Método de Reducción de Incertidumbre basado en HPC*

Germán Bianchini[1], Ana Cortés[2], Tomàs Margalef[2] y Emilio Luque[2]

[1] Departamento de Ingeniería en Sistemas de Información,
Universidad Tecnológica Nacional - Facultad Regional Mendoza,
CP 5500 (Mendoza) Argentina
gbianchini@frm.utn.edu.ar
[2] Dpto. de Arquitectura de Computadores y Sistemas Operativos,
Universitat Autònoma de Barcelona, CP 08193 Bellaterra (Barcelona) España
{ana.cortes,tomas.margalef,emilio.luque}@uab.es

Resumen La problemática existente a raíz de la falta de exactitud que se encuentra en los parámetros de entrada en cualquier modelo científico o físico, puede producir graves consecuencias en la salida del mismo si éste se trata de algún sistema crítico. Además, al citado problema deben sumarse las limitaciones impuestas por los propios modelos, las restricciones que agregan las soluciones numéricas y, por qué no, las provenientes de las propias implementaciones y versiones informáticas. Por tal motivo, resulta de gran interés el desarrollo de métodos informáticos que se enfoquen en el tratamiento de la incertidumbre de dichos valores de entrada para lograr así una predicción lo más confiable posible por parte del modelo en cuestión. En el presente trabajo se presenta un método basado en High Performance Computing en combinación con Cálculo Estadístico, el cual se ha evaluado y verificado en casos reales aplicándolo a un modelo de comportamiento de incendios forestales.

1. Introducción

El uso de modelos para representar sistemas físicos se ha vuelto una modalidad convencional en diversas áreas científicas. Generalmente, los modelos reciben un conjunto de parámetros de entrada representando condiciones particulares y proveen una salida que refleja la evolución del mismo en el tiempo. Además, es común que tales modelos se encuentren integrados en herramientas de simulación informática [8,10,11,16,17,19]. No obstante, en muchos casos, los modelos presentan una serie de limitaciones, las cuales suelen estar relacionadas con la gran cantidad de parámetros de entrada que manejan. Tales parámetros suelen presentar algún tipo de incertidumbre debido a la imposibilidad de medirlos en tiempo real, y por lo tanto se deben estimar a partir de medidas indirectas. Por otra parte, en ocasiones los modelos no pueden ser resueltos analíticamente y

* Este trabajo ha sido financiado por la Comisión Interministerial de Ciencia y Tecnología (CICYT) bajo contrato TIC2001-2592 y por la Comisión Europea bajo contrato EVG1-CT-2001-00043 SPREAD.

deben aplicarse métodos numéricos, los cuales no dejan de ser una aproximación de la realidad (aun sin considerar las limitaciones que presenta la traducción de estas soluciones cuando son efectuadas sobre computadoras).

Un enfoque prometedor para resolver este problema es el uso de asimilación de datos en tiempo real combinado con algún método computacional para analizar la desviación de la predicción de acuerdo al comportamiento real. De esta manera sería posible determinar los valores de los parámetros que reproducen el comportamiento correcto en el momento actual y usar dichos valores para un paso siguiente de simulación. Se han desarrollado varios métodos de asimilación de datos para optimizar los parámetros de entrada [1,4,12,18], los cuales, en general, operan sobre un gran número de valores de entrada, y, por medio de algún método de optimización, se enfocan en la búsqueda de un único conjunto de valores que describa el comportamiento previo de la mejor manera. Por lo tanto, es de esperar que el mismo conjunto de valores pueda ser usado en el futuro inmediato. A los sistemas que aplican este tipo de metodología se los conoce bajo el nombre de Métodos Conducidos por Datos (**Data-Driven Methods**).

Sin embargo, a pesar de que esta clase de métodos mejora los resultados que pueden obtenerse, los métodos conducidos por datos adolecen de un mismo problema: encuentran un único conjunto de valores, y, como se ha mencionado anteriormente, para aquellos parámetros que presentan un comportamiento dinámico, el valor hallado no resulta de utilidad para describir correctamente el futuro inmediato del modelo en cuestión.

Teniendo en mente el problema inicial planteado y las posibilidades existentes, el presente trabajo ofrece un método alternativo a las metodologías previamente comentadas. El método propuesto se basa en dos grandes pilares: **Análisis Estadístico** y **High Performance Computing** (HPC). Su propósito es hallar un patrón del comportamiento del modelo en el cual se aplica, independientemente de los valores particulares de los parámetros.

Dadas las características del método, se ha clasificado en una nueva rama que extiende a la clasificación anterior, la cual se ha dado en llamar Métodos Conducidos por Datos basados en Solución Solapada Múltiple[6] (**Data-Driven Methods using Multiple Overlapping Solution**).

El resto del presente artículo está organizado de la siguiente manera: En la sección 2 se comentan las principales características de los Métodos Conducidos por Datos y la descripción del método basado en Solución Solapada Múltiple. La sección 3 describe más en detalle el método propuesto aplicado a la predicción de incendios forestales y en la sección 4 se brindan detalles de implementación. Los experimentos y comparativas de la aplicación del método propuesto contra la aplicación clásica del mismo modelo se presenta en la sección 5. Finalmente, en la sección 6 se reportan las principales conclusiones.

2. Métodos Conducidos por Datos

Bajo esta denominación encontramos agrupados aquellos métodos que, buscando una solución al problema manifestado en los parámetros de entrada de

un modelo, hacen uso de técnicas de optimización que les permitan calibrar el conjunto de parámetros de entrada. El objetivo de tal optimización es hallar un conjunto de valores ideal. Si estos valores se aplican al modelo en cuestión (que normalmente puede haber sido implementado en un simulador), sería posible describir correctamente el comportamiento previo, es decir, el comportamiento que de alguna forma ha sido utilizado para calibrar o hallar el conjunto de parámetros. Por lo tanto, normalmente se espera que el mismo conjunto de valores pueda ser utilizado para describir el comportamiento de un futuro inmediato.

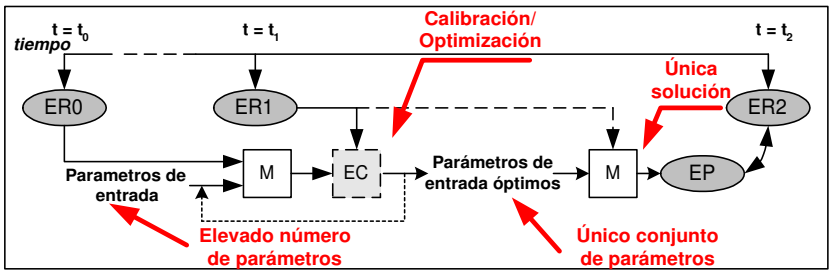

Figura 1. Diagrama esquemático de los Métodos Conducidos por Datos (ERX: estado real en tiempo X, M: modelo, EC: etapa de calibración, EP: estado propuesto por el modelo)

Esquemáticamente, los Métodos Conducidos por Datos operan sobre una etapa que llamaremos Etapa de Calibración (EC). En la figura 1 podemos apreciar las características. Los Métodos Conducidos por Datos trabajan sobre grandes cantidades de valores (diferentes combinaciones de las entradas que producen diversos escenarios). Esta característica es la que explica el tiempo extra que requieren para poder computar toda la información. Sin embargo, como se mencionó en la sección previa, cuando estamos frente a situaciones en las que los parámetros cambian dinámicamente, el conjunto de valores hallados genera una única solución que puede no coincidir con la situación real.

2.1. Métodos Conducidos por Datos basados en Solución Solapada Múltiple

En la figura 2 puede apreciarse un esquema gráfico de este tipo de metodología. Al igual que en la figura 1, la etapa encargada del procesamiento específico del método se ha rotulado como EC (Etapa de Calibración). De esta manera omitimos los detalles específicos, mostrando solamente la forma conceptual de operación. Entre ambas figuras, notamos que la principal diferencia radica en que los Métodos Conducidos por Datos realizan una predicción (EP) en función del resultado que han hallado en la etapa de calibración y los datos reales (ERX) del instante de tiempo precedente. En cambio, el caso de Solapamiento Múltiple

se basa en la totalidad de resultados obtenidos en la etapa de calibración y el estado real anterior.

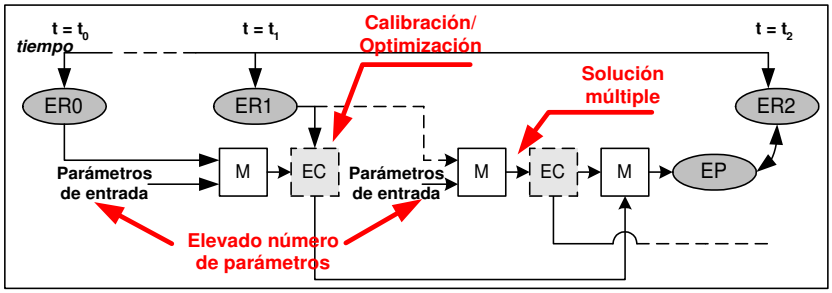

Figura 2. Diagrama esquemático de los Métodos Conducidos por Datos basados en Solapamiento Múltiple (ERX: estado real en tiempo X, M: modelo, EC: etapa de calibración, EP: estado propuesto por el modelo)

Seguidamente, veremos en más detalle las características de un sistema, denominado S^2F^2M [7], que pertenece a la categoría de método conducido por datos basado en solución solapada múltiple que ha sido implementado para la predicción de incendios forestales.

3. S^2F^2M - Sistema Estadístico para la Gestión de Incendios Forestales

El sistema S^2F^2M (Statistical System for Forest Fire Management) se basa en el concepto de experimento diseñado. En esta clase de experimento se realizan cambios deliberados o intencionados en las variables controladas de un sistema. Se observan los resultados obtenidos y luego se hace una inferencia o toma de decisión acerca de las variables responsables de los cambios. Cuando son varios los factores potencialmente importantes (en el caso de incendios forestales nos referimos a tipo de vegetación, humedad, velocidad del viento, etc.), una buena estrategia es usar algún tipo de experimento factorial. Un experimento factorial es aquel en el que los factores se hacen variar al mismo tiempo [15] (por ejemplo cambiando las condiciones del viento, el contenido de humedad en el combustible y ciertos parámetros de la vegetación). Cada situación particular resultante de una configuración de valores es lo que llamamos **escenario**.

Para cada parámetro en el modelo es preciso definir un rango y un valor de incremento con el cual recorrer el intervalo planteado. Para un parámetro dado i el intervalo e incremento asociado se expresa como:

$$[Cota\_Inferior\_i,\ Cota\_Superior\_i],\ Incremento\_i$$

Luego, de cada parámetro i, es posible obtener un número C_i (cardinalidad del dominio del parámetro), el cual es calculado de la siguiente manera:

$$C_i = ((Cota\_Superior\_i - Cota\_Inferior\_i) + Incremento\_i)/Incremento\_i$$

Finalmente, a partir de la cardinalidad de cada parámetro, es posible calcular el número total de escenarios obtenidos de las variaciones de todas las posibles combinaciones.

$$\#Escenarios = \prod_{i=1}^{p} C_i$$

siendo p el número de parámetros.

A partir de este punto, si queremos conocer si una porción del terreno (a la cual denominamos 'celda') se quemará o no en un determinado intervalo de tiempo, lo que hacemos es calcular su probabilidad de ignición. Si consideramos que n_A es el número de escenarios en el cual la celda A se ha quemado, entonces calculamos:

$$P_{ign}(A) = n_A/\#Escenarios$$

Generalizando este razonamiento, calculamos la probabilidad de ignición a todo el conjunto de celdas que conforman el terreno bajo estudio. En consecuencia, obtenemos una matriz con valores de probabilidad asociados a cada celda. De dicho conjunto, aquel subgrupo cuyo P_{ign} sea mayor o igual a un cierto valor que llamaremos P_K, $0 <= P_K <= 1$, conforma lo que definimos como Mapa de Probabilidad con probabilidad P_K.

Una vez obtenida la matriz de salida, la cual incluye a todos los mapas de probabilidad, el paso siguiente consiste en comparar el estado real contra la matriz. El objetivo que persigue tal comprobación es la búsqueda de un valor clave particular de P_K cuyo mapa de probabilidad concuerde de la forma más precisa posible con el estado actual. Dicho de otra manera, se busca un número de ignición clave (K_{ign}). Por lo tanto, el mapa de probabilidad asociado debe cumplir la siguiente condición:

$$\{x : P_{ign}(x) \geq K_{ign}/\#Escenarios \ \mid K_{ign} \in \mathbb{N}\}$$

con P_{ign} variando entre los valores $K_{ign}/\#Escenarios$ a 1. Dicho de otra manera, es el conjunto de celdas (x) que se han quemado al menos K_{ign} veces.

En el ejemplo de la figura 3 se puede comprender esta idea con más claridad. En dicha gráfica, el área quemada real se ha representado a través de una forma irregular. El objetivo es hallar un corte horizontal del cono (el cual representa al terreno en el cual cada celda indica con la altura el número de escenarios en los que se ha quemado) de forma tal que el área resultante concuerde de la mejor forma posible con el área real. Desde el punto de vista del método, la altura del cono respecto a la base será el valor K_{ign}.

Para la altura a, el valor del fitness es $fitness(a)$, para b es $fitness(b)$, etc. Sobre el margen derecho de la figura vemos la superposición de las secciones.

En este ejemplo, conformado por cuatro casos, es posible ver que el caso b es el que presenta el área más similar al caso real. Por lo tanto, el K_{ign} hallado es b, valor que define el área más semejante a la situación real. De esta manera, podemos utilizar este valor para predecir el siguiente paso. En otras palabras, la altura b hallada en el instante i constituye el K_{ign} que se usará en i+1 para efectuar la predicción (es decir, un nuevo corte a la altura b que ofrecerá una nueva superficie).

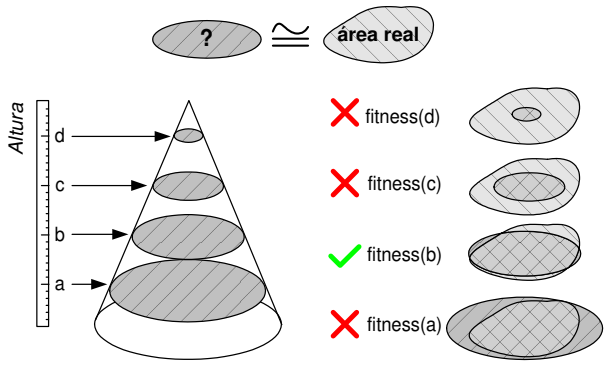

Figura 3. Esquema gráfico del método S^2F^2M

Un esquema del método se presenta en la figura 4. En ella se aprecia que el proceso de predicción requiere un paso de la etapa de calibración al comienzo de todo el proceso (entre t_0 y t_1) para obtener el primer valor K_{ign}. Una vez que esto ocurre por primera vez, ambas tareas (calibración para t_i+1 y predicción para t_i) serán solapadas en tiempo t_i. Este proceso se repetirá a lo largo de la ejecución a medida que el sistema sea alimentado con nueva información acerca del frente de fuego.

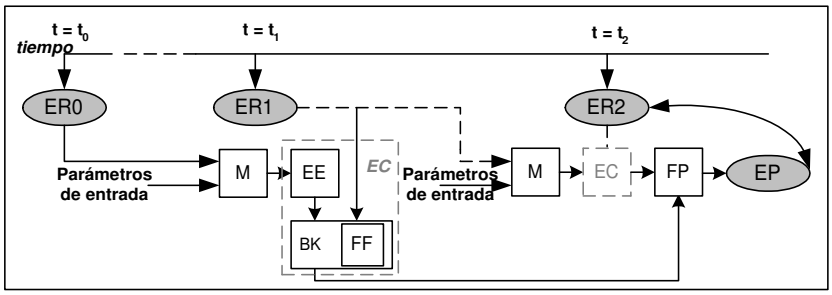

Figura 4. Diagrama esquemático del método S^2F^2M (ERX: estado real en tiempo X, M: modelo, EC: etapa de calibración, EE: etapa estadística, BK: búsqueda de K_{ign}, FF: función Fitness, FP: etapa de predicción, EP: estado propuesto por el modelo)

4. Implementación

Dada las características del método, el elevado número de iteraciones y cálculos que debe efectuar el mismo nos conducen hacia una implementación paralela, ya que la misma operación puede llevarse a cabo sobre grupos de datos disjuntos en forma simultánea. El incremento en la velocidad, y por ende la reducción del tiempo, son fundamentales para que el mismo se torne aplicable.

El paradigma utilizado ha sido el Master-Worker [9,13]. El proceso Master es el responsable de descomponer el problema en pequeñas tareas y distribuirlo entre el conjunto de procesos Workers, así como también de recuperar todos los resultados parciales para producir el resultado final del cómputo. Los Workers procesan los datos recibidos, efectuando las simulaciones pertinentes, y retornan el resultado al Master.

El método se ha implementado en un sistema operacional que incorpora un kernel de simulación [5] y aplica una metodología para evaluar el fitness. El sistema se ha desarrollado sobre un cluster de PC LINUX, utilizando MPI [21] como librería de paso de mensajes.

4.1. El kernel de simulación

S^2F^2M utiliza como núcleo de simulación el simulador propuesto por Collin D. Bevins, el cual está basado en la librería fireLib [5], la cual respeta los modelos de combustible definidos por Rothermel [20]. **fireLib** es una librería de funciones escrita en lenguaje C para predecir el índice de propagación e intensidad de un incendio forestal. Deriva directamente del algoritmo BEHAVE [2], pero con optimizaciones para aplicaciones altamente iterativas tales como las simulaciones basadas en celdas u ondas. En particular, el presente simulador utiliza una aproximación de autómata celular para evaluar el crecimiento del fuego. El terreno se divide en celdas cuadradas y se utiliza la relación con las celdas vecinas para evaluar si la celda se quemará o no, y en qué momento ocurrirá esto.

El simulador acepta como entrada mapas de terreno, características de la vegetación, el viento y el mapa de ignición, y como salida brinda una matriz en donde cada celda está rotulada con su tiempo de ignición.

4.2. Función de Fitness

Para evaluar la respuesta del método, se definió una función de fitness, la cual se especificó de la siguiente manera:

$$\text{Fitness} = \frac{(\#cells\bigcap - \#IgnitionCells)}{(\#cells\bigcup - \#IgnitionCells)}$$

donde, $\#cells\bigcap$ representa el número de celdas en la intersección entre el resultado de la simulación y el mapa real, $\#cells\bigcup$ es el número de celdas en la unión del resultado de la simulación y el caso real, y $\#IgnitionCells$ representa el número de celdas quemadas antes de iniciar la simulación. Un valor de fitness

igual a 1 se corresponde con la predicción perfecta ya que significa que el área predicha es igual al área quemada real. Por otra parte, un fitness igual a 0 indica el máximo error indicando que no existe ningún tipo de coincidencia.

5. Resultados de la experimentación

En la presente sección se comparan los resultados obtenidos tras aplicar el método S^2F^2M y el simulador fireSim de forma aislada (es decir, en la forma tradicional o clásica). Para tal propósito se ha utilizado un conjunto de cuatro casos de quemas controladas efectuadas en el campo, en particular en Serra de Louçã (Gestosa, Portugal).

A lo largo del desarrollo de las quemas, se definieron pasos discretos para representar el avance del frente de fuego. Por lo tanto, se consideran distintos instantes de tiempo t_0, t_1,... etc. En la tabla 1 pueden apreciarse las características (dimensiones y pendientes) de los terrenos utilizados en los cuatro experimentos.

Tabla 1. Dimensiones y pendientes de los terrenos utilizados en la experimentación

Experimento	Ancho (m)	Largo (m)	Pendiente (°)
1	89	91	18
2	95	123	21
3	75	126	19
4	20	30	6

Es importante notar que para la predicción clásica hemos utilizado valores medidos y el promedio de cada valor para aquellos parámetros que exhiben incertidumbre, provenientes de límites establecidos en BehavePlus [3]. Además, decidimos utilizar el modelo óptimo de vegetación para cada experimento con el objetivo de obtener el mejor resultado del caso clásico. Por otra parte, para el caso de S^2F^2M, se efectuaron un promedio de 60000 simulaciones por cada intervalo de tiempo.

Después de la aplicación de los métodos, los valores de fitness hallados se presentan en la Fig. 5. Tanto el experimento 1 como el 4 pertenecen a casos de incendios iniciados en la base del terreno a través de dispositivos pirotécnicos. Por su parte, en el experimento 2 y 3, el fuego se ha originado en un único punto. Puede apreciarse que en tres de los cuatro casos S^2F^2M obtiene mejores resultados en comparación con la aplicación clásica del simulador fireSim.

El experimento 3 resulta de gran interés porque es posible ver el gran problema que presentan los parámetros en extremo variables. En este experimento, el rango de la velocidad del viento es bastante amplio, y puede verse cómo ha disminuido la calidad de la predicción, ya que, aunque el método S^2F^2M propone mejores resultados, se advierte que tanto la primer predicción (minuto 5) como la última (minuto 9) están asociadas a valores de fitness no muy elevados. Por su parte, la aplicación clásica del simulador alcanza su máximo en el minuto 8 con 0.2402.

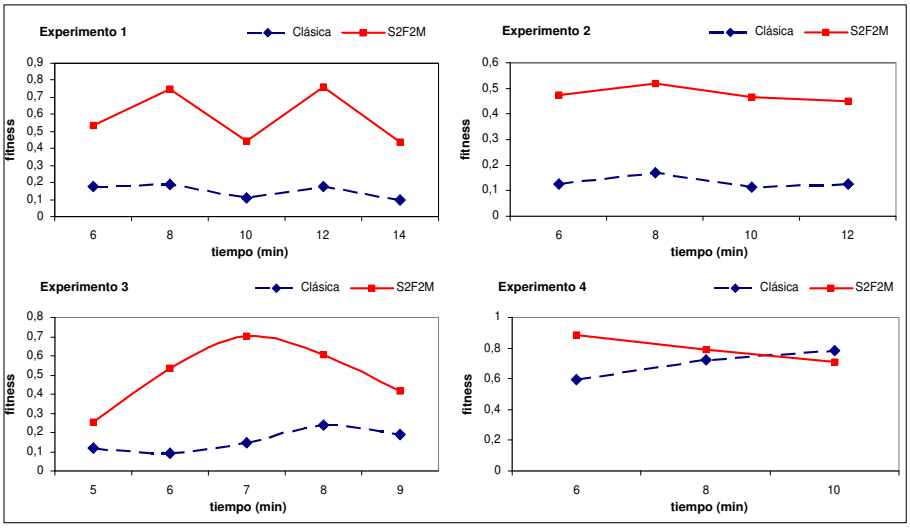

Figura 5. Comparación de fitness entre S^2F^2M y el método clásico

6. Conclusiones

Un problema de gran relevancia en innumerables sistemas es la capacidad de reflejar correctamente el modelo en cuestión sobre el cual operan. Estas discrepancias surgen por diferentes razones: imprecisión del modelo, limitaciones de las soluciones numéricas, incertidumbre en los valores de entrada, etc.

En el presente trabajo se ha descrito un método que se centra en el problema de la incertidumbre de los parámetros de entrada. El método, aplicado a la gestión de incendios forestales bajo el nombre de S^2F^2M, ha demostrado cómo, combinando conceptos estadísticos con el alto rendimiento alcanzable a través del cómputo paralelo, es posible mejorar la predicción de la propagación de incendios forestales. La comparación se efectuó contra la aplicación clásica del mismo simulador, fireSim, utilizado como núcleo de S^2F^2M, encontrando que la predicción estadística ofrece una predicción claramente superior a la del simulador en forma aislada, inclusive aplicando a éste los modelos de vegetación para los cuales la predicción brindaba un mayor valor de fitness.

Referencias

1. Abdalhaq B., "A methodology to enhance the Prediction of Forest Fire Propagation". Ph. D Thesis. Universitat Autònoma de Barcelona (Spain). 2004.
2. Andrews P. L. "BEHAVE: Fire Behavior prediction and modeling systems - Burn subsystem, part 1". General Technical Report INT-194. Odgen, UT, US Department of Agriculture, Forest Service, Intermountain Research Station. 1986.

3. Andrews P.L., Bevins C.D., Seli R.C., "BehavePlus fire modeling system, version 2.0: User's Guide" Gen. Tech. Rep. RMRS-GTR-106WWW. Ogden, UT: Department of Agriculture, Forest Service, Rocky Mountain Research Station. pp. 132. 2003.
4. Beven K. J., Freer J., "Equifinality, data assimilation, and uncertainty estimation in mechanistic modelling of complex environmental systems", Journal of Hydrology 249, pp. 11-49. 2001.
5. Bevins C.D., "FireLib User Manual & Technical Reference". 1996. http://www.fire.org.
6. Bianchini G., "Wildland Fire Prediction based on Statistical Analysis of Multiple Solutions". Ph. D Thesis. Universitat Autònoma de Barcelona (Spain). 2006.
7. Bianchini G., Cortés A., Margalef T., Luque E., "S^2F^2M - Statistical System for Forest Fire Management". LNCS 3514, pp. 427-434. 2005.
8. Chow T., He W., Chan A., Fong K., Lin Z., Ji J., "Computer modeling and experimental validation of a building-integrated photovoltaic and water heating system". Applied Thermal Engineering, Volume 28, Issues 11-12, pp. 1356-1364. 2008.
9. Grama A., Gupta A., Karypis G., Kumar V. "Introduction to Parallel Computing. Second Edition" Pearson Addison Wesley. 2003.
10. Ivanovskayaa V., Enjashina A., Sofronova A., Makurina Y., Medvedevab N., Ivanovskii A., "Quantum chemical simulation of the electronic structure and chemical bonding in (6,6), (11,11) and (20,0)-like metal-boron nanotubes". Journal of Molecular Structure: THEOCHEM. Volume 625, Issues 1-3, pp. 9-16. 2003.
11. Learmount J., Taylor M., Smith G., Morgan C., "A computer model to simulate control of parasitic gastroenteritis in sheep on UK farms". Veterinary Parasitology, Volume 142, Issues 3-4, pp. 312-329. 2006.
12. Mandel J., Bennethum L. S., Chen M., Coen J. L., Douglas C. C., Franca L. P., Johns C. J., Kim M., Knyazev A. V., Kremens R., Kulkarni V., Qin G., Vodacek A., Wu J., Zhao W., Zornes A., "Towards a Dynamic Data Driven Application System for Wildfire Simulation", LNCS 3515, pp. 632-639. 2005.
13. Mattson T., Sanders B., Massingill B., "Patterns for Parallel Programming". Addison-Wesley, 2004.
14. Montgomery D., Runger G., "Probabilidad y Estadística aplicada a la ingeniería", Limusa Wiley. 2002.
15. Douglas C. Montgomery, George C. Runger, "Applied statistics and probability for engineer", John Wiley & Sons, New York. D.L. 1994.
16. Mostaccio D., Suppi R., Luque E., "Simulation of Ecologic Systems Using MPI". LNCS 3666, pp. 449-456. 2005.
17. Odstrcil D., "Modeling 3-D solar wind structure".Advances in Space Research, Volume 32, Issue 4, pp. 497-506. 2003.
18. Piñol J., Salvador R., Beven K., "Model Calibration and uncertainty prediction of fire spread". Forest Fire Research & Wildland Fire Safety. On CD-ROM, Millpress. 2002.
19. Riffat S., Ma X., Wilson R., "Performance simulation and experimental testing of a novel thermoelectric heat pump system". Applied Thermal Engineering, Volume 26, Issues 5-6, pp. 494-501. 2006.
20. Rothermel, R. C., "A mathematical model for predicting fire spread in wildland fuels", USDA FS, Ogden TU, Res. Pap. INT-115, 1972.
21. Snir M., Otto S., Huss-Lederman S., Walker D., Dongarra J. "MPI: The complete reference". The MIT Press. Cambridge Massachusetts. London England. 1996.