

Performance de arquitecturas multiprocesador: técnicas de modelado y simulación en HPC y Cloud Computing.

Diego Encinas^{1,2}, Jimena Jara¹, Daniel Rosatto¹, Román Bond¹, Martín Morales^{1,3}

¹Instituto de Ingeniería y Agronomía - Universidad Nacional Arturo Jauretche

²Instituto de Investigación en Informática LIDI (III-LIDI) - Facultad de Informática - UNLP – Centro Asociado CIC

³ Unidad CodApli - Facultad Regional La Plata - UTN

dencinas@unaj.edu.ar, elchejime@gmail.com, danielrosatto@gmail.com,
roman.alejandro.b@gmail.com, martin.morales@unaj.edu.ar

Resumen

El objetivo de esta línea de investigación es el estudio de la performance de las arquitecturas multiprocesador y Cloud Computing a través de modelos de simulación. Enfocando a la obtención de herramientas que permitan predecir la eficiencia del sistema ante posibles escenarios y reconfigurar el sistema físico. Analizando los diferentes componentes del sistema que pueden influir en las prestaciones significativamente y pueden llegar a modelarse y/o reconfigurarse.

Palabras clave: *Arquitecturas Multiprocesador. Simulación. Sistema E/S paralela. Modelado y Simulación basado en agentes (Agent-Based Modeling and Simulation, ABMS). Cloud Computing. CloudSim.*

Contexto

Se presenta una línea de Investigación que es parte del Proyecto de Investigación “Simulación y tecnología en Cómputo de Altas Prestaciones (High Performance

Computing, HPC) para aplicaciones de interés social” de la Universidad Nacional Arturo Jauretche (UNAJ), acreditado por resolución interna 148/18. Además, el proyecto aporta al Programa “Tecnologías de la información y la comunicación (TIC) en aplicaciones de interés social” de la UNAJ.

En el tema existe un convenio de colaboración en actividades de Investigación y Postgrado con el Instituto de Investigación en Informática – LIDI de la Universidad Nacional de La Plata y el Área de Computación del Instituto de Ciencias de la Universidad Nacional de General Sarmiento.

Introducción

El crecimiento sostenido en la demanda del poder de cómputo remarca la necesidad de sistemas con enfoques de paralelización masiva y cómputo de alta performance (HPC, High Performance Computing) [1]. Los clusters se han convertido en uno de los enfoques principales para lograr paralelismo a bajo costo. Una noción extendida lo constituye la utilización de grid computing y más

recientemente cloud computing. Independientemente de la solución, estos sistemas constan de un gran número de componentes incluyendo nodos de procesamiento, bancos de memoria, discos, entre otros.

En cuanto a las herramientas de simulación, CloudSim es un framework desarrollado en Java que provee las APIs necesarias para que el usuario genere una simulación de un Data Center funcionando como servidor de nube capaz de simular la ejecución de CloudLets.

Sistemas de E/S Paralela

Las exigencias en los sistemas de E/S paralelos se han incrementado debido al aumento en número, velocidad y potencia de las unidades de procesamiento en los clusters. También las aplicaciones científicas que utilizan cómputo de altas prestaciones acrecientan estos requerimientos.

En muchos casos, el cuello de botella de los sistemas paralelos es la E/S de estos sistemas dada las exigencias que debe afrontar [2]. La E/S Paralela es esencial para emparejar el avance de las arquitecturas de los procesadores y el rápido crecimiento de la capacidad computacional. Aunque la arquitectura jerárquica de memoria multinivel puede evitar grandes pérdidas de prestaciones debido a los retardos de acceso a disco, la capacidad de memoria es limitada. Además, como la capacidad computacional aumentará, la disponibilidad de memoria por core decrecerá, especialmente si la escala de los sistemas de HPC se proyecta a millones de cores o más. Varias simulaciones científicas y de ingeniería de áreas críticas de investigación, tales como la nanotecnología, astrofísica, clima y energía física están convirtiéndose en aplicaciones intensivas de datos. Para

poder disminuir la brecha entre CPUs-E/S se deben identificar los factores que influyen en las prestaciones y proponer nuevas soluciones [3] [4].

En el área de tolerancia a fallas en sistemas de cómputo de alta prestaciones se puede notar la importancia de la unidad de E/S en las arquitecturas paralelas como un punto a mejorar para lograr cubrir las exigencias de las aplicaciones que utilizan HPC. Una manera de llevar a cabo este trabajo es utilizar técnicas de simulación para evaluar el efecto de los cambios de los factores con mayores influencias en las prestaciones del sistema de E/S paralelo.

Se puede disminuir la complejidad y la probabilidad de errores en la generación de sistemas híbridos desarrollando una simulación específica de éstos utilizando diferentes frameworks [5] [6] [7].

Las aplicaciones científicas con un uso intensivo de datos utilizan software de E/S paralelo para acceder a archivos. Contar con una herramienta que permita predecir el comportamiento de este tipo de aplicaciones en HPC es de gran utilidad para los desarrolladores de aplicaciones paralelas. Por otro lado, ABMS ha sido utilizado para modelar problemas y sistemas complejos en diversas áreas de la ciencia.

Evaluar las prestaciones del subsistema de E/S con diferentes configuraciones y la misma aplicación permite adaptar la configuración de E/S teniendo en cuenta el patrón de acceso de la aplicación. Pero también puede ser una gran ventaja analizar las necesidades de las aplicaciones antes de configurar el sistema físico. Una manera de predecir el comportamiento de las aplicaciones en el sistema de cómputo ante distintas configuraciones, es utilizando técnicas de modelado y simulación.

Se está desarrollando modelos e implementando una simulación de la

arquitectura de E/S paralela, por medio de técnicas de simulación basadas en agentes o Sistemas Multi-Agente, (MAS-MultiAgentSystems), para evaluar el efecto de dimensionar el sistema de E/S o cambiar componentes como la red de almacenamiento, dispositivos de E/S, entre otros [8].

Simulación de arquitecturas de Cloud Computing

CloudSim [9] [10] es un Framework de simulación generalizado y extensible que permite el modelado, la simulación y la experimentación de diferentes infraestructuras y servicios de aplicaciones de Cloud Computing. Un ejemplo de utilización es la simulación de muchos centros de datos.

Su arquitectura consiste en entidades específicas que se representan como clases Java que pueden ser heredadas o instanciadas para simular experimentos. Estas clases representan centros de datos, hosts físicos, máquinas virtuales, servicios a ejecutar en los centros de datos, servicios en la nube de usuarios, redes internas centro de datos y consumo de energía de los hosts físicos y elementos de los centros de datos [11]. Además, CloudSim soporta la inserción dinámica de los elementos de simulación y proporciona aplicaciones de paso de mensajes y la topología de la red del centro de datos.

Una definición importante de CloudSim es la de entidad. Una entidad es una instancia de un componente, que es una clase o un conjunto de clases que representan un modelo CloudSim (datacenter, host). El motor de simulación es capaz de simular el tiempo de ejecución de las apps ingresadas como Cloudlets con información básica [12].

La versatilidad de CloudSim es la principal ventaja del sistema. La

integración de nuevos parámetros y conceptos de la simulación es implementada desde abstracciones preestablecidas convenientemente por los autores. Las abstracciones principales son SimEvent [13], SimEntity [14], DataCenterCharacteristics y Vm.

En cada una de estas instancias es posible establecer el detalle del comportamiento y el estado de la simulación teniendo en cuenta el alcance del modelado que se planifica.

El aporte de un desarrollo de nuevos actores al componente Vm (Virtual Machine) posibilita que por medio de simulación se obtengan métricas de entrada/salida SAAS, PAAS o IAAS desarrollada con objetos que heredan de SimEvent y de SimEntity. En éste desarrollo se obtienen más detalles a la integración de éstos elementos en las características del objeto Vm que es el componente principal en los sistemas Cloud. Las estadísticas que aporta la nueva implementación de Vm, que se denomina SyntethicVm, son de gran interés ya que dan soporte a un espacio de memoria ram en tiempos de simulación. Dicho espacio de memoria principal está controlado por otro componente de SyntethicVm que es capaz de procesar instrucciones guardadas en el espacio de memoria sintético denominado RamEntity. Si las instrucciones se guardan lógicamente, la nueva versión de Vm es capaz de administrar procesos en la nueva capa de ejecución [15]. La idea general de la implementación consiste en la creación de una nueva SimEntity y sus agregaciones necesarias para manipular el espacio de memoria proporcionada por RamEntity, en tiempos de simulación. De tal forma que cada celda de RamEntity no sólo guarda datos, sino que también es posible que mantenga objetos interpretables como eventos para la nueva SimEntity que se denomina InitEntity.

Dichos eventos se planifican en tiempos de simulación, entonces se genera una nueva capa de ejecución en la simulación. La agrupación de eventos planificados para que sean ejecutados por InitEntity es definida como un ObjectCode, es decir lenguaje de máquina compilado listo para ser asignado a la pila de ejecución. En el caso del modelo de simulación estudiado, se establece que el código objeto se convierte en proceso. Ocurre cuando InitEntity lo recibe como dato del procesador de eventos de CloudSim, entonces es un proceso en estado listo. Cuando el procesador de eventos de InitEntity procesa un evento INIT_EXE_PR_ETC_RC puede tomar un proceso en estado listo, o también un proceso en estado bloqueado, si necesita un recurso no disponible y lo ejecuta. Los procesos que finalizan no se reasignan a la cola de procesos etc_rc. El código objeto es extensible para ser almacenado en el local storage para futuras entregas del modelo, es decir que durante la ejecución el proceso podría cambiar el código objeto almacenado en el local storage, en tiempos de simulación.

Lo destacable es que el entorno de la cola de procesos es en el espacio de memoria de RamEntity y que cada instrucción que compone a un código objeto del proceso es en realidad un evento con todos sus parámetros. La ejecución consiste en que el evento es programado en el motor de CloudSim, según el planificador del proceso Init. Cuando el proceso finaliza, el memory management de Init libera el fragmento de memoria asignado.

Líneas de Investigación, Desarrollo e Innovación

Temas de Estudio e Investigación

- Arquitecturas multiprocesador para procesamiento paralelo: multiprocesador de memoria compartida, multiprocesador on-chip de memoria distribuida. Multicore, Clusters, Clusters de multicore. Grid. Cloud.
- Arquitectura de E/S paralela considerando el software, hardware, comunicaciones entre módulos y dispositivos de almacenamiento.
- Modelado y simulación basada en agentes.
- Nuevos aportes de desarrollos que mejoren los modelos de simulaciones con CloudSim para el análisis de la performance en sistemas de arquitecturas de software de Cloud Computing.

Resultados y Objetivos

Investigación experimental

- Diseño y desarrollo de modelos mediante técnicas de Modelado y simulación basada en agentes (ABMS) para analizar el comportamiento de las distintas capas de la pila de software de E/S.
- Utilización de agentes para generar la funcionalidad de los elementos físicos (procesadores, memoria, buses, drivers, entre otros) como así también de las interfaces en las arquitecturas de E/S.
- Análisis, modelado e implementación de las operaciones típicas de E/S: read, write, open, close, flush.
- Análisis y modelado de librerías de archivos para aplicaciones que utilizan cómputo de altas

prestaciones-HPC. Se ha utilizado Amazon Web Services para creación de cluster virtuales y obtener métricas de la pila de software de E/S.

- Incorporación de tiempos de entrenamiento, obtenidos en AWS, en el simulador. Con esto se logra una salida más detallada y un método para validar tiempos y métricas del simulador con AWS.
- Implementación de comandos para ejecutar desde command center en NetLogo. Con esto se logró sintetizar el benchmark IOR correspondiente a la capa de aplicación de la pila de E/S, logrando introducir nuevos parámetros como tamaño de archivo y cantidad de nodos de E/S (metadata server y data server). De ejecutar este comando, se obtiene una nueva salida similar a la del benchmark IOR.
- Desarrollo de pruebas de conceptos con el modelo inicial del sistema de archivos paralelos utilizando el framework NetLogo.
- Los siguientes objetivos a tener en cuenta son: diferenciación de metadata server y data server, como dos parámetros diferentes; modelar e implementar la noción de IOPS, strip size y stride size.
- Obtención de un método de desarrollo de nuevos actores genéricos CloudSim que mejoran el modelado y la producción de estadísticas virtuales.
- Implementación de soportes de memoria principal en tiempos de simulación. Se los activa como componentes de las máquinas virtuales procesadoras de cloudlets.

- Implementación de la entidad InitEntity que procesa instrucciones en el espacio de memoria de las máquinas virtuales. Se vinculan exitosamente los tiempos de procesamiento de un cloudlet y las instrucciones asignadas a las máquinas virtuales.
- Mejorar detalles de la simulación, cada SyntethicVm inicializará automáticamente el ObjectCode con el prefijo Init_ seguido del nombre de la SyntethicVm, utilizando las ventajas de reflection de java.
- Lograr crear una nueva capa de ejecución en tiempos de simulación con los recursos de las propias entidades.
- El objetivo a mediano plazo es el contraste con las estadísticas de un despliegue de hardware que proporcione estadísticas de un sistema cloud computing privado.

Formación de Recursos Humanos

Dentro de la temática de la línea de I/D se participa en el dictado de la carrera de Ingeniería en Informática de la UNAJ. También aportan trabajos de alumnos de las materias Redes de Computadoras 2 y Programación en Tiempo Real.

Durante 2018 se han realizado 3 publicaciones nacionales. Además, se encuentran en desarrollo y concluidas varias Prácticas Profesionales Supervisadas (PPS) con las que concluyen sus estudios los alumnos de Ingeniería en Informática.

En esta línea de I/D existe cooperación a nivel nacional e internacional. Hay un investigador realizando su Doctorado, 2 becarios de grado (EVC CIN) y 2

alumnos avanzados de grado colaborando en las tareas.

Referencias

1. Grama A, Gupta A, Karypis G, Kumar V. "Introduction to parallel computing". Second Edition. Pearson Addison Wesley, 2003.
2. H Hennessy, J. L., Patterson, and D. A., Computer Architecture, Fourth Edition: A Quantitative Approach. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2006.
3. J. M. May, Parallel I/O for high performance computing. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001.
4. V. Balaji, Earth system modelling – Volume 4. IO and Postprocessing. Springer, 2013.
5. D. Encinas, Utilización de un reloj global para el modelado de un ambiente simulado distribuido. XVIII Congreso Argentino de Ciencias de la Computación. 2012
6. D. Encinas, Simulación de una red CAN para dimensionar las comunicaciones de una IMU. VII Congreso Argentino de Tecnología Espacial. 2013.
7. D. Black, SystemC: From the Ground Up. Second Edition, Springer, 2010.
8. D. Encinas et al., Modeling I/O System in HPC: An ABMS Approach. The Seventh International Conference on Advances in System Simulation (SIMUL), ISBN: 978-1-61208-442-8, 2015.
9. R. Calheiros, R. Ranjan, A. Beloglazov, C. De Rose and R. Buyya "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms" Published online 24 August 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/spe.995.
10. <http://www.cloudbus.org/cloudsim> 2018.
11. Hamza Ouarnoughi, Jalil Boukhobza, Frank Singhoff, Stephane Rubini, Erwann Kassis. "Considering I/O Processing in CloudSim for Performance and Energy Evaluation". OpenStack Cloud Software: Open source software for building private and public clouds. © Springer International Publishing AG 2016 M. Tafer et al. (Eds.): ISC High Performance Workshops 2016, LNCS 9945, pp. 591–603, 2016. DOI: 10.1007/978-3-319-46079-6 40.
12. Kushang Parikh, Nagesh Hawanna, Haleema. P.K, Jayasubalakshmi.R and N.Ch.S.N.Iyengar. School of Computing Science and engineering Vellore Institute of Technology, Tamil Nadu, "Virtual Machine Allocation Policy in Cloud Computing Using CloudSim in Java." 2015.
13. www.icsa.inf.ed.ac.uk/research/groups/hase/simjava 2018.
14. F. Howell, R Mc Nab. A discrete event simulation library for java. International Conference on Web-Based Modeling and Simulation. 1998.
15. D. Rosatto, R. Bond, M. Belizán, M. Morales, D. Encinas. Modelado y simulación de arquitecturas de Cloud Computing con CloudSim: comunicación entre entidades. XXIII Congreso Argentino de Ciencias de la Computación. ISBN 978-950-34-1539-9. 2017