Universidad de Chile, Facultad de Ciencias Físicas y Matemáticas

FI3104-01 Métodos Numéricos para la Ciencia e Ingeniería

Tarea N°1: Espectro y Luminosidad del Sol

Bruno Scheihing

24 de Septiembre de 2015

Resumen

En la presente tarea se calcula numéricamente la luminosidad total del sol, a partir del espectro de radiación del Sol, medido justo afuera de la atmósfera de la tierra. Paralelamente se calcula la integral de la función de Planck sobre todas las longitudes de onda para el cuerpo negro que mejor se ajusta al Sol (temperatura efectiva 5778 K), obteniendo la energía por unidad de área por unidad de tiempo emitida desde la superficie del cuerpo negro. Juntando ambos resultados, se obtiene una estimación del radio efectivo del Sol, la que entrega un valor de $R_{Sol} = 6{,}955 \cdot 10^8\ [m]$, estando esto muy cerca del valor de un radio solar.

1. Introducción

El espectro del Sol corresponde a la cantidad de energía emitida por unidad de área por unidad de tiempo (es decir, flujo), por tipo de luz emitida. En este caso, el tipo de luz está caracterizado por la longitud de onda, obteniéndose así un espectro de la forma *flujo por longitud de onda* v/s *longitud de onda*.

Para la presente tarea se entrega como dato el espectro del Sol medido desde afuera de la atmósfera de la Tierra, en forma de un archivo **sun_AMO.dat**. En primer momento, se busca graficar el espectro en unidades apropiadas.

Luego se quiere integrar numéricamente este espectro sobre todas las longitudes de onda, y el resultado de este proceso es el flujo de energía (energía por unidad de área y unidad de tiempo). Para calcular la luminosidad total del sol, se asume que el flujo de energía $\vec{J}$ es de la forma $\vec{J} = J(r)\hat{r}$, donde r es la distancia al centro del Sol, y $\hat{r}$ es la dirección radial.

Con esto, basta integrar sobre la superficie esférica en la que se conoce el flujo, que corresponde a una distancia $R = 1$ UA (una unidad astronómica, i.e. la distancia desde el Sol hasta la Tierra). Esto equivale a decir que la luminosidad total L_{Sol} del Sol es $L_{Sol} = J(R) \cdot 4\pi R^2$.

Por otra parte, se pide integrar numéricamente la función de Planck:

$$B_\lambda(T) = \frac{2\pi hc^2/\lambda^5}{e^{hc/\lambda k_B T} - 1}$$

que representa la radiación de un cuerpo negro en unidades de energía por unidad de tiempo por unidad de área por unidad de longitud de onda. h es la constante de Planck, c es la velocidad de la luz en el vacío, k_B es la constante de Boltzmann, T es la temperatura del cuerpo negro y λ es la longitud de onda. La integral es con respecto a la longitud de onda, para obtener el flujo de energía que sale del cuerpo negro. Se debe comparar el resultado numérico con el analítico, e implementar un algoritmo para refinar la convergencia de la integral numérica.

Usando el resultado anterior se puede estimar el radio del Sol, mediante conservación de energía, pues el flujo total en la superficie del Sol debe ser el mismo que a través de una superficie esférica de 1 UA de radio.

Finalmente se compara la velocidad de ejecución de los algoritmos implementados para la integración, con los de librerías ya existentes, en particular el módulo **scipy** de **Python**.

2. Procedimiento

2.1. Plot del espectro del Sol

Para hacer el gráfico del espectro del Sol, basta leer el archivo **sun_AMO.dat** usando el método **loadtext** de **numpy**. Se convierten los datos a unidades apropiadas, y se grafica usando el módulo **matplotlib**.

2.2. Cálculo de la luminosidad total del Sol

Para calcular la integral del espectro de radiación del Sol se usa la regla del trapecio. Es decir, se toman los datos de a pares consecutivos y se calcula el área I_i bajo ellos como:

$$I_i = (\lambda_{i+1} - \lambda_i)\frac{f(\lambda_{i+1}) + f(\lambda_i)}{2}$$

Donde λ_i es la i-ésima longitud de onda de los datos y $f(\lambda_i)$ es el valor del espectro para esa longitud de onda. Finalmente la integral I de todo el espectro es:

$$I = \sum_{i=1}^{N-1} I_i$$

Donde N es el número de longitudes de onda disponibles para la integración.

Con este resultado, y notando que $J(r = 1\text{ UA}) = I$, se calcula la luminosidad total del sol como $L_{Sol} = 4\pi \cdot (1\text{ UA})^2 \cdot I$.

2.3. Integral de función de Planck

Como se debe integrar la función de Planck sobre todas las longitudes de onda, la integral va desde 0 a ∞. Definiendo $x = hc/\lambda k_B T$, vemos que:

$$P = \int_0^\infty B_\lambda(T)d\lambda = \int_0^\infty \frac{2\pi hc^2/\lambda^5}{e^{hc/\lambda k_B T} - 1}d\lambda = \frac{2\pi h}{c^2}\left(\frac{k_B T}{h}\right)^4 \int_0^\infty \frac{x^3}{e^x - 1}dx$$

Y la integral que nos interesa calcular es:

$$\int_0^\infty \frac{x^3}{e^x - 1}dx$$

que tiene por resultado analítico $\pi^4/15$.

Para calcular esta integral numéricamente primero se debe normalizar el dominio de integración a un intervalo finito, para poder hacer la discretización. Con este propósito, se hace el cambio de variables $x = tan(y)$, o equivalentemente $y = arctan(x)$. Desarrollando, notamos que con este cambio de variables $dx = sec^2(y)dy$, y la integral se convierte en:

$$\int_0^{\pi/2} \frac{tan^3(y)sec^2(y)}{exp(tan(y)) - 1} dy$$

Notamos que, definiendo:

$$g(y) = \frac{tan^3(y)sec^2(y)}{exp(tan(y)) - 1}$$

se obtiene:

$$\lim_{y \to \pi/2^-} g(y) = 0 \quad \wedge \quad \lim_{y \to 0} g(y) = 0$$

con lo que la integral no diverge en sus límites y no requieren tratamiento especial. Para simplificar la expresión del cálculo, no se escriben estos términos en el programa final, calculando mediante la regla de los trapecios:

$$\int_0^{\pi/2} g(y)dy = \frac{\Delta y}{2} \cdot [g(0) + 2\sum_{i=2}^{n-1} g(y_i) + g(\pi/2)] = \Delta y \sum_{i=2}^{n-1} g(y_i)$$

donde n es el número de valores de y en los que se evalúa $g(\cdot)$ para hacer el cálculo. Se toman valores y_i equiespaciados en el intervalo $[0, \pi/2]$, incluyendo a los extremos al momento de hacer la partición. Es decir, $y_1 = 0$ e $y_n = \pi/2$.

Para refinar el resultado de esta integral en función de alguna tolerancia pedida, se puede calcular la integral con una partición de intervalos más pequeños sobre $[0, \pi/2]$. Para no repetir el trabajo ya hecho, se calculan los valores de $g(y)$ solo en los puntos medios de los intervalos de la partición anterior. Como la suma de los valores anteriores de $g(y_i)$ ya está calculada, basta agregar los nuevos valores y multiplicar por el nuevo paso de la partición Δy (que es la mitad del anterior), obteniendo así como resultado un nuevo valor de la integral, que se espera esté más cerca del valor analítico que la integral anterior.

Después de obtener el valor de la integral con la tolerancia deseada, de forma análoga a la parte anterior (datos medidos desde la Tierra), la luminosidad total del Sol estará dada por $L_{Sol} = 4\pi R^2 P$, pero en este caso $R = R_{Sol}$ es el radio del Sol, cantidad que queremos estimar. Como desde la parte anterior conoceríamos la luminosidad total, se despeja:

$$R_{Sol} = \sqrt{\frac{L_{Sol}}{4\pi P}}$$

2.4. Comparación algoritmos implementados v/s Python scipy

Para comparar y evaluar la velocidad de ejecución de los algoritmos implementados, se usa la funcionalidad **magic** de la consola **ipython**, comparando contra los métodos **scipy.integrate.trapz** para la integral del espectro medido desde la Tierra, y contra **scipy.integrate.quad** para la integral de la función de Planck. **.trapz** recibe como argumentos los arreglos de la función a integrar y el dominio de integración, mientras que **.quad** recibe como argumento la función completa, y por lo tanto puede alcanzar mayor fineza en el cálculo pues no depende de arreglos de datos fijos.

3. Resultados

3.1. Plot del espectro del Sol

La Figura 1 muestra el espectro de radiación del Sol medido desde justo afuera de la atmósfera de la Tierra. Se muestra el comportamiento de la función solamente hasta $3\mu m$ pues desde allí en adelante solo continúa acercándose a cero.

En el Anexo 1 se adjuntan plots del espectro en escala semilogarítmica y en escala logarítmica, donde se aprecia mejor la dependencia del flujo por longitud de onda vs longitud de onda cuando $\lambda \to \infty$.

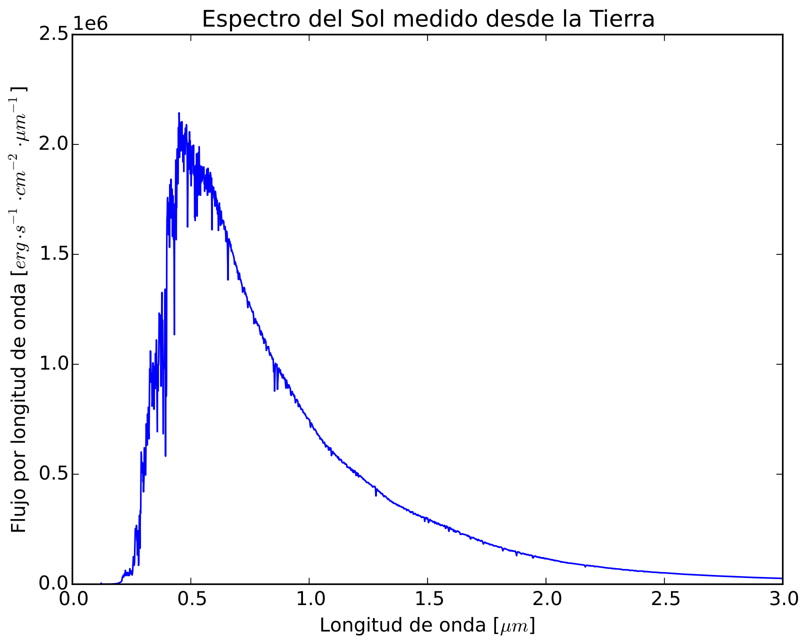

Figura 1: Espectro de longitud de onda del Sol medido desde la Tierra. Las unidades de flujo están en el sistema *cgs* y las de longitud de onda en micrones.

3.2. Cálculo de la luminosidad total del Sol

El resultado de la integración del espectro del Sol y el consiguiente cálculo de la luminosidad total del Sol entregó como resultado:

$$L_{Sol} = 3{,}842 \cdot 10^{26}\,[W]$$

que se compara bastante bien con el valor registrado en **astropy.constants** ($3{,}846 \cdot 10^{26}\,[W]$).

3.3. Integral de función de Planck y radio efectivo del Sol

Se calculó la siguiente integral, de acuerdo al algoritmo antes descrito, con una tolerancia de 10^{-12}:

$$\int_0^\infty \frac{x^3}{e^x - 1} dx = \int_0^{\pi/2} \frac{tan^3(y) sec^2(y)}{exp(tan(y)) - 1} dy$$

Y se obtuvo como resultado el número $\alpha = 6{,}4393940227$, que es igual a $\pi^4/15$ hasta la duodécima cifra significativa. El algoritmo implementado, dada la tolerancia antes definida, garantiza que $|\alpha - \pi^4/15| \leq 10^{-12}$.

Con esto, y usando el valor $T = 5778\,K$ para la temperatura efectiva del Sol (temperatura del cuerpo negro que mejor se ajusta a su espectro) y las constantes necesarias desde **astropy.constants**, se obtiene:

$$P = 63200679{,}7124\,[W/(s \cdot m^2)]$$

Con lo que

$$R_{Sol} = \sqrt{\frac{L_{Sol}}{4\pi P}} = 6{,}955 \cdot 10^8\,[m]$$

que es igual, hasta la cuarta cifra significativa, al valor para el radio del sol almacenado en **astropy.constants**. Esto nos dice que el radio efectivo del Sol como cuerpo negro no es más que su propio radio.

3.4. Comparación algoritmos implementados v/s Python scipy

El resultado de la integral del espectro del Sol, medido desde justo afuera de la atmósfera de la Tierra, usando el método **numpy.trapz**, es ideéntico, con toda la precisión disponible, al antes calculado implementando la regla de los trapecios.

Comparativamente, el método implementado es más lento que **numpy.trapz**. En el computador donde se desarrolló el código, el método implementado toma 2.3 ms para ejecutar, contra 296 μs que demora **numpy.trapz**. Esto es porque en la implementación del código no es óptima: se hace un ciclo for recorriendo cada valor para la longitud de onda y calculando el área entre ese valor y el siguiente, cuando se podría hacer una operatoria más simple: calcular el arreglo de las diferencias de las longitudes de onda y el arreglo de la altura promedio de cada trapecio a sumar. Luego se haría un producto componente a componente entre ambos, y se toma la suma sobre todas las componentes del resultado para obtener el valor de la integral. Esto sería mucho más eficiente que el algoritmo implementado ya que no se haría un ciclo for de forma explícita. También las librerías de **Python** podrían estar haciendo una mejor optimización que esto, aprovechándose de las particularidades del lenguaje.

El resultado de la integral

$$\int_0^\infty \frac{x^3}{e^x-1}dx$$

usando el método **integrate.quad** es igual, para la precisión disponible, al valor calculado usando el algoritmo implementado, y por consiguiente, es igual a $\pi^4/15$.

Comparativamente, el método implementado, pidiendo una tolerancia de 10^{-12} es más rápido que **integrate.quad**. El método implementado, en el computador donde se escribió el código, toma 211 μs, contra 928 μs que demora **integrate.quad**. Probablemente esto es porque **.quad** demora más tiempo manejando las potenciales singularidades que podría tener la función. En cambio, el método implementado asume que en los extremos del intervalo de integración la función toma un valor de cero (lo que es cierto en el límite). Además, que la partición definida al intervalo sea equiespaciada permite evitar las iteraciones, y solo repetir el proceso a la hora de refinar la integral. Esto hace que el método implementado sea bastante eficiente.

4. Conclusiones

Las integrales pedidas fueron calculadas numéricamente de forma satisfactoria, pues se obtuvieron resultados finales muy cercanos a los valores conocidos del radio o la luminosidad del Sol, con diferencias solamente a partir de la cuarta cifra significativa. Se implementó exitosamente la regla de los trapecios en el cálculo de 2 integrales, pues los resultados se comparan bien con los valores esperados y con los resultados obtenidos por otros medios. Una de ellas fue calculada desde datos almacenados, y la otra con los datos generados por una función. Para efectos de las comparaciones, se introdujo el uso de métodos ya implementados en **Python**.

La gráfica del espectro del Sol es suficientemente consistente con lo que se esperaría de un cuerpo negro.

Por lo general, los algoritmos ya implementados en librerías de **Python** son más rápidos para el cálculo de integrales. Sin embargo, bajo ciertas condiciones estos métodos pierden su eficiencia, y una implementación propia puede lograr hacer el cálculo de forma más rápida.

5. Anexos

5.1. Anexo 1: Plots del espectro del Sol en escala semilogarítmica y logarítmica

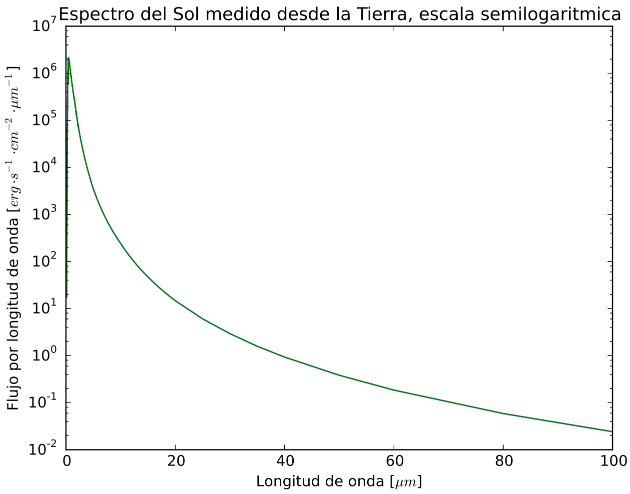

Figura 2: Espectro de longitud de onda del Sol en escala semilogarítmica, medido desde la Tierra

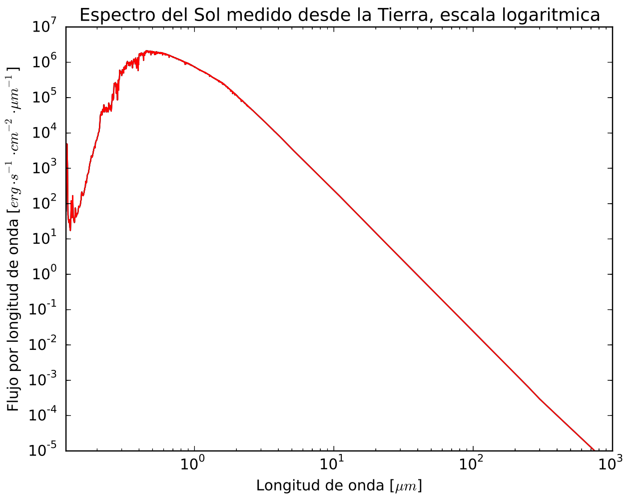

Figura 3: Espectro de longitud de onda del Sol en escala logarítmica, medido desde la Tierra