

Extensión de CluSim: Simulación de la arquitectura tolerante a fallos RADIC

Cecilia M. Lasserre, Nilda M. Pérez Otero, Abigaíl R. N. Verazay

Grupo de Ingeniería de Software (GIS) – Facultad de Ingeniería – Universidad Nacional de Jujuy
Italia esq. Martiarena – San Salvador de Jujuy – CP 4600 – TE 54+388+4221591 CEL
54+388+154870436
classerre@fi.unju.edu.ar – {nilperez, abigailrn}@gmail.com

Resumen

Los sistemas de Cómputo de Altas Prestaciones se utilizan para desarrollar software en una gran cantidad de campos. Es evidente el creciente predominio e impacto de las aplicaciones del Cómputo de Altas Prestaciones (*High Performance Computing* - HPC) en la sociedad moderna. Sin embargo, la presencia de fallos en el hardware o software de computadores paralelos hace necesario el uso de mecanismos tolerantes a fallos para asegurar que las aplicaciones finalicen exitosamente. Para ello se ha desarrollado RADIC, una arquitectura transparente, descentralizada, flexible y escalable para tolerancia a fallos que provee alta disponibilidad en sistemas de paso de mensajes. La falta de disponibilidad física de grandes clusters y el hecho de estar ligado a una implementación específica de MPI como base, son las principales dificultades con las que se encontraron los desarrolladores de RADIC. Como una solución a estos problemas el presente proyecto de investigación propone el desarrollo de un entorno de simulación para RADIC basado en OMNeT++, a partir de CLUSIM (Simulador de clusters basado en OM-Net++).

Palabras claves: Cómputo de Altas Prestaciones, HPC, Tolerancia a Fallos, OM-Net++, Simulación, RADIC.

Contexto

Este proyecto da continuidad a una línea de investigación iniciada en el año 2008 en el marco de los proyectos acreditados por

la Universidad Nacional de Jujuy denominados: APLICACIONES DEL CÓMPUTO DE ALTAS PRESTACIONES y SISTEMAS DE CÓMPUTO DE ALTAS PRESTACIONES CON ALTA DISPONIBILIDAD: EVALUACIÓN DE LA PERFORMANCE EN DIFERENTES CONFIGURACIONES.

El proyecto se encuentra acreditado y financiado por la Secretaría de Ciencia y Técnica y Estudios Regionales de la Universidad Nacional de Jujuy (SECTER-UNJu) y cuenta con el asesoramiento del Mg. Germán Montejano (Universidad Nacional de San Luis) y de la Dra. Jussara de Marques Almeida (Universidad Federal de Minas Gerais).

Introducción

Los sistemas de HPC se usan para desarrollar software en una gran cantidad de campos, incluyendo física nuclear, simulación de accidentes, procesamiento de datos de satélites, dinámica de fluidos, modelado del clima, bioinformática y modelado financiero. La gran variedad de organizaciones científicas, gubernamentales y comerciales presentes en esta lista ilustra el creciente predominio e impacto de las aplicaciones de HPC en la sociedad moderna [Carver, 2007].

La presencia de fallos en el hardware o software de computadores paralelos genera nuevas necesidades en el uso de mecanismos tolerantes a fallos para asegurar que las aplicaciones finalicen exitosamente. Recientemente, algunos centros de supercomputadores han publicado estadísticas acerca de fallos [Cappello, 2009]. El Labo-

ratorio Nacional de Los Álamos (LANL) provee información muy detallada con una descripción de 23.000 eventos que causaron paradas en la aplicación. Los datos de LANL corresponden a 22 clusters con hasta 4.096 CPUs, durante un período de 10 años. Esto representa alrededor de 5.000 nodos de cómputo y un total de 24.000 de CPUs (algunos nodos son multiprocesador). En un análisis de estos datos realizados por Schroeder y Gibson [Schroeder & Gibson, 2007] se puede observar que el número de fallos por año puede exceder los 1.000 para algunos sistemas. Para estos últimos, tres fallos por día implican que aplicaciones que utilizan todos los nodos de cómputo y demoran más de 8 horas tienen pocas posibilidades de finalizar correctamente.

Las estadísticas anteriores llevan a plantear la necesidad de implementar un sistema tolerante a fallos para HPC. Para alcanzar alta disponibilidad, tal sistema de tolerancia a fallos debe proveer una recuperación y detección de fallos automática y transparente. Además, para una tolerancia a fallos proactiva es también deseable que se puedan realizar tareas de mantenimiento preventivo, como, por ejemplo, reemplazar máquinas susceptibles a fallos sin interrupciones al sistema. [Santos *et al.*, 2008]

Considerando estos aspectos, Duarte y colegas [Duarte *et al.*, 2006; Duarte *et al.*, 2007] han propuesto y desarrollado RADIC (*Redundant Array of Distributed Independent Fault Tolerance Controllers*), una arquitectura transparente, descentralizada, flexible y escalable para tolerancia a fallos que provee alta disponibilidad en sistemas de paso de mensajes que basa su operación en el mecanismo de *rollback-recovery* basado en protocolo *log* pesimista. En tal protocolo, se realizan *checkpoints* regularmente y todos los mensajes recibidos por cada proceso de la aplicación deben ser guardados por el receptor para poder volver a utilizarlos en caso de fallo.

En la actualidad, es cada vez más frecuente el uso de modelos de simulación computacional en HPC, ya sea como ayuda al dise-

ño y modelado de prestaciones [Denzel *et al.*, 2008], como para explorar arquitecturas o aplicaciones [Hammond *et al.*, 2009; Minkenberg & Rodriguez, 2009] o como una herramienta de predicción de tráfico [Tikir *et al.*, 2009].

En el año 2010, [Valdiviezo *et al.*, 2010; Pérez Ibarra *et al.*, 2010; Lasserre *et al.*, 2011, García *et al.*, 2011] el GIS desarrolló el simulador CluSim, un simulador de clusters basado en OMNeT++, que al presente permite parametrizar la configuración de un cluster y hace posible evaluar y predecir el impacto en el rendimiento de diferentes configuraciones para aplicaciones tipo Master/Worker.

De igual modo, el realizar un simulador de RADIC permitiría disponer de una herramienta para analizar cómo afectan las características y parámetros de configuración de RADIC a la aplicación y al sistema. Por ejemplo, cuál sería el número ideal de nodos *spare* y su ubicación ideal dentro del cluster, investigar si es posible alcanzar mejores resultados distribuyendo los nodos *spare* de acuerdo con determinados criterios (nivel de degradación aceptable, límites de memoria de un nodo, topología de red), investigar acerca de las posibles políticas a usarse en las tareas de reemplazo de nodo, etc.

Líneas de investigación y desarrollo

- Incorporar la arquitectura tolerante a fallos RADIC a CluSim.
- Extender la funcionalidad de CluSim tolerante a fallos a aplicaciones paralelas tipo SPMD, Pipeline y Divide/Conquer.

Resultados y Objetivos

El proyecto ha comenzado en febrero del año en curso, de modo que, a la fecha aún no se tienen resultados para presentar.

El objetivo principal del proyecto es desarrollar un framework basado en OMNeT++ donde puedan simularse distintos esquemas de tolerancia a fallos y que permita

implementar los módulos de la arquitectura de RADIC de forma parametrizable y configurable. De esta manera, el simulador permitirá un mejor análisis y comprensión de las funciones de RADIC interactuando con el sistema de cómputo y con las aplicaciones. Es decir, el simulador:

- permitirá el desarrollo y prueba de nuevas políticas en sistemas que no están disponibles físicamente,
- permitirá el análisis del comportamiento del sistema (desbalanceo de carga, cuellos de botellas causados por fallos) mediante la inyección de diferentes patrones de fallos,
- ayudará en el proceso de toma de decisiones (por ejemplo, si se detecta un cuello de botella: cuántos nodos *spare* serán necesarios, dónde ubicarlos, cuál es la influencia del mapeo entre protectores y observadores) permitiendo así la evaluación de diferentes configuraciones de RADIC.

Formación de Recursos Humanos

El equipo de trabajo de la línea de I/D presentada está formado por:

Director: Lasserre, Cecilia María

Co-Director: Pérez Otero, Nilda María

Integrantes:

- Pérez Ibarra, Claudio Marcelo
- García, Adelina
- Verazay, Abigaíl Roxana Noemí
- Quispe, Gloria Lola
- Ovando, Pablo
- Martínez, Jorge
- Córdoba, Rafaela
- Argañaraz Azua, Fabio

Como se muestra a continuación algunos de los miembros del equipo realizan estudios de postgrado o desarrollan el proyecto de final de carrera de Ingeniería en Informática en temáticas afines al proyecto:

Tesis de maestría en curso: 1

Trabajos finales de especialización: 2

Especialización en curso en la UNLP: 1

Proyectos de final de carrera: 1

Referencias

- [Cappello, 2009] Cappello, F. (2009). Fault Tolerance in Petascale/ Exascale Systems: Current Knowledge, Challenges and Research Opportunities. *International Journal of High Performance Computing Applications*, 23(3):212–226.
- [Carver, 2007] Carver, J. C. Third international workshop on software engineering for high performance computing (HPC) applications. In *ICSE COMPANION '07: Companion to the proceedings of the 29th International Conference on Software Engineering*, p. 147, Washington, DC, USA. IEEE Computer Society.
- [Denzel *et al.*, 2008] Denzel, W. E.; Li, J.; Walker, P. & Jin, Y. A framework for end-to-end simulation of high-performance computing systems. In *Simutools '08: Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, pp. 1--10, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [Duarte, 2007] Duarte, A. RADIC: A Powerful Fault-Tolerant Architecture. PhD thesis, Departament d'Arquitectura de Computadors i Sistemes Operatius. Universitat Autònoma de Barcelona, <http://www.tdx.cat/TDX-1126107-101303>.
- [Duarte *et al.*, 2006] Duarte, A.; Rexachs, D. & Luque, E. Increasing the cluster availability using RADIC. *Cluster Computing, 2006 IEEE International Conference on*, vol., no., pp.1-8, 25-28 Sept. 2006
- [Duarte *et al.*, 2007] Duarte, A.; Rexachs, D. & Luque, E. (2007). Functional tests of the RADIC fault tolerance architecture. In *PDP*, pp. 278–287. IEEE Computer Society.
- [García *et al.*, 2011] García, A.; Pérez Otero, N. M.; Pérez Ibarra, C. M. y C.

- M. Lasserre. Simulación de Clusters: Integración de INET a CluSim. XVII Congreso Argentino de Ciencias de la Computación (CACIC 2011). ISBN 978-950-34-0756-1. Universidad Nacional de la Plata. Buenos Aires. pp. 367-373. Octubre 2011.
- [Hammond *et al.*, 2009] Hammond, S. D.; Mudalige, G. R.; Smith, J. A.; Jarvis, S. A.; Herdman, J. A. & Vadgama, A. Warpp: a toolkit for simulating highperformance parallel scientific codes. In Simutools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques, pp. 1--10, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [Lasserre *et al.*, 2011] Lasserre, C. M.; Pérez Ibarra, C. M.; Valdiviezo, L. M.; Verazay, A. R. N.; Quispe, G. L.; Nolasco, S. A.; Chosco, V. H. y N. M. Pérez Otero. Adaptación de CluSim a clusters heterogéneos. Investigaciones en Facultades de Ingeniería del NOA. Tomo 2 – 2011. ISSN 1853-7871. Editorial Científica Universitaria. San Fernando del Valle de Catamarca. pp 1053-1060. Octubre 2011.
- [Minkenberg & Rodriguez, 2009] Minkenberg, C. & Rodriguez, G. Tracedriven co-simulation of high-performance computing systems using OMNeT++. In Simutools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques, pp. 1--8, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [Pérez Ibarra *et al.*, 2010] Pérez Ibarra, C. M.; Valdiviezo L. M.; Pérez Otero N. M.; Liberatori, H. P.; Rexachs, D.; Luque E. y C. M. Lasserre. CLUSIM: Simulador de Clusters para aplicaciones de cómputo de altas prestaciones basado en OMNeT++. XVI Congreso Argentino de Ciencias de la Computación (CACIC 2010). ISBN 978-950-9474-49-9. Ed. Universidad de Morón. Buenos Aires. pp. 142-151. 2010. Octubre 2010.
- [Santos *et al.*, 2008] Santos, G.; Duarte, A.; Rexachs, D. & Luque, E. Providing non-stop service for message-passing based parallel applications with RADIC. In Luque, E.; Margalef, T. & Benitez, D., editores, EuroPar, volume 5168 of Lecture Notes in Computer Science, pp. 58–67. Springer.
- [Schroeder & Gibson, 2007] Schroeder, B. & Gibson, G. A. Understanding failures in petascale computers. Journal of Physics: Conference Series, 78:012022 (11pp).
- [Tikir *et al.*, 2009] Tikir, M. M.; Laurenzano, M. A.; Carrington, L. & Snaveley, A. PSINS: An open source event tracer and execution simulator for MPI applications. In Euro-Par '09: Proceedings of the 15th International Euro-Par Conference on Parallel Processing, pp. 135--148, Berlin, Heidelberg. Springer-Verlag.
- [Valdiviezo *et al.*, 2010] Valdiviezo, L. M.; Pérez Otero, N. M.; Pérez Ibarra, C. M. y C. M. Lasserre. Caracterización de una aplicación paralela con distintas configuraciones en CluSim. Investigaciones en Facultades de Ingeniería del NOA – 2010. ISSN 3367-5072. Ed. EdiUNJu. S. S. de Jujuy. pp. 499-504. Noviembre 2010.