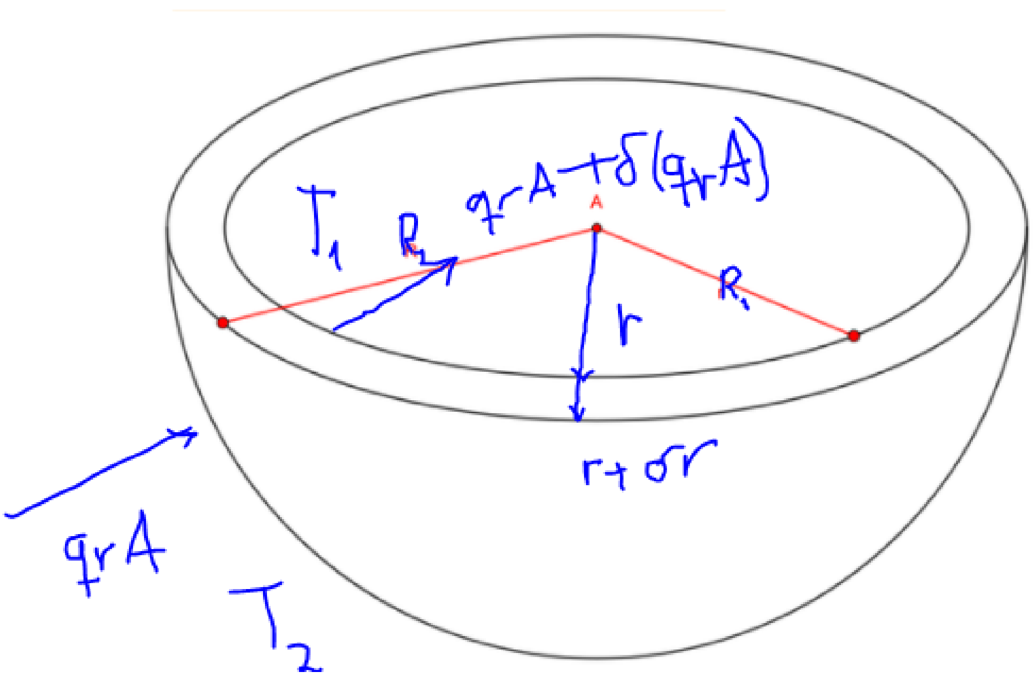
Trabajo Individual 2

June 25, 2021

[1]:
```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
```

1 Modelo Diferencial

1.1 Diagrama



Variables

1. Variable independiente: posición radial r, m.
2. Variable dependiente: temperatura T, K.
3. Variables intermedias:
   a. Flujo de calor: q_r, kW.
   b. Área transversal: A, m^2.
4. Variables fijas:
   a. Temperatura externa: T_2, K.

b. Temperatura interna: T_1, K.

c. Radio exterior: R_2, m.

d. Radio interior: R_1, m.

5. Parámetros:

a. Conductividad térmica: κ, kW/(m K).

1.1.1 Relaciones constitutivas:

$$\kappa = \kappa(T)$$

$$q_r = -\kappa \frac{dT}{dr}$$

Para el modelo se tomará κ constante.

1.1.2 Condiciones del modelo

Se tiene una semiesfera hueca de grosor $(R_2 - R_1)$, para la cual la temperatura en $R_2 = 1.6$ m es $T_2 = 1523.15$ K y en $R_1 = 1.5$ m $T_1 = 423.15$ K. Se está en estado estacionario, por lo que $V_A = 0$.

1.2 Balance de energía:

1. Velocidad de acumulación: $V_A = 0$
2. Flujo de entrada: $F_E = q_r A$
3. Flujo de salida: $F_S = q_r A + \delta(q_r A)$
4. Velocidad de generación: $V_G = 0$
5. Velocidad de consumo: $V_C = 0$

$$0 = q_r A - q_r A - \delta(q_r A)$$
$$\Rightarrow \delta(q_r A) = 0$$

1.2.1 Resolviendo

Se tiene que $dA = r^2 \sin\theta d\phi d\theta$:

$$\delta(q_r r^2 \sin\theta d\phi d\theta) = 0$$
$$\delta(q_r r^2) = 0$$
$$\frac{\delta(q_r r^2)}{\delta r} = 0$$

Ahora $\delta r \to 0$ y:

$$\Rightarrow r^2 \frac{dq_r}{dr} + 2rq_r = 0$$

Sustituyendo:

$$\Rightarrow r^2 \frac{d}{dr}\left(-\kappa \frac{dT}{dr}\right) + 2r\left(-\kappa \frac{dT}{dr}\right) = 0$$

$$\Rightarrow -\kappa r^2 \frac{d^2T}{dr^2} - 2\kappa r \frac{dT}{dr} = 0$$

Dividiendo entre $-\kappa r^2$:

$$\frac{d^2T}{dr^2} + \frac{2}{r}\frac{dT}{dr} = 0$$

2 Solucionando la ecuación diferencial

Se busca usar factor integrante:

$$F_I = \exp\left(\int_0^r \frac{2}{r'} dr'\right) = r^2$$

Esto permite reescribir la ecuación como:

$$\frac{d}{dr}\left(r^2 \frac{dT}{dr}\right) = 0$$

Integrando:

$$r^2 \frac{dT}{dr} = C_1$$

$$\Rightarrow \frac{dT}{dr} = \frac{C_1}{r^2}$$

$$\Rightarrow \int dT = \int \frac{C_1}{r^2} dr$$

$$T(r) = -\frac{C_1}{r} + C_2$$

$T(R_1) = T_1$, $T(R_2) = T_2$:

$$T_1 = \frac{-C_1}{R_1} + C_2$$

$$T_2 = \frac{-C_1}{R_2} + C_2$$

$$C_2 = T_2 + \frac{C_1}{R_2}$$

$$T_1 = \frac{-C_1}{R_1} + T_2 + \frac{C_1}{R_2}$$

$$T_2 - T_1 = \left(\frac{1}{R_1} - \frac{1}{R_2}\right) C_1$$

$$C_1 = (T_2 - T_1)\left(\frac{1}{R_1} - \frac{1}{R_2}\right)^{-1}$$

$$C_2 = T_2 + \frac{1}{R_2}(T_2 - T_1)\left(\frac{1}{R_1} - \frac{1}{R_2}\right)^{-1}$$

Por tanto, la expresión para la distribución radial de la temperatura queda como:

$$T(r) = T_2 + \frac{1}{R_2}(T_2 - T_1)\left(\frac{1}{R_1} - \frac{1}{R_2}\right)^{-1} - (T_2 - T_1)\left(\frac{1}{R_1} - \frac{1}{R_2}\right)^{-1}\frac{1}{r}$$

Reacomodando:

$$T(r) = T_2 + \frac{T_2 - T_1}{r\left(\frac{1}{R_2} - \frac{1}{R_1}\right)} - \frac{T_2 - T_1}{R_2\left(\frac{1}{R_2} - \frac{1}{R_1}\right)}$$

Sustituyendo con las condiciones de trabajo:

$$T(r) = 1523.15 + \frac{1100}{\frac{-r}{24}} - \frac{1100}{1.6 \cdot \frac{-1}{24}}$$

$$T(r) = 1523.15 - \frac{26400}{r} + 16500$$

$$T(r) = 18023.15 - \frac{26400}{r}$$

2.1 Comprobación del modelo

2.1.1 Dimensionalidad

Se tiene que $T(r)$ debe tener unidades de Kelvin, el primer término T_2 cumple con esto, el segundo término, tiene unidades de la forma $K/(m \cdot 1/m)$, de forma que también tiene unidades de Kelvin, y el último término también presenta estas unidades.

$$[T(r)] = K + \frac{K - K}{m \cdot \frac{1}{m}} - \frac{K - K}{m \cdot \frac{1}{m}}$$

$$\Rightarrow [T(r)] = K$$

2.1.2 Condición de contorno

Al evaluar el modelo en $R_1 = 1.5$, se obtiene como resultado $T_1 = 423.15$, lo que permite afirmar que el modelo cumple con la condición de contorno.

$$T(1.5) = 18023.15 - \frac{26400}{1.5}$$

$$\Rightarrow T(1.5) = 423.15$$

[2]:
```
T1=150+273.15 # K
T2=1250+273.15 # K
R1=1.5 # m
R2=1.6 # m
k=0.015 # kW/m.K
h = 0.5e-02 # Step size
r = np.arange(R2,R1,-h) # Numerical grid
```

3 Solución de la ecuación diferencial con métodos numéricos

Siguiendo la sugerencia, se busca resolver para dT/dr, entonces:

$$\frac{dq_r}{dr} = -\frac{2}{r}q_r$$

Con $q_r = -\kappa dT/dr$.

Se tomará un paso $h = 0.5 \times 10^{-2}$ m y el radio irá de 1.6 a 1.5, con $q_r(1.6) = 175 \times 10^3$ W/m^2

[3]:
```
# Define parameters
f = lambda r, q: -2*q/r# ODE
q0 = -175 # Initial Condition, kW/m^2.K
```

3.0.1 Método de Euler

Muestra de cálculo Con:

$$f(r, q_r) = \frac{-2q_r}{r}$$

Partiendo de $R_2 = 1.6$, con $q_0 = q_r = -175$ kW/m^2.K, entonces:

$$q_{i+1} = q_i + h \cdot \frac{-2 \cdot q_i}{r_i}$$

$$T_{i+1} = T_i - h \cdot \frac{q_i}{-\kappa}$$

$$T_1 = T_0 - 0.5 \times 10^{-2} \cdot \frac{q_0}{-\kappa} = 1523.15 - 0.5 \times 10^{-2} \cdot \frac{-175}{-0.015} = 1464.8167 \text{ K}$$

$$q_1 = -175 + 0.5 \times 10^{-2} \cdot \frac{-2 \cdot -175}{1.6} = -173.9 \text{ kW/m}$$

$$T_2 = 1464.8167 - 0.5 \times 10^{-2} \cdot \frac{-173.9}{-0.015} = 1406.8479 \text{ K}$$

$$q_2 = -173.9 + 0.5 \times 10^{-2} \cdot \frac{-2 \cdot -173.9}{1.6} = -172.8 \text{ kW/m}$$

Una vez que se calcularon todos los q_i, se divide entre $-\kappa$, para obtener el valor de dT/dr, entonces:

$$\frac{dT}{dr}_1 = \frac{q_1}{-\kappa} = \frac{-173.9}{-0.015} = 11593.75 \text{ K/m}$$

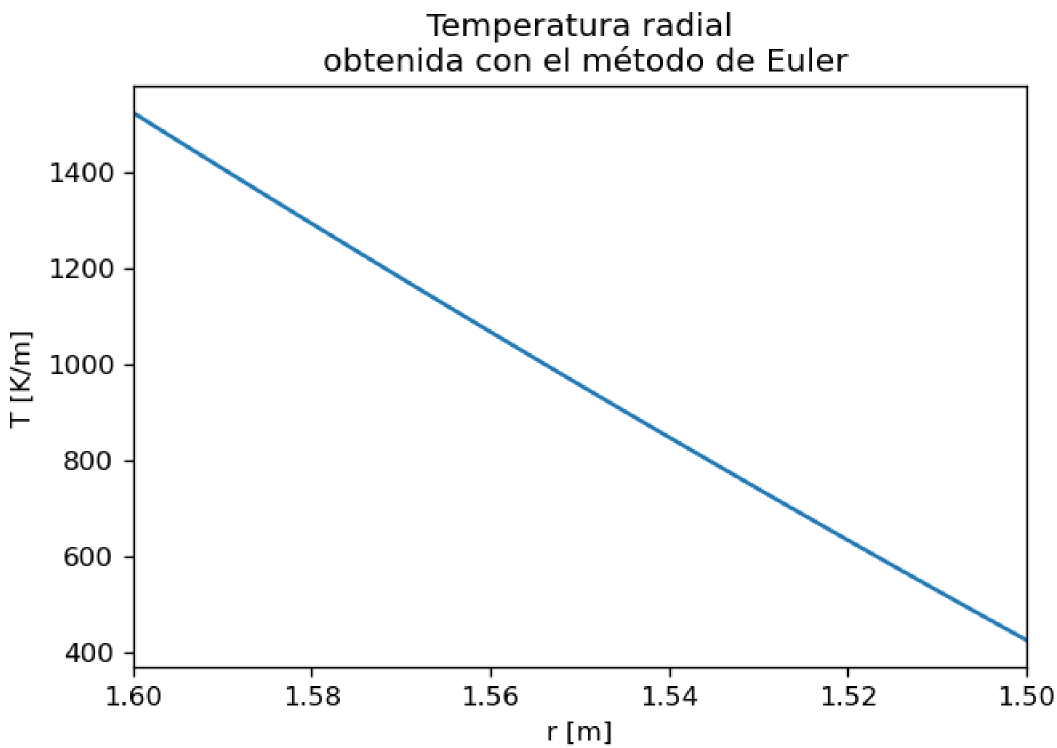
$$\frac{dT}{dr}_2 = \frac{q_2}{-\kappa} = \frac{-172.8}{-0.015} = 11521.29 \text{ K/m}$$

[4]:
```
def Euler(f):
    T=[T2]
    q=[q0]
    for j in range(len(r)-1):
        #Euler's method (explicit)
        T.append(T[j]-q[j]*h/-k)
        q.append(q[j]+f(r[j],q[j])*h)
    return np.array(q)/-k,np.array(T)
```

[5]:
```
_,TEuler=Euler(f)
tablaEuler=np.zeros((len(r),2))
tablaEuler[:,0]=r
tablaEuler[:,1]=TEuler
print('Las soluciones para la temperatura como función de la posición radial␣
 ↪con Euler, de 1.6 a 1.5 m,  son:\n          r,            T\n',tablaEuler)
```

```
Las soluciones para la temperatura como función de la posición radial con Euler,
de 1.6 a 1.5 m,  son:
          r,            T
 [[1.60000000e+00 1.52315000e+03]
 [1.59500000e+00 1.46481667e+03]
 [1.59000000e+00 1.40684792e+03]
 [1.58500000e+00 1.34924261e+03]
 [1.58000000e+00 1.29199960e+03]
 [1.57500000e+00 1.23511774e+03]
 [1.57000000e+00 1.17859589e+03]
 [1.56500000e+00 1.12243292e+03]
 [1.56000000e+00 1.06662766e+03]
 [1.55500000e+00 1.01117900e+03]
 [1.55000000e+00 9.56085769e+02]
 [1.54500000e+00 9.01346839e+02]
 [1.54000000e+00 8.46961063e+02]
 [1.53500000e+00 7.92927299e+02]
 [1.53000000e+00 7.39244403e+02]
 [1.52500000e+00 6.85911233e+02]
 [1.52000000e+00 6.32926646e+02]
 [1.51500000e+00 5.80289498e+02]
 [1.51000000e+00 5.27998648e+02]
 [1.50500000e+00 4.76052952e+02]
 [1.50000000e+00 4.24451267e+02]]
```

[6]:
```
plt.figure(dpi=120)
plt.plot(r,TEuler)
plt.xlim(R2,R1)
plt.xlabel('r [m]')
plt.ylabel('T [K/m]')
plt.title('Temperatura radial\n obtenida con el método de Euler')
plt.show()
```



3.0.2 Runge-Kutta 4

Muestra de Cálculo Con:

$$f(r, q_r) = \frac{-2q_r}{r}$$

Partiendo de $R_2 = 1.6$, con $q_0 = q_r = -175$ kW/m^2.K. Para cada paso de 0.5 cm, se calcula la solución de q como:

$$q_{i+1} = q_i + \frac{k_{1,i} + 2k_{2,i} + 2k_{3,i} + k_{4,i}}{6}$$
$$T_{i+1} = T_i - \frac{m_{1,i} + 2m_{2,i} + 2m_{3,i} + m_{4,i}}{6}$$

Donde:

$$m_{1,i} = 0.5 \times 10^{-2} \cdot \frac{q_i}{-\kappa}$$

$$k_{1,i} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot q_i}{r_i}$$

$$m_{2,i} = 0.5 \times 10^{-2} \cdot \left(\frac{q_i}{-\kappa} + \frac{k_{1,i}}{2} \right)$$

$$k_{2,i} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot (q_i + k_{1,i}/2)}{r_i + h/2}$$

$$m_{3,i} = 0.5 \times 10^{-2} \cdot \left(\frac{q_i}{-\kappa} + \frac{k_{2,i}}{2} \right)$$

$$k_{3,i} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot (q_i + k_{2,i}/2)}{r_i + h/2}$$

$$m_{1,i} = 0.5 \times 10^{-2} \cdot \left(\frac{q_i}{-\kappa} + k_3 \right)$$

$$k_{4,i} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot (q_i + k_{3,i})}{r_i + h}$$

Partiendo entonces de $q_0 = -175$:

$$m_{1,0} = 0.5 \times 10^{-2} \cdot \frac{-175}{-0.015} = 58.333$$

$$k_{1,0} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot q_0}{r_0} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot -175}{1.6} = 1.09375$$

$$m_{2,0} = 0.5 \times 10^{-2} \cdot \left(\frac{-175}{-0.015} + \frac{1.09375}{2} \right) = 58.336$$

$$k_{2,0} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot (q_0 + k_{1,0}/2)}{r_0 + h/2} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot (-175 + 1.09375)}{1.6 + 0.5 \times 10^{-2}/2} = 1.08863$$

$$m_{3,0} = 0.5 \times 10^{-2} \cdot \left(\frac{-175}{-0.015} + \frac{1.08863}{2} \right) = 58.336$$

$$k_{3,0} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot (q_0 + k_{2,0}/2)}{r_0 + h/2} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot (-175 + 1.0852)}{1.6 + 0.5 \times 10^{-2}/2} = 1.08865$$

$$m_{4,0} = 0.5 \times 10^{-2} \cdot \left(\frac{-175}{-0.015} + \frac{1.08865}{2} \right) = 58.336$$

$$k_{4,0} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot (q_0 + k_{3,0})}{r_0 + h} = 0.5 \times 10^{-2} \cdot \frac{-2 \cdot (-175 + 1.08865)}{1.6 + 0.5 \times 10^{-2}} = 1.08356$$

$$T_1 = 1523.15 - \frac{58.333 + 2 \cdot 58.336 + 2 \cdot 58.336 + 58.336}{6} = 1464.81394081$$

$$q_1 = -175 + \frac{1.09375 + 2 \cdot 1.08863 + 2 \cdot 1.08865 + 1.08356}{6} = -173.91$$

$$\frac{dT}{dr}_1 = \frac{q_1}{-\kappa} = 11594.1$$

[7]:
```
def RK4(f):
    '''
    Esta función corresponde a una función que soluciona una EDO por RK4␣
 ↪iterando sobre un y = np.arange(y0,yf,h)

    Parámetros
    ------------------------
    f: función al lado derecho de la EDO
    ------------------------
    np.array(q): arreglo con los valores de p asociados a los valores de r.
    '''
    q=[q0]
    T=[T2]
    for i in range(len(r)-1):
        m1=h*q[i]/-k
        k1=h*f(r[i],q[i])
        m2=h*(q[i]/-k+k1/2)
        k2 = h*f(r[i]+h/2,q[i]+k1/2)
        m3=h*(q[i]/-k+k2/2)
        k3=h*f(r[i]+h/2,q[i]+k2/2)
        m4=h*(q[i]/-k+k3)
        k4=h*f(r[i]+h,q[i]+k3)
        q_r=q[i]+(k1+2*k2+2*k3+k4)/6
        T_r=T[i]-(m1+2*m2+2*m3+m4)/6
        q.append(q_r)
        T.append(T_r)
    return np.array(q)/-k,np.array(T)
```

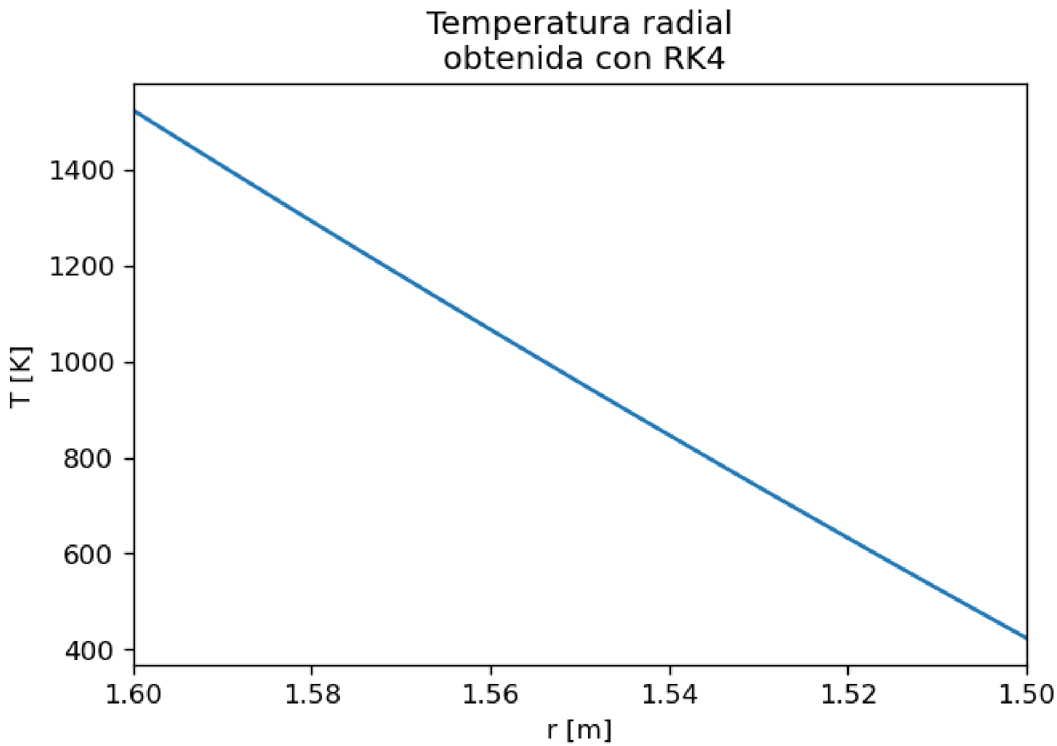
[8]:
```
_,TRK4=RK4(f)
tablaRK4=np.zeros((len(r),2))
tablaRK4[:,0]=r
tablaRK4[:,1]=TRK4
```

```
print('Las soluciones para la temperatura como función de la posición radial␣
↪con Runge-Kutta 4, de 1.6 a 1.5 m, son:\n          r,           ␣
↪T\n',tablaRK4)
```

```
Las soluciones para la temperatura como función de la posición radial con Runge-
Kutta 4, de 1.6 a 1.5 m, son:
          r,            T
 [[1.60000000e+00 1.52315000e+03]
 [1.59500000e+00 1.46481394e+03]
 [1.59000000e+00 1.40684077e+03]
 [1.58500000e+00 1.34922936e+03]
 [1.58000000e+00 1.29197857e+03]
 [1.57500000e+00 1.23508728e+03]
 [1.57000000e+00 1.17855435e+03]
 [1.56500000e+00 1.12237865e+03]
 [1.56000000e+00 1.06655904e+03]
 [1.55500000e+00 1.01109440e+03]
 [1.55000000e+00 9.55983587e+02]
 [1.54500000e+00 9.01225476e+02]
 [1.54000000e+00 8.46818934e+02]
 [1.53500000e+00 7.92762826e+02]
 [1.53000000e+00 7.39056022e+02]
 [1.52500000e+00 6.85697388e+02]
 [1.52000000e+00 6.32685793e+02]
 [1.51500000e+00 5.80020105e+02]
 [1.51000000e+00 5.27699190e+02]
 [1.50500000e+00 4.75721918e+02]
 [1.50000000e+00 4.24087155e+02]]
```

[9]:

```
plt.figure(dpi=120)
plt.plot(r,TRK4)
plt.xlim(R2,R1)
plt.xlabel('r [m]')
plt.ylabel('T [K]')
plt.title('Temperatura radial\n obtenida con RK4')
plt.show()
```



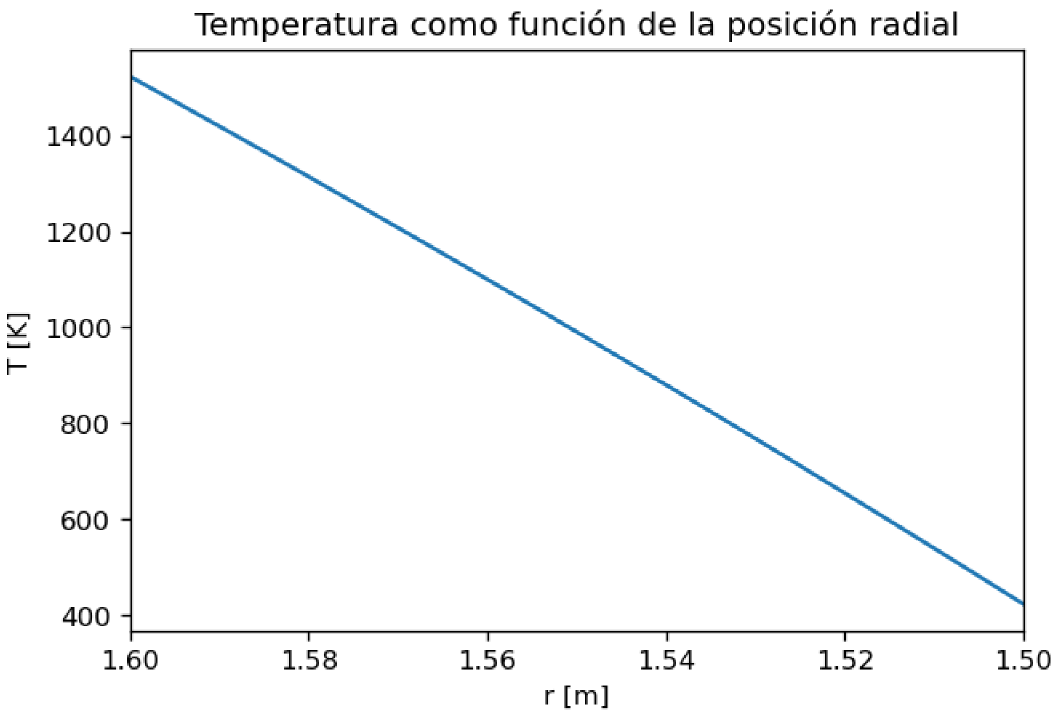
Como se conocen los valores obtenidos para dT/dr, se puede integrar para conocer el cambio en la temperatura ΔT

4 Temperatura entre 1.6 m y 1.5 m

Se toma el modelo analítico y se evalúa entre 1.6 y 1.5 m.

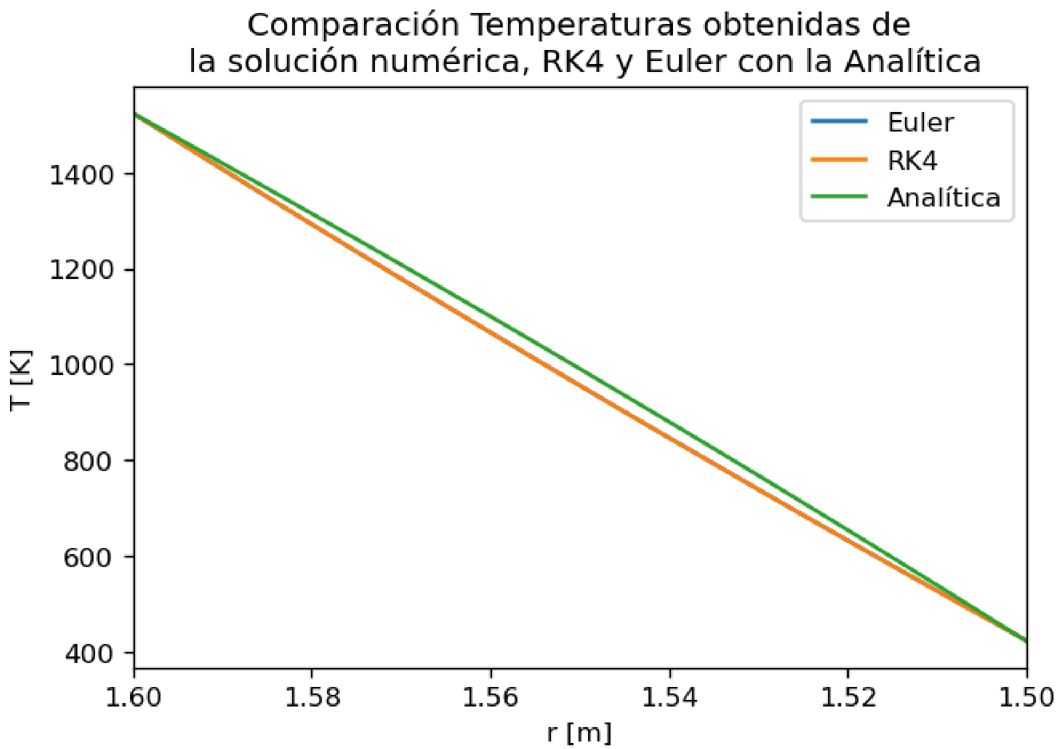
[10]:
```
def T(r):
    return T2+(T2-T1)/(r*(1/R2-1/R1))-(T2-T1)/(R2*(1/R2-1/R1))
```

[11]:
```
plt.figure(dpi=120)
plt.plot(r,T(r))
plt.xlim(R2,R1)
plt.title('Temperatura como función de la posición radial')
plt.xlabel('r [m]')
plt.ylabel('T [K]')
plt.show()
```

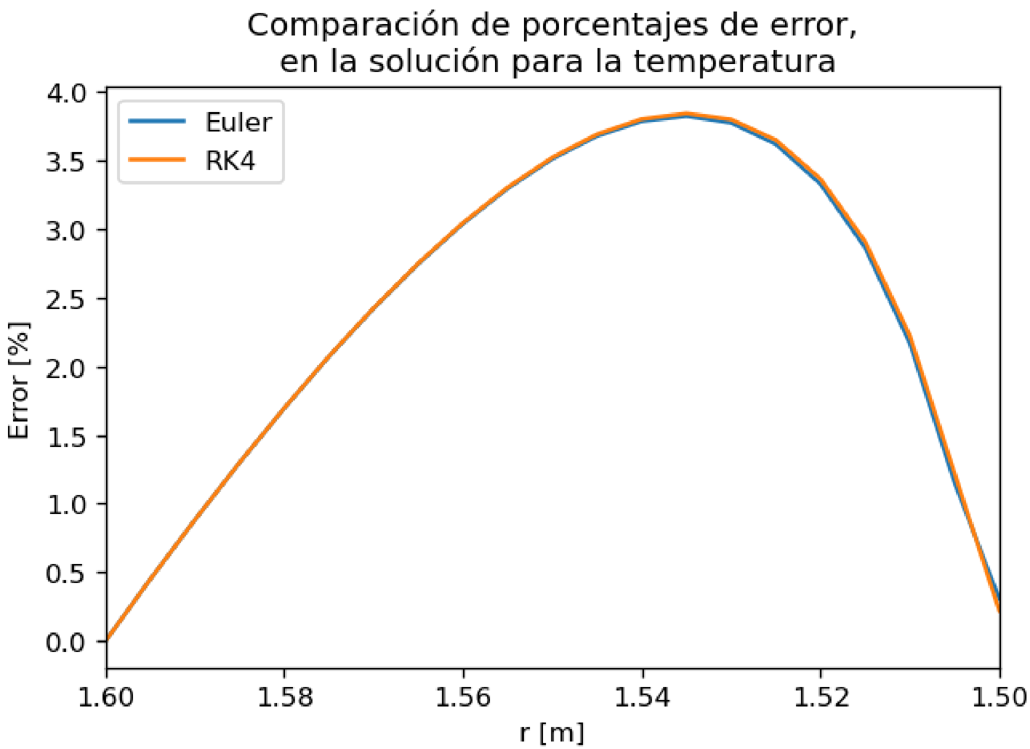


5 Comparación de las soluciones analítica y numéricas

[12]:
```
plt.figure(dpi=120)
plt.plot(r,TEuler)
plt.plot(r,TRK4)
plt.plot(r,T(r))
plt.xlim(R2,R1)
plt.legend(['Euler','RK4','Analítica'])
plt.xlabel('r [m]')
plt.ylabel('T [K]')
plt.title('Comparación Temperaturas obtenidas de\n la solución numérica, RK4 y␣
↪Euler con la Analítica')
plt.show()
```



[13]:
```
errorEuler=np.abs(TEuler-T(r))/T(r)*100
errorRK4=np.abs(TRK4-T(r))/T(r)*100
plt.figure(dpi=120)
plt.plot(r,errorEuler)
plt.plot(r,errorRK4)
plt.xlim(R2,R1)
plt.legend(['Euler','RK4'])
plt.ylabel('Error [%]')
plt.xlabel('r [m]')
plt.title('Comparación de porcentajes de error,\n en la solución para la␣
↪temperatura')
plt.show()
```



Observando la gráfica del porcentaje de error entre 1.6 y 1.5, para la Temperatura, aunque los métodos están cercanos entre sí, parece que no hay diferencia entre ninguno de los dos métodos, tal vez Runge-Kutta de un grado superior podría dar mejores resultados. Parecen ser lo suficientemente buenos, pues se acercan bastante a la anática, pero para posiciones más cercanas a los bordes, mientras que hacia el centro se alejan hasta un 4%