Universidad Centroamericana "José Simeón Cañas"
Facultad de Ingeniería y Arquitectura
Departamento de Matemática
Análisis numérico Ciclo 01/2021
Ing. Daniel Augusto Sosa (Secciones 1 y 2)
Instructor: Kevin Aquino

Guía de ejercicios 2

1. Considere la función $f(x) = x - \tan(x)$ en el intervalo $[1, 2]$. Tenemos que $f(1) < 0$ y $f(2) > 0$. Al ver esto, se decide aplicar el método de bisección con una tolerancia $\texttt{TOL} = 10^{-15}$. Para lograr la tolerancia, se calculan 50 iteraciones, obteniendo los siguientes resultados:

n	a_n	b_n	p_n	$f(p_n)$
0	1	2	1.5	−12.6014199471717
1	1.5	2	1.75	7.27037992250933
2	1.5	1.75	1.625	20.0558627623696
3	1.5	1.625	1.5625	−118.970005722542
⋮	⋮	⋮	⋮	⋮
49	1.57079625129699	1.57079637050628	1.57079625129699	−13245400.0360663

¿Por qué falló el método?

2. Muchas ecuaciones de estado han sido desarrolladas para describir la relación $P-V-T$ (presión, temperatura, volumen) de los gases. Una de las ecuaciones mejor conocidas es la ecuación de Beattie-Bridgeman,

$$P = \frac{RT}{V} + \frac{\beta}{V^2} + \frac{\gamma}{V^3} + \frac{\delta}{V^4}$$

donde P es la presión, V el volumen molar, T la temperatura, β, γ y δ son parámetros dependientes de la temperatura y característicos del gas, y R es la constante universal de los gases, en unidades compatibles. Considere los siguientes valores:

- $\beta = -1.16583607818894$
- $\delta = -1.25100322759761 \times 10^{-4}$
- $R = 0.08205$
- $\gamma = 0.0542253936905836$
- $T = 273.15$
- $P = 200$

(a) Formule el problema como un problema de búsqueda de raíces. Determine la función $f(V)$ a utilizar en el método de Newton y su derivada.

(b) Implemente el método de Newton con aproximación inicial,

$$V_0 = \frac{RT}{P}$$

y tolerancia $\texttt{TOL} = 10^{-9}$.

3. Se desea conocer al ángulo de rotación ideal de una cámara, dado el radio de esta (r), medido desde el centro de rotación, como función del ángulo x en radianes.

$$r(x) = 0.5 + 0.5e^{-x/2\pi}\sin(x) \quad 0 \leq x \leq 2.1.$$

Para un ángulo de rotación dado, la función de desplazamiento del montaje de dicha cámara viene dada por $d(x)$, y se desea que sea igual al radio de la cámara, $r(x)$. Por lo tanto el ángulo deseado x que produce el desplazamiento deseado D es la solución de la ecuación

$$f(x) = r(x) - D = 0.5 + 0.5e^{-x/2\pi}\sin(x) - D = 0.$$

Asuma que $D = 0.75$.

(a) Utilice el método de bisección para aproximar la raíz de la ecuación con una tolerancia $\texttt{TOL} = 0.2$.

(b) Refine la aproximación anterior $p_{\text{bisección}}$ utilizándola como valor inicial para el método de la secante, es decir, tome $p_0 = p_{\text{bisección}}$. Para p_1 use $p_0 + 0.1$. Itere hasta lograr una tolerancia $\texttt{TOL} = 10^{-12}$.

4. Sea A un número real positivo y considere la función $g(x) = 2x - Ax^2$. Demuestre que si la iteración de punto fijo para esta función converge a un número p diferente de 0, entonces ese número es $p = 1/A$.

5. De las siguientes iteraciones de punto fijo, indique cuáles convergen al punto fijo α (dado que x_0 está suficientemente cerca de α). De ser convergente, indique el orden de convergencia; para convergencia lineal, encuentre la constante asintótica de convergencia. En cualquiera de los casos, realice 10 iteraciones utilizando los parámetros x_0 para cada ejercicio.

(a) $x_{n+1} = -16 + 6x_n + \dfrac{12}{x_n} \quad \alpha = 2, \quad x_0 = 1$

(b) $x_{n+1} = \dfrac{2}{3}x_n + \dfrac{1}{x_n^2} \quad \alpha = 3^{\frac{1}{3}}, \quad x_0 = 2.5$

(c) $x_{n+1} = \dfrac{12}{1 + x_n} \quad \alpha = 3, \quad x_0 = 2.5$

6. La constante de Littlewood-Salem-Izumi α_0, definida como la única solución en $0 < \alpha < 1$ de

$$(\star) \qquad \int_0^{3\pi/2} \frac{\cos(t)}{t^\alpha} \mathrm{d}t = 0$$

resulta de interés en la teoría de series trigonométricas. Para aproximar su valor, proceda de la siguiente forma:

(a) Convierta $(\star)$ a una integral sobre el intervalo $(0, 1)$ aplicando la sustitución $t = \frac{3\pi}{2}x$.

(b) Note que el lado derecho de $(\star)$ es una función de α, de manera que podemos calcular su derivada respecto a α. Para calcular la derivada, puede utilizar la *regla de derivación bajo el signo de la integral*:

$$\frac{\mathrm{d}}{\mathrm{d}y} \int_a^b f(x, y) \mathrm{d}x = \int_a^b \frac{\partial}{\partial y} f(x, y) \mathrm{d}x.$$

Haciendo uso de la regla anterior, calcule la derivada respecto a α de la función encontrada en el literal anterior.

(c) Implemente el método de Newton para encontrar α_0 usando las funciones encontradas en los literales anteriores. Use $\alpha_{\text{inicial}} = 0.5$ y $\texttt{TOL} = 10^{-15}$.

Nota: Para este ejercicio, necesitará calcular integrales de forma numérica en cada iteración. Se recomienda implementar el método de Newton en Python y usar la función `quad` del módulo `scipy.integrate`.

Para algo de historia sobre este problema, puede entrar aquí, páginas 15, 16 y 17.

7. (Newton en $\mathbb{C}$) El método de Newton se puede utilizar también para aproximar raíces complejas. La derivada de una función de variable compleja se define de forma análoga a la derivada de una función de variable real, de modo que al considerar la función

$$f(z) = z^3 - 2z + 2,$$

tenemos que $f'(z) = 3z^2 - 2$, por lo que la fórmula iterativa del método de Newton resulta

$$p_{n+1} = p_n - \frac{p_n^3 - 2p_n + 2}{3p_n^2 - 2}.$$

(a) Modifique su implementación del método de Newton para permitir aritmética compleja. Recuerde que el valor absoluto de un número complejo $z = a + bi$ es $|\, z \,| = \sqrt{a^2 + b^2}$. [1]

(b) Aproxime una de las raíces de la función f dada arriba, usando como valor inicial $1 + i$, hasta llegar a una tolerancia de 10^{-10}.

[1] Si está usando Python no tiene que preocuparse por el tipo de datos, solo basta con escribir, por ejemplo, `1 + j`. Si, en cambio, está usando C++, en lugar de incluir las funciones del archivo de cabecera `<cmath>`, debe hacerlo de `<complex>` y usar, por ejemplo, el tipo de dato `std::complex<double>`.

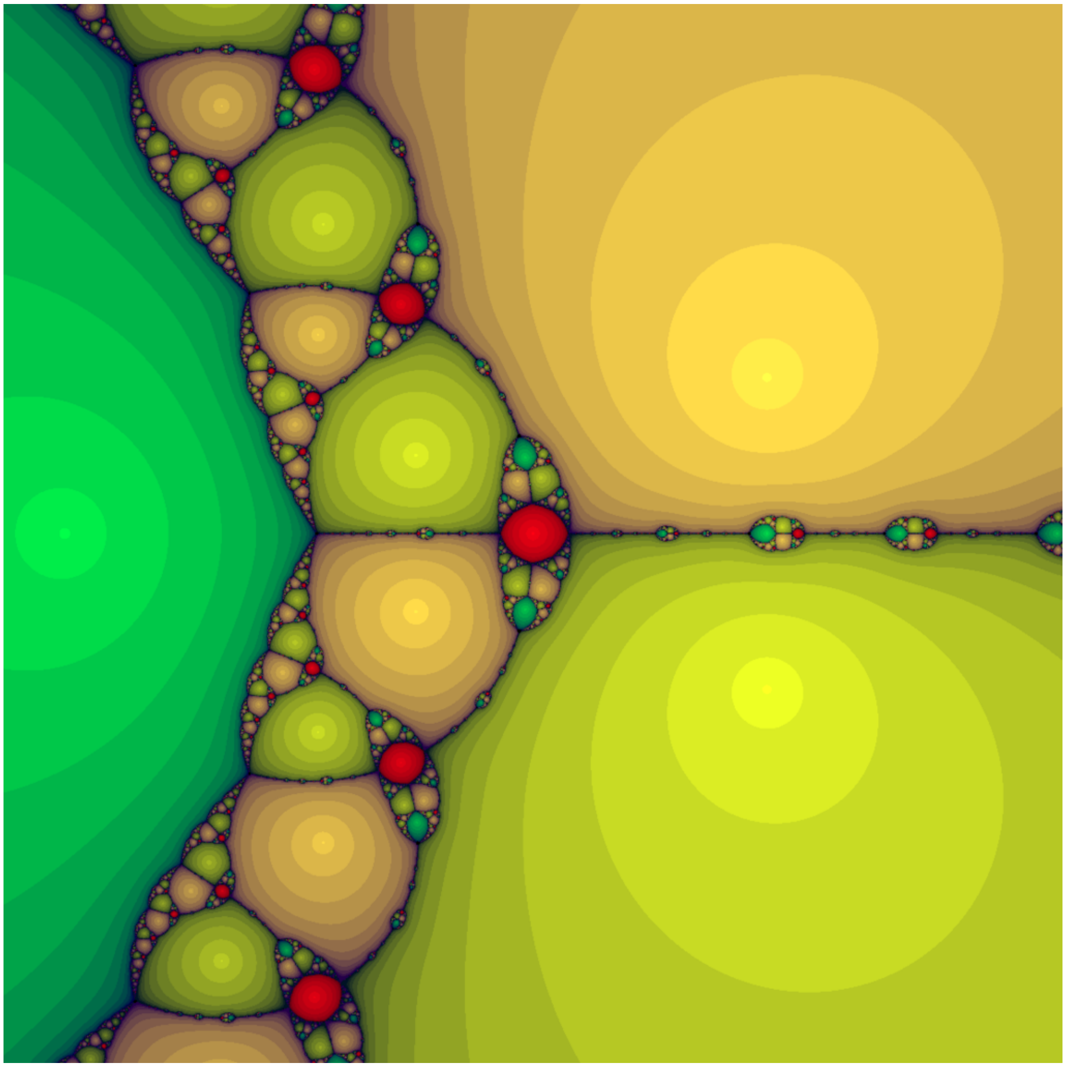

Figura 1: El método de Newton para f.

La imagen anterior muestra el plano complejo $\mathbb{C}$ coloreado de acuerdo a las regiones de convergencia del método de Newton. La función f tiene tres raíces: una real y dos complejas conjugadas. Todos los puntos en la región verde convergen a la raíz real de f, mientras que las dos zonas coloreadas de la derecha corresponden a puntos que convergen a las raíces complejas conjugadas. La convergencia en el método de Newton es un asunto **delicado:** si elige un punto en las regiones en rojo, el método no converge. En este ejercicio se le da un punto en la región amarilla y luego de ciertas iteraciones, este converge a la raíz.

La imagen anterior se llama **fractal de Newton** para f. Los fractales son un caso curioso: comenzaron a ser estudiados mucho antes que la gente los pudiera visualizar. Las primeras imágenes de los fractales fueron posibles hasta los años 70, gracias a las computadoras.

Selected solutions

1. Una gráfica de la función puede ayudar.

2. $p_6 = 0.080306265$, con los siguientes resultados:

n	p_n	p_{n+1}	$E_{\text{abs}} = \mid p_{n+1} - p_n \mid$
0	**0.112059787**	0.063346630	0.048713158
1	0.063346630	0.074770281	0.011423651
2	0.074770281	0.079718027	0.004947747
3	0.079718027	0.080299685	0.000581658
4	0.080299685	0.080306264	0.000006579
5	0.080306264	**0.080306265**	0.000000001

3.

(a) $p_3 = p_{\text{bisección}} = 0.65625$, obteniendo los siguientes resultados:

n	a_n	b_n	p_n	$f(p_n)$	$E_{\text{abs}} = \frac{b-a}{2}$
0	0.000000000	2.1	1.049999952	0.1169652352	1.049999952
1	0.000000000	1.049999952	0.5249999762	-0.01948230977	0.5249999762
2	0.5249999762	1.049999952	0.7874999642	0.0625602922	0.2624999881
3	0.5249999762	0.7874999642	**0.65625**	0.02481890211	0.131249994

(b) Bastan 6 iteraciones. $p_7 = 0.580438364567228$, obteniendo los siguientes resultados:

n	p_n	p_{n+1}	p_{n+2}	$E_{abs} = \mid p_{n+2} - p_{n+1} \mid$
0	**0.65625**	**0.75625**	0.57175440163901	0.18449559836099
1	0.75625	0.57175440163901	0.581366932491075	0.0096125308520655
2	0.57175440163901	0.581366932491075	0.580442649993477	0.000924282497598699
3	0.581366932491075	0.580442649993477	0.58043836243865	$4.28755482701588 \cdot 10^{-6}$
4	0.580442649993477	0.58043836243865	0.580438364567233	$2.12858299215857 \cdot 10^{-12}$
5	0.58043836243865	0.580438364567233	**0.580438364567228**	$4.87755452346916 \cdot 10^{-15}$

7. Datos calculados usando `std::complex<double>` en C++. Bastan 6 iteraciones. $p_6 = 0.88464617712 + 0.58974280502i$.

n	p_n	p_{n+1}	$E_{\text{abs}} = \mid p_{n+1} - p_n \mid$
0	$\mathbf{1 + i}$	$0.9 + 0.7i$	0.31622776602
1	$0.9 + 0.7i$	$0.88366948601 + 0.60026024723i$	0.10106781866
2	$0.88366948601 + 0.60026024723i$	$0.88458896742 + 0.58983589987i$	0.010464820311
3	$0.88458896742 + 0.58983589987i$	$0.8846461653+0.5897428062i$	0.00010926128464
$\vdots$	$\vdots$	$\vdots$	$\vdots$
5	$0.88464617712 + 0.58974280502i$	$\mathbf{0.88464617712+ 0.58974280502i}$	$1.1102230246 \cdot 10^{-16}$