

La solución a esta tarea debe cargarse a su repositorio en GitHub en la carpeta /MC/Tareas/HW6/ como un cuaderno de nombre `HW6.ipynb`. En el cuaderno dedicar una sección a cada ejercicio con subsecciones para cada literal. Es requisito que en todo lo hecho se pongan comentarios que expliquen lo que se está haciendo.

La fecha límite de entrega es el **viernes 10 de julio a las 23:59**. Puede trabajarse en parejas.

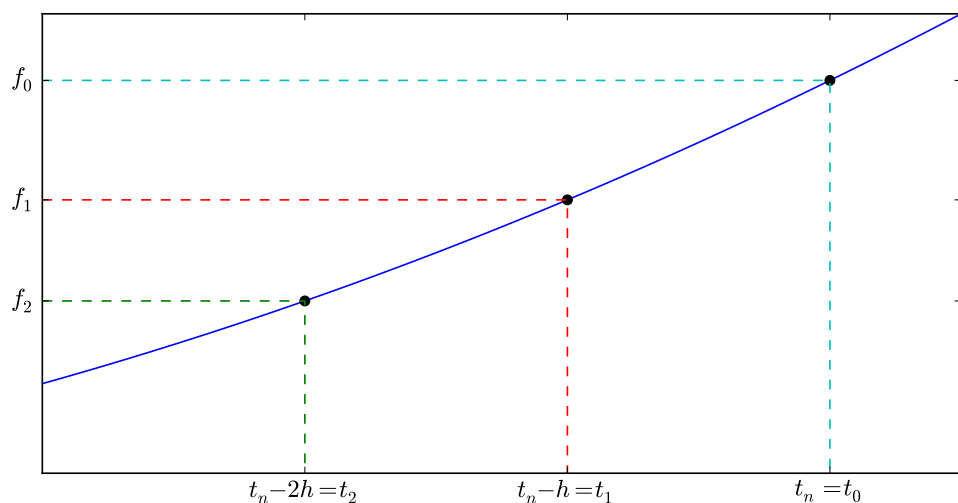
1. 40 pt **Cálculo Simbólico**

En este ejercicio queremos usar `Sympy` para calcular las ecuaciones que definen a los métodos explícitos de Adams-Bashforth para cualquier orden.

- (a) 20 pt Programe una función de `python` llamada `lagrange` que reciba el número N de puntos $\{t_i, f_i\}$ para $i = 0 \dots, N - 1$, y que entregue de regreso una expresión simbólica para el polinomio de grado $N - 1$ que los interpola. Puede usar `var` para definir símbolos de `SymPy` de forma programática y `eval` para convertir strings en nombres de objetos. Abajo a la izquierda se muestra el output en algunos casos.

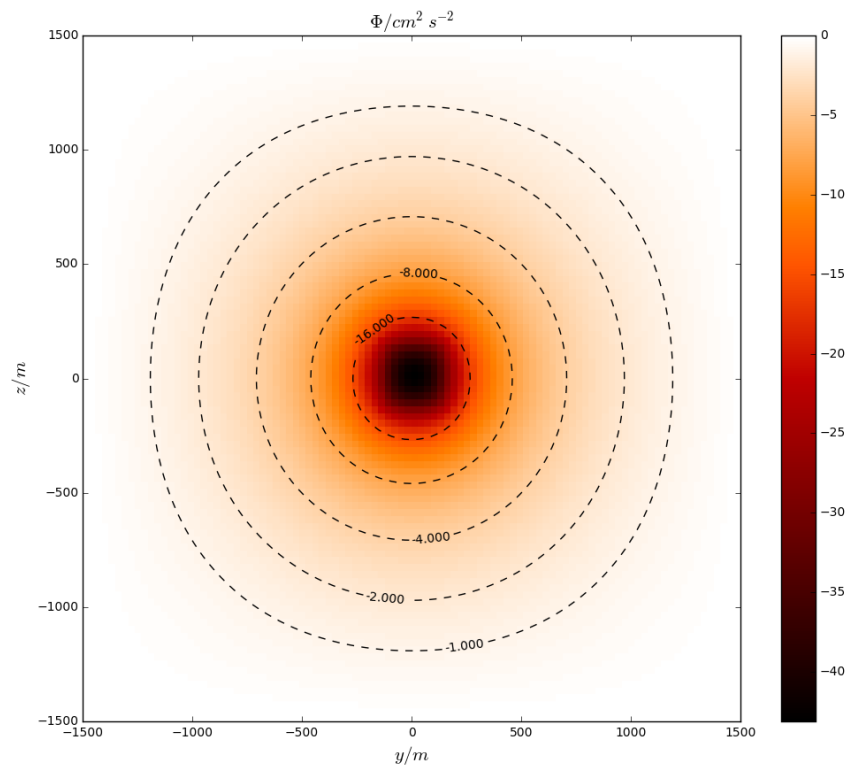
| | |
|---|---|
| <code>lagrange(1)</code> | <code>adams(2)</code> |
| f_0 | $\frac{h}{2}(3f_0 - f_1)$ |
| <code>lagrange(2)</code> | <code>adams(3)</code> |
| $\frac{f_0(t-t_1)}{t_0-t_1} + \frac{f_1(t-t_0)}{-t_0+t_1}$ | $\frac{h}{12}(23f_0 - 16f_1 + 5f_2)$ |
| <code>lagrange(3)</code> | <code>adams(4)</code> |
| $\frac{f_0(t-t_1)(t-t_2)}{(t_0-t_1)(t_0-t_2)} + \frac{f_1(t-t_0)(t-t_2)}{(-t_0+t_1)(t_1-t_2)} + \frac{f_2(t-t_0)(t-t_1)}{(-t_0+t_2)(-t_1+t_2)}$ | $\frac{h}{24}(55f_0 - 59f_1 + 37f_2 - 9f_3)$ |
| | <code>adams(5)</code> |
| | $\frac{h}{720}(1901f_0 - 2774f_1 + 2616f_2 - 1274f_3 + 251f_4)$ |

- (b) 20 pt Ahora, en el contexto de los métodos *multistep* de Adams-Bashforth defina una función llamada `adams` que reciba el orden m a considerar y de regreso entregue el incremento correspondiente. Para ello tome $t_i = t_n - i * h$ y $f_i = f(t_i)$ siendo h el *timstep*. Para ello debe utilizar la función definida en el anterior literal, y utilizar `integrate` y `simplify` de `SymPy`. Arriba a la derecha se muestra el output en algunos casos.

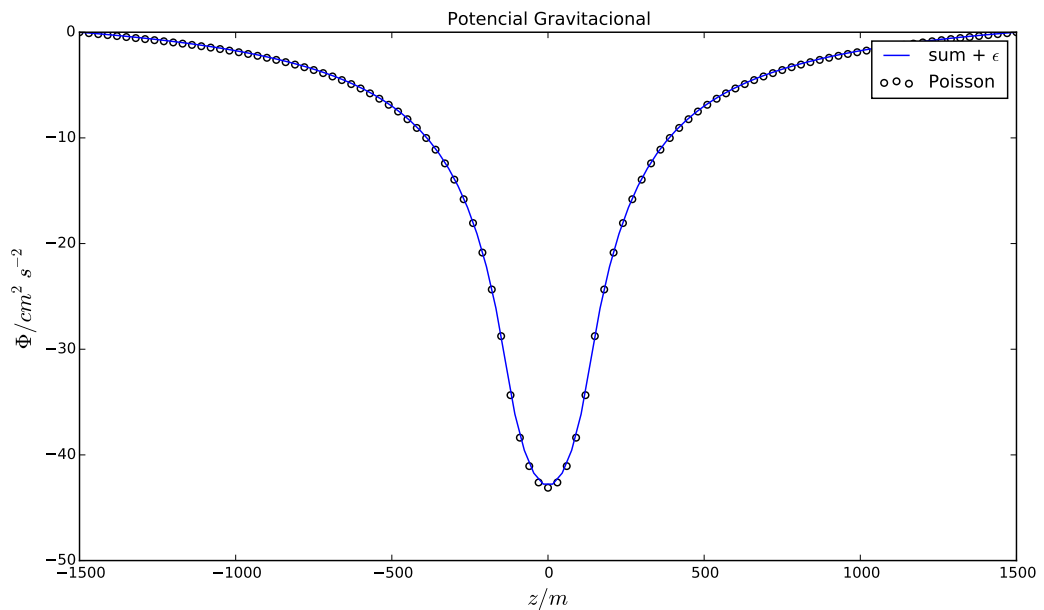


2. 60 pt En este ejercicio queremos encontrar la solución a la ecuación de Poisson para el caso del potencial gravitacional y usar el resultado para resolver la ecuación de movimiento de una pequeña partícula. Para poder hacerlo debe modificar lo visto para la ecuación de Poisson en el caso electromagnético para el caso gravitacional.

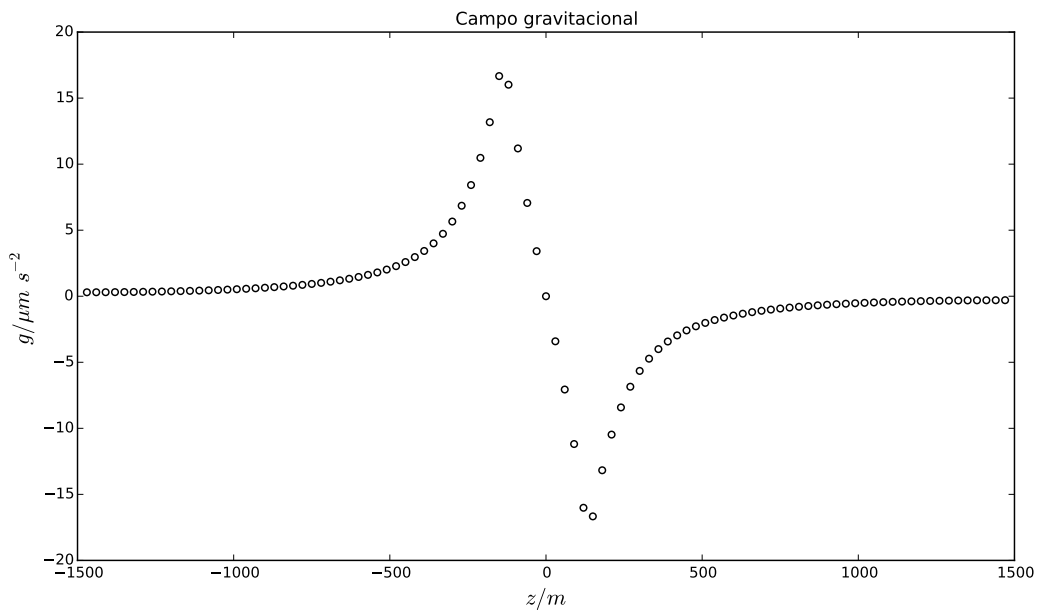
- (a) 20 pt Encuentre el potencial gravitacional de un cubo de densidad $400 \cdot \frac{\text{kg}}{\text{m}^3}$ y lado $270 \cdot \text{m}$ usando el método de relajación. Resuelva la ecuación de Poisson para el potencial en un cubo de 3.00 km de lado usando una cuadrícula en cada dirección de $30 \cdot \text{m}$ de lado. Use condiciones de frontera de Dirichlet tomando $\phi = 0$ en la frontera y tome como *ansatz* el potencial de una masa puntual. Use un sistema de coordenadas con el centro del cubo en el origen y los ejes xyz perpendiculares a las caras. Al final haga una gráfica de densidad del potencial en el plano yz con líneas equipotenciales superpuestas.



- (b) 15 pt También haga una gráfica del potencial a lo largo del eje z y compare con lo que se obtiene de segmentar el cubo en 9^3 masas puntuales y sumar los potenciales correspondientes.



- (c) 15 pt Luego calcule el campo gravitacional a lo largo del mismo eje haciendo derivadas numéricas.



- (d) 10 pt Finalmente encuentre el tiempo que le tomaría a una partícula de masa pequeña en chocar contra el cubo si parte del reposo sobre el eje z y a una distancia de 500 m de la cara más próxima. Para ello resuelva la ecuación de movimiento usando lo encontrado para el campo gravitacional en el anterior literal.