



TECNOLÓGICO DE MONTERREY

MECÁNICA CUÁNTICA
PROYECTO/TAREA 2DO PARCIAL

Sistemas simples en 3D (Pozo esférico de potencial infinito)

Nombres:

Carla Judith López Zurita	A00822301
Rubén Darío Casso de León	A01196975
Alejandro Salinas de León	A01282503

Profesor:

Dr. Julio César Gutiérrez Vega

Lunes 18 de mayo, 2020

1. Introducción

Se busca analizar un sistema simple en 3D de una partícula de masa M que se encuentra atrapada en una esfera de radio a de paredes impenetrables. Su potencial, está entonces dado por:

$$U(r) = \begin{cases} 0, & r < a \\ \infty, & r \geq a \end{cases}$$

Para empezar, se parte de la asunción que se conocen de antemano las soluciones analíticas al problema del pozo esférico de potencial infinito en cuestión. Estas están dadas por las multiplicaciones de las funciones esféricas de Bessel de primera clase $j_l(kr)$ ($l = 0, 1, 2, \dots$) y los armónicos esféricos $Y_l^m(\theta, \phi)$, como se muestra a continuación:

$$\Psi_{nlm}(r, \theta, \phi) = A_l^n j_l(k_l^n r) Y_l^m(\theta, \phi), \quad k_n = \sqrt{\frac{2ME_n}{\hbar^2}} \quad (1)$$

Donde k_l^n se determina al aplicar la condición de frontera en la pared esférica $j_l(k_l^n a) = 0$, y A_{nl} es la constante de normalización de la función. Si β_l^n es el n -ésimo cero de la función esférica de Bessel, tal que $j_l(\beta_l^n) = 0$, entonces reescribimos:

$$\Psi_{nlm}(r, \theta, \phi) = A_l^n j_l(\beta_l^n \frac{r}{a}) Y_l^m(\theta, \phi) \quad (2)$$

$$\text{Con } E_{n,l} = \frac{\hbar^2}{2Ma^2} (\beta_{n,l})^2$$

2. Normalización

Primeramente, nos disponemos a obtener la constante de normalización A_l^n que se encuentra adjunta en la solución del problema, así como lo indica la ecuación (2). Conviene recordar que la función de normalización en coordenadas esféricas está dada por:

$$\int |\Psi_{nlm}|^2 r^2 \sin(\theta) dr d\theta d\phi = 1 \quad (3)$$

Notamos que se puede separar la solución en su ecuación radial y su ecuación armónica esférica quedando la normalización como se muestra a continuación:

$$\int \Psi_{nlm}^* \Psi_{nlm} r^2 \sin(\theta) dr d\theta d\phi = \int |R_{nl}|^2 r^2 dr \int |Y_l^m|^2 \sin(\theta) d\theta d\phi = 1 \quad (4)$$

Esta ecuación puede simplificarse atendiendo a la conocida ortonormalidad de las funciones armónicas:

$$\int (Y_l^m)^* Y_l^m \sin(\theta) d\theta d\phi = 1 \quad (5)$$

Estando en un pozo esférico infinito de radio a y regresando a la definición de la ecuación radial planteada en la ecuación (2), tenemos:

$$\int_0^a |R_{nl}|^2 r^2 dr = \int_0^a |A_l^n j_l(\beta_l^n \frac{r}{a})|^2 r^2 dr = 1 \quad (6)$$

A fin de resolver esta integral, nos apoyamos en el *Mathematical Methods for Physicists* [1] y adoptamos la convención de Sturm-Liouville del problema de dos eigenfunciones $J_v(kp)$ y $J_v(k'p)$ con v no siendo necesariamente un número entero, lo que permite aplicar esta fórmula indiscriminadamente a funciones Bessel esféricas sin problema alguno. En este caso le hacemos un ligero cambio pues la variable r está cuadrada, quedando la ecuación de la siguiente manera:

$$\frac{a^2[k'J_v(ka)J'_v(k'a) - kJ'_v(ka)J_v(k'a)]}{k^2 - k'^2} = \int_0^a J_v(kr)J_v(k'r)r^2dr$$

Usando la regla de l'Hôpital (derivando funciones de la derecha que dependan k) y haciendo que k' tienda a k como límite, esta ecuación se transforma a:

$$\int_0^a [J_v(kr)]^2 r^2 dr = \lim_{k' \rightarrow k} \frac{a^2[J_v(ka)\frac{d}{dk'}(k'J'_v(k'a)) - kJ'_v(ka)\frac{d}{dk'}(J_v(k'a))]}{\frac{d}{dk'}(k^2 - k'^2)}$$

Simplificamos para el caso en que $ka = \alpha_{vi}$, y posicionamos $J_v(ka) = 0$.

$$\int_0^a [J_v(\alpha_{vi}\frac{r}{a})]^2 r^2 dr = \frac{-a^3 k [J'_v(ka)]^2}{-2k} = \frac{a^3}{2} [J'_v(\alpha_{vi})]^2$$

Sin embargo, reconocemos que al ser α_{vi} un cero de J_v entonces $J'_v(\alpha_{vi}) = -J_{v+1}(\alpha_{vi})$ restando la ecuación:

$$\int_0^a [J_v(\alpha_{vi}\frac{r}{a})]^2 r^2 dr = \frac{a^3}{2} [J_{v+1}(\alpha_{vi})]^2 \quad (7)$$

Finalmente, regresamos a nuestra ecuación (6) para insertar este resultado quedando la ecuación:

$$\int_0^a |A_l^n j_l(\beta_l^n \frac{r}{a})|^2 r^2 dr = (A_l^n)^2 \frac{a^3}{2} [j_{l+1}(\beta_l^n)]^2 = 1$$

Después de despejar y sacar raíz en ambos lados nuestra constante de normalización para el problema del pozo esférico infinito resulta en:

$$\boxed{A_l^n = \sqrt{\frac{2}{a^3}} \frac{1}{j_{l+1}(\beta_l^n)}} \quad (8)$$

3. Armónicos esféricos

En esta sección se calcula el armónico esférico $Y_l^m(\theta, \phi)$ para los índices (l, m) y (θ, ϕ) . Conviene recordar la propiedad de la base ortonormal que suponen los armónicos esféricos y que les permite expandir cualquier función $g(\theta, \phi)$ sobre una esfera unitaria. Asimismo, en este apartado se encuentran los coeficientes de la expansión de la siguiente función en términos de los armónicos esféricos:

$$g(\theta, \phi) = \exp[-\cos(\phi + 2\theta)](\pi - \theta)\theta \quad (9)$$

a) En este inciso se muestra el comportamiento de la función $g(\theta, \phi)$ sobre la superficie de la esfera.

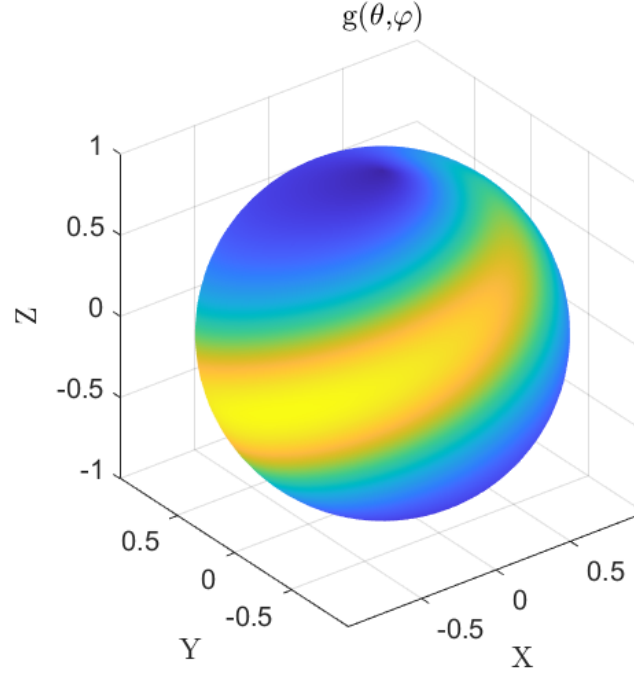


Figura 1: Gráfica de la función g sobre la superficie de la esfera

b) Ahora, se tabulan los coeficientes de expansión de la función comentada a priori, en términos de armónicos esféricos. Para calcularlos, se utilizó el producto interno, haciendo la integral de la función g con la función **trapz** en MATLAB. Los resultados más significativos se muestran a continuación, con l creciendo en la vertical de 0 a 10, y m en la horizontal de -2 a 2, por mencionar algunos valores.

l	$m=-2$	$m=-1$	$m=0$	$m=1$	$m=2$
0	0	0	8.9062	0	0
1	0	-2.4092	3.3276×10^{-16}	2.4092	0
2	0.1487	-2.7614i	-2.2128	-2.7614i	0.1487
3	0.4356i	1.7568	2.9631×10^{-16}	-1.7568	-0.4356i
4	-0.6382	0.5515i	-0.4750	0.5515i	-0.6382
5	-0.5421i	0.0611	-2.0926×10^{-15}	-0.0611	0.5421i
6	0.2304	0.1024i	-0.1864	0.1024i	0.2304
7	-0.0180i	0.0127	6.6308×10^{-16}	-0.0127	0.0180i
8	0.0600	0.0351i	-0.0947	0.0351i	0.0600
9	-0.0035i	0.0045	1.6425×10^{-15}	-0.0045	0.0035i
10	0.0272	0.0158i	-0.0557	0.0158i	0.0272

Cuadro 1: Coeficientes de expansión de $g(\theta, \phi)$, $l \in [0, 10]$ y $m \in [-2, 2]$

Es importante recalcar que los valores tabulados previamente no corresponden a los números completos; se decidió considerar únicamente la parte del número complejo que fuera más significativa comparada con la otra. Es por ello que no aparecen las partes reales y complejas juntas en ningún caso. Es de mayor interés para nosotros la visualización del comportamiento general de los coeficientes en la tabla, y no la precisión de los números, aunque esta última sí fue tomada en cuenta para incisos subsecuentes a este problema.

Fácilmente se observa también, que el armónico esférico más dominante de la serie es $Y_0^0(\theta, \phi)$, ubicado en el primer renglón, tercera columna de la tabla 1. Su respectivo coeficiente tiene un valor de: $C_0^0=8.9062$.

En su totalidad, la serie de coeficientes contiene tantos términos que se volvería ineficiente computacionalmente usar una cantidad grande y arbitraria para generar la función. Para tener un criterio de truncamiento de la serie, calculamos el error relativo de la expansión. Es decir, si $F(\theta, \phi)$ es la función reconstruida, entonces el error relativo para cada punto angular debe ser menor a una tolerancia, digamos, del 1 %:

$$\left| \frac{F(\theta, \phi) - f(\theta, \phi)}{f(\theta, \phi)} \right| < tol = 0,01$$

Así, para poder cumplir esa tolerancia, se necesitaron valores de hasta $l = 40$, y la razón es porque muchos coeficientes de la expansión poseen valores cercanos a 0; cualquier variación mínima significaba entonces una desviación de más del 1 % de tolerancia. Aunque como notaremos en la siguiente imagen, usar hasta $l = 20$ ya se asemeja bastante a la función original gráficamente.

c) A continuación, la imagen 2 muestra una comparativa entre la función sintetizada sobre la superficie de la esfera a diferentes rangos de l , y la función original.

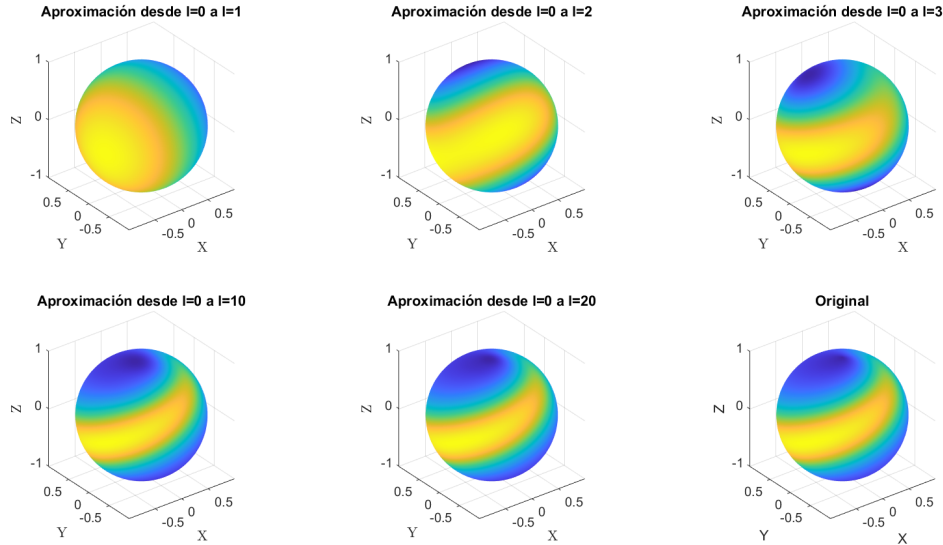


Figura 2: Comparación de función reconstruida con función original

4. Pozo esférico

Esta sección tiene como fin calcular las eigenfunciones $\Psi_{nlm}(r, \theta, \phi)$ del pozo esférico infinito. A fin de simplificar algunas constantes en la solución numérica, se considera que $a = 1$, $\hbar = 1$, $M = \frac{1}{2}$, por lo que $\hbar^2/2Ma^2 = 1$, que viene a dar las mismas unidades que las energías. Tomamos en consideración una partícula atrapada con la función siguiente en el estado inicial:

$$Q(r, \theta, \phi) = \mathcal{B}(a - r^2)g(\theta, \phi) = \mathcal{B}(a - r^2)e^{-\cos(\phi+2\theta)}(\pi - \theta)\theta \quad (10)$$

Donde \mathcal{B} es una constante de normalización y a es el radio unitario de la esfera.

(a) *Se calcula la constante de normalización \mathcal{B} :*

Así como en el problema pasado, se sabe que el valor de \mathcal{B} va a ser tal que se cumpla la siguiente relación:

$$\iiint_{\mathcal{V}} \mathcal{B}^2 |Q(r, \theta, \phi)|^2 r^2 \sin \theta dr d\theta d\phi = 1$$

Con el fin de integrarlo rápido numéricamente, se recurrió al lenguaje de Mathematica, y se empleó el código que se muestra a continuación:

```
In[1]:= a = 1;

In[24]:= G[phi_, theta_] := Exp[-Cos[phi + 2*theta]]*(Pi - theta)*theta

In[25]:= Int = NIntegrate[(a - r^2)*G[phi, theta]^2*r^2*Sin[theta], {r, 0, a},
{theta, 0, Pi}, {phi, 0, 2*Pi}];

In[26]:= B = 1/Sqrt[Int]

0.32792284570150765
```

Figura 3: Código de Mathematica para cálculo de \mathcal{B}

Así, conocemos que el valor de \mathcal{B} es de:

$$\mathcal{B} = 0.3279228$$

Si se quisiese calcular analíticamente el valor de \mathcal{B} , habría que primero descomponer totalmente Q en sus eigenfunciones con sus respectivos coeficientes C_{nlm} (que serán obtenidos en el siguiente inciso), cuadrar cada uno y sumarlos; así, habríamos resuelto la integral de la figura 3. Sin embargo, como en general puede haber infinitas eigenfunciones, es mejor ni siquiera preocuparnos por hacerlo analítico cuando tenemos herramientas computacionales que realizan el trabajo de forma precisa y rápida.

- (b) *Encontramos la expansión de Q en términos de las eigenfunciones $\Psi_{nlm}(r, \theta, \phi)$:*

Se volvió a usar la función **trapz** para integrar numéricamente, y se realiza la operación $C_{nlm} = \langle \Psi_{nlm} | Q \rangle$ para conseguir los coeficientes de la expansión. Los datos son guardados en una matriz tridimensional; por lo que la tabla siguiente muestra los valores más significativos de l y m , para $n = 1$.

l	m=-2	m=-1	m=0	m=1	m=2
0	0	0	0.7991	0	0
1	0	-0.2146	-7.3362×10^{-19}	0.2146	0
2	0.0126	-0.2348i	-0.1881	-0.2348i	0.0126
3	0.0350i	0.1411	3.6054×10^{-17}	-0.1411	-0.0350i
4	-0.0484	0.0418i	-0.0360	0.0418i	-0.0484
5	-0.0388i	0.0044	6.45×10^{-17}	-0.0044	0.0388i
6	0.0156	0.0069i	-0.0126	0.0069i	0.0156
7	-0.0012i	0.0008	7.09×10^{-17}	-0.0008	0.0012i
8	0.0037	0.0021i	-0.0058	0.0021i	0.0037
9	-0.0002i	0.0003	8.20×10^{-17}	-0.0003	0.0002i
10	0.0015	0.0009i	-0.0031	0.0009i	0.0015

Cuadro 2: Coeficientes de expansión de Q para $n = 1$, $l \in [0, 10]$ y $m \in [-2, 2]$

Una vez más, así como en la tabla pasada, se incluyeron solamente los términos dominantes con respecto al otro, ya sea la parte real o la parte imaginaria, y no ambas.

- (c) *Obtenemos el valor más probable después de medir la energía de la partícula en el estado Q :*

Se sabe, por la tabla 2, que el coeficiente con mayor valor corresponde a Ψ_{100} , y además, que las energías de una eigenfunción Ψ_{nlm} salen de $(\beta_{nl})^2$. Así, se obtiene que la energía más probable está dada por:

$$E = 9.8696$$

- (d) *Calculamos el valor esperado de la medición de energía de la partícula en Q :*

El método de hacerlo es multiplicar cada valor de energía, que ya mencionamos es $(\beta_{nl})^2$ por su respectivo coeficiente $|C_{nlm}|^2$ de la expansión. De esta forma, se consigue que el valor esperado sea:

$$\langle E \rangle = 24.1326$$

- (e) *Ahora obtenemos el valor más probable después de medir la componente z del momento angular orbital, L_z .*

De antemano, sabemos que los eigenvalores de L_z son: $m\hbar$, por lo que al operar L_z sobre una función, en este caso la eigenfunción Ψ_{nlm} , resulta en:

$$L_z \Psi_{nlm} = m\hbar \Psi_{nlm}$$

Sin embargo, hay que recordar que ya nos habíamos percatado del coeficiente mayor de la expansión, que corresponde al valor más probable en una medición. Volviendo a ver la tabla 2, dicho coeficiente se trata de Ψ_{100} , donde $m = 0$. Por lo tanto, podemos concluir que:

$$L_z = 0$$

(f) *Obtenemos el valor esperado de la medición de $|\mathbf{L}|^2$ de la partícula:*

Se conocen los eigenvalores de $|\mathbf{L}|^2$: $\hbar^2 l(l+1)$, por lo que el operar en la eigenfunción Ψ_{nlm} , resultaría en:

$$|\mathbf{L}|^2 \Psi_{nlm} = \hbar^2 l(l+1) \Psi_{nlm}$$

Ahora, usando el mismo procedimiento del inciso (d), se multiplica cada valor de $|\mathbf{L}|^2 \Psi_{nlm}$ por su respectivo coeficiente $|C_{nlm}|^2$. De esta forma, se consigue que el valor esperado sea:

$$\langle |\mathbf{L}|^2 \rangle = 2.3309$$

(g) *Considerando que $Q(r, \theta, \phi)$ está en la condición de $t=0$, se realiza una simulación de la evolución temporal de la función de estado. Asimismo, se muestra la densidad de probabilidad $|Q|^2$ en función del tiempo según dos posibles casos:*

Las simulaciones de las evoluciones temporales para los siguientes dos casos se encuentran dentro de las respectivas ligas de Google Drive.

(I) *Estando sobre la superficie de una esfera de radio $\frac{a}{2}$*

<https://drive.google.com/open?id=1qWNlcDRxNb6z5eLqznEjGUqyiwwYvJaw>

(II) *Estando sobre el plano $z=0$*

<https://drive.google.com/open?id=12b5IdmXgaucPYIHANauB-P4fPDGo9bmo>

5. Conclusiones

El proyecto representó un reto demandante para nosotros, no solo por la programación de las funciones y obtención de coeficientes precisos de las expansiones, sino porque, en general, nos vimos en la necesidad de utilizar los conocimientos de la clase en un problema práctico que profundizó más que las tareas. Gran parte de la dificultad sin duda recayó en lo abstracto de los conceptos, como sucede mucho en Mecánica Cuántica; teníamos que revisar varias veces nuestro código y el conocimiento teórico, ya que no conocíamos con certeza los resultados correctos de, por ejemplo, las simulaciones de la evolución temporal de la sección 3.

Aun así, el trabajo realizado resultó una experiencia enriquecedora que ayudó a hacer más tangible lo aprendido en clase y a mejorar la visualización (y por lo tanto, el entendimiento) de los conceptos del curso.

Referencias

- [1] G. Arfken. *Mathematical Methods for Physicists*. Academic Press, Inc., San Diego, third edition, 1985.

6. Apéndice: Códigos implementados

6.1. Función Y_{lm} para armónicos esféricos

```
1 function Ylm = ylm(l,M,phi,theta)
2 m = abs(M);
3 L = legendre(l,cos(theta));
4 if l == 0
5     Lm = L;
6 else
7     Lm = reshape(L(m+1,:,:),size(phi));
8 end
9 norm = sqrt((2*l+1).*factorial(l-m)/(4*pi*factorial(l+m)));
10 phase = (-1)^m.*exp(1i.*m.*phi);
11
12 Ylm = norm.*Lm.*phase;
13 if M<0
14     Ylm = (-1)^m.*conj(Ylm);
15 end
16 end
```

6.2. Función $g(\theta, \phi)$

```
1 % Generacion del espacio
2 N=2^6;
3 pol=linspace(0,pi,N);
4 az=linspace(0,2*pi,N);
5 [phi,theta] = meshgrid(az,pol);
6
7 % Funcion a analizar
8 g = exp(-cos(phi+2*theta)).*(pi-theta).*theta;
9
10 % Descomposicion en eigenestados
11 L=40;
12 coeff= zeros(size(-L:L,2),L); gY = 0;
13 for l=0:L
14     i=1;
15     for m = -l:l
```

```

16     Yl = ylm(l,m,phi,theta);
17     coeff(i,l+1) = trapz(trapz(conj(Yl).*g.*sin(theta)))*2*(
        pi/N).^2;
18     gY = gY + coeff(i,l+1).*Yl;
19     i=i+1;
20     end
21 end
22
23 % Coordenadas cartesianas
24 R = ones(size(phi)); %radio
25 [X,Y,Z] = sph2cart(phi,pi/2-theta,R);
26
27 %% Graficas
28 % Funcion reconstruida a diferentes rangos de l
29 image(1); subplot(2,3,1)
30 s1 = surf(X,Y,Z,real(gY));
31 direction = [0 0 1];
32 rotate(s1,direction,200);
33 set(gca,'FontSize',12);
34 title('Aproximaci n desde l=0 a l=1')
35 axis equal
36 xlabel('X'); ylabel('Y'); zlabel('Z');
37 colormap parula
38 shading interp
39
40 % Funcion original g sobre esfera
41 figure(2);
42 s = surf(X,Y,Z,real(g));
43 direction = [0 0 1];
44 rotate(s,direction,200);
45 set(gca,'FontSize',12);
46 set(0,'defaultTextInterpreter','latex');
47 title('g($\theta$, $\varphi$)')
48 axis equal
49 xlabel('X');
50 ylabel('Y');
51 zlabel('Z');
52 colormap parula
53 shading interp

```

6.3. Función *betaln* para cálculo de β_{nl}

```

1 function betal = betaln(l,n)
2 N=2^8; up = 50; h = up/N;

```

```

3 r = linspace(0,up,N);
4 esf = @(l,x) sqrt(pi./(2.*x)).*besselj(l+1/2,x);
5 betal = r(esf(l,r).*esf(l,r+h)<0) + h/2;
6 betal = betal(n);
7 end

```

6.4. Función $Q(r, \theta, \phi)$

```

1 % condiciones inciales
2 a = 1; h = 1; M = 1/2;
3
4 N=2^6;
5 pol=linspace(0,pi,N);
6 az=linspace(0,2*pi,N);
7 R = linspace(0.0000001,a,N);
8 [phi,theta,r] = meshgrid(az,pol,R);
9
10 g = exp(-cos(phi+2*theta)).*(pi-theta).*theta;
11 B = 0.32792284570150765;
12 Q = B.*(1-r.^2).*g;
13
14 % bessel esferica
15 esf = @(l,x) sqrt(pi./(2.*x)).*besselj(l+1/2,x);
16 % normalizacion
17 Aln =@(l,n) sqrt(2/a^3)/esf(l+1,betaln(l,n));
18 % eigenestados
19 Psi =@(n,l,m) Aln(l,n).*esf(l,betaln(l,n).*r./a).* ylm(l,m,
    phi,theta);
20
21 % decomposicion en eigenestados
22 L=10; nmax = 3;
23 coeff2 = zeros(nmax,size(-L:L,2),L+1); qY = 0;
24
25 for n = 1:nmax
26 for l=0:L
27     i=1;
28
29     for m = -l:l
30         coeff2(n,i,l+1) = trapz(trapz(trapz(conj(Psi(n,l,m)).*Q
            .*r.^2.*sin(theta)))).*(2.*a./N).*(pi/N).^2;
31         qY = qY + coeff2(n,i,l+1).*Psi(n,l,m);
32
33     i=i+1;
34     end

```

```

35 end
36 end
37
38 % el coeficiente maximo corresponde a
39 MaxC = max(max(max(coeff2)));
40 % que est dado por n=1,l=0,m=0

```

6.5. Valores esperados

```

1 Et = 0;
2 for n = 1:nmax
3     for l=0:L
4         i=1;
5         for m = -l:l
6             %para el valor esperado de la energia
7             En = h^2/(2*M*a)*betaln(l,n)^2.*conj(coeff2(n,i,l+1))
                .*coeff2(n,i,l+1);
8             Et = Et + En;
9
10            %para el valor esperado del momentum angular L^2
11            Ln = l*(l+1).*conj(coeff2(n,i,l+1)).*coeff2(n,i,l+1)
                ;
12            Lt = Lt + Ln;
13
14            i=i+1;
15        end
16    end
17 end

```

6.6. Evolución temporal

6.6.1. Sobre esfera radio $a/2$

```

1 % condiciones inciales
2 a = 1; h = 1; M = 1/2;
3
4 % mandamos a llamar los coeficientes
5 S = load('coeff2');
6 coeff2 = cell2mat(struct2cell(S));
7
8 % generacion del espacio
9 N = 2^6;
10 pol = linspace(0,pi,N);

```

```

11 az = linspace(0,2*pi,N);
12 r = a/2;
13 [phi,theta] = meshgrid(az,pol);
14
15 % descomposicion en eigenestados
16 L = 5; nmax = 5; qt=0;
17
18 k= 50; seg = 10;
19 saveq = zeros(k,N,N);
20 p = 1;
21
22 for t = 0:seg/k:seg %tiempo
23 for n = 1:nmax
24 for l=0:L
25
26     i=1;
27     % bessel esferica
28     J = sqrt(pi./(2.*betaln(l,n).*r)).*besselj(l+1/2,betaln(
        l,n).*r);
29     % normalizacion
30     Aln = sqrt(2)/(sqrt(pi./(2.*betaln(l,n))).*besselj(l
        +3/2,betaln(l,n)));
31     % energia
32     En = betaln(l,n).^2;
33
34     for m = -l:l
35         % eigenestados
36         Psi = Aln.*J.*ylm(l,m,phi,theta);
37         qt = qt + coeff2(n,i,l+1).*Psi.*exp(-1i*En*t);
38
39         i=i+1;
40     end
41 end
42 end
43 saveq(p, :, :) = qt;
44 disp(p)
45 p = p + 1;
46 end
47
48 %% Grafica
49 R = ones(size(phi)); %radio
50 [X,Y,Z] = sph2cart(phi,pi/2-theta,R);
51 figure(1);
52
53 for p = 1:k+1

```

```

54 figure(1);
55 kj = reshape(saveq(p, :, :), 64, 64);
56 s = surf(X, Y, Z, abs(kj).^2);
57 direction = [0 0 1];
58 rotate(s, direction, 200);
59 set(gca, 'FontSize', 12);
60 set(0, 'defaultTextInterpreter', 'latex');
61 title(['$|Q(t=$' num2str(seg/k*(p-1)) '$)|^2$']);
62 axis equal
63 xlabel('X'); ylabel('Y'); zlabel('Z');
64 colormap parula
65 shading interp
66 pause(0.1)
67 end

```

6.6.2. Sobre plano $z=0$

```

1  % condiciones iniciales
2  a = 1; h = 1; M = 1/2;
3
4  % mandamos a llamar los coeficientes
5  S = load('coeff2');
6  coeff2 = cell2mat(struct2cell(S));
7
8  % generacion del espacio
9  N = 2^6;
10 rho = linspace(0, a, N);
11 az = linspace(0, 2*pi, N);
12 theta = pi/2;
13 [phi, r] = meshgrid(az, rho);
14
15 % descomposicion en eigenestados
16 L = 5; nmax = 5; qt=0;
17 k= 50; seg = 20;
18 saveq = zeros(k, N, N);
19 p = 1;
20 for t = 0:seg/k:seg %tiempo
21 for n = 1:nmax
22 for l=0:L
23
24     i=1;
25     %bessel esferica
26     J = sqrt(pi./(2.*betaln(l, n).*r)).*besselj(l+1/2, betaln(
        l, n).*r);

```

```

27      % normalizacion
28      Aln = sqrt(2)/(sqrt(pi./(2.*betaln(1,n))).*besselj(1
      +3/2,betaln(1,n)));
29      % energia
30      En = betaln(1,n).^2;
31
32      for m = -1:1
33          % eigenstados
34          Psi = Aln.*J.*ylm2d(1,m,phi,theta);
35          qt = qt + coeff2(n,i,l+1).*Psi.*exp(-1i*En*t);
36
37          i=i+1;
38      end
39  end
40  end
41  saveq(p,:,:) = qt;
42  disp(p)
43  p = p + 1;
44  end
45
46  %% Grafica
47  X = r.*cos(phi);
48  Y = r.*sin(phi);
49
50  for p = 1:k+1
51      figure(1);
52      kj = reshape(saveq(p,:,:),64,64);
53      surf(X,Y,abs(kj).^2);
54      view(2)
55      set(gca,'FontSize',12);
56      set(0,'defaultTextInterpreter','latex');
57      title(['$|Q(t=$' num2str(seg/k*(p-1)) '$)|^2$'])
58      axis equal
59      xlabel('X'); ylabel('Y');
60      colormap parula
61      shading interp
62      pause(0.2)
63  end

```