

Método de Reducción de Incertidumbre Aplicado a la Predicción del Comportamiento de Incendios Forestales bajo una Implementación Heterogénea MPI/CUDA

Miguel Méndez-Garabetti ^{a,b}, Germán Bianchini ^b, Paola Caymes-Scutari ^{b,c}
y María Laura Tardivo ^d

^aFacultad de Ingeniería, Universidad Atlántida Argentina, Buenos Aires, Argentina.

^bLaboratorio en Cómputo Paralelo Distribuido (LICPaD), UTN-FRM, Mendoza, Argentina.

^cConsejo Nacional de Investigaciones Científicas y Técnicas (CONICET)

^dDepartamento de Computación, UNRC, Río Cuarto, Córdoba, Argentina

miguel.mendez@atlantida.edu.ar, gbianchini@frm.utn.edu.ar, pcaymesscutari@frm.utn.edu.ar,
lauratardivo@dc.exa.unrc.edu.ar

Resumen

Los incendios forestales causan grandes pérdidas y daños en todo el mundo. Por esta razón, la predicción de este tipo de fenómeno se considera una tarea muy importante que implica un alto grado de complejidad. Debido a esto, tales métodos deben configurarse para operar de la manera más eficiente posible, tanto en términos de calidad de los resultados como así también en relación al tiempo de obtención de los resultados. En este trabajo, se presenta una implementación MPI/CUDA de un método paralelo de reducción de incertidumbre aplicado a la predicción del comportamiento de incendios forestales. Los resultados revelan grandes incrementos en la eficiencia computacional del método.

1. Introducción

El fuego ha sido una herramienta fundamental en el desarrollo de las civilizaciones, Hough en su trabajo [1], menciona que el hombre ha sido el único mamífero en superar el miedo al fuego, siendo esta cualidad una característica humana de gran importancia, incluso en la misma obra se deja abierto el interrogante sobre si la no adopción del fuego -por parte del ser humano- no hubiera dejado a la humanidad en decadencia destinada a la

extinción. Posteriormente destaca que: "La única fuerza esencial disponible en la naturaleza unida al ingenio del hombre se convirtió en una combinación de potencialidades inimaginables" [1]. Cuando el fuego, por falta de control, se extiende sobre una superficie se conoce como incendio, éste fenómeno al consumir vegetación de áreas forestales, se denomina incendio forestal [2]. Cuando se habla de incendios forestales comúnmente se connota con los aspectos negativos que éstos poseen. Sin embargo, éstos juegan un rol fundamental en los cambios naturales que ocurren en los ecosistemas de nuestro planeta. Los efectos del fuego, sobre bosques, praderas, etc., propician la diversidad de la vida vegetal y animal. Incluso ciertos tipos de plantas no tienen posibilidad de reproducirse sin el fuego.

El fuego inicia procesos naturales que permiten convertir la materia orgánica en nutrientes que posteriormente la lluvia entrega al suelo, proporcionando un semillero fértil rejuvenecido para las plantas [3]. Sin embargo, los incendios forestales fuera de control suelen generar grandes daños, entre ellos: pérdidas de vidas humanas, daños a la flora, la fauna y al suelo [4]. Las pérdidas económicas son difíciles de estimar debido a la gran cantidad de sectores que pueden verse afectados a corto, mediano o largo plazo, pero en general los recursos ambientales de toda zona afectada suelen ser

considerablemente damnificados. En base a esto desde diversas organizaciones a nivel mundial se estudia el desarrollo de herramientas y técnicas con el propósito de minimizar los efectos negativos causados por los incendios forestales. Estas investigaciones se han enfocado principalmente en herramientas para la prevención, detección y predicción de incendios forestales. Cada una de éstas corresponde a distintas fases del proceso de lucha contra incendios. La prevención tiene como objetivo lograr que los incendios no alcancen a materializarse; la detección se concentra en determinar la ubicación de los focos de incendio antes de que éstos adquieran una magnitud tal que lleve a perder el control sobre ellos; y por último la predicción, que tiene dos acepciones: predicción de incendios forestales y predicción del comportamiento de incendios forestales. La primera de ellas intenta predecir la ocurrencia de incendios antes de que estos sucedan, mientras que la segunda, busca determinar el posible comportamiento de un incendio forestal una vez que éste ya se ha iniciado, permitiendo tomar decisiones acertadas en el plan de acción a tomar.

Uno de los retos más importantes que enfrenta un modelo de predicción de comportamiento de incendios forestales es disminuir los efectos causados por la incertidumbre i.e., intentar conocer con el mayor grado de precisión posible los valores de los parámetros de entrada que determinan el comportamiento del fenómeno. El comportamiento del fuego en un entorno forestal depende de diferentes variables: el clima, el combustible, la topografía del terreno, el tipo de vegetación, entre otros. Generalmente, al momento de realizar la predicción de un incendio, existe dificultad para adquirir en tiempo real los valores de los factores que determinan el comportamiento del incendio. Esta carencia de información se conoce como incertidumbre en los parámetros de entrada y afecta de forma considerable la capacidad de predicción del sistema. De esta manera, la reducción de incertidumbre constituye un proceso necesario e importante para lograr predicciones más acordes a la realidad, que permitan

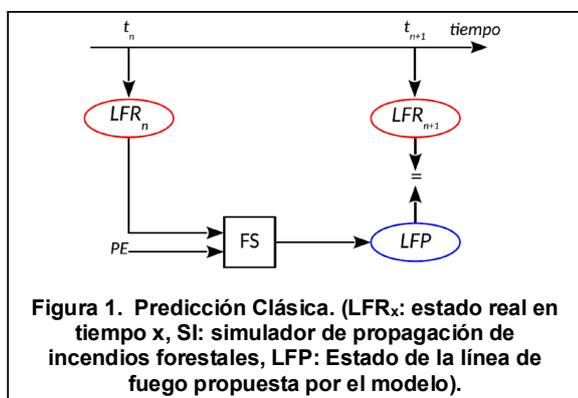
constituir herramientas confiables para asistir a los sistemas de toma de decisiones.

Claramente, lograr reducir la incertidumbre facilita la obtención de predicciones más cercanas a la realidad permitiendo cumplir uno de los principales objetivos de todo sistema de predicción, sin embargo, este tipo de sistemas también debe garantizar generar las predicciones dentro de un lapso razonable de tiempo. En este sentido la computación de alto rendimiento (HPC, High Performance Computing) [5] juega un rol fundamental, ya que permite resolver problemas cada vez más grandes y computacionalmente costosos de manera más eficiente. Este trabajo presenta una implementación paralela heterogénea (memoria compartida y distribuida) de un método de reducción de incertidumbre aplicado a la predicción del comportamiento de incendios forestales, la cual permite reducir el tiempo de procesamiento respecto a la versión homogénea.

A continuación, el trabajo se estructura de la siguiente manera: la Sección 2 presenta los conceptos de predicción clásica y sus limitaciones, la Sección 3 describe el método de reducción de incertidumbre desarrollado, sus componentes y funcionamiento. A continuación, la Sección 4 presenta la experimentación desarrollada y finalmente se presentan las conclusiones y trabajos futuros.

2. Predicción Clásica

En general, un sistema de predicción es un software que, a partir del conocimiento de ciertas variables iniciales, y que, mediante la ejecución de diferentes operaciones, permite realizar estimaciones futuras acerca del comportamiento de un fenómeno determinado [6]. En este trabajo se aborda este concepto en relación a la predicción del comportamiento de incendios forestales, el cual puede verse representado gráficamente en la Fig. 1, donde un Simulador de Incendios (SI), tratado como una caja negra, es alimentado con ciertos parámetros de entrada (PE) que representan condiciones actuales del incendio, tales como: velocidad y dirección del viento, vegetación, características de la superficie, entre otros. El simulador necesita, además de los PE, la Línea de Fuego Real (LFR) del incendio en el instante de tiempo t_0 (LFR_0). Una vez que el simulador es ejecutado, éste proporcionará en su salida la predicción de la línea de fuego o Línea de Fuego Predicha (LFP) para el instante de tiempo posterior, t_1 . Por supuesto, se espera que dicha predicción coincida, de la mejor manera posible, con el estado real del incendio para dicho instante de tiempo (LFR_1). Sin embargo, debido a la complejidad e incertidumbre del modelo de comportamiento del fuego, la incertidumbre en los parámetros de entrada, y dado que la predicción está basada en una única simulación, esta metodología no suele



ofrecer una predicción lo suficientemente aproximada a la realidad como para ser considerada una herramienta confiable para la toma de decisiones. Algunos ejemplos de predicción clásica son [7], [8], [9], [10], [11], [12].

Las limitaciones presentes en el enfoque de predicción clásica hacen evidente la necesidad de incorporar técnicas de reducción de incertidumbre que permitan mejorar el rendimiento (en calidad y tiempo de respuesta) de los sistemas de predicción actuales. En este contexto, los desarrollos de métodos de reducción de incertidumbre cobran vital importancia ya que han demostrado, tras ser aplicados en diferentes áreas de la ciencia, robustez a la hora de minimizar los efectos negativos causados por las diferentes incertidumbres permitiendo mejorar la calidad de salida de los modelos tratados. Además, debido a que estos sistemas suelen trabajar con grandes conjuntos de datos sobre los cuales es necesario realizar complejas operaciones, éstos sistemas suelen requerir de sistemas de computación de alto rendimiento, los cuales permiten reducir el tiempo de procesamiento al utilizar múltiples unidades de cómputo al mismo tiempo.

3. Método de Reducción de Incertidumbre

Los métodos de reducción de incertidumbre se desarrollan para asistir y participar en el proceso de resolución de problemas de alta magnitud y complejidad computacional, tal como la predicción y prevención de desastres naturales. Este tipo de problemas puede ser clasificado en la categoría de Grand Challenge Problems [13], es decir, problemas de alta complejidad, pero a la vez importantes (desde el punto de vista social y económico) donde se requieren varios órdenes de magnitud de recursos para resolverlos. Un ejemplo puntual de este tipo de problemas es el desarrollo de métodos, sistemas y/o herramientas que aborden la predicción del comportamiento de incendios forestales, teniendo en cuenta las limitaciones presentadas por la predicción clásica.

Las principales limitaciones de la predicción clásica devienen principalmente de la complejidad del modelo de comportamiento del fuego, la incertidumbre en los parámetros de entrada, y de que la predicción se basada en una única simulación. Dado este antecedente, el tratamiento de la incertidumbre se vuelve un aspecto crucial en el desarrollo de sistemas y herramientas de predicción eficaces y eficientes. En este contexto, se ha trabajado en el desarrollado diversos métodos de reducción de incertidumbre que podrían ser aplicados a diferentes fenómenos con características de propagación (i.e., avalanchas, inundaciones, aludes, incendios forestales, etc.). Los métodos desarrollados se alinean con el concepto de predicción guiada por datos, el cual involucra una visión diferente, que requiere la incorporación de una nueva etapa o procedimiento dentro

de la cadena de predicción. Dicha etapa se encarga de efectuar una calibración o ajuste antes de realizar la predicción. Esto persigue como único objetivo, obtener un cierto grupo de combinaciones de parámetros que permita realizar una predicción lo más cercana a la realidad posible. Debido a la falta de precisión en los parámetros de entrada y la dificultad existente para medirlos en tiempo real, los Métodos Guiados por Datos (Data Driven Methods, DDM) consideran un gran número de combinaciones de valores para cada parámetro. O sea que éstos métodos realizan un ajuste para obtener estos valores óptimos de parámetros de entrada. Sin embargo, los DDM obtienen un único conjunto de valores de parámetros de entrada, y para aquellos parámetros de comportamiento dinámico, el valor encontrado en general no resulta de utilidad para describir correctamente el comportamiento del modelo. Esta categoría de métodos se denomina Métodos Guiados por Datos de Única Solución (Data Driven Methods with Unique Solution, DDM-US) [14], [15].

Otra clasificación de los DDM trabaja con solapamiento de diversos casos o combinaciones de parámetros para efectuar las predicciones, esta categoría se denomina Métodos Guiados por Datos con Solución Múltiple Solapada, (Data Driven Methods with Multiple Overlapping Solutions, DDM-MOS). En esta clasificación se encuentra el Sistema Estadístico Híbrido Evolutivo con Modelo de Islas (HESSIM, Hybrid Evolutionary-Statistical System with Island Model) [16]. Su nombre deviene de la utilización de metaheurísticas evolutivas paralelas bajo un enfoque híbrido de colaboración basado en poblaciones y migración.

3.1. Metaheurísticas

La gran mayoría de los problemas de optimización poseen un grado de complejidad tal que no pueden ser resueltos de manera exacta sin hacer uso de grandes cantidades de tiempo. Sin embargo, en algunos problemas o situaciones particulares, no es estrictamente necesario alcanzar una solución exacta, pudiendo ser éstos resueltos haciendo uso de técnicas de optimización aproximadas, lo que permite solucionar el problema en cuestión en períodos considerablemente menores de tiempo.

Los métodos de optimización aproximados brindan soluciones de alta calidad, pero no garantizan obtener una solución óptima global, en contraste con los métodos de optimización exactos los cuales sí aseguran encontrar soluciones óptimas a cambio de un alto costo computacional [17]. De los métodos de optimización aproximados las metaheurísticas han cobrado importancia en los últimos veinte años debido a su capacidad de brindar soluciones satisfactorias aplicadas a problemas de gran tamaño en plazos razonables de tiempo.

Las metaheurísticas son técnicas de optimización aproximadas que suelen utilizarse para resolver problemas de optimización, con rasgos de incertidumbre y dinamicidad, que no pueden ser resueltos de manera exacta sin hacer uso de grandes cantidades de recursos y tiempo [18]. Las metaheurísticas son estrategias generales e inteligentes que tienen como objetivo mejorar y/o diseñar procedimientos heurísticos para resolver problemas de alta complejidad [19].

HESS-IM ha sido implementado haciendo uso de tres metaheurísticas de forma paralela, estas son: Algoritmos Evolutivos (Evolutionary Algorithms, EA) [19], Evolución Diferencial (Differential Evolution, DE) [20] y Optimización por Cúmulo de Partículas (Particle Swarm Optimization, PSO) [21], cada una de ellas es descrita brevemente a continuación.

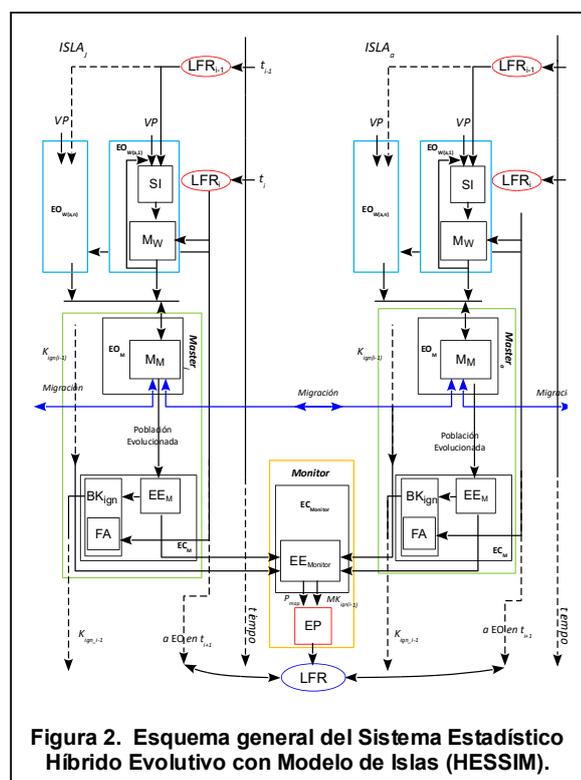


Figura 2. Esquema general del Sistema Estadístico Híbrido Evolutivo con Modelo de Islas (HESSIM).

3.1.1. Algoritmos Evolutivos. Los EAs son métodos de optimización estocásticos cuyo funcionamiento se encuentra inspirado en la teoría de evolución natural de las especies [22]. Pertenecen a la clasificación de metaheurísticas poblacionales, las cuales se caracterizan por conformar una población de soluciones candidatas, que son sometidas a un proceso iterativo que permite imitar el principio de selección natural y supervivencia.

Esto permite la evolución de los individuos a soluciones de mejor calidad mediante la aplicación de ciertos operadores evolutivos: selección, cruce, mutación y migración. Cada individuo posee un valor que permite cuantificar la calidad de la solución que dicho individuo representa para el problema. Este valor se conoce como aptitud o fitness y es calculado utilizando la función matemática que describe el problema. En cada iteración evolutiva, se selecciona cierta cantidad individuos que son perturbados generando así nuevas soluciones candidatas. Posteriormente, se lleva a cabo un procedimiento de reemplazo con el objetivo de determinar aquellos individuos que sobrevivirán o aquellos que serán reemplazados [23]. Al finalizar el ciclo de iteración evolutiva, se espera que algún individuo represente una solución aceptable para el problema tratado.

3.1.2. Evolución Diferencial. El algoritmo de Evolución Diferencial utiliza una población de soluciones candidatas para explorar el espacio de búsqueda, las cuales son creadas de forma aleatoria. El mecanismo central de DE esta basado en la diferencia de vectores, mecanismo por el cual se modifica de forma iterativa la población mediante el uso de operadores. A diferencia de EA, todos los operadores (i.e., mutación, cruzamiento y selección) son aplicados sobre cada uno de los individuos de la población. En primer lugar, se realiza la operación de mutación, la cual modifica la población mediante la aplicación de diferencias de vectores entre los miembros diferentes individuos de la población, de esta forma se determina tanto el grado como la dirección de los nuevos individuos.

Posteriormente, se somete cada individuo mutado junto con el individuo de la población actual al operador de cruce, generando un nuevo vector, llamado vector de prueba. A continuación, se realiza la selección para mantener en la población siguiente (i.e., próxima generación), a aquellos individuos con mejores características, para esto se evalúa para cada vector de prueba el valor de su función objetivo y se contrasta con su correspondiente de la población actual. Una vez comparados, si el vector de prueba tiene mejor o igual valor de función objetivo, se reemplaza el vector actual en la siguiente generación.

3.1.3. Optimización por Cúmulo de Partículas. PSO pertenece a la clasificación de metaheurísticas bioinspiradas, es decir que mantiene cierta analogía con distintos sistemas sociales o presentes en la naturaleza. PSO representa las soluciones candidatas mediante una población de partículas con las cuales explora el espacio de búsqueda. Las partículas poseen un valor de posición y de velocidad los cuales son actualizados en cada iteración del algoritmo. En términos generales, PSO maneja la población de soluciones como un cúmulo de partículas

donde cada partícula i consta de tres componentes. Por un lado, x_i representa la posición de partícula dentro del espacio de búsqueda, v_i describe la velocidad de la partícula y $pbest_i$ es la memoria de la partícula, lo cual representa la mejor posición encontrada hasta el momento. Por otro lado, el algoritmo maneja una componente social (o global a la población) denominada $pbest_g$, que representa la mejor posición encontrada por alguna partícula en toda la población. La velocidad de cada partícula y su posición son actualizadas en cada iteración del algoritmo.

3.2. HESSIM

HESSIM es un método general de reducción de incertidumbre que basa su funcionamiento en un esquema colaborativo de metaheurísticas evolutivas poblacionales. Un esquema general del funcionamiento interno de HESSIM se puede observar en la Fig. 2, en ella se representa un esquema de paralización basado en dos niveles de jerarquía master-worker. En el nivel superior un proceso denominado *monitor* es el encargado de iniciar la cadena de predicción, enviando a cada isla la información del incendio a ser evaluado (i.e., mapa real del incendio, intervalos de tiempo a considerar, parámetros, etc.). El envío de esta información se produce entre el *monitor* y los procesos *master* de cada isla.

Cada proceso *master* (en cada isla particular) instancia una Etapa de Optimización (EOM), donde se da inicio a la metaheurística que cada isla tiene asignada (i.e., EA, DE o PSO). Tal como se puede observar, la Etapa Metaheurística se encuentra dividida en dos subetapas: por un lado, la Etapa Metaheurística del proceso master (M_M), y por el otro, la Etapa Metaheurística de los procesos workers (M_W). En términos generales, diremos que la etapa M_W se encarga de la evaluación de aptitud de los individuos, y la etapa M_M se ocupa del resto de las operaciones involucradas en cada metaheurística.

Los *workers* de cada isla realizan la evaluación de aptitud de los individuos mediante la utilización del simulador de incendios (SI). El SI se basa en el modelo definido por Rothermel [24] e implementado mediante la librería fireLib [25]. Para realizar la evaluación de aptitud en t_i es necesario contar con la línea de fuego real del incendio (LFR) en t_{i-1} (i.e., LFR _{t_{i-1}}) y los valores de los parámetros de entrada, los cuales se almacenan en los vectores de parámetros (VP). Una vez evaluados los individuos, éstos son enviados junto a su correspondiente valor de aptitud, a la etapa M_M. Esta etapa, además de realizar el resto de las operaciones de cada metaheurística (i.e., alteración de individuos, evolución de la población, etc.), se encarga de la migración de individuos hacia las islas vecinas. El proceso de migración se encuentra dividido en tres pasos: selección, envío/recepción y reemplazo. El proceso de *selección* consiste en escoger, de la población

actual, aquellos individuos que serán enviados al resto de las islas. Se utiliza un criterio semi-elitista, donde el 50% de los individuos a migrar corresponde a los mejores y el otro 50% se selecciona de forma aleatoria.

El siguiente paso, *envío/recepción*, es el mecanismo que efectúa la comunicación entre las diferentes islas, el cual envía y recibe individuos cada cierta cantidad de generaciones, también llamado frecuencia de migración, siguiendo una topología de anillo. La frecuencia de migración está establecida como un parámetro variable que depende de la cantidad máxima de generaciones.

El proceso de *reemplazo* determina los individuos de la población actual que serán reemplazados por los que arriben provenientes de otras islas; el criterio utilizado es similar al proceso de selección, donde los peores individuos son reemplazados por los mejores y el resto son reemplazados de forma aleatoria.

Finalmente, cuando las poblaciones de las distintas islas han evolucionado, éstas son enviadas a la Etapa de Calibración del proceso master (EC_M). En esta etapa se calcula un mapa de probabilidad, en función de todos los individuos, el cual se utiliza para generar el valor clave de ignición (K_{ign}, por sus siglas en inglés). El K_{ign} representa el patrón de comportamiento del incendio y es obtenido en la etapa Búsqueda del K_{ign} (BK_{ign}).

En todo instante de tiempo, en cada isla se genera un valor clave de ignición y un mapa de probabilidad, los cuales son enviados a la etapa de calibración del proceso *monitor* (EC_M), donde en la etapa estadística (EE_M) indica el número de isla cuyo par de valores ha obtenido el mejor valor de fitness. Éstos valores finalmente son ingresados en la etapa de predicción (EP) la cual realiza el cálculo de predicción (i.e., línea de fuego predicha) para el próximo instante de tiempo LFR _{$t+1$} .

Es importante mencionar que el rendimiento del método se evalúa tanto en términos de la calidad de predicción, como así también en relación al tiempo total de procesamiento. Si bien la calidad de predicción es considerada la medida de mayor importancia, el rendimiento computacional no deja de tener menor relevancia, ya que todo sistema de predicción debe garantizar obtener las predicciones con antelación suficiente a la ocurrencia del fenómeno en la realidad.

3.2.1. Pasos de simulación y pasos de predicción.

Debido al funcionamiento de este método, es importante destacar la diferencia entre un paso de simulación y uno de predicción: para ello observemos la Fig. 3. Aquí podemos ver que, para un incendio de duración determinada, existen t_s muestras representativas del avance del fuego, por lo tanto, $s-1$ pasos de simulación y $s-2$ pasos de predicción. Esto es debido a que éste método precisa de una etapa de calibración de parámetros de entrada, por lo que no pueden efectuar predicciones en el primer instante de tiempo (o sea en el primer paso de

simulación comprendido entre t_1 y t_2). El primer paso de simulación es utilizado para realizar la primera calibración de los parámetros de entrada. Luego de realizar dicha calibración el sistema se encuentra en condiciones para efectuar la primera predicción correspondiente al segundo paso de simulación, que equivale al primer paso de predicción (es decir, entre t_2 y t_3). Por lo tanto, para todo

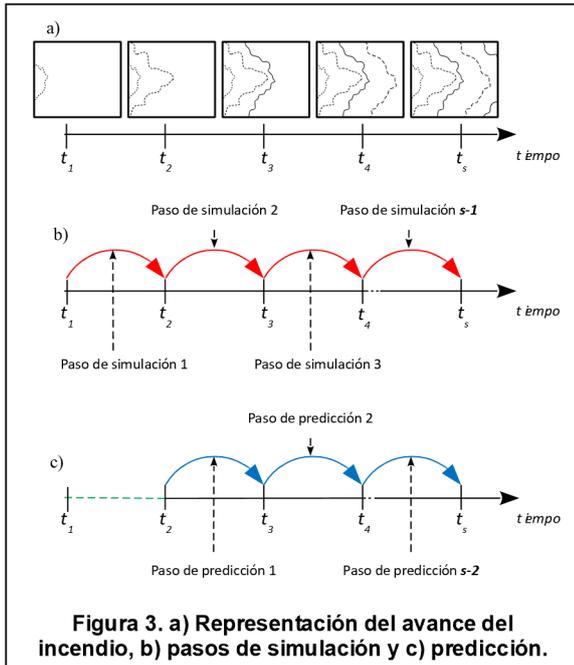


Figura 3. a) Representación del avance del incendio, b) pasos de simulación y c) predicción.

caso de prueba se obtienen s-2 valores de calidad de predicción, uno para cada paso de predicción.

3.3. Computación de Alto Rendimiento

La predicción de la propagación de incendios forestales constituye un desafío desde el punto de vista computacional, dada la complejidad que involucran los modelos, la necesidad de métodos numéricos y la eficiente administración de los recursos para obtener los resultados. En este sentido, la computación de alto rendimiento ha permitido abordar problemas cada vez más grandes y complejos, beneficiado por el avance tecnológico de los últimos años.

Tal como se mencionó en la sección anterior, HESSIM utiliza una arquitectura de doble jerarquía master/worker la cual en su implementación original incorporó únicamente paralelismo de memoria distribuida bajo la Interfaz de Paso de Mensajes (MPI, Message Passing Interface) [26]. En la Fig. 4 se puede observar la existencia de tres tipos de procesos: a) *monitor o master global*, b) *worker de nivel 1 o master de isla*, y c) *worker de isla*. Es decir, en el nivel superior (nivel 1) el proceso *monitor*

coordina a sus respectivos procesos *workers*. En el nivel inferior de jerarquía (nivel 2) cada uno de éstos procesos operan como *masters de isla*. Finalmente, cada isla contiene su conjunto de *workers de isla*. El proceso de comunicación entre los procesos se ve reflejado en las

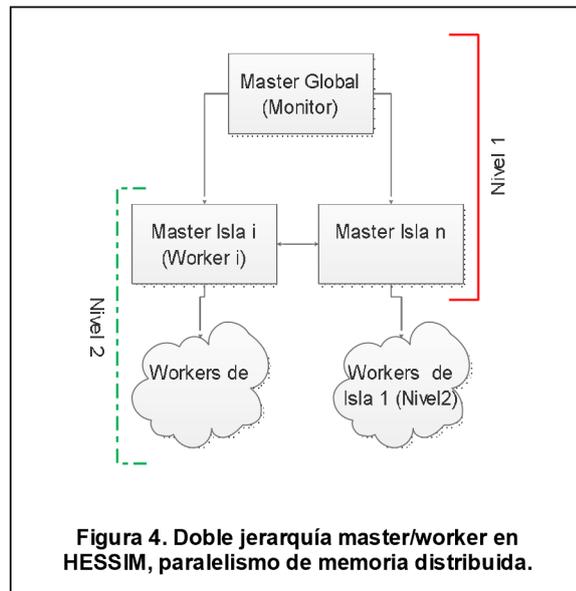


Figura 4. Doble jerarquía master/worker en HESSIM, paralelismo de memoria distribuida.

conexiones entre cada entidad (Fig. 4) y se corresponde con el esquema de la Fig. 2, donde se puede apreciar que la migración de soluciones candidatas se efectúa entre las islas a cargo de cada proceso *master de isla*.

Debido a que las metaheurísticas involucradas en el proceso de optimización necesitan someter cada solución candidata al simulador de incendios, para así obtener su fitness (i.e., función de fitness), resulta indispensable reducir la cantidad de evaluaciones de la función objetivo, o bien el tiempo de procesamiento asociado a este proceso. Siguiendo este objetivo se decidió implementar HESSIM bajo un esquema de paralelismo heterogéneo de memoria distribuida y memoria compartida bajo MPI/CUDA (Compute Unified Device Architecture) [29].

Debido a que HESSIM utiliza el simulador fireLib, el cual originalmente fue implementado para CPU, y dado que la simulación es la operación de mayor costo computacional se realizó una adaptación de fireLib para GPU basada en el trabajo de [27] (CudaFGM).

4. Experimentación

La evaluación de HESS-IM se llevó a cabo mediante su aplicación en un incendio real ocurrido en Capilla del Monte, Córdoba (Argentina) entre el 29 de agosto al 3 de septiembre de 2008, donde el fuego consumió un total de 51.852,063 hectáreas. El período de duración total del

incendio ha sido dividido en lapsos menores de tiempo denominados pasos de simulación, en la Tabla 1 se presenta toda la información detallada de cada instante de tiempo. Es importante observar que los valores del tiempo corresponden a intervalos de 1440 minutos (un día).

ta

Tabla 1: Detalle del incendio evaluado.

Paso	Fecha	Min. inicial	Min. final
1	29/08/2008	0	1440
2	30/08/2008	1440	2880
3	31/08/2008	2880	4320
4	01/09/2008	4320	5760
5	02/09/2008	5760	7200
6	03/09/2008	7200	8640

Por lo tanto, la comparación de calidad se lleva cabo en cada paso de predicción de los experimentos.

4.1.1. Evaluación del método. El rendimiento del método ha sido evaluado en base a dos criterios: a) tiempo de procesamiento y b) calidad de predicción. La evaluación de tiempo consiste en instrumentar el código de tal manera que se permita registrar y comparar el tiempo de cómputo (para el mismo incendio) total de la implementación basada en MPI versus el tiempo de procesamiento de la implementación heterogénea MPI/CUDA. Respecto a la calidad de predicción del método, ésta es evaluada mediante una función de aptitud la cual se basa en el índice de Jaccard [28] y se describe en la ecuación 1, donde A representa el conjunto de celdas en el mapa real sin el subconjunto de celdas quemadas antes de comenzar las simulaciones, y B representa el conjunto de celdas en el mapa de simulación sin el subconjunto de celdas quemadas antes de comenzar la simulación. El valor resultante estará entre 0 y 1, donde un valor igual a 1 corresponde a una predicción perfecta, ya que significa que el área predicha es igual a la superficie real quemada. Y un valor de aptitud igual a 0 indica que el error ha sido máximo debido a que la predicción no coincide con la realidad en absoluto. Debido a que el resultado numérico de ambas implementaciones de HESSIM es idéntico, la calidad de predicción ha sido contrastada contra una implementación previa del método la cual utilizaba como técnica de optimización un Algoritmo Evolutivo, este método se denomina ESSIM (para más información ver [29]). Es importante destacar que otras metodologías similares existentes en la literatura no han liberado su código fuente por lo que no es posible realizar estudios comparativos de forma directa, además en [29] ESSIM fue contrastado contra otras metodologías utilizando casos de quemas controladas.

$$fitness = \frac{A \cap B}{A \cup B} \quad (1)$$

Debido al comportamiento no determinístico de HESSIM, y con el propósito de evitar cualquier amenaza a la validez de la experimentación, para ambas evaluaciones se efectuó un total de 30 ejecuciones de dicho experimento y tomando el promedio de éstas, registrando la calidad de predicción en cada instante de tiempo, como así también el tiempo total consumido. La experimentación se llevó a cabo en un cluster Linux compuesto por seis nodos con las características de hardware descritas en la Tabla 2.

Tabla 2: Detalle de hardware de cada nodo.

Nodo	CPU	RAM	GPU
1	AMD FX 6300	16GB	TESLA 2075
2	AMD FX 6300	16GB	TESLA 2075
3	AMD FX 6300	16GB	GTX 480
4	AMD FX 6300	16GB	GTX 480
5	AMD FX 6300	16GB	GTX 650
6	AMD FX 6300	16GB	GTX 650

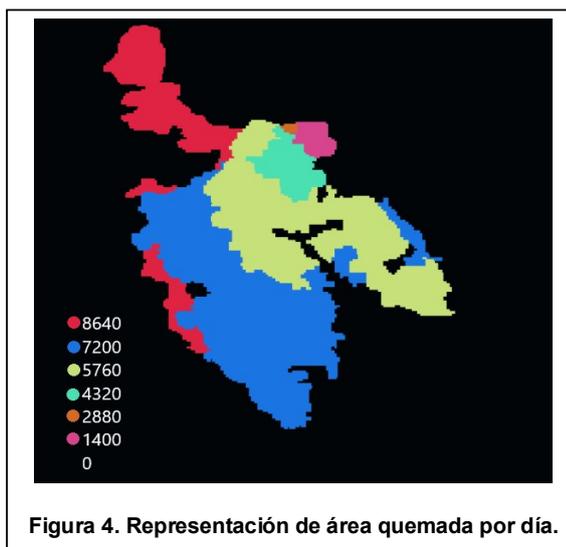


Figura 4. Representación de área quemada por día.

Los nodos se interconectaron mediante una Gigabit Ethernet bajo un entorno MPI/CUDA, modelo de programación preferido al diseñar códigos para aplicaciones GPGPU (General Purpose Graphics Processing Units).

Los resultados obtenidos en términos de tiempo de procesamiento se pueden observar en la Tabla 3, tal resultado representa el promedio de 30 ejecuciones debido a que HESSIM posee un comportamiento estocástico. Tal como se puede observar, en esta primera aproximación, se deja evidencia del incremento de rendimiento en la implementación heterogénea sobre la versión de memoria compartida (sólo CPU). Es importante destacar que no se

dispone por el momento de una implementación basada en MPI/CUDA del método ESSIM, debido a ello sólo el tiempo de la implementación MPI es considerado.

En términos de calidad de predicción, los resultados pueden observarse en la Fig. 5 donde se ha graficado el promedio de las 30 ejecuciones para cada paso de predicción. En el eje x se representa el tiempo de avance del incendio en minutos correspondientes al paso de predicción evaluado, y en el eje y el valor de aptitud o fitness.

Tabla 3: Comparación de tiempo de perecimiento.

Método	MPI	MPI/CUDA
HESSIM	17 horas 33 min	6 horas 20 min
ESSIM	18 horas 20 min	-

Tal como se puede observar en la Fig. 5, el primer valor de predicción (paso de predicción 1) corresponde al paso 2 de la Tabla 1 (paso de simulación 2), esto se debe a lo mencionado anteriormente, que el método no puede efectuar predicciones en el primer instante de tiempo; ya que al comenzar el proceso predictivo éste se encuentra calculando el primer valor K_{ing} (en la Etapa de Calibración), eso sucede en el primer paso de simulación (ver Fig. 5). La predicción comienza en el paso de simulación 2, donde se registra valor de calidad de predicción elevado, posteriormente la calidad de predicción se mantiene por encima de ESSIM y en el paso de simulación 5 se registra el mayor valor de calidad de predicción. Es importante notar que la evaluación de calidad de predicción corresponde a un incendio real donde se desconocen las variables que condicionan el comportamiento del fuego.

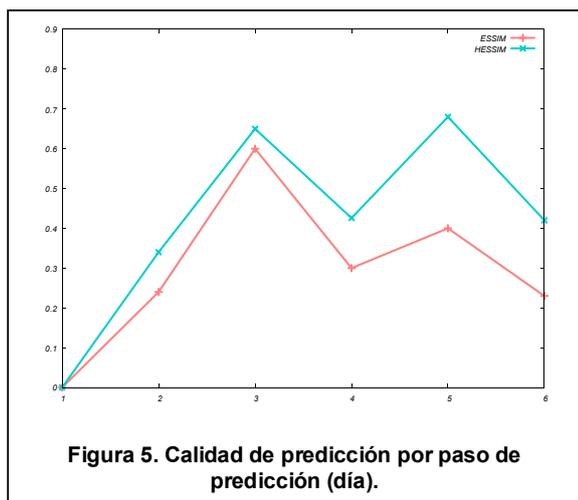


Figura 5. Calidad de predicción por paso de predicción (día).

5. Conclusiones

En el presente trabajo se ha presentado la primera implementación del método de reducción de incertidumbre HESSIM bajo un esquema de paralelismo heterogéneo basado en MPI/CUDA. Además, se ha evaluado el desempeño de esta implementación en un caso real de incendio forestal ocurrido en Córdoba en el año 2008. La experimentación se ha desarrollado bajo condiciones de completa incertidumbre obteniendo valores de predicción superiores a una implementación previa. En término de tiempo de utilización de recursos, la implementación MPI/CUDA ha logrado obtener un tiempo de respuesta superior. Evidentemente queda aún mucho trabajo por realizar, entre ellos se destaca la aplicación de HESSIM a nuevos casos de incendios reales, como así también realizar un proceso de sintonización de los parámetros de las metaheurísticas participantes con el objetivo de sacar el máximo provecho al método. Por otro lado, también se considera importante trabajar en el procesamiento automático de las imágenes satelitales, con el propósito de poder alimentar al método de forma automática y aproximarnos aún más al objetivo de poder predecir este tipo de fenómenos en tiempo real. Y finalmente, y no menos importante, se considera pertinente continuar trabajando en la optimización del método en términos de utilización de recursos, ya que el propósito de todo método de predicción es obtener resultados con suficiente tiempo de antelación como para poder tomar medidas al respecto y minimizar los efectos negativos.

6. Referencias

- [1] W. Hough, "Fire and Human Civilization", *Proc. Am. Philos. Soc.*, vol. 71, n° 6, pp. 403–406, 1932.
- [2] D. M. Molina Terrén, *Incendios forestales: fundamentos, lecciones aprendidas y retos de futuro*. AIFEMA, 2010.
- [3] S. Graham, "Global Fire Monitoring", 22-oct-1999. [En línea]. Disponible en: https://earthobservatory.nasa.gov/Features/GlobalFire/fire_2.php. [Accedido: 01-may-2018].
- [4] P. Morgan *et al.*, "Mapping fire regimes across time and space: Understanding coarse and fine-scale fire patterns", *Int. J. Wildl. Fire*, vol. 10, pp. 329–342, 2001.
- [5] R. Buyya, *High Performance Cluster Computing: Architectures and Systems*. Prentice Hall, PTR, NJ, USA, 1999.
- [6] M. R. Arahall, M. Berenguel Soria, y F. Rodríguez Díaz, *Técnicas de predicción con aplicaciones en ingeniería*. Sevilla: Secretariado de Publicaciones, Universidad de Sevilla, 2006.
- [7] G. Wallace, "A Numerical Fire Simulation-Model", *Int. J. Wildl. Fire*, vol. 3, n° 2, p. 111, 1993.
- [8] M. A. Finney, "FARSITE: Fire Area Simulator — Model Development and Evaluation", *USDA For. Serv.*

- Res. Pap.*, n° February, p. 47, 1998.
- [9] P. L. Andrews, C. D. Bevens, R. C. Seli, A. Andrews, P. L.; Bevens, y C. D.; Seli, "BehavePlus fire modeling system, version 4.0: User's Guide Revised", 2008.
- [10] F. A. Heinsch y P. L. Andrews, "BehavePlus fire modeling system, version 5.0: Design and Features", 2010.
- [11] A. M. . Lopes, M. G. Cruz, y D. X. Viegas, "FireStation -An integrated software system for the numerical simulation of fire spread on complex topography".
- [12] P. Mangana, P. Moura, L. M. Ribeiro, D. X. Viegas, y A. G. Lopes, "Operational application of a decision support tool in fire management in Portugal", *For. Ecol. Manag.*, vol. 234, n° Supplement 1, p. 243, 2006.
- [13] B. Wilkinson y C. M. Allen, *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers*. Pearson/Prentice Hall, 1999.
- [14] K. Beven y A. Binley, "The future of distributed models: Model calibration and uncertainty prediction", *Hydrol. Process.*, vol. 6, n° 3, pp. 279–298, jul. 1992.
- [15] B. Abdalhaq, *A Methodology to enhance the prediction of forest fire propagation*. Universitat Autònoma de Barcelona, 2004.
- [16] M. Méndez-Garabetti, G. Bianchini, M. L. Tardivo, P. G. Caymes Scutari, y G. V. Gil Costa, "Hybrid-Parallel Uncertainty Reduction Method Applied to Forest Fire Spread Prediction", *J. Comput. Sci. Technol. ISSN-e 1666-6038, Vol. 17, N°. 1, 2017, págs. 12-19*, vol. 17, n° 1, pp. 12–19, 2017.
- [17] E.-G. Talbi, *Metaheuristics: from design to implementation*. John Wiley & Sons, 2009.
- [18] L. T. Bui, H. A. Abbass, y J. Branke, "Multiobjective optimization for dynamic environments", *2005 IEEE Congr. Evol. Comput.*, vol. 3, pp. 2349–2356, 2005.
- [19] F. Glover, "Future paths for integer programming and artificial intelligence", *Comput. {&} Oper. Res.*, vol. 13, n° 5, pp. 533–549, 1986.
- [20] R. Storn y K. Price, "Differential Evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces", Berkeley, CA, 1995.
- [21] F. Marini y B. Walczak, "Particle swarm optimization (PSO). A tutorial", *Chemom. Intell. Lab. Syst.*, vol. 149, pp. 153–165, dic. 2015.
- [22] C. Darwin, *On the origins of species by means of natural selection*. 1859.
- [23] X. Yu y M. Gen, *Introduction to Evolutionary Algorithms*, 1ª ed., vol. 0. London: Springer-Verlag London, 2010.
- [24] R. C. Rothermel, *A mathematical model for predicting fire spread in wildland fuels*, vol. II. Res. Pap. INT-115, US Dept. of Agric., Forest Service, Intermountain Forest and Range Experiment Station. (Ogden, UT.), 1972.
- [25] Collin D. Bevens, "fireLib User Manual and Technical Reference", 1996.
- [26] W. Gropp, E. Lusk, y A. Skjellum, *Using MPI: portable parallel programming with the Message-Passing-Interface*. 2014.
- [27] F. A. Sousa, R. J. N. dos Reis, y J. C. F. Pereira, "Simulation of surface fire fronts using fireLib and GPUs", *Environ. Model. Softw.*, vol. 38, pp. 167–177, dic. 2012.
- [28] R. Real y J. M. Vargas, "The Probabilistic Basis of Jaccard's Index of Similarity", *Syst. Biol.*, vol. 45, n° 3, pp. 380–385, sep. 1996.
- [29] M. Méndez-Garabetti, G. Bianchini, P. Caymes-Scutari, y M. L. Tardivo, "Increase in the quality of the prediction of a computational wildfire behavior method through the improvement of the internal metaheuristic", *Fire Saf. J.*, vol. 82, pp. 49–62, 2016.