

FASES PARA EL DESARROLLO DE PROGRAMAS PARALELOS Y SU APLICACIÓN EN LA PREDICCIÓN DEL COMPORTAMIENTO DE INCENDIOS FORESTALES

Ana Laura Molina¹, Nelson Rodriguez¹, María A. Murazzo¹, Fernando Pincioli² y Miguel Méndez-Garabetti³

¹Departamento de Informática, Facultad de Ciencias Exactas, Físicas y Naturales
Universidad Nacional de San Juan

²Instituto de Investigaciones, Facultad de Informática y Diseño
Universidad Champagnat

³Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

lauramolina@outlook.com, nelson@iinfo.unsj.edu.ar, marite@unsj-cuim.edu.ar,
pincirolifernando@uch.edu.ar, mendez-garabettimiguel@uch.edu.ar

RESUMEN

En la actualidad, la programación paralela está siendo ampliamente utilizada en aplicaciones tanto científicas como comerciales. A través de esta se busca lograr una reducción del tiempo necesario para resolver un problema complejo y un aumento en la calidad de la respuesta. Las propuestas existentes para guiar el desarrollo de programas paralelos plantean únicamente metodologías para su diseño y en algunos casos herramientas para su implementación, sin hacer uso de buenas prácticas propias de la Ingeniería de Software.

Esta línea de investigación intenta elaborar una propuesta de fases que incluirá por una parte las de diseño e implementación -ya investigadas previamente por diversos autores-, y por otra parte la adaptación de fases propias de la Ingeniería de Software como la captura de requisitos y las pruebas, no utilizadas hasta el momento en la programación paralela. Esta propuesta será aplicada a la predicción del comportamiento de incendios forestales, donde el poder computacional ofrecido por los programas paralelos contribuiría en la obtención de un resultado de calidad en el menor tiempo posible.

Palabras clave: programas paralelos, Ingeniería de Software, predicción de incendios forestales.

CONTEXTO

El presente trabajo se encuadra en la línea de I+D denominada “Procesamiento Distribuido y Paralelo” y se desarrolla en el marco del proyecto de investigación denominado: “Orquestación de servicios para la Continuidad Edge al Cloud”.

La unidad ejecutora de dicho proyecto es el Departamento de Informática de la Facultad de Ciencias Exactas, Físicas y Naturales (FCEfyN) de la Universidad Nacional de San Juan (UNSJ).

1. INTRODUCCIÓN

En un mundo donde el tiempo es, quizás, el bien más preciado, la velocidad resulta esencial y la eficiencia es traducida en cuán rápido y qué tan bien podemos solucionar un problema [1]. En este contexto, es donde el procesamiento paralelo adquiere importancia. Esta, es un área de las ciencias de la computación que aprovecha los recursos de hardware a través de la construcción de programas en los que múltiples tareas

cooperan estrechamente para resolver un problema [2].

Un programa paralelo divide el problema a resolver en piezas que serán ejecutadas por un procesador multi-núcleo o múltiples procesadores mono-núcleo, dependiendo de si se trata del modelo de memoria compartida o distribuida. Algunas de estas piezas lo harán en simultáneo, es decir en paralelo, logrando así una reducción del tiempo necesario para completar el trabajo [3]. Además de mejorar el tiempo de ejecución, también se puede aumentar la calidad de la solución obtenida respecto del mejor algoritmo secuencial tratando el mismo problema [1]. Sin embargo, no todas las aplicaciones que sean programadas en paralelo obtendrán una importante mejora en el desempeño. Esto se debe, en la mayoría de los casos, a la necesidad de los núcleos de coordinar su trabajo. Esta coordinación involucra comunicación, balanceo de carga y sincronización [2].

La evolución de los procesadores a lo largo del tiempo, demostró que existe una continua demanda de incremento del poder computacional. Otra fuente histórica de incremento del poder computacional es el paralelismo que con el paso de los años, se está volviendo omnipresente y, en consecuencia, la programación paralela se convierte así en un aspecto central para las empresas de desarrollo de software.

A medida que aumenta el poder computacional, aumenta la cantidad de problemas que pueden ser considerados para su resolución. Entre estos se encuentran los pertenecientes a campos como: modelado climático, descubrimientos medicinales, análisis de datos en genética, física, medicina, Internet, entre otros [2]. Dentro del conjunto de problemas que pueden ser resueltos a partir de un incremento del poder de cómputo, existe un subgrupo que requiere su resolución en un período de tiempo razonable y otro que incluso posee un tiempo máximo de resolución, denominado en inglés *deadline*, donde posterior a este, el dato deja de ser útil, como es el caso de la predicción meteorológica [4] y la predicción del

comportamiento de incendios forestales [5][6][7]. En este último caso, para poder tomar decisiones preventivas sobre una distribución de recursos de extinción lo más eficiente posible, se necesita un sistema de predicción cuyos resultados deben producirse antes de que ocurra el fenómeno [5]. Por esto, se vuelve fundamental lograr una reducción en el tiempo de cálculo. Si bien el tiempo de respuesta es clave, también lo es la calidad de la predicción debido a la criticidad de las decisiones que se van a tomar en función de la salida [6]. En este caso se busca determinar el posible comportamiento de un incendio forestal una vez que este ya se ha iniciado, es decir, obtener información de la dirección y velocidad de la propagación, la forma del frente de fuego y la intensidad de la llama, utilizando la mayor cantidad de información posible [7][8]. De acuerdo con estos requisitos, se torna primordial la utilización de sistemas de procesamiento de alto rendimiento y la implementación de programas paralelos [7]. Esta línea de investigación se basa en la aplicación de los programas paralelos a la predicción del comportamiento de incendios forestales.

Si bien, tradicionalmente, los principales desarrollos de programas paralelos estuvieron orientados a aplicaciones científicas e ingenieriles, actualmente, las aplicaciones comerciales que deben procesar grandes volúmenes de datos también requieren del paralelismo para obtener un buen desempeño.

En cualquiera de los casos mencionados, desarrollar programas paralelos no es una tarea sencilla, ya que la mayoría de los problemas tienen muchas soluciones paralelas y resulta difícil encontrar la óptima o, incluso, una aceptable. De hecho, la programación paralela ha sido pensada tradicionalmente como una actividad intensiva en tiempo y esfuerzo [9]. Tal es así que el progreso de los componentes de hardware implicados en la computación paralela, debe ir acompañado de un progreso en los aspectos de diseño, análisis y aplicación de los programas paralelos para que estos beneficios sean observables en la práctica [1].

Tradicionalmente, la Ingeniería de Software ha estado más enfocada en el dominio empresarial y de la tecnología de la información que en el dominio de la ciencia y la ingeniería. Muchos investigadores pertenecientes a esta rama de la Ingeniería han asumido que cualquiera de sus soluciones podría aplicarse al ámbito de las ciencias de la computación, mientras que un gran número de investigadores de dicho ámbito han creído que sus inconvenientes eran diferentes a los surgidos en entornos tradicionales de la Ingeniería de Software y, por lo tanto, no han aplicado sus métodos.

Actualmente, en la mayoría de los casos, el desarrollo de software en las ciencias de la computación no sigue las buenas prácticas tradicionales de la Ingeniería de Software y esto se debe a que muchas de ellas no son aplicables sin una previa adaptación. Esta tendencia debe ser abordada para mejorar tanto la productividad del desarrollo como la fiabilidad del software a desarrollar [10]. No tendría sentido buscar notaciones, métodos o técnicas universales para la Ingeniería de Software ya que diferentes tipos de software requieren de distintos enfoques. Sin embargo, hay ciertos principios fundamentales que se aplican a todo tipo de software como son: desarrollar utilizando un proceso comprendido y gestionado, entender y gestionar los requerimientos de clientes y usuarios, y usar eficientemente los recursos [11].

A partir del análisis de propuestas de diversos autores [1][2][9][12], es posible afirmar que todos ellos plantean únicamente metodologías para el diseño de programas paralelos y en algunos casos herramientas para la implementación, dejando de lado las fases frecuentemente utilizadas en el área de la Ingeniería de Software, como son la captura de requisitos y las pruebas.

Estas dos últimas fases mencionadas se presentan como un espacio sin investigar en el ámbito de la programación paralela, por lo cual se transforma en el problema que se intenta abordar en esta línea de investigación. El problema planteado se relaciona específicamente con la conveniencia de

integrar aportes de la Ingeniería de Software tradicional al desarrollo de programas paralelos, reuniendo conocimientos hasta el momento inconexos. Esta investigación se propone definir y caracterizar las fases para el desarrollo de programas paralelos y aplicar la propuesta a la predicción del comportamiento de incendios forestales.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

Siguiendo la línea de investigación en que se enmarca este trabajo, se llevarán a cabo actividades relacionadas con los siguientes espacios:

- Ingeniería de software
- Ciencias de la computación, en el área de Procesamiento Distribuido y Paralelo.

3. RESULTADOS OBTENIDOS/ESPERADOS

En esta línea de I+D se intenta elaborar una propuesta de fases que incluya tanto las de diseño e implementación que suelen utilizarse frecuentemente en el ámbito de la programación paralela como la adaptación de fases propias de la Ingeniería de Software, como son la captura de requisitos y las pruebas de verificación y validación, que no se han utilizado aún en dicho ámbito.

Estas dos últimas fases serán adaptadas en función de las necesidades específicas y características particulares de los programas paralelos y sus entornos de desarrollo. Para llevar a cabo esta tarea, se seleccionarán entre las diversas técnicas existentes y difundidas, aquellas que mejor se ajusten, y serán modificadas en aspectos que se considere que contribuirán a mejorar la calidad del software resultante.

Se busca contribuir en el área de la programación paralela produciendo una guía útil para que quienes no tienen conocimientos en el área de la Ingeniería de Software puedan llevar a cabo dichas fases adecuadamente. Para cada fase de la propuesta se plantearán

una serie de actividades a realizar y un conjunto de buenas prácticas que deberían tener en cuenta las partes interesadas tanto pertenecientes al equipo de desarrollo como ajenas (clientes, usuarios, sponsors, etc.).

Luego, se pretende aplicar la propuesta de fases elaborada al desarrollo de un programa paralelo para la predicción del comportamiento de incendios forestales y así poder analizar sus limitaciones y potencialidades.

Se espera que a través del uso de la propuesta en el desarrollo de programas paralelos se logre satisfacer las necesidades y objetivos de todas las partes interesadas.

4. FORMACIÓN DE RECURSOS HUMANOS

Esta investigación se encuadra en el proyecto que se ejecuta bajo la dirección del Lic. Nelson Rodríguez y la codirección de la Lic. María A. Murazzo.

En el marco de la línea de I+D presentada, se desarrolla una tesina de grado por parte de la estudiante de la Licenciatura en Sistemas de Información de la Universidad Nacional de San Juan, Ana Laura Molina.

Dicha tesina es asesorada por el director del proyecto antes mencionado, el Lic. Fernando Pinciroli y el Mgter. Ing. Miguel Méndez-Garabetti.

5. REFERENCIAS

- [1] S. G. Akl y M. Nagy, "Introduction to Parallel Computation" en *Parallel Computing: Numerics, Applications, and Trends*, R. Trobec, M. Vajteršic and P. Zinterhof, London: Springer, 2009, pp. 43-80.
- [2] P. Pacheco, *An introduction to parallel programming*. USA: Elsevier, 2011.
- [3] A. Freeman, *Pro. NET 4 parallel programming in C*. USA: Apress, 2011.
- [4] B. Wilkinson, y M. Allen, *Parallel programming: techniques and applications using networked workstations and parallel computers*, 2nd ed. New York: Prentice hall, 2005.
- [5] M. Méndez-Garabetti, G. Bianchini, M. L. Tardivo, y P. Caymes-Scutari, "Comparative analysis of performance and quality of prediction between ESS and ESS-IM", *Electronic Notes in Theoretical Computer Science*, vol. 314, pp. 45-60, 2015.
- [6] M. Méndez-Garabetti, G. Bianchini, M. L. Tardivo, P. Caymes Scutari y G. V. Gil Costa, "Hybrid-Parallel Uncertainty Reduction Method Applied to Forest Fire Spread Prediction", *Journal of Computer Science & Technology*, vol. 17, pp. 12-19, 2017.
- [7] M. Mendez-Garabetti, M. L. Tardivo, G. Bianchini, P. Caymes Scutari, "Predicción del Comportamiento de Incendios Forestales mediante un Método de Reducción de Incertidumbre basado en HPC y Evolución Diferencial", en *XVI Workshop de Investigadores en Ciencias de la Computación*, 2014, pp. 690-694.
- [8] M. Mendez-Garabetti, G. Bianchini, P. Caymes-Scutari, M. L. Tardivo, "Increase in the quality of the prediction of a computational wildfire behavior method through the improvement of the internal metaheuristic", *Fire Safety Journal*, vol. 82, pp. 49-62, 2016.
- [9] A. Grama, A. Gupta, G. Karypis, y V. Kumar, *Introduction to Parallel computing*. Harlow, UK: Pearson, 2003.
- [10] J. C. Carver, "Software Engineering for Computational Science and Engineering", *Computing in Science & Eng*, vol. 14, no. 2, pp. 8-11, 2012.
- [11] I. Sommerville, *Software Engineering*. Boston: Addison Wesley Publishing Company, 2011.
- [12] I. Foster, *Designing and building parallel programs*. Boston: Addison Wesley Publishing Company, 1995.