Tarea 1

Gustavo Hernández Angeles[1]

[1] *Universidad Autonoma de Nuevo Leon*

Publication date: 10/02/2023

Abstract— En esta tarea se utilizaron los distintos métodos para derivación numérica con el fin de encontrar soluciones a problemas físicos reales, además de programar una expresión para la n-ésima derivada y para la serie binomial. Al final se encuentran los códigos utilizados en la realización de la tarea.

Keywords— Fortran, Derivación numérica

I. Problemas de Física

a. Molécula diatómica

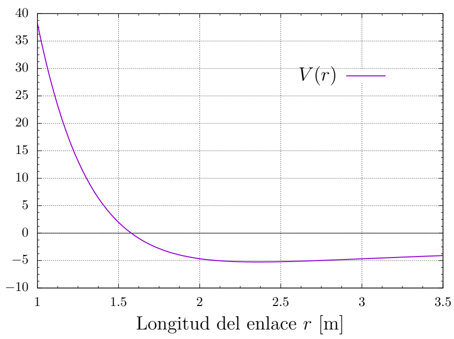
Encontrar la posición de equilibrio de los átomos dentro de una molécula, implica conocer los mínimos de una función. Para encontrar la longitud de enlace de una molécula diatómica NaCl es necesario conocer el potencial de interacción entre ambos iones (Na+ y Cl-). Asumiendo que el potencial de interacción es $V(r)$ cuando los dos iones que están separados una distancia r, la longitud de enlace en el equilibrio ocurre cuando se tiene el mínimo de $V(r)$. El potencial $V(r)$ está dado como:

$$V(r) = -\frac{e^2}{4\pi\varepsilon_0 r} + V_o \exp(-r/r_0) \tag{1}$$

Siendo e la carga del protón, ε_0 la permitividad eléctrica en el vacío y V_o, r_o los parámetros de la interacción efectiva. Asumiendo los valores $V_o = 1.09 \times 10^3$ eV, $r_o = 0.33$Å. Además, aproximando $e^2/4\pi\varepsilon_0 = 14.4$ÅeV. Encuentre la distancia donde la fuerza es cero y por lo tanto, la distancia de equilibrio Na+ - Cl-. Esto es:

$$F(r) = -\frac{\mathrm{d}V(r)}{\mathrm{d}r} = 0 \tag{2}$$

1. Gráfica del potencial $V(r)$



2. Solución numérica

Con ayuda de la gráfica podemos ver un intervalo en donde se encuentra el mínimo del potencial de interacción y por tanto, el punto de equilibrio. Podemos proponer el intervalo [1.5,3.0]. Si introducimos esta información en el programa para calcular máximos y mínimos resulta en $r_{enlace} \simeq 2.3605$Å.

```
PS C:\Users\Equipo\Desktop\Gus_FCFM\8vo_semestre\Computacional
\Tareas\T2> .\a.exe
 Dame los extremos del intervalo que contienen el maximo o min
imo:
1.5,3.0
 Se encontro un minimo en x =    2.3605386179649921
```

Fig. 1: Solución numérica a $V'(r) = 0$.

b. Función de distribución radial

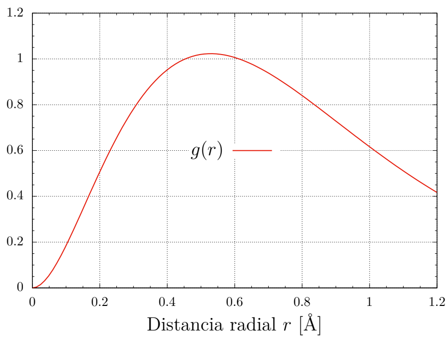
La función de distribución radial que da la probabilidad de encontrar un electrón dentro del átomo de hidrógeno está dada por la siguiente función:

$$g(r) = \frac{4}{a_0^3} r^2 e^{-2r/a_0} \tag{3}$$

¿Cuál es la distancia donde es mayor la probabilidad de encontrar al electrón?

1. Gráfica de la función $g(r)$

Para graficar la función, debemos tomar en cuenta que en el átomo de hidrógeno, a_0 es igual al radio de Bohr.



2. *Solución numérica*

De la gráfica podemos proponer un intervalo de [0.2,0.8] para el máximo, introduciendo esta información en el programa de máximos y mínimos nos resulta en que el resultado es $r \simeq a_0 = 0.529167$Å.

```
 Dame los extremos del intervalo que contienen el maximo o min
imo:
0.2,0.8
 Se encontro un maximo en x =    0.528999984458507735
```

Fig. 2: Solución numérica a $g'(r) = 0$.

3. *Solución analítica*

$$g(r) = \frac{4}{a_0^3} r^2 e^{-2r/a_0}$$

Derivando:

$$g'(r) = \frac{8}{a_0^3} r e^{-2r/a_0} - \frac{8}{a_0^4} r^2 e^{-2r/a_0}$$

$$g'(r) = \frac{8}{a_0^3} r e^{-2r/a_0} \left[1 - \frac{r}{a_0} \right] = 0$$

Obtenemos 2 soluciones:

$r = 0$; $r = a_0$

Físicamente, la única admisible es:

$$\boxed{r = a_0}$$

c. *Ley de desplazamiento de Wien*

La ley de desplazamiento de Wien se obtiene derivando la fórmula de radiación de Planck con respecto a la longitud de onda para el cuerpo negro. Al igualar la derivada a cero se obtiene la siguiente relación:

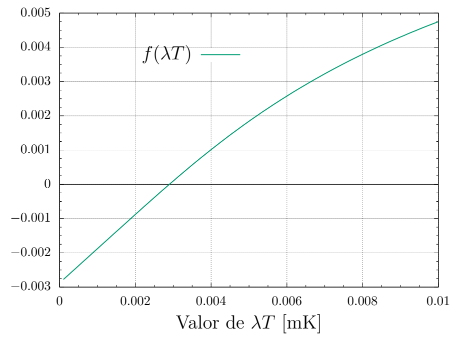
$$\lambda T = \frac{a}{5(1 - e^{-a/\lambda T})} \tag{4}$$

Encontrar el valor de λT.

1. *Gráfica*

De la literatura podemos determinar que $a = 1.4385 \times 10^{-2}$ mK. Utilizando esta información, graficamos:

$$f(\lambda T) = \lambda T(1 - \exp(-a/(\lambda T)) - a/5 \tag{5}$$



2. *Solución numérica*

De la gráfica vemos que un buen intervalo de la raíz es [0.002,0.004], introduciendo esta información al método, nos da un resultado aproximado al de la literatura: $\lambda_{máx} T = 2897.6\mu$mK.

```
 Dame los extremos del intervaloque contienen la raiz:
0.002,0.004
   4       0.00289721       0.00000000
```

Fig. 3: Solución numérica a $f(\lambda T) = 0$.

d. *Valor de la incógnita*

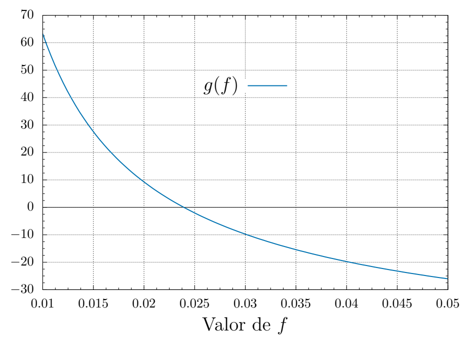
Encuentre el valor de f de la siguiente ecuación, con $n = 0.4$ y $Re = 6000$.

$$\frac{1}{f} = \frac{4}{n^{0.75}} \ln\left(Re \cdot f^{1-0.5n}\right) - \frac{0.4}{n^{1.2}} \tag{6}$$

1. *Gráfica*

Realizamos una función auxiliar $g(f)$ tal que sus raíces sean la solución a nuestro problema, en este caso:

$$g(f) = \frac{1}{f} - \frac{4}{n^{0.75}} \ln\left(Re \cdot f^{1-0.5n}\right) + \frac{0.4}{n^{1.2}} \tag{7}$$



2. *Solución numérica*

De la gráfica podemos observar que un buen intervalo de la raíz sería [0.02,0.03], el programa del método de secante nos arroja el resultado $f \simeq 0.023932$.

```
 Dame los extremos del intervaloque contienen la raiz:
0.02,0.03
   5       0.02393231       0.00000000
```

Fig. 4: Solución numérica a $g(f) = 0$.

II. Programación en FORTRAN

a. N-ésima derivada

Deducir la expresión de la enésima derivada, utilizando diferencias centradas. Hacer el programa respectivo en FORTRAN.

1. Deducción

Problema 1

Deducción de la fórmula de la n-derivada.

Sea $f(x)$ de clase C^{n+1}. Entonces:

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h} \qquad (1)$$

- Para la segunda derivada:

$$f''(x) = \frac{f'(x+h) - f'(x-h)}{2h} \qquad (2)$$

De 1 en 2:

$$f''(x) = \frac{(f(x+2h) - f(x))/2h - (f(x) - f(x-2h))/2h}{2h}$$

$$f''(x) = \frac{f(x+2h) - 2f(x) + f(x-2h)}{4h^2} \qquad 3$$

- Para la tercera derivada:

$$f^3(x) = \left(f^2(x+h) - f^2(x-h)\right)/2h \qquad 4$$

De 3 en 4

$$f^3(x) = \left[(f(x+3h) - 2f(x+h) + f(x-h))/4h^2 - (f(x+h) - 2f(x-h) + f(x-3h))/4h^2\right]/2h$$

$$\Rightarrow f^3(x) = \frac{f(x+3h) - 3f(x+h) + 3f(x-h) - f(x-3h)}{8h^3} \qquad 5$$

- Para la 4ta derivada:

$$f^4(x) = \frac{f^3(x+h) - f^3(x-h)}{2h} \quad (6)$$

De (5) en (6):

$$f^4(x) = [(f(x+4h) - 3f(x+2h) + 3f(x) - f(x-2h))/8h^3 - (f(x+2h) - 3f(x) + 3f(x-2h) - f(x-4h))/8h^3]/2h$$

$$\Rightarrow f^4(x) = \frac{f(x+4h) - 4f(x+2h) + 6f(x) - 4f(x-2h) + f(x-4h)}{16h^4}$$

Como podemos observar en las 4 derivadas existe un patrón:

- El denominador es de la forma $2^n h^n = (2h)^n$, siendo n el orden de la derivada
- Los coeficientes de los términos $f(x+kh)$ alternan de signo.
- Si n es par, solo hay términos $f(x+kh)$ con k par y además $k \leq n$.
- Lo mismo con n impar
- Finalmente, los coeficientes pertenecen a una serie binomial, es decir, son las combinaciones $\binom{n}{k}$.

Por lo tanto, podemos proponer:

$$f^{(n)}(x) = \frac{1}{(2h)^n} \sum_{k=0}^{n} \binom{n}{k} (-1)^k f(x+(n-2k)h)$$

2. Programa en FORTRAN

```fortran
program ndercentral
    implicit none
    integer, parameter :: qp = selected_real_kind(33, 4931)
    real(qp) :: x, nder
    integer :: n !, i
    print*,'Orden de la derivada: ';read*,n
    print*,'Punto a evaluar: ';read*,x
    write(*,'(a,i2,a,f5.2,a,f50.33)')'La derivada de orden ',n,' en el punto ',x,' es igual a: ',nder(x,n)
end

function f(x)
    implicit none
    integer, parameter :: qp = selected_real_kind(33, 4931)
    real(qp) :: x, f
    f = exp(x)
    return
end

recursive function factorial(n) result(fact)
    implicit none
   integer :: N,fact
   if (n/=1.and.n/=0) then
   fact = factorial(n-1)*N
   return
   else
   fact = 1
    end if
   return
end

integer function combination(n,k)
    implicit none
    integer :: n,k,factorial
    combination = factorial(n)/(factorial(n-k)*factorial(k))
    return
end

function nder(x,n)
    implicit none
    integer :: n, k, combination
    integer, parameter :: qp = selected_real_kind(33, 4931)
    real(qp) :: f, x, sum = 0, nder
    real(qp), parameter :: h=1.0e-1
    do 1 k=0,n
        sum = sum + (-1)**(k) * combination(n,k) * f(x+(n-2*k)*h)
    1 continue
    nder = sum / ((2*h)**n)
    return
end
```

3. Resultado

Para analizar si la derivada funcionaba, utilicé la función $f(x) = e^x$ evaluando en el punto $x = 1$ donde la derivada n-ésima evaluada en ese punto será siempre $f^n(1) = e$. Como hipotesis, tengo que el orden hasta donde sea efectiva la derivada depende del valor de h, ya que solo es posible almacenar hasta 33 decimales. Podemos lograr una buena aproximación de la derivada hasta el orden 12 utilizando un $h = 0.1$.

```
 Orden de la derivada:
12
 Punto a evaluar:
1
La derivada de orden 12 en el punto  1.00 es igual a:
       2.773176289442187888496330292251408
```

b. Serie binomial

1. Programa en FORTRAN

```
program t1_2
  implicit none
  integer :: x , combi1
  integer*4 :: combi2
  real :: y
  print*, 'Valor de n : '
  read*, y
  if (modulo(y,1.0) .eq. 0) then
    select case (int(y))
      case (1 : )
        print*, 'Entero positivo'
        do x = 0 , int(y)
          print*, 'Combinacion', x, combi1(int(y), x)
        end do
      case ( : -1)
        print*, 'Entero negativo'
        do x = 0 , int(-y)
          print*, 'Combinacion', x,  combi2(x, int(-y))
        end do
      end select
  else
    select case (int(y))
      case (1 : )
        print*, 'Real positivo'
        call combi3(y)
      case ( : -1)
        print*, 'Real negativo'
        call combi3(y)
    end select
  end if
end program t1_2


function factorial (n)    ! no mas grande que 12
  implicit none
  integer, intent (in) :: n
  integer :: factorial
  integer :: j
  factorial = product ((/(j, j = 1, n)/))
  return
end function factorial

function combi1 (n, k) result (res)
  implicit none
  integer, intent (in) :: n
  integer, intent (in) :: k
  integer :: res, factorial
  res = factorial (n) / (factorial (k) * factorial (n - k))
end function combi1

integer*4 function combi2(r, n)
  implicit none
  integer*4, intent(in):: r, n
  integer*4 i, j/1/
  do i = max(n-r, r) + 1, n
    j = j * i
  enddo
  do i = min(n - r, r), 2, -1
    j = j / i
  end do
  combi2 = j
  return
end function combi2


subroutine combi3(r)
  real, intent(in) :: r
  real :: nume
  integer :: limsup, n, factorial
  print*, 'Orden de la expansion (0,1,...,inf): '
  read*,  limsup
  n=0
```

```
  do while (n.le.limsup)
    nume=1.0
    if (n==0) then
      combireal = 1
    else
      a = n-r+1
      do while (a.le.r)
        nume = nume*a
        a = a + 1
      enddo
      combireal = nume/(1.0*factorial(n))
    end if
    write(*,*)'Combinacion', n , combireal
    n = n+1
  enddo
  return
end
```

III. Conclusión

La diferenciación numérica tiene distintos métodos y varias teorías que logran hacer una aproximación numérica a las derivadas de las funciones en puntos dados. La eficiencia de los distintos métodos suelen recaer en la forma de la función y otros aspectos intrínsecos a esta. Sin embargo, la mayoría de aproximaciones coinciden en que el acercamiento resulta más eficiente mediante una **diferenciación central**. Con estos acercamientos, pudimos resolver problemas de distintas ramas de la física. Durante la realización de esta tarea aprendí nuevos detalles de FORTRAN como la recursividad en funciones, el empleo de las subrutinas y el poder aplicar la cuádruple precisión para las variables reales. En GNUPLOT, aprendí que se pueden almacenar funciones y valores de parámetros lo que representa mayor facilidad en la realización de una gráfica, además de mover la leyenda de cada gráfica.

IV. Códigos empleados para los problemas

a. Secante para raíces de funciones

```
PROGRAM SECANTE
    DL = 1.0E-08
    WRITE (*,*)'Dame los extremos del intervaloque contienen la raiz:'
    READ(*,*)A,B
    DX = (B-A)/10.
    X0 = (A+B)/2.0
    CALL SECANT (DL,X0,DX,ISTEP)
    WRITE (6,999) ISTEP,X0,DX
    STOP
999 FORMAT (I4,2F16.8)
END

SUBROUTINE SECANT (DL,X0,DX,ISTEP)
    ISTEP = 0
    X1 = X0 + DX
    DO   100  WHILE (ABS(DX).GT.DL)
      D  = F(X1) - F(X0)
      X2 = X1 - F(X1)*(X1-X0)/D
      X0 = X1
      X1 = X2
      DX = X1 - X0
      ISTEP = ISTEP + 1
100 END DO
    RETURN
END

FUNCTION F(X)
    real*4, parameter :: n = 0.44, re = 6000
    F = (1/x)-(4*log(re*x**(1-0.5*n)))/(n**0.75)+0.4/(n**1.2)
    RETURN
END
```

b. Máximos y mínimos

```fortran
program maxmin
    real*8 :: dl,dx, a,b,x0,der2
    integer :: istep
    DL = 1.0E-07
    WRITE (*,*)'Dame los extremos del intervalo que contienen el maximo o minimo:'
    READ(*,*)A,B
    DX = (B-A)/10.
    X0 = (A+B)/2.0
    CALL SECANT (DL,X0,DX,ISTEP)
    if (der2(x0).gt.0) then
        print*,'Se encontro un minimo en x = ',x0
    else
        print*,'Se encontro un maximo en x = ',x0
    endif
    STOP
end

SUBROUTINE SECANT (DL,X0,DX,ISTEP)
    real*8 :: dl,x0,dx, x1,x2,d,der1
    integer :: ISTEP
    ISTEP = 0
    X1 = X0 + DX
    DO    100  WHILE (ABS(DX).GT.DL)
      D  = der1(X1) - der1(X0)
      X2 = X1 - der1(X1)*(X1-X0)/D
      X0 = X1
      X1 = X2
      DX = X1 - X0
      ISTEP = ISTEP + 1
 100 END DO
    RETURN
end

real*8 function f(x)
    real*8 :: x
    f = cos(x)
    return
end

real*8 function der1(x)
    real*8 :: x,h=1.0e-7,f
    der1 = (f(x+h)-f(x-h))/(2*h)
    return
end

real*8 function der2(x)
    real*8 :: x,h=1.0e-7,f
    der2 = (f(x+h)-2*f(x)+f(x-h))/(h**2)
    RETURN
END
```

c. Gráficas GNUPLOT

```gnuplot
set terminal epslatex color
set output 'plot.tex'
set zeroaxis lt -1 lc -1
set key at 3.2,30.0
set xlabel '$x$'
set grid xtics
set grid ytics
set mytics 4
set mxtics 4

a =
b =
f(x) =
plot [a:b] f(x) title "$f(x)$" w l lw 3 lc 0
```