



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

INFORME PARA PRESENTACIÓN

LABORATORIO DE MODELACIÓN I MAT-282

Nicolás Boyardi

nicolas.boyardi@usm.cl

Roberto González

roberto.gonzalezf@sansano.usm.cl

17 DE DICIEMBRE DE 2021

CASA CENTRAL

1 Descripción del problema

1.1 Introducción al tema de trabajo, motivación y relevancia.

En el presente trabajo, buscamos extender una propuesta Open-source Chilena sobre la modelación de propagación de incendios forestales, en la cual nos enfocaremos en intentar ampliar éste, a la vez que analizaremos si hay cambios significativos en los tiempos de cómputo del algoritmo nuevo frente al original. Las motivaciones para tratar este tema son varias, y se describen a continuación:

Primero y más importante: tenemos 2 diagnósticos sobre el impacto de los siniestros en el mundo. Uno es que la cantidad de incendios forestales en Chile, en los últimos 60 años, ha sido de alrededor de 230 mil incendios, los que afectaron a 3 millones de hectáreas. El otro es que los bosques en el mundo, a falta de incendios naturales debido a las medidas preventivas promovidas desde hace más de 100 años, son mucho más volátiles que antes [1],

"...entonces, asumimos que las actuales zonas forestadas aportan, en promedio, (más)combustible (para eventuales incendios) ..."

debido a una mayor concentración de vegetación (o debido a una falta de "manchas"sin vegetación). Los modelos propuestos por los expertos u otros científicos, entre otras cosas, pueden sugerir las zonas de mayor riesgo, permitiendo establecer políticas de prevención, e incluso se puede llegar a abarcar análisis en tiempo real, que permiten obtener información precisa en el momento, con el cual poder reevaluar la situación en el momento.

Segundo, pero no menos importante, en el trabajo de los expertos [2] se da a conocer que de los modelos conocidos sobre propagación de incendios solo 1, de 12, (y el más antiguo) es de código abierto. Este paper supone un gran avance en la materia.

Tercero, y particular de este estudio, es de explorar el uso de al menos 2 combustibles distintos dentro del modelo, pues el modelo base con el partimos solo considera la existencia de un único combustible. Esta suposición es, en general, falsa; pero se partió por ahí por motivos de implementación. Agregar nuevos combustibles es un pequeño desafío dentro de la estructura del modelo.

1.2 Estado del arte del problema

1.2.1 Modelos relacionados

Los modelos más recurrentes, y a los que nos enfocaremos, de incendios forestales suelen ser sistemas dinámicos (sistemas de EDPs) que involucran los 3 ingredientes fundamentales de un incendio: oxígeno, calor y combustible. En ausencia de cualquiera de los 3 anteriores simplemente no se produce ni continúa el incendio.

El aspecto principal de estos modelos es caracterizar el comportamiento particular que tienen los incendios. El enfoque típico es caracterizar a través de "la magnitud, dirección e intensidad de la propagación del fuego, que dependen de condiciones ambientales, como la vegetación, tiempo y topografía", según [3] (pp. 1350).

Hasta el momento, la atención de las investigaciones se ha centrado principalmente en modelos simplificados de naturaleza semiempírica, que es una simbiosis entre uso teórico de leyes físicas y la constatación del modelo con datos estadísticos reales. Se prefiere ésta, mezcla lo mejor de dos mundos: modelos teóricos puros son de naturaleza a priori, i.e., viven de la precisión de nuestros supuestos. Por otro lado, los modelos estadísticos puros son de naturaleza a posteriori, i.e., se

cimentan en la experiencia, en los datos que podamos recoger, lo que le da un espíritu heurístico. Además, se trabaja considerando sectores quemados y no quemados, limitada a no poder anticipar situaciones como aparición de focos.

En relación al software computacional para realizar simulación se aprecian dos acercamientos, modelos discretos como autómatas celulares(CA), y modelos continuos como Ecuaciones Diferenciales Parciales(EDP).A continuación, se presenta una tabla con modelos, tomada de [2]. En ese mismo artículos se encuentran explicaciones de algunos de esos modelos.

Tipo de modelo	Author	Componentes			Discretización		¿El código es reproducible?
		Fuego	Viento	Topografía	Espacio	Tiempo	
CA	Hansen (1993) [4]	sí	no	no	-	-	sí
	Karafyllidis et al. (1997) [5]	sí	sí	sí	-	-	no
	Quartieri et al. (2010) [6]	sí	sí	sí	-	-	no
	Almeida et al. (2011) [7]	sí	no	no	-	-	no
	Ghisu et al. (2015) [8]	sí	sí	sí	-	-	no
	Fernandez-Anez et al. (2017) [9]	sí	no	no	-	-	no
PDE	Montenegro et al. (1997) [10]	sí	sí	sí	FEM adaptativo	Semi-implicit Euler	no
	Asensio et al. (2002) [11]	sí	sí	sí	FEM	Semi-implicit Euler	no
	Ferragut et al. (2006) [12]	sí	sí	sí	FEM	Semi-implicit Euler	no
	Ferragut et al (2007) [13]	sí	sí	sí	FEM	Semi-implicit Euler	no
	Mandel et al. (2008) [14]	sí	sí	sí	Diferencias finitas	Explicit Euler	no
	Ferragut et al. (2015) [15]	sí	sí	sí	FEM	Semi-implicit Euler	no
	Eberle et al. (2015) [3]	sí	sí	sí	colocación RBF	Crank-Nicolson	no
	San Martín y Torres (2018) [2]	sí	sí	sí	Diferencias finitas	Runge-Kutta cuarto orden	sí

Además de esos artículos también encontramos el modelo descrito en [16], en el cual se modela un incendio en la escala de bosques, pero modelando la combustión de cada uno de los árboles por separado, e incluye dinámica atmosférica y topografía.Un problema que puede tener un modelo de esta naturaleza, es la alta necesidad de recursos computacionales(Se necesita ocupar un computador con una tarjeta gráfica dedicada, se demora 95 milisegundos en calcular un paso de tiempo de 16 milisegundos de la simulación en el caso más complejo presentado).

Para este trabajo, se considerará un modelo que toma como base leyes físicas (de transferencia de masa y calor), como también químicas (combustión). Así, con un modelo propuesto, y con datos tales como la topografía del lugar y el campo vectorial del viento, podemos proceder a estimar soluciones a través de métodos numéricos, los cuales son explicados en la sección 1.5.

1.2.2 Software existente

A continuación se incluye una tabla de software usados comúnmente:

Nombre	Modelos	Es Open Source?
BehavePlus	Semi-empírica, modelar geometría de incendios forestales	Sí
Farsite	Semi-empírica, modelar geometría de incendios forestales	No
FLAMMAP	Semi-empírica, modelar geometría de incendios forestales	No
CAWFE	Fuego + EDPs de atmosféricas	No
MesoNH/ ForeFire	Modelar geometría de incendios forestales	No
FDS	modelo 3D EDPs + Atmosférica	Si
WRF-Fire	Semi-empírica	No
SFIRE	EDPs + Atmosférica	Si
FIRETEC/HIGRAD	EDPs + Atmosférica	No
QUIC-fire	EDPs + Atmosférica	No

En el caso de los modelos presentados con anterioridad, uno podría tomar los artículos de investigación y replicar los resultados. En algunos de los Software presentados, como no son open-source, no se pueden replicar. Por eso es relevante la Opensource-ness del Modelo que tomamos como base del trabajo de los expertos. Además, notamos también que algunos de estos modelos ocupan recursos computacionales de alta performance (como supercomputadoras), en específico se señala en la página web de FIRETEC que “ocupa una gran cantidad de recursos en el laboratorio computacional de Los Alamos, por lo que es solamente una herramienta de investigación” [17]. Recientemente se desarrollo un paquete de software llamado QUIC-fire [18], que puede obtener resultados que concuerdan con los FIRETEC, pero ocupando los recursos computacionales de un computador personal.

1.2.3 Modelo con el que se va a trabajar

Sea $u(\mathbf{x}, t)$ el valor numérico de la temperatura y $\beta_a(\mathbf{x}, t), \beta_b(\mathbf{x}, t)$ la cantidad de combustible a y b respectivamente, todo en coordenadas espaciales $\mathbf{x} = (x, y)$ y de tiempo t , donde $\mathbf{x} \in \Omega = [x_{min}, x_{max}] \times [y_{min}, y_{max}] \subseteq \mathbb{R}^2$ y $t \in [0, t_{max}]$. Sea $\mathbf{v}(\mathbf{x}, t) = \mathbf{w}(\mathbf{x}, t) + \nabla T(\mathbf{x})$ el campo vectorial que modela los efectos del viento y la topografía, y $f(u, \beta)$ una fuente no lineal de calor. Entonces el modelo queda dado por:

$$\begin{aligned}
u_t &= \bar{\kappa} \Delta u - \mathbf{v} \cdot \nabla u + f(u, \beta^a, \beta^b), & \text{en } \Omega \times]0, t_{\max}], \\
\beta_t^a &= g^a(u, \beta^a), & \text{en } \Omega \times]0, t_{\max}], \\
\beta_t^b &= g^b(u, \beta^b), & \text{en } \Omega \times]0, t_{\max}], \\
u(\mathbf{x}, t) &= h_1(\mathbf{x}, t), & \text{sobre } \Gamma \times]0, t_{\max}], \\
\beta^a(\mathbf{x}, t) &= h_2(\mathbf{x}, t), & \text{sobre } \Gamma \times]0, t_{\max}], \\
\beta^b(\mathbf{x}, t) &= h_3(\mathbf{x}, t), & \text{sobre } \Gamma \times]0, t_{\max}], \\
u(\mathbf{x}, 0) &= u_0(\mathbf{x}), & \text{en } \Omega, \\
\beta^a(\mathbf{x}, 0) &= \beta_0^a(\mathbf{x}), & \text{en } \Omega \\
\beta^b(\mathbf{x}, 0) &= \beta_0^b(\mathbf{x}), & \text{en } \Omega
\end{aligned}$$

donde Γ representa el borde de Ω y,

$$f(u, \beta^a, \beta^b) = r_{Ca} H_{pc}^a(u) \beta^a \exp\left(\frac{u}{1 + \varepsilon^a u}\right) + r_{Cb} \frac{r_u}{r_T} H_{pc}^b(u) \beta^b \exp\left(\frac{ur_u}{1 + \varepsilon^b ur_u}\right) - \alpha u$$

$$g^a(u, \beta^a) = -H_{pc}^a(u) \frac{\varepsilon^a}{q^a} \beta^a \exp\left(\frac{u}{1 + \varepsilon^a u}\right)$$

$$g^b(u, \beta^b) = -\frac{1}{r_t} H_{pc}^b(u) \frac{\varepsilon^b}{q^b} \beta^b \exp\left(\frac{ur_u}{1 + \varepsilon^b ur_u}\right)$$

con

$$H_{pc}^a(u) = \begin{cases} 1, & \text{si } u \geq u_{pc}^a, \\ 0, & \text{en otro caso.} \end{cases}$$

$$H_{pc}^b(u) = \begin{cases} 1, & \text{si } u \geq u_{pc}^b, \\ 0, & \text{en otro caso.} \end{cases}$$

con r_{Ca}, r_{Cb} como los ratios de calores volumétricos, r_u el cambio de escala de la temperatura y r_T el cambio de escala del tiempo. En este caso, el gradiente es definido como $\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right)$, y el operador Laplaciano como $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$.

1.3 deducción del modelo

Se seguirán los pasos utilizados en Aberle et al. para obtener el modelo utilizado. Se modelarán dos combustibles, que denotaremos por combustibles a y b. Asumiremos que el combustible a es una parte p_a del total y el combustible b es una parte $p_b = 1 - p_a$ del total.

En el caso de la quema de cada combustible, sabemos que sigue la ecuación de Arrhenius, de esta manera se tiene:

$$\begin{aligned}
\frac{\partial Y_a}{\partial t} &= H(T - T_{pca}) A Y_a \exp\left(-\frac{E_{Aa}}{RT}\right) \\
\frac{\partial Y_b}{\partial t} &= H(T - T_{pcb}) A Y_b \exp\left(-\frac{E_{Ab}}{RT}\right)
\end{aligned}$$

Ahora veamos como varía la Temperatura.

- La energía térmica asociada al dominio Ω es:

$$\int_{\Omega} (C_a \bar{\rho}_a + C_b \bar{\rho}_b) T dV$$

con C_1, C_2 los calores específicos y $\bar{\rho}_1, \bar{\rho}_2$ es:

$$\begin{aligned}\bar{\rho}_a &= \rho_a p_a \\ \bar{\rho}_b &= \rho_b p_b\end{aligned}$$

con ρ_a, ρ_b las densidades de los materiales.

- El flujo de calor asociado a difusión en la frontera es:

$$\int_{\partial\Omega} j \cdot n ds$$

- energía de liberada por la reacción es:

$$\int_{\Omega} (s_a + s_b) dV$$

donde s_a, s_b son la densidad de energía liberada por cada combustible

- Por último aproximamos enfriamiento vertical con:

$$\int h(T - T_{\infty}) dV$$

Teniendo en cuenta lo anterior la variación de la energía térmica en el tiempo es igual a la energía que fluye por la frontera más la energía liberada por la reacción.

$$\begin{aligned}\frac{d}{dt} \left(\int_{\Omega} (C_a \bar{\rho}_a + C_b \bar{\rho}_b) T dV \right) &= - \int_{\partial\Omega} j \cdot n ds + \int_{\Omega} (s_a + s_b) dV \\ &= - \int \nabla \cdot j dV + \int (s_a + s_b) dV - \int h(T - T_{\infty}) dV\end{aligned}$$

Notemos podemos reescribir la derivada como:

$$\begin{aligned}\frac{d}{dt} \left(\int_{\Omega} (C_a \bar{\rho}_a + C_b \bar{\rho}_b) T dV \right) &= \int \left[\frac{\partial}{\partial t} ((C_a \bar{\rho}_a + C_b \bar{\rho}_b) T) + v \cdot \nabla ((C_a \bar{\rho}_a + C_b \bar{\rho}_b) T) \right] dV \\ &= \int \left[(C_a \bar{\rho}_a + C_b \bar{\rho}_b) \frac{\partial T}{\partial t} + (C_a \bar{\rho}_a + C_b \bar{\rho}_b) v \cdot \nabla T \right] dV\end{aligned}$$

Luego tenemos que :

$$\int \left[(C_a \bar{\rho}_a + C_b \bar{\rho}_b) \frac{\partial T}{\partial t} + (C_a \bar{\rho}_a + C_b \bar{\rho}_b) v \cdot \nabla T \right] dV = - \int \nabla \cdot j dV + \int (s_a + s_b) dV - \int h(T - T_{\infty}) dV$$

Y por lo tanto:

$$(C_a \bar{\rho}_a + C_b \bar{\rho}_b) \frac{\partial T}{\partial t} + (C_a \bar{\rho}_a + C_b \bar{\rho}_b) v \cdot \nabla T = - \nabla \cdot j + s_a + s_b - h(T - T_{\infty})$$

Tenemos que las enerrgías liberadas por cada combustible se comportan según:

$$s_a = H(T - T_{pca})AY_a\bar{\rho}_a \exp\left(-\frac{E_{Aa}}{RT}\right)$$

$$s_b = H(T - T_{pcb})AY_b\bar{\rho}_b \exp\left(-\frac{E_{Ab}}{RT}\right)$$

por simplicidad, aproximamos:

$$j \approx (\kappa_a p + \kappa_b(1 - p))\nabla T \Rightarrow j \cdot \nabla T \approx (\kappa_a p + \kappa_b(1 - p))\nabla^2 T = D\nabla^2 T$$

Con lo que obtenemos finalmente que:

$$(C_a\bar{\rho}_a + C_b\bar{\rho}_b)\frac{\partial T}{\partial t} = -(C_a\bar{\rho}_a + C_b\bar{\rho}_b)v \cdot \nabla T + D\nabla^2 T + h(t - T_{pca})AY_a\bar{\rho}_a \exp\left(-\frac{E_{Aa}}{RT}\right) + h(t - T_{pcb})AY_b\bar{\rho}_b \exp\left(-\frac{E_{Ab}}{RT}\right)$$

Luego de aplicar la siguiente adimensionalización:

- En el espacio y tiempo:

$$x = \frac{\bar{x}}{l_{a0}}, \quad y = \frac{\bar{y}}{l_{a0}}, \quad t = \frac{\bar{t}}{t_{a0}}$$

- De los componentes del modelo:

$$u = \frac{T - T_\infty}{\varepsilon_a T_\infty}, \quad \beta_a = \frac{Y_a}{Y_{a0}}, \quad \beta_b = \frac{Y_b}{Y_{b0}}, \quad \mathbf{w} = \frac{t_{a0}}{l_{a0}} \bar{\mathbf{v}}$$

Se obtiene:

$$\frac{\partial u}{\partial t} = w \cdot \nabla u + \bar{\kappa} \nabla^2 u + r_{Ca} H_{pca}^1(u) \beta_a \exp\left(\frac{u}{1 + \varepsilon_a u}\right) + r_{Cb} \frac{r_b}{r_b} H_{pcb}^2(u) \beta_b \exp\left(\frac{u r_u}{1 + \varepsilon_b u r_u}\right) - \alpha u$$

donde:

$$r_{Ca} = \frac{C_a \bar{\rho}_a}{C_a \bar{\rho}_a + C_b \bar{\rho}_b}$$

$$r_{Cb} = \frac{C_b \bar{\rho}_b}{C_a \bar{\rho}_a + C_b \bar{\rho}_b}$$

$$r_u = \frac{\varepsilon_b}{\varepsilon_a}$$

$$r_t = \frac{t_{b0}}{t_{a0}}$$

$$\bar{\kappa} = \frac{4\sigma \delta u_\infty^3}{D}$$

$$\alpha = \frac{t_{0a} h}{C_a \bar{\rho}_a + C_b \bar{\rho}_b}$$

1.4 Objetivos generales y específicos

1.4.1 Objetivos generales

- Extender el modelo presentado en [2] , considerando ya no sólo uno, sino, al menos, 2 tipos de combustible.

1.4.2 Objetivos específicos

- Entender el contexto en que se encuentra el modelo, comprendiendo porqué es importante su resolución.
- Comprender qué leyes científicas existen que nos permitan darle una base/justificación a nuestro modelo.
- Estudiar qué herramientas computacionales son más útiles para la resolución del problema (calcular numéricamente una solución, o crear una solución explícita desde cero).
- Implementar el modelo en al menos 1 lenguaje de programación, teniendo en preferencia a `python`.
- Comparar los costos computacionales del nuevo modelo construido con el antiguo.
- Evaluar cómo se compara la solución numérica con las propiedades de una solución esperada. Por ejemplo, si la distribución de combustible es simétrica, ¿la solución numérica obtenida también es simétrica?

1.5 Herramientas matemáticas a emplear

1.5.1 Naturaleza del modelo

En nuestro proyecto buscamos encontrar una solución numérica para un sistema de ecuaciones diferenciales parciales con valor inicial y condiciones de frontera. Para esto, se utilizará el método de las líneas. En este caso, se aproxima la ecuación en el espacio usando **diferencias finitas**, con lo que obtenemos una EDO en el tiempo. Luego, podemos utilizar **métodos de resolución numérica de EDOs** como los Runge-Kutta, para obtener la solución de la EDO previamente obtenida. En la siguiente subsección se explicarán estos métodos con más detalle.

1.5.2 Aproximación en el espacio: Diferencias finitas

El método de diferencias finitas aquí se aplica a la función $u : \mathbb{R}^2 \rightarrow \mathbb{R}$, que nos entrega la temperatura en una posición dada del plano (vista aérea), en una temperatura fija t_k . Particularmente, se establece una grilla de rectángulos congruentes en dicho plano, es decir, se particionan equidistantemente los ejes X e Y , generando $N_x + 1$ y $N_y + 1$ subintervalos respectivamente. Así, se define la matriz $U := [u((x_i, y_j), t_k)]$ que determina la temperatura de cada rectángulo en la grilla, en un tiempo t_k fijo, con $i \in \{0, \dots, N\}$.

Para aproximar las derivadas $\frac{\partial u}{\partial x}$, $\frac{\partial u}{\partial y}$, $\frac{\partial^2 u}{\partial x^2}$ y $\frac{\partial^2 u}{\partial y^2}$ en los $N_x - 1$ y $N_y - 1$ puntos interiores, se hace uso de distintas aproximaciones de alto orden, que en estos problemas multivariados nos permiten construir mejores aproximaciones como la dada por:

$$\begin{aligned}\frac{\partial u}{\partial x}(x_i, y_j) &\approx \frac{u(x_{i+1}, y_j) - u(x_i, y_j)}{\Delta x} \\ \frac{\partial u}{\partial y}(x_i, y_j) &\approx \frac{u(x_i, y_{j+1}) - u(x_i, y_j)}{\Delta y} \\ \frac{\partial^2 u}{\partial x^2}(x_i, y_j) &\approx \frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j)}{\Delta x^2} \\ \frac{\partial^2 u}{\partial y^2}(x_i, y_j) &\approx \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1})}{\Delta y^2}\end{aligned}$$

donde $\Delta x, \Delta y$ corresponde al tamaño de las particiones de la variable x e y respectivamente.

1.5.3 Aproximación en el tiempo: Métodos de resolución numérica de EDOs

Mostraremos algunos métodos de resolución numérica de EDOs. Sea un problema de valor inicial de forma

$$\begin{aligned} \mathbf{y}_t(t) &= f(\mathbf{y}, t) \text{ para } t \in]0, t_{\max}] \\ \mathbf{y}(0) &= \mathbf{y}_0 \end{aligned}$$

con $y_0 \in \mathbb{R}^S$ e $y : [0, t_{\max}] \rightarrow \mathbb{R}^S$ Este problema puede ser resuelto de varias formas, que pueden ser:

- **implícito/explicito:** Es implícito si es necesario resolver una ecuación/sistema que puede ser no lineal, para determinar la siguiente iteración del método. En caso contrario, es explícito.
- **paso simple/paso múltiple:** Si la ecuación de diferencia asociada al método depende de \mathbf{y}_{n+j} , $j = 0, 1, \dots, k$ con $k > 1$, es de paso múltiple. Si $k = 1$ es de paso simple. [19]

A continuación se describen algunos métodos iterativos:

- **Euler explícito:** Corresponde a un método explícito de paso simple. Se calcula la siguiente iteración con la siguiente fórmula:

$$\begin{aligned} t_{n+1} &:= t_n + h \\ \mathbf{y}_{n+1} &:= \mathbf{y}_n + hf(t_n, \mathbf{y}_n) \end{aligned}$$

- **Euler implícito:** Corresponde a un método implícito de paso simple. Se calcula la siguiente iteración con la siguiente fórmula:

$$\begin{aligned} t_{n+1} &:= t_n + h \\ \mathbf{y}_{n+1} &:= \mathbf{y}_n + hf(t_{n+1}, \mathbf{y}_{n+1}) \end{aligned}$$

- **Runge Kutta:** Pueden ser explícitos o implícitos, son de paso simple. Uno de los más usados en RK-4, para este, se calcula la siguiente iteración con la siguiente fórmula:

$$\begin{aligned} k_1 &:= f(t_n, \mathbf{y}_n) \\ k_2 &:= f(t_n + \frac{h}{2}, z_n + \frac{h}{2}k_1) \\ k_3 &:= f(t_n + \frac{h}{2}, z_n + \frac{h}{2}k_2) \\ k_4 &:= f(t_n + h, z_n + hk_3) \\ \mathbf{y}_{n+1} &:= \mathbf{y}_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \end{aligned}$$

- **Adams-Bashforth:** Son métodos explícitos de paso múltiple. El método de Adams-Bashforth de 2 pasos es:

$$\begin{aligned} t_{n+1} &:= t_n + h \\ \mathbf{y}_{n+1} &:= \mathbf{y}_n + \frac{h}{2}(3f(t_n, \mathbf{y}_n) - f(t_{n-1}, \mathbf{y}_{n-1})) \end{aligned}$$

En los modelos que vimos h es constante. Podemos variar el tamaño de cada paso según la iteración, es decir que $t_{n+1} = t_n + h_n$, con lo que tenemos una selección de paso adaptativa. Para esto es necesario tener una estimación local del error.

1.5.4 Error del modelo

Tenemos que fuera del **error de modelamiento**, vamos a tener **errores computacionales**:

- errores de discretización: aproximar el modelo en el espacio y tiempo
- errores de redondeo

Para controlar el error obtenido tenemos que cambiar los parámetros Δs y h o el modelo numérico para la resolución de EDO. Va a existir un **trade-off** entre tener parámetros Δs y h más pequeño, con menor error en cada iteración, pero con un mayor costo computacional en una iteración. De la misma forma, elegir un método numérico para EDO, que otorgue un menor error por iteración, por lo general, va a tener un mayor costo computacional por iteración.

Para la implementación se usó euler explícito en Python y euler semi-implícito en Julia.

2 Implementación numérica

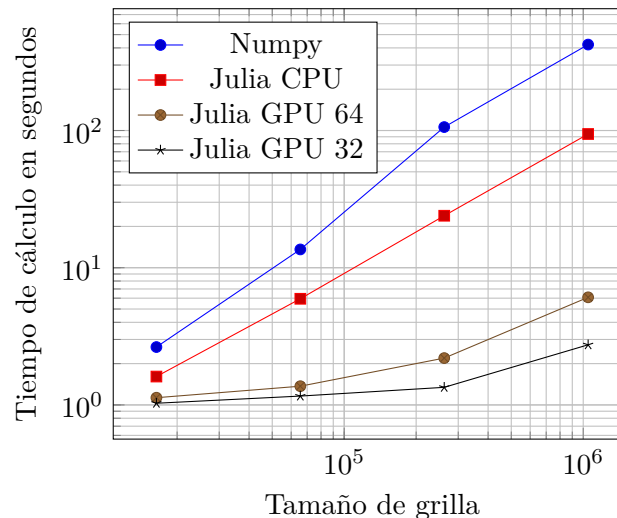
2.1 Tiempos de ejecución

Se implementó en `python` usando el paquete `numpy` y en `Julia` usando el paquete `ParallelStencil.jl`. Para la implementación de `Julia` se calculó usando solo la CPU y usando la GPU en dos casos: uno usando números con `Float32` y otro con `Float64`. Se utilizaron los siguientes parámetros:

Se tomó el promedio de los tiempos de ejecución variando el tamaño de la grilla de 128×128 , 256×256 , 512×512 y 1024×1024 . Los resultados se pueden apreciar en esta tabla, con los tiempos en segundos:

Tamaño de grilla	juliaGPU32	juliaGPU64	juliaCPU	numpy
16,384	1,03	1,13	1,61	2,64
65,536	1,16	1,37	5,94	13,59
$2,62 \cdot 10^5$	1,34	2,19	23,93	105,95
$1,05 \cdot 10^6$	2,74	6,09	94,3	423,01

y en este gráfico:



Bibliografía

- [1] Raymond D., Yang Z., Yost A., Belongie C., Cohen W.: The normal fire environment—Modeling environmental suitability for large forest wildfires using past, present, and future climate normals, *Forest Ecology and Management*, Volume 390, 2017, Pages 173-186, <https://doi.org/10.1016/j.foreco.2017.01.027>
- [2] San Martín, D., & Torres, C. E. (2018). Ngen-Kütral: Toward an Open Source Framework for Chilean Wildfire Spreading. In 2018 37th International Conference of the Chilean Computer Science Society (SCCC) (pp. 1–8). <https://doi.org/10.1109/SCCC.2018.8705159>
- [3] Eberle, S., Freeden, W., & Matthes, U. (2015). Forest fire spreading. In Freeden Willi, M. Z. and Nashed, & and Sonar Thomas (Eds.), *Handbook of Geomathematics: Second Edition* (pp. 1349–1385). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-54551-1_70
- [4] P. B. Hansen, “Parallel cellular automata: A model program for computational science,” *Concurrency: Practice and Experience*, vol. 5, no. 5, pp. 425–448, 1993.
- [5] I. Karafyllidis and A. Thanailakis, “A model for predicting forest fire spreading using cellular automata,” *Ecological Modelling*, vol. 99, no. 1, pp. 87–97, jun 1997.
- [6] J. Quartieri, N. E. Mastorakis, G. Iannone, and C. Guarnaccia, “A cellular automata model for fire spreading prediction,” *Latest Trends on Urban Planning and Transportation*, pp. 173–178, 2010
- [7] R. M. Almeida and E. E. N. Macau, “Stochastic cellular automata model for wildland fire spread dynamics,” *Journal of Physics: Conference Series*, vol. 285, no. 1, p. 12038, 2011
- [8] T. Ghisu, B. Arca, G. Pellizzaro, and P. Duce, “An Improved Cellular Automata for Wildfire Spread,” *Procedia Computer Science*, vol. 51, pp. 2287–2296, jan 2015.
- [9] N. Fernandez-Anez, K. Christensen, and G. Rein, “Two-dimensional model of smouldering combustion using multi-layer cellular automaton: The role of ignition location and direction of airflow,” *Fire Safety Journal*, vol. 91, pp. 243–251, jul 2017.
- [10] R. Montenegro, A. Plaza, L. Ferragut, and M. I. Asensio, “Application of a nonlinear evolution model to fire propagation,” *Nonlinear Analysis: Theory, Methods & Applications*, vol. 30, no. 5, pp. 2873–2882, dec 1997.
- [11] M. I. Asensio and L. Ferragut, “On a wildland fire model with radiation,” *International Journal for Numerical Methods in Engineering*, vol. 54, no. 1, pp. 137–157, 2002.
- [12] L. Ferragut, M. I. Asensio, and S. Monedero, “Modeling Radiation and Moisture Content in Fire Spread,” in *Numerical Mathematics and Advanced Applications*, de Castro Alfredo Bermudez, , D. G ´ omez, and ´ Quintela Peregrina, and and Salgado Pilar, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 601–608.
- [13] —, “A numerical method for solving convection-reaction-diffusion multivalued equations in fire spread modelling,” *Advances in Engineering Software*, vol. 38, no. 6, pp. 366–371, jun 2007.

- [14] . Mandel, L. S. Bennethum, J. D. Beezley, J. L. Coen, C. C. Douglas, M. Kim, and A. Vodacek, “A wildland fire model with data assimilation,” *Mathematics and Computers in Simulation*, vol. 79, no. 3, pp. 584–606, dec 2008.
- [15] L. Ferragut, M. I. Asensio, J. M. Cascon, and D. Prieto, “A Wildland Fire ´ Physical Model Well Suited to Data Assimilation,” *Pure and Applied Geophysics*, vol. 172, no. 1, pp. 121–139, 2015.
- [16] Torsten Hädrich, Daniel T. Banuti, Wojtek Pałubicki, Sören Pirk, and Dominik L. Michels. 2021. Fire in Paradise1 : Mesoscale Simulation of Wildfires. *ACM Trans. Graph.* 40, 4, Article 163 (August 2021), 15 pages. <https://doi.org/10.1145/3450626.3459954>
- [17] página de FIRETEC. Accedido el 11 de Noviembre, 2021. Disponible en: <https://www.frames.gov/firetec/home>
- [18] Linn, R.R.; Goodrick, S.L.; Brambilla, S.; Brown, M.J.; Middleton, R.S.; O’Brien, J.J.; Hiers, J.K. 2020. QUIC-fire: A fast-running simulation tool for prescribed fire planning. *Environmental Modelling & Software*. 125: 104616-. <https://doi.org/10.1016/j.envsoft.2019.104616>
- [19] Lambert J.D. , *Numerical Methods for Ordinary Differential Systems*, John Wiley & Sons, Chichester, 1991. ISBN 0-471-92990-5.