

Para la solución de los siguientes ejercicios, cree una copia del [repositorio del curso complementario](#) utilizando **fork** con el título **MetCompCompl-202320_{APELLIDO1}_{APELLIDO1}**. En la carpeta **Taller_2**, guarde los archivos producidos en el taller como se describe en los enunciados a continuación.

1. Derivadas [20 pt]

Un bucle de cobre de 25 cm de diámetro y resistencia $R = 1,75 \text{ k}\Omega$ se encuentra en una región de campo magnético alternante $\mathbf{B} = B_0 \cos(2\pi ft)(-\hat{\mathbf{z}})$ como se muestra en la figura. En $t = 0$, el bucle empieza a girar a una velocidad angular constante Ω .

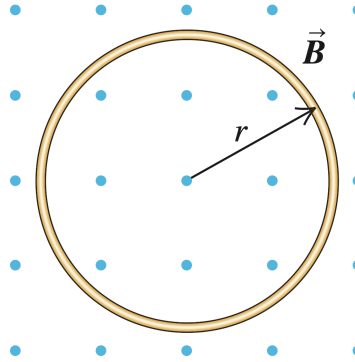


Figura 1: Bucle de cobre en un campo magnético no uniforme.

Utilizando $\Phi_B = \mathbf{B} \cdot \mathbf{A} = BA \cos \theta$ donde θ es el ángulo entre el campo magnético y el área, se puede demostrar que el flujo magnético a través del bucle se puede escribir como

$$\Phi_B = \pi r^2 B_0 \cos(\Omega t) \cos(2\pi ft)$$

1. Grafique la corriente inducida sobre el bucle en función del tiempo para $B_0 = 0,05T$, $f = 7Hz$, $\Omega = 3,5rad/s$. Recuerde que por la ley de Faraday-Lenz, $I = -\frac{1}{R} \frac{d\Phi_B}{dt}$. Solo grafique para dos períodos de rotación del bucle.
2. Encuentre los primeros tres instantes de tiempo en los que la corriente sobre el bucle es cero.

2. Fórmulas de Newton-Cotes [20 pt]

Las fórmulas de Newton-Cotes para la integración de una función $f(x)$ en el intervalo $[a, b]$ se construyen dividiendo este en subintervalos de m puntos igualmente espaciados $\{x_i, x_{i+1}, \dots, x_m\}$ e interpolar un polinomio de Lagrange de grado $m - 1$ y hacer esta integral. Posteriormente, se suman los resultados de las integrales de todos los puntos y se obtendrá un resultado de la forma

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n-1} c_i f(x_i) \quad (1)$$

donde c_i se conocen como los pesos. A partir de esta notación, escoja **SOLAMENTE UNO** de los siguientes puntos para resolver.

2.1. Regla de Simpson de 3/8

La fórmula de Simpson de 3/8 consiste en dividir el intervalo en subintervalos de $m = 4$ puntos cada uno. Demuestre que la integral de cualquiera de estos subintervalos está dada por

$$\int_{x_i}^{x_{i+3}} f(x)dx \approx \frac{3h}{8} (f(x_i) + 3f(x_{i+1}) + 3f(x_{i+2}) + f(x_{i+3}))$$

2.2. Regla de Boole

La fórmula de Boole consiste en dividir un intervalo $[a, b]$ en subintervalos de $m = 5$ puntos cada uno. Utilizando estos 5 puntos, se interpola un polinomio de Lagrange de grado cuatro y se integra desde x_i hasta x_{i+4} para obtener

$$\int_{x_i}^{x_{i+4}} f(x)dx = \frac{2h}{45} (7f(x_i) + 32f(x_{i+1}) + 12f(x_{i+2}) + 32f(x_{i+3}) + 7f(x_{i+4}))$$

Demuestre a partir de este resultado que para la regla de Boole, los pesos en la ecuación 1 estarán dados por

$$c_i = \begin{cases} 7 & i = 0, i = n - 1 \\ 32 & i \in \{1, 3, 5, n - 2\} \\ 12 & i \in \{2, 6, 10, n - 3\} \\ 14 & i \in \{4, 8, 12, n - 5\} \end{cases}$$

A partir de esto, ¿qué condición se debe cumplir para el número de subintervalos $n - 1$?

3. Cuadratura Gaussiana [30 pt]

Realice los siguientes puntos en el notebook `Punto3_CuadraturaGauss.ipynb`.

3.1. Cuadratura Gauss-Laguerre

La aproximación de Gauss-Laguerre consiste en encontrar los k nodos y pesos tal que

$$\int_0^\infty e^{-x} f(x) dx = \sum_{k=0}^{n-1} c_k f(x_k)$$

Se puede demostrar que para n nodos, estos están dados por los ceros de los *Polinomios de Laguerre*, definidos mediante la fórmula de Rodrigues como

$$L_n(x) = \frac{e^x}{n!} \frac{d^n}{dx^n} (e^{-x} x^n)$$

A partir de esto, se puede crear una función en Python que retorne el n -ésimo polinomio de Laguerre. Los polinomios de Laguerre también satisfacen la ecuación de recursión

$$L_{k+1} = \frac{(2k+1-x)L_k(x) - kL_{k-1}(x)}{k+1}$$

por lo que se puede escribir la función anterior de forma recursiva teniendo en cuenta que $L_0(x) = 1, L_1(x) = 1 - x$.

1. Escriba una función `GetLaguerre(n,x)` que retorne el n -ésimo polinomio de Laguerre de forma recursiva.
2. Escriba una función `GetAllRootsGLag(n)` que encuentre todas las raíces del n -ésimo polinomio de Laguerre. Para esto, utilice el hecho de que las raíces están acotadas en el intervalo $(0, n + (n-1)\sqrt{n})$
3. Escriba una función `GetWeightsGLag(n)` que encuentre los n -pesos del n -ésimo polinomio de Laguerre. Tenga en cuenta que estos pesos están dados por

$$c_k = \frac{x_k}{(n+1)^2 [L_{n+1}(x_k)]^2}$$

3.2. Cuadratura Gauss-Hermite

La aproximación de Gauss-Hermite consiste en encontrar los k nodos y pesos tal que

$$\int_{-\infty}^{\infty} e^{-x^2} f(x) dx = \sum_{k=0}^{n-1} c_k f(x_k)$$

Se puede demostrar que para n nodos, estos están dados por los ceros de los *Polinomios de Hermite*, definidos mediante la fórmula de Rodrigues como

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} e^{-x^2}$$

A partir de esto, se puede crear una función en Python que retorne el n -ésimo polinomio de Hermite. Los polinomios de Hermite también satisfacen la ecuación de recursión

$$H_{n+1}(x) = 2xH_n(x) - H'_n(x)$$

por lo que se puede escribir la función anterior de forma recursiva teniendo en cuenta que $H_0(x) = 1$, $H_1(x) = 2x$.

1. Escriba una función **GetHermite(n,x)** que retorne el n -ésimo polinomio de Hermite de forma recursiva.
2. Escriba una función **GetAllRootsGHer(n)** que encuentre todas las raíces del n -ésimo polinomio de Hermite. Para esto, utilice el hecho de que las raíces están acotadas en el intervalo $(-\sqrt{4n+1}, \sqrt{4n+1})$
3. Escriba una función **GetWeightsGHer(n)** que encuentre los n -pesos del n -ésimo polinomio de Hermite. Tenga en cuenta que estos pesos están dados por

$$c_k = \frac{2^{n-1} n! \sqrt{\pi}}{n^2 [H_{n-1}(x_k)]^2}$$

3.3. Aplicación

Utilizando la cuadratura **Gauss-Laguerre**, resuelva:

En 1852, el físico escocés James Clerk Maxwell describió la distribución de velocidades de moléculas de un gas como

$$P(v) = 4\pi \left(\frac{M}{2\pi RT} \right)^{3/2} v^2 \exp \left(-\frac{Mv^2}{2RT} \right)$$

1. Realizando la sustitución $u = \frac{Mv^2}{2RT}$, demuestre que $\int_0^\infty P(v)dv = 1$, es decir, que es una distribución de probabilidad.
2. Grafique $P(v)$ para distintas temperaturas. ¿Qué puede decir de la velocidad más probable (con mayor $P(v)$) a medida que aumenta la temperatura?
3. Para 10 distintas temperaturas, encuentre la *velocidad promedio*

$$v_{avg} = \int_0^\infty v P(v) dv$$

Grafique (en escala logarítmica) v_{avg} y T y verifique que se cumple

$$v_{avg} = \sqrt{\frac{8RT}{\pi M}}$$

4. Para 10 distintas temperaturas, encuentre la *velocidad media cuadrática*

$$v_{rms} = \sqrt{\int_0^\infty v^2 P(v) dv}$$

Grafique (en escala logarítmica) v_{rms} y T y verifique que se cumple

$$v_{rms} = \sqrt{\frac{3RT}{M}}$$

5. Usando lo anterior, demuestre que la energía interna de un gas está dada por $E_{int} = \frac{3}{2} nRT$



Figura 2: Representación del Laboratorio Nacional de los Álamos, donde se desarrolló el método Monte Carlo en la película Oppenheimer (2023).

4. Método de Monte Carlo [30 pt]

Realice los siguientes puntos en el notebook `Punto4.MonteCarlo.ipynb`. Este punto lo haremos en la clase del viernes 22 de septiembre.

El método de Monte Carlo fue desarrollado de forma independiente por Enrico Fermi y por Stanisław Ulam (en conjunto al matemático John von Neumann) en el Laboratorio Nacional de Los Álamos al estudiar la difusión de neutrones para el desarrollo de la bomba de hidrógeno[2] [3].

1. Grafique el error porcentual para la integral $\int_0^\pi e^{-x} \sin x dx$ utilizando el método de Monte Carlo para su cálculo en función del número de muestras N de este método.
2. Determine la masa, centro de masa y momento de inercia que ocupa la región D y tiene la función de densidad dada, donde D está encerrada por el lazo derecho de la rosa de cuatro pétalos $r = \cos 2\theta$ con $\rho(x, y) = x^2 + y^2$.
3. La ecuación del transporte de neutrones requiere calcular la tasa de producción de neutrones [1] por medio de la integral

$$\iiint_{\Omega} \sin(x^2 + y^2 + z^2) e^{x^2 + y^2 + z^2} dV$$

donde Ω es la esfera de radio 1 centrada en el origen. Utilizando el método de Monte Carlo para N puntos aleatorios, aproxime el valor de esta integral.

Referencias

- [1] Elmer Eugene Lewis y Warren F Miller. «Computational methods of neutron transport». En: (1984).
- [2] Nicholas Metropolis y Stanislaw Ulam. «The monte carlo method». En: *Journal of the American statistical association* 44.247 (1949), págs. 335-341.
- [3] Eckhardt Roger. «Stan Ulam, John Von Neumann, and the Monte Carlo Method». En: *Los Alamos Science* 15 (1987), págs. 131-137.