente nuevo. En realidad aún no está bien definido como tal, y lo más conocido suele ser el resultado de técnicas no mucho más allá de las estándares de estadística, como las correlaciones. Lamentablemente desde hace algunos años se viene haciendo demasiado énfasis en los casos de "éxito", casi a nivel promoción/propaganda y no se divulga tanto el procesamiento real/de fondo, al menos a nivel masivo. Actualmente, dos de las mayores áreas de desarrollo del procesamiento relacionado con Big Data avanzan alrededor de los algoritmos de procesamiento más básicos (como el mencionado del área de estadística) y visualización. Dado que la visualización tiene que ver con aspectos más de interpretación humana que de procesamiento en sí mismo, no se lo considera de importancia desde el punto de vista de HPC (no es que no sea de importancia en sí mismo, hasta el momento no parecen haber algoritmos que relacionen el cómputo de HPC con la visualización más allá del propio volumen de datos involucrados).

Se considera importante avanzar en áreas completamente relacionadas con HPC, y quizás agregando aspectos específicos de HPC como el de entrada/salida y formato de datos, que no necesariamente han sido muy estudiados en el ámbito de cómputo científico, que normalmente se "reduce" a cómputo numérico de punto flotante de la mayor precisión posible (cantidad de bytes, usualmente resuelto en norma IEEE 754 de precisión doble) y con almacenamiento de "checkpoints" para retomar el procesamiento a partir de puntos predefinidos del avance de las simulaciones o algún tipo de cómputo en particular. En este contexto, todo parece indicar que el procesamiento sobre de bases de datos no estructuradas (usualmente denominadas bases de datos NoSQL) tiene un gran potencial. Dentro de las alternativas de procesamiento con bases de datos NoSQL, el ambiente provisto por Hadoop es uno de los más completos y más popular a la vez.

Dentro de las funcionalidades de Hadoop son muy atractivas las características de almacenamiento y tolerancia a fallas provista por su sistema de archivos. Como punto de partida, el sistema de archivos es distribuido y replicado, lo cual le da una base muy ventajosa tanto en lo referente a tolerancia a fallas (si un nodo falla el sistema sigue funcionando con las réplicas) como en lo referente a paralelización (se pueden hacer diferentes cálculos en nodos diferentes aún sobre datos replicados). Dentro de las tareas a llevar a cabo es necesaria alguna caracterización de rendimiento o aún de penalización de rendimiento de todo el sistema en función de los factores de replicación posibles. Al día de hoy solamente se afirma que el sistema de archivos de Hadoop es distribuido y replicado, pero no se dan mayores precisiones de cómo, por qué y qué efecto sobre el rendimiento tienen diferentes factores de replicación o la relación con el procesamiento. Específicamente relacionado con el procesamiento en Hadoop, se han publicado múltiples casos de éxito o "bondades" del modelo map-reduce, pero casi ninguno avanza en datos técnicos sobre rendimiento o aún sobre los límites de aplicación de ese modelo de procesamiento, que dada su simplicidad por lo menos tiene una gran probabilidad de ser imposible de aplicar en algunos problemas. Sin embargo, han habido tantas publicaciones de casos de éxito (al menos a nivel de publicaciones "masivas", que a priori parecería que se pueden resolver todos los problemas con este modelo. Es necesaria, por lo tanto, una caracterización realista en

cuanto a qué problemas efectivamente se pueden resolver y qué problemas no, o las características de los problemas que "facilitarían" o "imposibilitarían" su aplicación.

La popularidad de los sistemas de cómputo NoSQL no debe entenderse a nivel "casos de éxito" sino en lo que significa el desarrollo open source. En este sentido, tanto Hadoop como otros ambientes de cómputo sobre bases de datos NoSQL como MongoDB son especialmente populares entre desarrolladores, lo cual les da una dinámica muy especial y atractiva en cuanto a la resolución de problemas o incluso bugs existentes como a la incorporación de nueva funcionalidad.

c) Procesamiento para problemas de tiempo real/robótica. Aunque la problemática de tiempo real y robótica no ha sido tradicionalmente asociada a HPC, sí está naturalmente relacionada con optimización, procesamiento distribuido y aún con procesamiento paralelo. La propia idea de optimización como el uso más eficiente posible de los recursos disponibles es muy cercana a los problemas de tiempo real y su aplicación en robótica: normalmente se debe optimizar al máximo para responder en tiempo y forma a para responder/reaccionar a eventos cuya respuesta debe ser provista con un límite máximo de tiempo. En robótica es muy usual encontrar múltiples casos en los cuales deben ser tenidos en cuenta los tiempos de respuesta, desde casos tan sencillos como los de PWM (Pulse Width Modulation) para un motor DC para las ruedas de un automóvil, por ejemplo, hasta casos tan complejos donde se deben calcular los movimientos de un brazo robótico, por ejemplo, teniendo en cuenta cálculos odométricos necesarios (movimientos más "naturales", optimización de fuerzas de motores, etc.). Si además se consideran robots con mayor complejidad como por ejemplo los que incorporan cálculos de sensores (ej. GPS,) y/o procesamiento en tiempo real de video, la conexión con cómputo de alto rendimiento es mucho más evidente. Se podría decir que el cómputo distribuido es casi imposible de independizar de las aplicaciones de robótica. Desde los problemas de menor nivel de abstracción hasta los más complejos, más relacionados con los sistemas robóticos como un todo. En el caso de los menores niveles de abstracción, se podría decir que cada sensor o actuador que se utiliza en robótica tiene algún tipo de control o procesamiento local, independiente (en ese sentido "distribuido") de todo lo demás que se debe procesar en el robot. En el caso de mayores niveles de abstracción, los sistemas con más de un robot, cada uno de los robots involucrados necesariamente se deben coordinar para llevar adelante una o varias tareas a llevar a cabo. En el sistema distribuido más sencillo, los robots suelen estar conectados a un sistema de procesamiento que es capaz de alterar o incluso detener/inicial algún tipo de tarea, con lo cual como mínimo los robots suelen estar conectados y controlados por algún tipo de sistema de cómputo externo, lo que se podría asociar a un sistema distribuido en cuanto a comunicaciones y coordinación se refiere. En este contexto, o teniendo en cuenta estas definiciones, se considera adecuado el aprovechamiento de la experiencia en de optimización, paralelización y procesamiento distribuido para el ámbito de la robótica. Aunque se podría considerar en un estadio muy inicial, la conexión de esta temática con una de las asignaturas de la carrera Ingeniería en Computación (Taller de Proyecto II, del último cuatrimestre de la carrera) de la que es profesor responsable quien suscribe es un ámbito más que propicio para desarrollar e implementar ideas con estudiantes que están próximos a insertarse en el ámbito laboral. Todo desarrollo de implementación e investigación tiene un ámbito inicial de evaluación con estudiantes próximos a ser profesionales y la transferencia de estas experiencias enriquece tanto a los estudiantes como al ámbito de investigación de quien suscribe.

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:
- a. Una copia en papel A-4 (puntos 1 al 22).
 - b. 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".
 - c. Informe del Director de tareas (en los casos que corresponda), en sobre cerrado.
- B. Envío por correo electrónico:
- a. Se deberá remitir por correo electrónico a la siguiente dirección: ininvest@cic.qba.gob.ar (puntos 1 al 22), en formato .doc zipeado, configurado para papel A-4 y libre de virus.
 - b. 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.
- C. Sistema SIBIPA:
- a. Se deberá peticionar el informe en la modalidad on line, desde el sitio web de la CIC, sistema SIBIPA (ver instructivo).

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.