Programa web para visualización de conjuntos en sistemas dinámicos discretos en $\mathbb{C}$

*i*magi

Renato Leriche Vázquez[1]

[1]Facultad de Ciencias,
UNAM

Seminario de Dinámica Holomorfa, Mayo 2016

Índice

*i*magi

- "Imagi" significa imaginar, en esperanto.
- Programa web (sobre HTML5), en JavaScript $\Longrightarrow$ independente de plataforma.
- Creado inicialmente para graficación de telarañas e itinerarios en familias de transformaciones Möbius en 2 pedazos.
- Graficación de Mandelbrots, Julias y órbitas en familias generales $\{h_\kappa : \mathbb{C} \to \mathbb{C}\}$.

*i*magi

*i*magi

- "Imagi" significa imaginar, en esperanto.
- Programa web (sobre HTML5), en JavaScript $\implies$ independente de plataforma.
- Creado inicialmente para graficación de telarañas e itinerarios en familias de transformaciones Möbius en 2 pedazos.
- Graficación de Mandelbrots, Julias y órbitas en familias generales $\{h_\kappa : \mathbb{C} \to \mathbb{C}\}$.

*i*magi

- "Imagi" significa imaginar, en esperanto.
- Programa web (sobre HTML5), en JavaScript $\Longrightarrow$ independente de plataforma.
- Creado inicialmente para graficación de telarañas e itinerarios en familias de transformaciones Möbius en 2 pedazos.
- Graficación de Mandelbrots, Julias y órbitas en familias generales $\{h_\kappa : \mathbb{C} \to \mathbb{C}\}$.

*i*magi

- "Imagi" significa imaginar, en esperanto.
- Programa web (sobre HTML5), en JavaScript $\Longrightarrow$ independente de plataforma.
- Creado inicialmente para graficación de telarañas e itinerarios en familias de transformaciones Möbius en 2 pedazos.
- Graficación de Mandelbrots, Julias y órbitas en familias generales $\{h_\kappa : \mathbb{C} \to \mathbb{C}\}$.

Secciones

- Canvas izquierdo: Mandelbrot de $\{f_\kappa\}$.
- Canvas derecho: Julia, órbitas, telarañas, etc, de f_κ.
- Menú y controles.
    - Familias transformaciones Möbius en 2 pedazos.
    - Familias de funciones generales
    - Configuración de algoritmos
    - Configuración general
    - Ayuda

Canvases

- Acercamiento: **Ctrl + Mouse Left** $\searrow$.
- Alejamiento: **Ctrl + Mouse Left** $\nwarrow$.
- Indicación de coordenadas: **Move Mouse**.
- Elección de parámetro κ: **Double Click Mouse Left** en canvas izquierdo $\Longrightarrow$ redibujo de canvas derecho.
- Guardar imagen: **Mouse Right**.

Möbius en 2 Pedazos

- Especificación de transformaciones Möbius:

$$f_\kappa(z) = \frac{a_f(\kappa)z + b_f(\kappa)}{c_f(\kappa)z + d_f(\kappa)}, \quad g_\kappa(z) = \frac{a_g(\kappa)z + b_g(\kappa)}{c_g(\kappa)z + d_g(\kappa)}$$

$$F_{\kappa,R}(z) = \begin{cases} f_\kappa(z) & \text{si } z \in R \\ g_\kappa(z) & \text{si } z \notin R \end{cases}$$

- Pedazo R:
  - $r > 0 \implies R = D(c, r)$, disco con centro en c y radio $r > 0$.
  - $r \leq 0 \implies R$ semiplano definido por la recta que pasa por c con inclinación $-r\pi$
    $R = \{z \mid n \cdot (z - c) < 0\}$, $n = ie^{-r\pi i}$

Möbius en 2 Pedazos

- Especificación de transformaciones Möbius:

$$f_\kappa(z) = \frac{a_f(\kappa)z + b_f(\kappa)}{c_f(\kappa)z + d_f(\kappa)}, \quad g_\kappa(z) = \frac{a_g(\kappa)z + b_g(\kappa)}{c_g(\kappa)z + d_g(\kappa)}$$

$$F_{\kappa,R}(z) = \begin{cases} f_\kappa(z) & \text{si } z \in R \\ g_\kappa(z) & \text{si } z \notin R \end{cases}$$

- Pedazo R:
  - $r > 0 \implies R = D(c,r)$, disco con centro en c y radio $r > 0$.
  - $r \leq 0 \implies R$ semiplano definido por la recta que pasa por c con inclinación $-r\pi$
    $R = \{z|\ n \cdot (z - c) < 0\}$, $n = ie^{-r\pi i}$

Algoritmos

Aproximaciones de:

- Julia lleno $\mathscr{J}(F_{\kappa,R}) = \{z | \mathscr{O}(z, F_{\kappa,R})\,\text{acotada}\}$.
- Órbita $\mathscr{O}(A, F_{\kappa,R}) = \bigcup_{n\geq 0} F^n_{\kappa,R}(A)$, $A \subset \mathbb{C}$.
- Telaraña $Spid(F_{\kappa,R}) = \overline{\bigcup_{n\geq 0} F^{-n}_{\kappa,R}(\partial R)}$.
- “Estimación” de telaraña $Spid(F_{\kappa,R})$ y de itinerarios.
- Itinerarios $I_{F_{\kappa,R}}(z) \in \Sigma^2 = \{0,1\}^{\mathbb{N}}$, donde

$$(I_{F_{\kappa,R}}(z))_n = \begin{cases} 0 & \text{si } F^n_{\kappa,R}(z) \in R \\ 1 & \text{si } F^n_{\kappa,R}(z) \notin R \end{cases}.$$

Siempre se dibuja $\mathscr{M}(\{F_{\kappa,R}\}, s_0) = \{\lambda | \mathscr{O}(s_0, F_{\lambda,R})\,\text{acotada}\}$.

Generales

- $h_\kappa : \mathbb{C} \to \mathbb{C}$.
- Operaciones en $\mathbb{C}$
  - + : Suma.
  - - : Resta.
  - * : Multiplicación.
  - / : División.
  - ^ : Potencia.
  - % : Módulo, $(x + yi)\%(a + bi) = (x\%a) + (y\%b)i$.
- Ejemplos:
  - z*z + k
  - z^2 - (2 - 0.5i)*z + k/(z^2)

Generales

- $h_\kappa : \mathbb{C} \to \mathbb{C}$.
- Operaciones en $\mathbb{C}$
  - + : Suma.
  - - : Resta.
  - * : Multiplicación.
  - / : División.
  - ^ : Potencia.
  - % : Módulo, $(x + yi)\%(a + bi) = (x\%a) + (y\%b)i$.
- Ejemplos:
  - z*z + k
  - z^2 - (2 - 0.5i)*z + k/(z^2)

Generales

- $h_\kappa : \mathbb{C} \to \mathbb{C}$.
- Operaciones en $\mathbb{C}$
  - $+$: Suma.
  - - : Resta.
  - ***** : Multiplicación.
  - / : División.
  - ^ : Potencia.
  - **%** : Módulo, $(x + yi)\%(a + bi) = (x\%a) + (y\%b)i$.
- Ejemplos:
  - **z*z + k**
  - **z^2 - (2 - 0.5i)*z + k/(z^2)**

Funciones permitidas

- **re im abs** $(=|z|)$ **arg** $(=\sphericalangle(z))$ **conj** $(=\overline{z})$ **neg** $(=-z)$ **norm** $(=\frac{z}{|z|})$
- **floor** $(=\lfloor x\rfloor+\lfloor y\rfloor i)$ **ceil** $(=\lceil x\rceil+\lceil y\rceil i)$ **round** $(=\lfloor x+0{,}5\rfloor+\lfloor y+0{,}5\rfloor i)$ **iPart** (parte entera) **fPart** (parte fraccionaria)
- **square** $(=z^2)$ **cube** $(=z^3)$ **sqrt** $(=z^{\frac{1}{2}})$ **cbrt** $(=z^{\frac{1}{3}})$
- **exp** $(=e^z)$ **log** $(=ln(z))$ **gamma** $(=\Gamma(z))$ **fact** $(=\prod(x-n)+\prod(y-n)i)$
- **cos sin tan sec csc cot arccos arcsin arctan arcsec arccsc arccot cosh sinh tanh sech csch coth arccosh arcsinh arctanh arcsech arccsch arccoth.**
- Constantes: **i** $(i^2=-1)$, **pi** $(=\pi)$ y **e**.

Ejemplo

- Ejemplo $h_\kappa : \mathbb{R}^2 \to \mathbb{R}^2$.
- Mapeo de Hénon:

$$h_{a,b}(x, y) = (a - by + x^2,\ x),\ a, b \in \mathbb{R}$$

- $\Longrightarrow$ **re(k) - im(k)*im(z) + re(z)*re(z) + i*re(z)**

Ejemplo

- Ejemplo $h_\kappa : \mathbb{R}^2 \to \mathbb{R}^2$.
- Mapeo de Hénon:

$$h_{a,b}(x,y) = (a - by + x^2,\ x),\ a,b \in \mathbb{R}$$

- $\Longrightarrow$ re(k) - im(k)*im(z) + re(z)*re(z) + i*re(z)

Ejemplo

- Ejemplo $h_\kappa : \mathbb{R}^2 \to \mathbb{R}^2$.
- Mapeo de Hénon:

$$h_{a,b}(x,y) = (a - by + x^2,\ x),\ a, b \in \mathbb{R}$$

- $\Longrightarrow$ **re(k) - im(k)*im(z) + re(z)*re(z) + i*re(z)**

Algoritmos

Aproximaciones de:

- Julia lleno $\mathscr{J}(h_\kappa)$.
- Órbita $\mathscr{O}(A, h_\kappa)$, $A \subset \mathbb{C}$.

Siempre se dibuja $\mathscr{M}(\{h_\kappa\}, s_0)$.

Parámetros para algoritmos

Para cualquier algoritmo (Mandelbrot, Julia, órbita, telaraña...)

- Iteraciones máximas N.
- Cota para el criterio de escape M.
- Criterio de escape:
  - $|z| > M$ (internamente usa $|z|^2 > M^2$).
  - $|Re(z)| > M$.
  - $Re(z) > M$. (Ejemplo con **k*exp(z)**).
- Semilla s_0 para el Mandelbrot.
- Parámetro κ.

Nota: s_0 y κ aceptan cualquier expresión, por ejemplo
-1+i, **i/2**, **(0.5-i)/(-2+0.25i)** ó **exp(i*pi/3)** ($\in S^1$).

Parámetros para órbitas

- Conjunto $A \subset \mathbb{C}$: Punto, Segmento de Recta, Circunferencia o Rectángulo Relleno.
- Número de puntos en A.
- Puntos z_0 y z_1 para definir A:
  - Punto: $A = \{z_0\}$.
  - Segmento de Recta: $A = \{z_0 + t(z_1 - z_0)\}_{t \in [0,1]}$.
  - Circunferencia: $A = \{z_0 + re^{it}\}_{t \in [0,2\pi]}$, donde $r = |z_0 - z_1|$.
  - Rectángulo: $A = [x_0, y_0] \times [x_1, y_1]$.
- "Grosor" de puntos.

Elementos de Dibujo

- Mapeos de color fijos, uno personalizable (usar **custom** en la lista), pueden invertirse.
- Se pueden mostrar/ocultar: Ejes coordenados, etiquetas, frontera del pedazo D.
- Colores de fondo y de frente.
- “Resetear” canvases izquierdo y derecho por separado.

To Do

- Conseguir tesistas para hacer el trabajo que falta ☺.
- Mejorar algoritmos.
- Añadir soporte para **WebGL** $\Longrightarrow$ Generación de dibujo más rápido, uso de 3D.
- Diagrama de bifurcaciones, 2D y 3D.
- Visualización en $\hat{\mathbb{C}}$ (esfera en 3D).
- Escribir manual de usuario.
- Varios idiomas (añadir español, al menos).
- ¿Sugerencias?

To Do

- Conseguir tesistas para hacer el trabajo que falta ☺.
- Mejorar algoritmos.
- Añadir soporte para **WebGL** $\Longrightarrow$ Generación de dibujo más rápido, uso de 3D.
- Diagrama de bifurcaciones, 2D y 3D.
- Visualización en $\hat{\mathbb{C}}$ (esfera en 3D).
- Escribir manual de usuario.
- Varios idiomas (añadir español, al menos).
- ¿Sugerencias?

To Do

- Conseguir tesistas para hacer el trabajo que falta ☺.

- Mejorar algoritmos.
- Añadir soporte para **WebGL** $\implies$ Generación de dibujo más rápido, uso de 3D.
- Diagrama de bifurcaciones, 2D y 3D.
- Visualización en $\hat{\mathbb{C}}$ (esfera en 3D).
- Escribir manual de usuario.
- Varios idiomas (añadir español, al menos).
- ¿Sugerencias?

Fin

¡Gracias!