

# Modelo de Klausmeier

Claudio Armas - ANIII - 2024

FAMAF

Facultad de Matemática,  
Astronomía, Física y  
Computación



UNC

Universidad  
Nacional  
de Córdoba

28/11/2024

# Resumen

## 1 Introducción

## 2 VisualPDE

- Breve introducción a VisualPDE
- Probando VisualPDE para el modelo Klausmeier

## 3 Esquema e implementación

- Esquema para el modelo de Klausmeier
- Implementación y resultados numéricos

## 4 Conclusiones

# Problema

- La desertificación es la degradación de las tierras de zonas áridas y semiáridas resultante de diversos factores, tales como las variaciones climáticas y las actividades humanas.
- A finales de la década de 1960 comenzó aemerger como un problema relevante.
- En la actualidad, es un problema que relaciona la economía, la sociedad y la ecología.

# Problema

- El modelo de Klausmeier se centra en la dinámica de vegetación en áreas semidesérticas.
- Muchos investigadores han intentado resolver el modelo de Klausmeier pues obtener una solución precisa o exacta no es tarea fácil ya que se trata de una ecuación diferencial parcial no lineal (EDPNL).

# Propuesta

- Resolver el modelo por medio de métodos numéricas simples.
- Utilizar una herramienta llamada VisualPDE, que permite realizar simulaciones interactivas de ecuaciones diferenciales parciales (EDP) con respecto del tiempo en una y dos dimensiones espaciales.
- Plantear y desarrollar una implementación del modelo de Klausmeier usando esquemas de diferencias finitas con el objetivo de replicar las simulaciones de VisualPDE.

# Breve introducción a VisualPDE

- El acceso generalizado a la computación ha tenido un profundo impacto en la ciencia y en la sociedad en general.
- Los modelos matemáticos relativamente avanzados se han vuelto ampliamente utilizados en numerosos campos de la ciencia.
- Los creadores de VisualPDE buscaban crear una plataforma que abstraiga las complejidades de los métodos numéricos y su implementación, de modo que un estudiante tenga la libertad de explorar modelos a los que normalmente solo se puede acceder mediante simulación.

## Breve introducción a VisualPDE

## Ejemplos

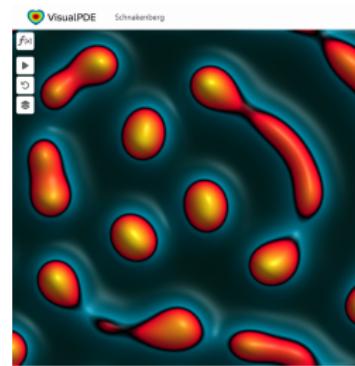
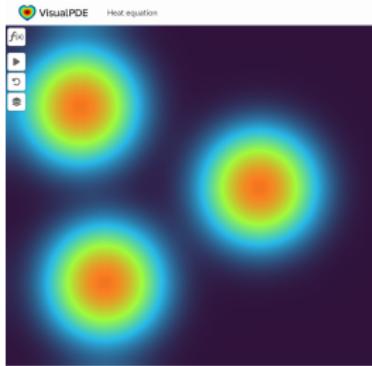


Figura: Ilustración de tres modelos de EDP del sitio wed VisualPDE.

# El modelo de Klausmeier

Es una EDP de difusión-advección-reacción de la forma,

$$\begin{cases} \frac{\partial w}{\partial t} = a - w - wn^2 + v \frac{\partial w}{\partial x} + \nabla^2 w \\ \frac{\partial n}{\partial t} = wn^2 - mn + \nabla^2 n \end{cases} \quad (1)$$

# Muestra del Simulador

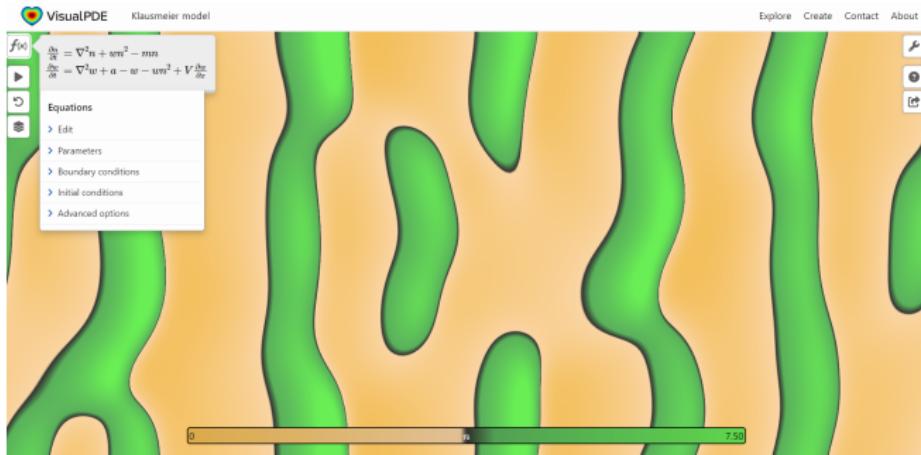
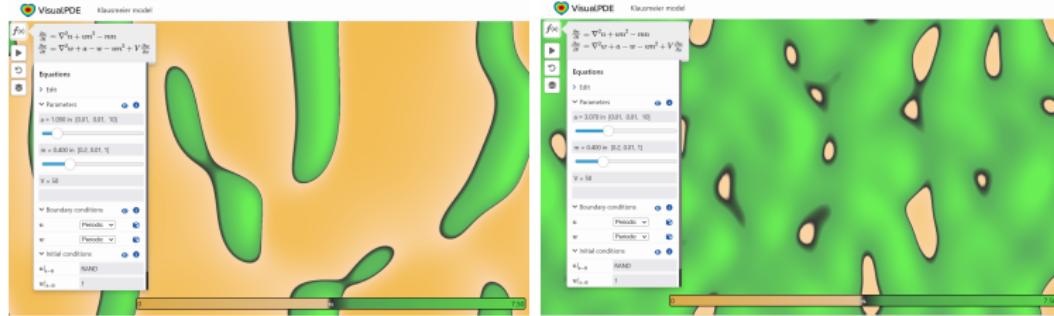


Figura: Captura pausada del simulador del sitio web visualPDE.

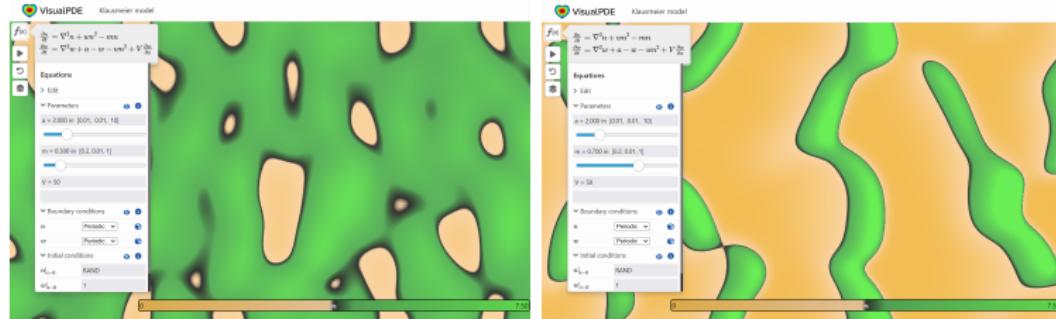
## Probando VisualPDE para el modelo Klausmeier

## Ejemplo 1

Figura: Distintos valores del parámetro  $a$  (tasa de lluvia).

## Probando VisualPDE para el modelo Klausmeier

## Ejemplo 2

Figura: Distintos valores del parámetro  $m$  (pérdida de vegetación).

# Planteo

Reordenando el modelo de la siguiente forma,

$$\begin{cases} \frac{\partial w}{\partial t} = \nabla^2 W + v \frac{\partial w}{\partial x} + a - w - wn^2 \\ \frac{\partial n}{\partial t} = \nabla^2 n + wn^2 - mn \end{cases} \quad (2)$$

# Planteo

- Sea  $\Omega = [0, L] \times [0, L]$  (la cuadrícula),
- $\Delta x$  y  $\Delta y$  pasos espaciales entre nodos,
- un entero  $R_{nd}$  la cantidad de nodos,
- el paso del tiempo definimos un  $\Delta t$ ,
- cada nodo espacial es de la forma  $x_i = i\Delta x$  e  $y_j = j\Delta y$ , para  $i, j = 0, 1, 2, \dots, R_{nd}$ ,
- y lo mismo para el tiempo,  $t_k = k\Delta t$  para  $k = 0, 1, 2, \dots, T$ , donde  $T$  es el tiempo máximo.

# Aplicando diferencias finitas adelantada

$$\begin{cases} \frac{\partial w}{\partial t}(t, x, y) \approx \frac{W_{i,j}^{k+1} - W_{i,j}^k}{\Delta t}, & \frac{\partial w}{\partial x}(t, x, y) \approx \frac{W_{i,j}^{k+1} - W_{i,j}^k}{\Delta t}, \\ \frac{\partial n}{\partial t}(t, x, y) \approx \frac{N_{i,j}^{k+1} - N_{i,j}^k}{\Delta t}. \end{cases} \quad (3)$$

# Aplicando diferencias finitas centradas

$$\begin{cases} \nabla^2 w(t, x, y) \approx \frac{W_{i+1,j}^k - 2W_{i,j}^k + W_{i-1,j}^k}{\Delta x^2} + \frac{W_{i,j+1}^k - 2W_{i,j}^k + W_{i,j-1}^k}{\Delta y^2}, \\ \nabla^2 n(t, x, y) \approx \frac{N_{i+1,j}^k - 2N_{i,j}^k + N_{i-1,j}^k}{\Delta x^2} + \frac{N_{i,j+1}^k - 2N_{i,j}^k + N_{i,j-1}^k}{\Delta y^2}. \end{cases} \quad (4)$$

# Primer esquema

Reemplazando en (2) se obtiene,

$$\left\{ \begin{array}{l} \frac{W_{i,j}^{k+1} - W_{i,j}^k}{\Delta t} = \frac{W_{i+1,j}^k - 2W_{i,j}^k + W_{i-1,j}^k}{\Delta x^2} + \frac{W_{i,j+1}^k - 2W_{i,j}^k + W_{i,j-1}^k}{\Delta y^2} \\ \quad + v \frac{W_{i+1,j}^k - W_{i,j}^k}{\Delta x} + a - W_{i,j}^k - W_{i,j}^k N_{i,j}^k \\ \\ \frac{N_{i,j}^{k+1} - N_{i,j}^k}{\Delta t} = \frac{N_{i+1,j}^k - 2N_{i,j}^k + N_{i-1,j}^k}{\Delta x^2} + \frac{N_{i,j+1}^k - 2N_{i,j}^k + N_{i,j-1}^k}{\Delta y^2} \\ \quad + W_{i,j}^k (N_{i-1,j}^k)^2 - m N_{i,j}^k. \end{array} \right. \quad (5)$$

## Segundo esquema

Aplicando el método de Euler a (5) se obtiene,

$$\left\{ \begin{array}{l} W_{i,j}^{k+1} = W_{i,j}^k + \Delta t \left( \frac{W_{i+1,j}^k - 2W_{i,j}^k + W_{i-1,j}^k}{\Delta x^2} + \frac{W_{i,j+1}^k - 2W_{i,j}^k + W_{i,j-1}^k}{\Delta y^2} \right) \\ \quad + \Delta t \left( v \frac{W_{i+1,j}^k - W_{i,j}^k}{\Delta x} + a - W_{i,j}^k - W_{i,j}^k N_{i,j}^k \right) \\ \\ N_{i,j}^{k+1} = N_{i,j}^k + \Delta t \left( \frac{N_{i+1,j}^k - 2N_{i,j}^k + N_{i-1,j}^k}{\Delta x^2} + \frac{N_{i,j+1}^k - 2N_{i,j}^k + N_{i,j-1}^k}{\Delta y^2} \right) \\ \quad + \Delta t (W_{i,j}^k (N_{i-1,j}^k)^2 - m N_{i,j}^k) \end{array} \right. \quad (6)$$

# Simulador para el modelo de Klausmeier

```
1 def simulador_klausmeier(a, m, v):
2
3     num_nodos = 100 # Cantidad de nodos para x e y (malla en 2D)
4     long_xy = 100.0 # Longitud del dominio (cuadrado)
5     delta_x = long_xy/num_nodos # paso espacial
6     delta_y = long_xy/num_nodos # paso espacial
7
8     # Condiciones iniciales
9     W_0 = np.zeros((num_nodos, num_nodos))+1.0
10    N_0 = (np.random.rand(num_nodos, num_nodos)<0.05)*1.0 + 1.0
11
12    time_max = 50.0 # Tiempo maximo
13    delta_t = 0.001 # paso de tiempo
14    num_plot = 17    # numeros de plots a graficar
15
16    # Generamos los frame para plotear las simulaciones
17    fig, axes = plt.subplots(4, 4, figsize=(12, 12))
```

## Implementación y resultados numéricos

## Simulador para el modelo de Klausmeier

```
19 # Numero maximo de iteraciones
20 iter_max = int(time_max / delta_t)
21
22 # Vamos a plotear algunos frame entonces definimos un paso de plot
23 step_plot = iter_max // num_plot
24
25 # Empezamos a iterar
26 for idx in range(int(time_max / delta_t)):
27
28     # Metodo de Euler
29     W = W_0 + delta_t * ( a - W_0 - W_0*N_0*N_0 + v*Dx(W_0, delta_x)
30         + dw*DDxy(W_0, delta_x, delta_y))
31     N = N_0 + delta_t * ( W_0*N_0*N_0 - m*N_0 + dn * DDxy(N_0, delta_x, delta_y))
32
33     # Rescribimos para la siguiente iteracion
34     W_0 = W
35     N_0 = N
36
37     # Si se satisface las condiciones ploteamos
38     if idx % step_plot == 0 and idx < ((num_plot-1) * step_plot):
39         ax = axes.flat[idx // step_plot]
40         ax.imshow(N, cmap=VegMap, interpolation='bilinear', extent=[0, 1, 0, 1])
41         ax.set_axis_off()
42         ax.set_title(f'$t={idx * delta_t:.2f} s$')
```

## Implementación y resultados numéricos

## Resultados numéricos

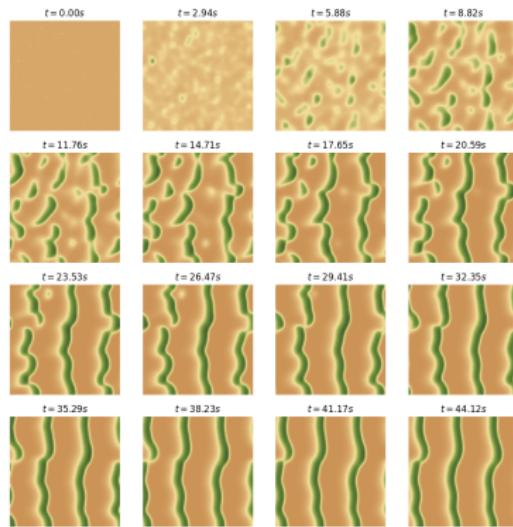


Figura: Una muestra del algoritmo ejecutado para un tiempo maximo 50s.

# Conclusiones

- No siempre es necesario recurrir a métodos numéricos que ofrezcan una solución extremadamente precisa; en muchos casos, basta con capturar la esencia del modelo.
- VisualPDE entiende esta idea y se enfoca en la simplicidad de los métodos numéricos.
- Este tipo de herramienta son necesarias en la actualidad para que investigadores de distintas disciplinas puedan aprender e implementar modelos de EDP de forma práctica y accesible

# Referencias

-  Lv, Jianping and Li, Chunguang and Dong, Jianqiang.  
*A high accuracy compact difference scheme and numerical simulation for a type of diffusive plant-water model in an arid flat environment.*  
AIMS Mathematics 2024.
-  Walker, Benjamin J and Townsend, Adam K and Chudasama, Alexander K and Krause, Andrew L.  
*VisualPDE: rapid interactive simulations of partial differential equations.*  
Bulletin of Mathematical Biology. Springer 2023.