

Modelado y Simulación de Sistemas de Gran Escala

Rubén Apolloni¹, Alicia Castro¹, Verónica Gil-Costa², Fernando Loor², Cristian Tisera¹
and Guillermo Trabes¹

⁽¹⁾Departamento de Informática
Facultad de Ciencias Físico
Matemáticas y Naturales
Universidad Nacional de San Luis
Ejército de los Andes 950, 1° piso.
(02652-420823)

⁽²⁾CCT-XONICET San Luis
Almirante Brown 907
(02664-421654)

Resumen

Las simulaciones computacionales permiten entender los elementos y patrones que pueden alterar un sistema y puede ser utilizada para estudiar sistemas complejos, incluyendo aquellos que son analíticamente intratables. La simulación también es utilizada cuando no se tienen los recursos computacionales disponibles. En estos casos se puede estimar diferentes parámetros como la escalabilidad de los sistemas, a medida que se simula el aumento del número de recursos asignados. En otras palabras, utilizando técnicas de modelado y simulaciones eficientes es posible diseñar, desarrollar y evaluar sistemas de gran escala. En este trabajo, se presentan los objetivos, trabajo realizado y desafíos que aborda el grupo de investigación de la Universidad Nacional de San Luis, para abordar los temas que involucra el diseño de simulaciones de sistemas complejos y de gran escala para procesar grandes volúmenes de datos e información.

Palabras clave: *Sistemas Complejos y de Gran Escala, Simulación, Modelado.*

1. Contexto

La línea de investigación que se presenta en este trabajo “Modelado y Simulación de Sistemas de Gran Escala” pertenece al proyecto PROICO “Tecnologías Avanzadas aplicadas al Procesamiento de Datos Masivos” de la UNSL. Esta línea de

investigación, se basa en el uso de técnicas de simulaciones eficientes para diseñar y desarrollar sistemas de gran escala como lo son los sistemas de búsqueda Web y sistemas de evaluación y soporte a las decisiones para situaciones de desastres naturales. Con estas simulaciones se pueden probar alternativas en el diseño y predecir su impacto sobre un sistema real por medio de métricas de rendimiento, lo cual permite determinar el costo-beneficio e implementar la aplicación en un hardware real.

Existen diferentes técnicas de simulación como la Simulación Aproximada [Marin13], Redes de Petri [Petri62] [Peterson77], simuladores orientados a procesos, simulación paralela, entre otros. En particular, las investigaciones abordadas se enfocan en considerar el modelado de aplicaciones complejas y de gran escala donde intervienen diferentes variables de incertidumbre para plataformas paralelas y aplicadas a sistemas, que no pueden ser probadas en sistemas y hardware reales sin acarrear un costo económico que puede ser de gran impacto en el caso de que las nuevas aplicaciones o los algoritmos no cumplan con sus objetivos.

2. Introducción y Motivación

El desarrollo e implementación de sistemas de gran escala es una tarea compleja y más aún cuando se espera que sean capaces de seguir funcionando en situaciones de desastre natural. Varios son

los factores a considerar bajo esta premisa, como por ejemplo: el tiempo de desarrollo, arquitectura de software, el comportamiento dinámico de los usuarios, cortes de luz, destrucción de infraestructura de telecomunicación, etc., lo cual finalmente se traducen en una costosa inversión, que de no llevarla a cabo correctamente podría poner vidas en riesgo. Para disminuir dichos riesgos se hace necesario contar, de ante mano, con herramientas que permitan modelar y analizar las diversas situaciones que se podrían producir al usar el sistema, libre de riesgos y en un ambiente seguro. Es en este punto donde se destaca el uso de la simulación computacional, una disciplina ampliamente estudiada cuando el análisis de un sistema complejo ya no puede ser llevado a cabo mediante métodos analíticos.

La Simulación de Eventos Discretos, desde ahora DES (en inglés Discrete Event Simulation), se utiliza para representar sistemas complejos como una secuencia ordenada de eventos y donde cada evento posee el time-stamp del sistema físico representado. Cuando dichos sistemas son muy costosos en términos de procesamiento, disco y uso de memoria, su simulación también puede ser costosa, siendo éste el caso, se puede decir que el resultado es una simulación de gran escala que suele enfrentarse mediante el uso de Simulación Paralela de Eventos Discretos, desde ahora PDES (en inglés Paralell Discrete Event Simulation). Este tipo de simulación busca aprovechar al máximo las capacidades de los clusters de cómputo implementando diversas técnicas, por ejemplo: la distribución y balance de carga de trabajos en las unidades de cómputo y sincronización de eventos concurrentes mediante los protocolos conservativos (evita violar las condiciones de causalidad) y optimistas (avanza permitiendo violar las condiciones de causalidad y realiza roll-back cuando esto sucede).

El objetivo principal, de esta línea de investigación, es investigar el uso de simuladores eficiente y eficaces para evaluar el rendimiento de nuevos

algoritmos paralelos y distribuidos para aplicaciones de datos masivos. Un punto crucial es probar algoritmos y estrategias que permitan balancear la carga de trabajo de los componentes a evaluar. En particular, se propone (a) Investigar el enfoque forward-in-time para simulaciones de evoluciones de poblaciones. (b) Obtener simuladores que sean escalables, reduzcan el consumo de memoria y entreguen resultados en el menor tiempo posible. (c) Investigar la aplicación de BSP y Multi-BSP para modelar problemas de gran escala y estimar los tiempos de ejecución utilizando los parámetros provistos por los modelos. (d) Continuar estudiando y desarrollado simuladores basados en autómatas celulares para situaciones de evacuaciones. (e) Continuar aplicando el conocimiento adquirido en esta línea para realizar actividades de servicio y poner en práctica los nuevos algoritmos/enfoques desarrollados.

A continuación, se describen algunas de las técnicas, herramientas y plataformas utilizadas.

3. Investigación y Desarrollo

3.1 Capacity Planning

Capacity planning [Klosterboer11] es un proceso iterativo, que explota el comportamiento histórico de la gente para predecir el crecimiento de las necesidades de éstos a través del tiempo y prever futuros requerimientos de hardware en alzas inesperados de peticiones. La empresa Amazon cuenta con metodologías de capacity planning privadas, sin embargo sólo se sabe que para poder predecir el crecimiento de las necesidades de sus centros de datos actualmente utilizan Aprendizaje Máquina (Machine Learning), [geekwire17].

De manera similar, Google utiliza Aprendizaje Máquina para hacer un capacity planning enfocándose en optimizar el Uso de Energía Eficiente (Power Usage Efficient, PUE) [Gao14].

En algunos trabajos previos, se han desarrollado modelos y simuladores para analizar la cantidad de recursos que deben asignarse para un motor de búsqueda Web. En particular, en [Gil12] se propone utilizar un modelo de Redes de Petri.

Posteriormente, en [Gil13] se evalúan diferentes algoritmos de despliegues para mejorar el rendimiento del sistema.

3.2 Simulación Paralela Aproximada

La sincronización de eventos en PDES suele llevarse a cabo mediante estrategias conservativas y optimista. Sin embargo, ambas estrategias suelen ser costosas en términos computacionales y en tiempo de ejecución. Otra estrategia válida es la propuesta en [Marin13], donde se ha expuesto un nuevo enfoque mediante una estrategia de simulación Aproximada, la cual plantea realizar una sincronización optimista que permite ejecutar eventos con time-stamp que violan la condición de causalidad. Es decir, ejecutar eventos con time-stamp menor al tiempo global de la simulación, esto sin realizar roll-back. Su lógica se basa en ir calculando periódicamente barreras tiempo dinámicas, gracias a una técnica de ventanas deslizantes, que posibilitan lo anterior. Con este método lograron demostrar una eficiente paralelización de los eventos.

Los modelos de simulación proveen una mejora en términos de tiempo de ejecución. Bajo este paradigma el modelo de simulación es particionado en diferentes procesadores físicos que se comunican entre sí intercambiando mensajes (eventos).

En particular, en el trabajo presentado en [Marin13] se propone priorizar los tiempos de ejecución y perder hasta un 5% de precisión en los resultados finales de la simulación. Este trabajo se basa en la utilización del modelo de computación paralelo BSP [Val90], donde el cómputo se organiza en una secuencia de supersteps. Durante un superstep, cada

procesador realiza operaciones sobre sus datos locales y/o envía mensajes a otros procesadores. Al final de cada superstep se ejecuta una barrera de sincronización que permite que los mensajes enviados en el superstep actual, se encuentren disponibles en las colas de entrada de los procesadores destinos en el próximo superstep.

3.3 Modelos de Simulación

Los modelos de computación se han usado como un puente entre algoritmos paralelos y arquitecturas de hardware. El modelo Bulk-Synchronous Parallel (BSP) [Val90] es un modelo muy conocido originalmente diseñado para ejecutar algoritmos distribuidos en clusters de procesadores de un solo núcleo. El modelo Multi-BSP, que amplía el modelo BSP clásico, fue propuesto recientemente para procesadores multi-core.

Sin embargo, el modelo Multi-BSP presenta algunas restricciones tales como las sincronizaciones explícitas entre los cores, presentando algunos desafíos que se deben tener en cuenta para modelar adecuadamente los algoritmos paralelos. Por lo tanto, es importante explorar la utilidad y el alcance de estos modelos para las arquitecturas multi-core [Trabes18].

3.4 Simulación de Evacuaciones

En los últimos años se ha incrementado el interés en desarrollo de modelos de procesos de evacuaciones de emergencias. Nuestra propuesta, se enfoca en el estudio de evacuación de edificaciones, el cual radica en que la totalidad de sus ocupantes, en cualquier instante, deben tener la posibilidad de desplazarse hasta un lugar seguro en el tiempo adecuado con las suficientes garantías de seguridad.

Debido a la dificultad en cómputo para resolver este tipo de problemas, un modelo de simulación complejo puede requerir de modernas plataformas computacionales de alta performance para acelerar a gran escala las simulaciones de

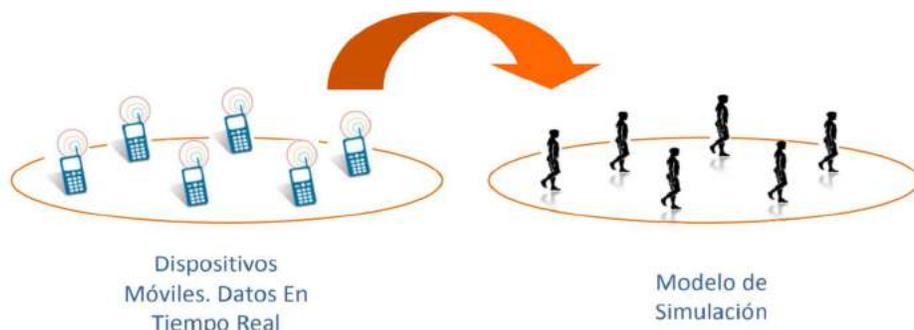


Figura 1: Datos en tiempo real enviados a un modelo de simulación

eventos discretos y promover la reutilización y la interoperabilidad de los componentes de simulación.

Actualmente se cuenta con la implementación de un modelo de simulación que permite simular el comportamiento de un conjunto de individuos en un ambiente cerrado con un número fijo de salidas. El principal objetivo de este modelo, es el de recrear la situación en la cual un conjunto de personas debe desalojar un recinto debido a la amenaza de fuego. En este modelo es factible obtener la información necesaria a través de sensores o dispositivos móviles, que será utilizada como entrada para nuestro modelo (Fig. 1).

También se han desarrollado modelos de simulación híbrida [TPL12] que consiste en dos sub-modelos denominados ambiental (EsM) y peatón [Psm]. Este modelo junto con la metodología computacional permite construir ambientes artificiales poblados con agentes autónomos que son capaces de interactuar entre sí.

4. Resultados

La línea de investigación y las técnicas descritas en la sección anterior involucra una serie de desarrollos individuales que en su conjunto logran obtener el objetivo planteado. Este objetivo contempla el modelado y diseño de sistemas complejos mediante diferentes herramientas, algunas de las cuales han sido desarrolladas por los integrantes del proyecto.

Mediante estas herramientas se pretende facilitar el entendimiento del sistema, realizar profiling a la ejecución de los algoritmos y mejorar el análisis de resultados.

Los resultados obtenidos hasta el momento son:

- Evaluar el beneficio de la colaboración entre un motor de búsqueda Web y un proveedor de servicio de internet (ISP) [Ros14]. En particular, en este trabajo se propone utilizar una jerarquía de memorias cachés ubicadas en los nodos de las redes P2P y en el servidor ISP, con el objetivo de reducir el tráfico de consultas dirigidas al motor de búsqueda. A cambio, el motor de búsqueda puede reducir los costos por acceder a él y entregar información más precisa respecto del tiempo de vida de los resultados.
- En [Hid14] y [Gil14b] se realiza una evaluación del consumo de energía requerido tanto por la red de comunicación interna al ISP y la red de comunicación entre el ISP y el Motor de búsqueda.

Los resultados esperados son:

- Generar un modelo de las aplicaciones de gran escala. Modelar los componentes más relevantes y la comunicación entre los mismos.
- Desarrollar simuladores secuenciales de las aplicaciones de gran escala. Validar los resultados de los simuladores y verificar que el comportamiento (flujo de datos y operaciones) sea correcto.

- Proponer enfoques paralelos eficientes de los simuladores desarrollados. Utilizar estos simuladores para realizar estudios de capacity planning.
- Transferencia de logros obtenidos.

5. Formación de Recursos Humanos

Actualmente, se cuenta con tres doctores en ciencias de la computación realizando la investigación y dirección de los doctorandos involucrados en la línea. Se cuenta con tres estudiantes de doctorando. Dos de ellos se encuentran en su segundo año de trabajo. El tercer doctorando se encuentra en su primer año, de iniciación a la investigación. El primer estudiante realiza su tesis en el desarrollo y estudio de modelos para la automatización del despliegue de aplicaciones en plataformas paralelas. El segundo estudiante, realiza su tesis doctoral en el estudio y desarrollo de plataformas escalables para casos de desastres naturales utilizando técnicas de crowdsourcing.

Mediante este trabajo de investigación se podrán formar profesionales que puedan modelar, diseñar e implementar algoritmos eficientes (previamente evaluados y analizados con un mínimo porcentaje de error mediante las técnicas de modelado y simulación) que se ejecuten en sistemas complejos donde intervienen una gran cantidad de variables y requieren el procesamiento de datos masivos.

6. Bibliografía

- [Gil12] V. Gil-Costa, J. Lobos, A. Inostrosa-Psijas and M. Marin. "Capacity Planning for Vertical Search Engines: An approach based on Coloured Petri Nets" Petri Nets 2012.
- [Gil13] A. Inostrosa-Psijas, V. Gil-Costa, M. Marin and E. Feuestein. "Service Deployment Algorithms for Vertical Search Engines". PDP 2013.
- [Gil14b] V. Gil-Costa, E. Rosas, N. Hidalgo and M. Marin. "Simulation Model for energy assessment in ISP and Web Search Engine collaboration". Summer Simulation Multi-Conference, pp. 23:1--23:8, 2014. [Geekwire17] <https://www.geekwire.com/2017/amazon-web-services-uses-machine-learning-make-capacity-planning-decisions/>, at Pacific Science Center's 14th Annual Foundations of Science Breakfast
- [Gao14] Gao, J., Jamidar, R. (2014). Machine learning applications for data center optimization. Google White Paper
- [Hid14] "Assessing energy efficiency in ISP and Web Search Engine collaboration". N. Hidalgo, E. Rosas, V. Gil-Costa and M. Marin. FINA. pp. 299-304. May 13 - 16, 2014.
- [Klosterboer 11] Klosterboer, L. (2011). ITIL capacity management. Upper Saddle River, NJ: IBM Press. [Google Scholar]
- [Marin13] M. Marin, V. Gil-Costa, C. Bonacic and R. Solar. pproximate Parallel Simulation of Web Search Engines. PADS 2013
- [Peterson77] J.L. Peterson. Petri nets. Computing Surveys, 9:223-252, 1977.
- [Petri62] C.A. Petri. "Communication with Automata" New York: Griffiss Air Force Base. Tech. Rep. RADC-TR-65-377, vol.1, Suppl. 1, 1962.
- [Ros14] E. Rosas, N. Hidalgo, M. Marin, V. Gil-Costa. Web Search Results Caching Service for Structured P2P Networks. Future Generation Computer Systems (Elsevier). Volume 30, Pages 254-264, January 2014.
- [TPL12] P. Tisera A. M. Printista, and E. Luque. A Hybrid Simulation Model to Test Behaviour Designs in an Emergency Evacuation. ICCS, (2012)
- [Trabes18] G. Trabes, V. Gil-Costa, M. Printista and M. Marin. Multi-BSP vs. BSP: A Case of Study for Dell AMD Multicores. PDP 2018-
- [Val90] L. G. Valiant. A bridging model for parallel computation. Comm. ACM, 33(8) 1990.