Daftar Isi

BAB I Pendahuluan **3**

1.1 Latar Belakang 3

1.2 Rumusan Masalah 3

1.3 Tujuan 3

BAB II Landasan Teori **4**

2.1 Persamaan Schrodinger Atom Elektron Tunggal 4

2.2 Fungsi Legendre Terasosiasi 5

2.3 Fungsi Laguerre Terasosiasi 5

2.4 Solusi Persamaan Schrodinger 5

2.5 Diferensial Numerik Orde Tinggi 7

2.6 Integral Trapezoid 8

2.7 Interpolasi Lagrange 8

2.8 Integral Simpson 9

2.9 Skrip MATLAB 9

2.9.1 Fungsi Faktorial 9

2.9.2 Diferensial Numerik Orde Tinggi 9

2.9.3 Fungsi Azimuthal 10

2.9.4 Fungsi Legendre Terasosiasi 11

2.9.5 Interpolasi Lagrange 12

2.9.6 Fungsi Laguerre Terasosiasi 14

2.9.7 Integral Simpson 3/8 15

2.9.8 Harmonik Bola 17

2.9.9 Distribusi Probabilitas dan Fungsi Gelombang 18

BAB III Metodologi Penelitian **23**

3.1 Jenis Penelitian 23

3.2 Peralatan Yang Digunakan 23

3.3 Lokasi dan Waktu Penelitian 23

3.4 Prosedur Penelitian 23

3.5 Diagram Alir 24

BAB IV Hasil dan Pembahasan **25**

4.1 Hasil 25

4.2 Pembahasan 31
BAB V Kesimpulan dan Saran **33**
5.1 Kesimpulan 33
5.2 Saran 33
Daftar Pustaka **34**
Lampiran **35**
1. Fungsi Legendre Terasosiasi 35
2. Fungsi Laguerre Terasosiasi 36
3. Harmonik Bola 37
4. Isotop Atom Hidrogen Deuterium 40
5. Isotop Atom Hidrogen Tritium 42
6. Atom Helium 44
7. Atom Lithium 45
8. Atom Berilium 47

BAB I

PENDAHULUAN

1.1 Latar Belakang

Permasalahan-permasalahan yang semakin kompleks dari waktu ke waktu menuntut manusia untuk selalu berkembang dan mencari pemecahan dari permasalahan tersebut. Hal ini mendorong semakin berkembang pula ilmu pengetahuan dan teknologi yang dapat membantu manusia dalam menyelesaikan permasalahan-permasalahannya. Salah satu disiplin ilmu tersebut adalah matematika, dimana dalam matematika terdapat suatu kajian tentang pemodelan dan simulasi dengan metode numerik yang sedikit banyak dapat membantu manusia untuk menyelesaikan masalahnya.

Pada laporan ini, dicoba untuk mengetahui distribusi probabilitas untuk atom berelektron tunggal dengan cara numerik dengan bantuan MATLAB. Dengan menggunakan metode numeric, solusi exact daripersoalan yang dihadapi tidak akan diperoleh. Metode numeric hanya bisa memberikan solusiyang mendekati atau menghampiri solusi sejati sehingga solusi numeric dinamakan juga solusi hampiran (approximation solution). Pendekatan solusi ini tentu saja tidak tepat sama dengan solusi sejati, sehingga ada selisih antara keduanya.

1.2 Perumusan Masalah

Rumusan masalah yang didapatkan adalah:

1. Bagaimana distribusi probabilitas elektron tunggal pada atom?
2. Bagaimana bentuk fungsi gelombang atom?
3. Bagaimana bentuk harmonik bola untuk fungsi gelombang anguler?

1.3 Tujuan

Tujuan diadakannya penelitian ini yaitu untuk mengetahui:

1. Distribusi probabilitas elektron tunggal pada atom.
2. Fungsi gelombang atom.
3. Harmonik bola untuk fungsi gelombang anguler.

BAB II

LANDASAN TEORI

2.1 Persamaan Schrodinger Atom Elektron Tunggal

Persamaan Schrodinger tiga dimensi tak bergantung waktu memiliki bentuk

$$-\frac{\hbar^2}{2\mu}\nabla^2\Psi + V\Psi = E\Psi \tag{1}$$

Untuk kasus Laplacian koordinat bola diberikan oleh

$$\nabla^2 = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial}{\partial r}\right) + \frac{1}{r^2 \sin\theta}\frac{\partial}{\partial\theta}\left(sin\theta\frac{\partial}{\partial\theta}\right) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2}{\partial\varphi^2} \tag{2}$$

Dalam sistem atom, maka ketika sebuah elektron mengorbit inti atom, maka akan terjadi dua interaksi antara elektron dengan inti, yaitu interaksi gravitasi dan elektrostatik. Karena gaya gravitasi antara keduanya sangat kecil dibandingkan dengan interaksi elektrostatiknya, maka interaksi gravitasi dapat diabaikan, sehingga potensial yang dialami elektron hanyalah potensial elektrostatik. Apabila dianggap hanya ada satu elektron yang mengorbit inti yang memiliki nomor atom Z, maka potensial elektrostatiknya diberikan oleh

$$V = -\frac{Ze^2}{4\pi\varepsilon_0 r} \tag{3}$$

Sehingga persamaan Schrodinger untuk atom terionisasinya,

$$-\frac{\hbar^2}{2\mu}\left(\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial\Psi}{\partial r}\right) + \frac{1}{r^2 \sin\theta}\frac{\partial}{\partial\theta}\left(sin\theta\frac{\partial\Psi}{\partial\theta}\right) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2\Psi}{\partial\varphi^2}\right) - \frac{Ze^2}{4\pi\varepsilon_0 r}\Psi = E\Psi \tag{4}$$

Untuk menyelesaikan persamaan diferensial parsial tersebut dilakukan pemisahan variabel.

$$\Psi = R(r)\Theta(\theta)\Phi(\varphi) \tag{5}$$

Bentuk persamaan Schrodinger menjadi

$$-\frac{\hbar^2}{2\mu R}\left(\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial R}{\partial r}\right) + \frac{1}{r^2 \sin\theta\Theta}\frac{\partial}{\partial\theta}\left(sin\theta\frac{\partial\Theta}{\partial\theta}\right) + \frac{1}{r^2\sin^2\theta\Phi}\frac{\partial^2\Phi}{\partial\varphi^2}\right) - \frac{Ze^2}{4\pi\varepsilon_0 r} = E \tag{6}$$

Dengan demikian akan didapatkan tiga buah persamaan diferensial yang terkopel

$$\frac{d}{dr}\left(r^2\frac{dR}{dr}\right) + \frac{2\mu r^2}{\hbar^2}\left(E + \frac{e^2}{4\pi\varepsilon_0 r}\right)R = l(l+1)R \tag{7}$$

$$\frac{d^2\Phi}{d\varphi^2} = -m^2\Phi \tag{8}$$

$$sin\theta\frac{d}{d\theta}\left(sin\theta\frac{d\Theta}{d\theta}\right) - m^2\Theta = -l(l+1)\sin^2\theta\,\Theta \tag{9}$$

2.2 Fungsi Legendre Terasosiasi

Apabila sebuah persamaan diferensial orde dua linear memiliki bentuk

$$(1-x^2)\frac{d^2y}{dx^2} - 2x\frac{dy}{dx} + \left[l(l+1) - \frac{m^2}{1-x^2}\right]y = 0 \qquad (10)$$

Persamaan diferensial ini disebut dengan persamaan diferensial Legendre Terasosiasi. Apabila kasus khusus dimana m = 0, maka persamaan ini akan menjadi persamaan Legendre. Solusi persamaan diferensial Legendre Terasosiasi merupakan sebuah polinomial dengan orde l dan dengan derajat m atau dapat ditulis sebagai $P_l^m(x)$. Bentuk polinomial Legendre Terasosiasi dapat dinyatakan dalam persamaan Rodrigues.

$$P_l^m(x) = \frac{(-1)^m}{2^l l!}(1-x^2)^{\frac{m}{2}}\frac{d^{l+m}}{dx^{l+m}}(x^2-1)^l \qquad (11)$$

Polinomial Legendre Terasosiasi ini hanya memperbolehkan nilai derajat m berkisar antara $-l \leq m \leq l$. Untuk nilai m yang negatif, persamaan Rodrigues untuk polinomial Legendre Terasosiasi adalah

$$P_l^{-m}(x) = (-1)^m \frac{(l-m)!}{(l+m)!}P_l^m(x) \qquad (12)$$

$$= (-1)^m \frac{(l-m)!}{(l+m)!}\frac{(-1)^m}{2^l l!}(1-x^2)^{\frac{m}{2}}\frac{d^{l+m}}{dx^{l+m}}(x^2-1)^l \qquad (13)$$

2.3 Fungsi Laguerre Terasosiasi

Apabila sebuah persamaan diferensial orde dua linear memiliki bentuk

$$x\frac{d^2y}{dx^2} + (k+1-x)\frac{dy}{dx} + ny = 0 \qquad (14)$$

Persamaan diferensial ini disebut dengan persamaan diferensial Laguerre Terasosiasi. Apabila $k = 0$, maka persamaan tersebut menjadi persamaan Laguerre. Solusi dari persamaan diferensial Laguerre Terasosiasi dengan orde n dan derajat k, atau dapat ditulis sebagai $L_n^k(x)$. Bentuk polinomial Laguerre Terasosiasi ini dapat dinyatakan dalam persamaan Rodrigues.

$$L_k^n(x) = \frac{x^{-k}e^x}{n!}\frac{d^n}{dx^n}(e^{-x}x^{n+k}) \qquad (15)$$

Polinomial Laguerre Terasosiasi ini hanya memperbolehkan nilai $n \geq 0$ dan nilai $k \geq -1$.

2.4. Solusi Persamaan Schrodinger

Tiga persamaan terpisah dari persamaan Schrodinger untuk atom terionisasi dapat diselesaikan satu per satu secara independen. Untuk solusi Azimuthal, meninjau persamaan (8).

$$\frac{d^2\Phi}{d\varphi^2} = -m^2\Phi \qquad (16)$$

$$\frac{d^2\Phi}{d\varphi^2} + m^2\Phi = 0 \qquad (17)$$

Maka didapatkan basis solusinya adalah

$$\Phi = e^{im\Phi} \quad (18)$$

Dimana nilai m yang diperbolehkan adalah semua bilangan bulat negatif maupun positif. Kemudian untuk menyelesaikan persamaan Orbital, meninjau persamaan (9).

$$sin\theta \frac{d}{d\theta}\left(sin\theta \frac{d\Theta}{d\theta}\right) - m^2\Theta = l(l+1)\sin^2\theta\,\Theta \quad (19)$$

$$\frac{d^2\Theta}{d\theta^2} - \frac{\cos\theta}{sin\theta}\,\Theta + [l(l+1) - \frac{m^2}{\sin^2\theta}]\ \Theta = 0 \quad (20)$$

Apabila dilakukan substitusi $x = cos\theta$, maka solusina akan menyadi persamaan Legendre Terasosiasi yaitu

$$\Theta(\theta) = P_m^l(cos\theta) \quad (21)$$

Solusi radial persamaan Schrodinger dari persamaan (7) akan menghasilkan suku polinomial Laguerre terasosiasi, yaitu

$$R(r) = e^{\frac{r}{na_0}}\left(\frac{2r}{na_0}\right)^l L_{n-l-1}^{2l+1}(\frac{2r}{na_0}) \quad (22)$$

Dimana a_0 adalah jari-jari Bohr yang memiliki besar $a_0 = \frac{4\pi\varepsilon_0\hbar^2}{\mu e^2}$. Sehingga solusi fungsi gelombang atom terionisasinya adalah

$$\Psi_{nlm}(r,\theta,\varphi) = C_{nlm}R(r)\Theta(\theta)\Phi(\varphi) \quad (23)$$

Dengan C_{nlm} adalah koefisien normalisasi yang bergantung pada ketiga bilangan kuantum n,m,l. dimana,

$$P = \int_0^{2\pi}\int_0^{\pi}\int_0^{\infty}|\Psi_{nlm}(r,\theta,\varphi)|^2 r^2 sin\theta drd\theta d\varphi = 1 \quad (24)$$

Atau

$$\int_0^{2\pi}\int_0^{\pi}\int_0^{\infty}|C_{nlm}|^2|R(r)|^2|\Theta(\theta)|^2|\Phi(\varphi)|^2 r^2 sin\theta drd\theta d\varphi = 1 \quad (25)$$

Sehingga

$$|C_{nlm}|^2 = \frac{1}{\int_0^{2\pi}\int_0^{\pi}\int_0^{\infty}|R(r)|^2|\Theta(\theta)|^2|\Phi(\varphi)|^2 r^2 sin\theta drd\theta d\varphi} \quad (26)$$

$$C_{nlm} = \sqrt{\frac{1}{\int_0^{2\pi}\int_0^{\pi}\int_0^{\infty}|R(r)|^2|\Theta(\theta)|^2|\Phi(\varphi)|^2 r^2 sin\theta drd\theta d\varphi}} \quad (27)$$

Didefinisikan juga yang dinamakan radial probability yang merupakan probabilitas partikel pada rentang r hingga r + rd.

$$P(r) = \int_0^{2\pi}\int_0^{\pi}|\Psi_{nlm}(r,\theta,\varphi)|^2 r^2 sin\theta d\theta d\varphi \quad (28)$$

2.5 Diferensial Numerik Orde Tinggi

Apabila sebuah fungsi

$$y = f(x) \quad (28)$$

Akan memiliki deret Taylor disekitar h, yaitu

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \frac{h^4}{4!}f^{(4)}(x) + \cdots \quad (29)$$

Apabila diambil $f'(x)$ di sisi kiri persamaan, maka didapatkan

$$f'(x) = \frac{f(x+h)-f(x)}{h} - \frac{h}{2!}f''(x) - \ldots \quad (30)$$

Apabila hanya diambil suku pertama di sisi kanan, maka didapatkan turunan orde pertama

$$f'(x) = \frac{f(x+h)-f(x)}{h} + O(h) \quad (31)$$

Dengan cara yang sama, mengubah fungsi $f(x)$ dengan $f'(x)$, maka didapatkan turunan keduanya,

$$f''(x) = \frac{f'(x+h)-f'(x)}{h} + O(h) \quad (32)$$

$$f''(x) = \frac{f(x+2h)-2f(x+h)+f(x)}{h^2} + O(h) \quad (33)$$

Untuk mencari turunan ke-3 nya,

$$f'''(x) = \frac{f''(x+h)-f''(x)}{h} + O(h) \quad (34)$$

$$f'''(x) = \frac{f(x+3h)-3f(x+2h)+3f(x+h)-f(x)}{h^3} + O(h) \quad (35)$$

Dari sini terlihat pola dalam mencari turunan ke-n dengan metode numerik.

$$f^n(x) = \sum_{k=0}^{n} C_{nk} f(x+kh) + O(h) \quad (36)$$

Dari bentuk fungsi turunan pertama, kedua dan ketiga terlihat bahwa nilai C_{nk} berubah tanda selang seling bergantung pada nilai n dan k nya, sehingga

$$C_{nk} = (-1)^{n-k} A_{nk} \quad (37)$$

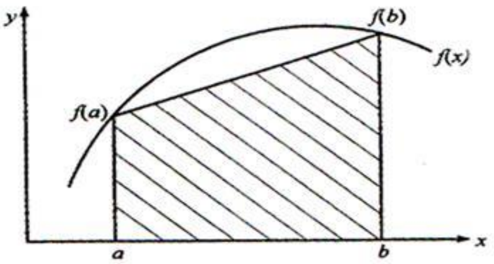
Pada turunan pertama, koefisiennya secara berturut-turut adalah 1 dan 1, untuk turunan kedua secara berturut-turut adalah 1,2,dan 1, untuk turunan ke3 memiliki 1,3,3,dan 1. Dari pola ini terlihat bahwa A_{nk} mengikuti bentuk Segitiga Pascal, dimana

$$A_{nk} = \binom{n}{k} = \frac{n!}{k!(n-k)!} \quad (38)$$

Sehingga secara umum,

$$f^n(x) = \frac{1}{h^n} \sum_{k=0}^{n} (-1)^{n-k} \frac{n!}{k!(n-k)!} f(x+kh) + O(h) \quad (39)$$

2.6 Integral Trapezoid



Gambar Metode trapesium

Apabila sebuah fungsi $f(x)$ ingin dicari luasannya di antara batas atas b dan batas bawah a, maka cara paling mudah adalah membagi luasan di bawah fungsi tersebut menjadi trapesium-trapesium yang sangat kecil dan tak berhingga jumlahnya. Luasan di bawah kurva adalah jumlahan dari semua luas trapesium yang ada.

$$L = \sum \frac{1}{2}(f(x_i) + f(x_{i+1}))\ \Delta x \tag{40}$$

Apabila nilai Δx diambil sangat kecil, maka, Luasan akan sama seperti yang diharapkan.

$$L = \lim_{\Delta x \to 0} \sum_{i=0}^{\frac{b-a}{\Delta x}} \frac{1}{2}\big(f(x_i) + f(x_{i+1})\big)\Delta x \tag{41}$$

2.7 Interpolasi Lagrange

Bentuk paling sederhana dalam interpolasi adalah bentuk polinomial. Ini sangat mungkin untuk membentuk polinom berorde $n-1$ yang melewati n data. Cara paling sederhana adalah dengan menggunakan metoda Lagrange.

$$P_{n-1}(x) = \sum_{i=1}^{n} y_i L_i(x) \tag{42}$$

Dimana

$$L_i(x) = \frac{x-x_1}{x_i-x_1} \cdot \frac{x-x_2}{x_i-x_2} \cdots \frac{x-x_{i-1}}{x_i-x_{i-1}} \cdot \frac{x-x_{i+1}}{x_i-x_{i+1}} \cdots \frac{x-x_n}{x_i-x_n} \tag{43}$$

$$L_i(x) = \prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{x-x_i}{x_i-x_j} \tag{44}$$

2.8 Integral Simpson

Integral dengan metode Simpson dapat diturunkan dengan substitusi Interpolasi Lagrange orde-3 Dimana untuk batas dari a, dan $x_1 = a + \Delta x, x_2 = a + 2\Delta x$,... h, maka

$$\int_a^b f(x)dx = \int_{x_0}^{x_2} P_3(x)dx + O(h^4) =$$

$$\frac{3\Delta x}{8}\left[f(a) + 3\sum_{i=4,7,\dots}^{n-2} f(x_i) + 3\sum_{i=2,5,8,,\dots}^{n-1} f(x_i) + 2\sum_{i=3,6,9,\dots}^{n-3} f(x_i) + f(b)\right] \quad (45)$$

2.9 Skrip MATLAB

2.9.1 Fungsi Faktorial

Fungsi faktorial diberikan oleh

$$n! = n(n-1)(n-2)\dots 3.2.1 \quad (46)$$

Membentuk fungsi pada matlabnya

```
function fakt = faktorial(x)%masukkan nilai yang akan dicari faktorialnya pada x
```

nilai faktorial khusus jika n = 0, maka n! = 1,

```
if x == 0;
    fakt = 1;
```

untuk yang lainnya,

```
else
fakt = 1; %nilai awal faktorial(1) adalah 1
for i = 1:x; %definisikan nilai i bergerak dari 1 sampai x
    fakt = fakt*i; %looping nilai faktorial
end
fakt; %menampilkan hasil akhir faktorial
end
end
```

2.9.2 Diferensial Numerik Orde Tinggi

Mula-mula dibuat fungsinya yang menunjukkan output yaitu turunan ke-n pada x untuk fungsi f. Yaitu:

```
function ddx = turunanke(f,n,x)
```

Buat hasil output dari fungsi turunanke adalah desimal dengan nilai dibelakang desimal besar (tidak dibulatkan).

```
format long
```

Dari persamaan (39), sebelumnya perlu dideklarasikan nilai h terlebih dahulu. Nilai h agar ketika n = 1 dan n = besar tidak meledak, maka nilai h berubah-ubah bergantung pada nilai n yang digunakan.

```
h = 10^(-12/n);
```

Dapat dilihat dari persamaan (39) bahwa iterasi dimulai dari $i = 0$, sedangkan untuk MATLAB membaca sebagai matriks, dan tidak ada matriks dengan suku ke-0, maka perlu dideklarasikan terlebih dahulu suku ke-0 dari persamaan (39) dengan menggesernya

$$f^n(x) = \sum_{k=0}^{n}(-1)^{n-k}\frac{n!}{k!(n-k)!}f(x+kh) \quad (39)$$

$$f^n(x) = \frac{1}{h^n}\left(\frac{(-1)^{n-0}(n!)}{0!(n-0)!}f(x) + \sum_{k=1}^{n}(-1)^{n-k}\frac{n!}{k!(n-k)!}f(x+kh)\right) \quad (47)$$

$$f^n(x) = \frac{1}{h^n}((-1)^n f(x) + \sum_{k=1}^{n}(-1)^{n-k}\frac{n!}{k!(n-k)!}f(x+kh)) \quad (48)$$

Sehingga inisiasi untuk melakukan iterasinya adalah

```
k = (-1)^n*f(x);
```

Kasus khusus apabila fungsi itu diturunkan nol kali, maka fungsi itu tidak akan berubah, sehingga;

```
if n == 0;
    ddx = k;
```

Untuk kasus yang lain,

```
else
```

dapat dimulai iterasi karena jumlahan sudah diawali dari 1. Juga dengan memanggil fungsi faktorial yang telah dibentuk sebelumnya, maka didapat

```
for i = 1:n;
    k    =    k+((-1)^(n-i)*faktorial(n)/(faktorial(i)*faktorial(n-
i))*f(x+i*h));
end
ddx = k/h^(n);
end
```

Sehingga Skripnya menjadi

```
function ddx = turunanke(f,n,x)
%----inisialisasi h----
format long
h = 10^(-12/n)     ;
%----inisialisasi hasil turunan----
k = (-1)^n*f(x);
%----dapatkan turunan ke-0----
if n == 0;
    ddx = k;
else
    %----dapatkan turunan ke -n ----
for i = 1:n;
    k    =    k+((-1)^(n-i)*faktorial(n)/(faktorial(i)*faktorial(n-
i))*f(x+i*h));
end
ddx = k/h^(n);
end
```

2.9.3 Fungsi Azimuthal

Fungsi Azimuthal yang diberikan dalam persamaan (18) dapat dibuat skripnya dengan menggunakan MATLAB dengan mendefinisikan fungsinya terlebih dahulu.

```
function solusi_harmonic = Harmonicc(m,x);
```

untuk f bergantung pada x, dimana x sebenarnya adalah φ yang diganti dengan peubah dummy, dan i = $\sqrt{-1}$, maka

```
f = @(x) exp(sqrt(-1)*m*x);%(exp(sqrt(-1)*m*x));
solusi_harmonic = f(x);
```

2.9.4 Fungsi Legendre Terasosiasi

Dalam mendefinisikan fungsi Legendre dalam MATLAB, didefinisikan nama fungsinya terlebih dahulu.

```
function P = Legendre(l,m,x)
```

Dari persamaan (11) terlihat bahwa persamaan Rodrigues untuk fungsi Legendre bergantung pada fungsi $(1-x^2)^{\frac{m}{2}}$ dan $(x^2-1)^l$, sehingga perlu didefinisikan kedua fungsi tersebut terlebih dahulu sebagai fungsi f dan g.

```
f = @(x) (x^2-1)^l;
g = @(x) (1-x^2)^(m/2);
```

Terlihat dari hubungan persamaan (11) dan persamaan (12) bahwa nilai m yang diperbolehkan adalah $-l \leq m \leq l$, dan yang pasti nilai l dan m haruslah bulat. Maka definisikan bahwa nilai l dan m haruslah bulat. Untuk membuktikannya, maka ketika l dan m dibagi dengan 1, maka tidak akan sisa, atau sisanya sama dengan nol. Jika hal itu tidak memenuhi, maka tidak perlu dikerjakan, karena nilai l atau m tidak memenuhi. Sehingga dapat dibuat dalam skrip matlab untuk modulo l dan m terhadap 1 haruslah nol.

```
if mod(l,1) ~= 0;
    return
end
if mod(m,1) ~= 0 ;
    return
end
```

Selanjutnya adalah memastikan bahwa nilai |m| harus kurang dari l. Jika tidak memenuhi, maka juga tidak perlu dikerjakan.

```
if abs(m)>l;
    return
```

Kasus pertama jika |m| kurang dari l, namun lebih dari nol, maka digunakan persamaan(11) dengan memanggil fungsi turunanke dan fungsi faktorial sebelumnya.

```
elseif l>=m && m >= 0;
    P = (-1)^m/(2^l*faktorial(l))*g(x)*turunanke(f,m+l,x);
```

Kasus kedua jika |m| kurang dari l, namun kurang dari nol, maka digunakan persamaan (12) dengan memanggil fungsi turunanke dan fungsi faktorial.

```
elseif -l<=m && m<0;
k = abs(m);
    P          =          (-1)^(m)*faktorial(l-m)/faktorial(l+m)*(-
1)^k/(2^l*faktorial(l))*(g(x))^(k/2)*turunanke(f,k+l,x);
```

```
End
```

Sehingga Skripnya menjadi

```
function P = Legendre(l,m,x)
%----inisiasi fungsi yang diperlukan----
f = @(x) (x^2-1)^l;
g = @(x) (1-x^2)^(m/2);
%----batasi nilai l dan m----
if mod(l,1) ~= 0;
    return
end
if mod(m,1) ~= 0 ;
    return
end
if abs(m)>l;
    return
    %----dapatkan hasil polinom Legendre untuk m positif----
elseif l>=m && m >= 0;
    P = (-1)^m/(2^l*faktorial(l))*g(x)*turunanke(f,m+l,x);
    %----dapatkan hasil polinom Legendre untuk m negatif----
elseif -l<=m && m<0;
    P = faktorial(l-m)/faktorial(l+m)*1/(2^l*faktorial(l))*g(x)*turunanke(f,m+l,x);
end
```

2.9.4 Interpolasi Lagrange

Mula-mula didefinisikan fungsi interpolasi Lagrangenya.

```
function L = Lagrange(x,y,x1)
```

Dimana, input dari Lagrange haruslah data x, data y, dan nilai x1, yaitu nilai yang diinginkan. Sehingga, jumlah data dari x dan y yang dimasukkan kedalam input haruslah sama. Input data x dan y dibuat sedemikian ruma menjadi matriks baris.

```
n = length(x);
m = length(y);
if n ~= m;
    return;
end
```

Kemudian, sebelum mencari nilai hasil interpolasinya, maka perlu dicari nilai L_i untuk masing-masing suku interpolasi. Dengan inisialisasi perkalian dan syarat bahwa yang dilakukan interpolasi tidak boleh sama dengan i, maka

```
kali = 1;
for j = 1:n;
    if j~=i;
        kali = kali*(x1-x(j))/(x(i)-x(j));
    end
end
```

Sehingga untuk mencari interpolasinya, perlu dimasukkan dalam persamaan (42), didapat

```
yi = 0;
for i = 1:n;
    kali = 1;
    for j = 1:n;
        if j~=i;
            kali = kali*(x1-x(j))/(x(i)-x(j));
        end
    end
    yi = yi + kali*y(i);
end
```

dan hasil interpolasinya

```
L = yi
```

Sehingga fungsi Interpolasinya adalah

```
function L = Lagrange(x,y,x1)
%----ukur banyak data----
n = length(x);
m = length(y);
%----wajibkan banyak data x dan y sama----
if n ~= m;
    return;
end
%----mulai mencari hasil interpolasinya----
yi = 0;
for i = 1:n;
    %----mulai mencari koefisien interpolasinya----
    kali = 1;
    for j = 1:n;
        if j~=i;
            kali = kali*(x1-x(j))/(x(i)-x(j));
        end
    end
    %----dapatkan koefisien interpolasinuya----
    yi = yi + kali*y(i);
end
%----dapatkan hasil interpolasinya----
L = yi
```

2.9.5 Fungsi Laguerre Terasosiasi

Pertama definisikan fungsinya.

```
function Ass_Laguerre = Laguerre(n,k,x);
```

Fungsi-fungsi lain yang bergantung pada x yaitu $e^{-x}x^{n+k}$ dan $e^{x}x^{-k}$ didefinisikan terlebih dahulu.

```
f = @(x) (exp(-x)*x^(n+k));
u = @(x) (exp(x)*x^(-k));
```

nilai n dan k haruslah bulat dan lebih dari -1.

```
if mod(n,1) ~= 0;
    return
end
if mod(k,1) ~= 0 ;
    return
end
if n<0;
    return
end
if k<0;
    return
end
```

Sehingga solusinya dapat diiterasikan dengan menggunakan persamaan (15).

```
Ass_Laguerre = u(x)/faktorial(n)*turunanke(f,n,x);
```

Terdapat kasus khusus dimana nilai Ass_Laguerre(x=0) = inf. Sedangkan limit kanan maupun kirinya ada, sehingga nilai pada x = 0 dapat dicari melalui interpolasi dari nilai fungsi pada titik-titik lain disekitar nol dari kiri maupun kanan untuk selain nol.

Mula-mulai definisikan fungsi selain nolnya.

```
function h = kecualinol(x);
g = 0.1
m = -x:g:-1e-2;
N = length(m);
n = 1e-2:g:x
for i = 1:N;
    h(i) = m(i);
end
for i = N+1:2*N;
    h(i) = n(i-N);
end
```

Kemudian definisikan fungsilain yang merupakan bentuk yang sama dengan fungsi Laguerre. Misalkan fungsinya adalah Laguerre2, untuk mencari nilai pada x = 0.

```
function Ass_Laguerre = Laguerre2(n,k)
f = @(x) (exp(-x)*x^(n+k));
u = @(x) (exp(x)*x^(-k));
x = kecualinol(3);
N = length(x);
for i = 1:N;
Ass_Laguerre(i) = u(x(i))/faktorial(n)*turunanke(f,n,x(i));
```

```
end
```

Sehingga didapat nilai untuk x = 0.

```
if x ==0;
    Ass_Laguerre = Lagrange(kecualinol(3),Laguerre2(n,k),0);
end
```

Fungsi Laguerrenya menjadi

```
function Ass_Laguerre = Laguerre(n,k,x);
%----inisiasi fungsi-fungsi yang dibutuhkan----
f = @(x) (exp(-x)*x^(n+k));
u = @(x) (exp(x)*x^(-k));
%----batasi nilai n dan k----
if mod(n,1) ~= 0;
    return
end
if mod(k,1) ~= 0 ;
    return
end
if n<0;
    return
end
if k<0;
    return
end
%----dapatkan nilai Assosiasi Laguerre-----
Ass_Laguerre = u(x)/faktorial(n)*turunanke(f,n,x);
%----dapatkan nilai Assosiasi Laguerre pada x=0----
if x ==0;
    Ass_Laguerre = Lagrange(kecualinol(3),Laguerre2(n,k),0);
end
```

2.9.6 Metode Integral Simpson 3/8

Mula-mula definisikan fungsi yang akan digunakan

```
function int =  simpson8_3_mod(f,a,b)
```

lakukan inisasi untuk banyaknya pembagian dari fungsi yang akan diintegralkan

```
N = 1e3;
```

Buat agar hasil desimalnya banyak dibelakang koma

```
format long;
```

inisiasi nilai h atau Δx pada batas antara a dan b

```
h=(b-a)/N;
```

inisiasi nilai x_i

```
x=a:h:b;
```

pada persamaan (45) terlihat bahwa suku pertama dalam kurung kotak adalah $f(a)$. Maka didefinisikan terlebih dahulu untuk suku pertama.

Awal iterasi didefinisikan nilai D yang akan diiterasi kan

```
D=0;
```

Untuk suku pertama

```
if i==1;
        D=D+f(x(1));
```

Kemudian untuk suku kedua yaitu untuk nilai $i = 4,7,10, \ldots..$ Suku i ini terlihat dia adalah kelipatan 3n+1, sehingga modulo dari i terhadap 3 adalah 1. Sehingga iterasi untuk suku kedua dari kurung kotak adalah

```
elseif mod(i,3)==1;
        D=D+3*(f(x(i)));
```

Kemudian untuk suku ketiga berlaku untuk nilai $i = 2,5,8, \ldots.$ Suku i ini terlihat bahwa dia adalah kelipatan 3n+2, sehingga modulo dari i terhadap 3 adalah 2. Iterasi suku ketiga

```
elseif mod(i,3)==2;
        D = D+3*f(x(i));
```

Untuk suku keempat berlaku untuk nilai $i = 3,6,9, \ldots.$ Suku i ini terlihat bahwa dia adalah kelipatan 3n, sehingga modulo dari i terhadap 3 adalah nol. Iterasi suku keempatnya,

```
elseif mod(i,3)==0;
        D = D+2*f(x(i));
```

Kemudian yang terakhir yaitu untuk suku ke N. Karena tidak tahu apakah pada N modulonya berapa, maka didefinisikan sendiri. Karena ini mengakhiri syarat dalam kurung maka, perlu diakhiri juga.

```
      elseif i == N;
            D = D+f(x(N));
    end
```

Kemudian dilakukan penjumlahan dari semuanya untuk i dari 1 hingga ke N.

```
for i=1:N;
    if i==1;
        D=D+f(x(1));
    elseif mod(i,3)==1;
        D=D+3*(f(x(i)));
    elseif mod(i,3)==2;
        D = D+3*f(x(i));
    elseif mod(i,3)==0;
        D = D+2*f(x(i));
    elseif i == N;
        D = D+f(x(N));
    end
end
```

Definisikan hasil integral Simpson seperti pada persamaan (45) dengan mengali dengan konstanta $\frac{3}{8}h$, maka hasil akhir integral didapat

```
int=3*h*D/8
```

Skrip integral Simpson dari awal hingga akhir dapat ditulis dengan

```
function int =  simpson8_3_mod(f,a,b)
%----inisialisasi N dan h----
N = 1e5;
format long;
h=(b-a)/N;
x=a:h:b;
%----inisialisasi iterasi----
D=0;
for i=1:N;
    %----inisialisasi syarat penjumlahan----
    if i==1;
        D=D+f(x(1));
```

```
    elseif mod(i,3)==1;
        D=D+3*(f(x(i)));
    elseif mod(i,3)==2;
        D = D+3*f(x(i));
    elseif mod(i,3)==0;
        D = D+2*f(x(i));
    elseif i == N;
        D = D+f(x(N));
        %----akhiri syarat jumlahan----
    end
    %----akhiri iterasi----
end
%----dapatkan hasil integral----
int=3*h*D/8
```

2.9.7 Harmonik Bola

Mula-mula definisikan fungsinya

```
function Y32 = harmonikbola(l,m)
```

dapatkan nilai untuk iterasi yaitu azimuthal dan altitude(orbital). Untuk azimuth bergerak dari $0 \rightarrow 2\pi$ dan altitude bergerak dari $-\frac{\pi}{2} \rightarrow \frac{\pi}{2}$.

```
dx = pi/60;
alt = -pi/2:dx:pi/2;
az = 0:dx:2*pi;
```

kemudian buat grid 2-D untuk azimuthal dan altitude

```
[theta, phi] = meshgrid(alt,az);
```

Ukur ukuran matriks alt dan az

```
h = length(alt)
t = length(az);
```

Buat fungsi orbital yang bergantung pada alt, yaitu persamaan (21) dari fungsi Legendre terasosiasi yang telah dibuat sebelumnya.

```
for i = 1:h;
    Pjk(1,i) = Legendre(l,m,cos(alt(i)));
end
```

Hanya saja, ukuran dari fungsi orbital harus seukuran dengan thetha dalam grid 2-D. Sehingga Pjk dikopi sedemikian rupa sebanyak ukuran az.

```
for i = 1:t;
    Plm(i,:) = Pjk(1,:);
end
```

Kemudian diperlukan untuk menormalisasi fungsi orbital maupun fungsi azimuthal dengan menggunakan persamaan yang mirip dengan persamaan (27) tanpa menggunakan nilai $R(r)^2r^2$. Awalnya perlu didefinisikan terlebih dahulu untuk norm fungsi harmonik $|Y_{lm}|^2 = P_m^l(\cos(\theta))^2 sin\theta\ |e^{im\varphi}|^2$

```
Integran_Azimuth = @(x) ((abs(Harmonicc(m,x)))^2);
Integran_Orbital = @(x) ((Legendre(l,m,cos(x)))^2*sin(x));
```

Kemudian dicari koefisien normasilasinya dengan mengintegralkan melalui metoda Simpson yang telah didefinisikan sebelumnya.

```
Integral_Azimuth = simpson8_3_mod(Integran_Azimuth,0,2*pi);
Integral_Orbital = simpson8_3_mod(Integran_Orbital,0,pi);
```

Koefisien normalisasinya dapat ditentukan dengan persamaan (27).

```
C = sqrt(1/(Integral_Azimuth*Integral_Orbital));
```

Sehingga didapatkan fungsi harmonik bola yang ternormalisasi. Mengingat fungsi orbital dan azimuthal sudah berupa matriks grid, maka perkalian dengan koefisiennya juga perkalian suku matriks.

```
Y32 = C.*Plm .* Harmonicc(m,phi)
```

Untuk memplot harmonik bola yang merupakan bentuk dalam koordinat bola, maka perlu diubah ke korrdinat persegi. Mengingat fungsi harmonik bola adalah fungsi kompleks, maka perlu didefinisikan untuk semua real dan imajinernya.

```
[Xm,Ym,Zm] = sph2cart(phi, theta, real(Y32));
[Xn,Yn,Zn] = sph2cart(phi, theta, imag(Y32));
```

Kemudian plot permukaan harmonik bola untuk yang real.

```
surf(Xm,Ym,Zm)
```

Pada saat yang bersamaan plot juga bagian imajinernya.

```
hold on
surf(Xn,Yn,Zn)
```

Berikan judul harmonik bola.

```
title('Y_l^m Harmonik Bola')
```

Skrip MATLAB secara keseluruhan untuk harmonik bola adalah

```
function Y32 = harmonikbola(l,m)
%----inisiasi sudut orbital dan azimuthal----
dx = pi/60;
alt = -pi/2:dx:pi/2;
az = 0:dx:2*pi;
%----inisialisasi grid orbital dan azimuthal----
[theta, phi] = meshgrid(alt,az);
%----ukur alt az----
h = length(alt)
t = length(az);
%----hitung fungsi gelombang orbitalnya----
for i = 1:h;
    Pjk(1,i) = Legendre(l,m,cos(alt(i)));
end
%----kopi hingga berukuran sama dengan grit thetha----
for i = 1:t;
    Plm(i,:) = Pjk(1,:);
end
%----buat integran untuk normalisasi----
Integran_Azimuth = @(x) ((abs(Harmonicc(m,x)))^2);
Integran_Orbital = @(x) ((Legendre(l,m,cos(x)))^2*sin(x));
%----hitung hasil integralnya-----
Integral_Azimuth = simpson8_3_mod(Integran_Azimuth,0,2*pi);
Integral_Orbital = simpson8_3_mod(Integran_Orbital,0,pi);
%----dapatkan koefisien normalisasinya----
C = sqrt(1/(Integral_Azimuth*Integral_Orbital));
%----dapatkan fungsi harmonik bolanya----
Y32 = C.*Plm .* Harmonicc(m,phi);
%---ubah dari bola ke kartesian----
[Xm,Ym,Zm] = sph2cart(phi, theta, real(Y32));
[Xn,Yn,Zn] = sph2cart(phi, theta, imag(Y32));
%---plot permukaan----
surf(Xm,Ym,Zm)
```

```
hold on
surf(Xn,Yn,Zn)
title('Y_l^m Harmonik Bola')
```

2.9.8 Distribusi Probabilitas dan Fungsi Gelombang

Mula-mula definisikan fungsinya.

```
function Probabilitas = Kuantum(n,l,m,A,Z)
```

Buat agar hasilnya desimal dengan ketelitian tinggi.

```
format long
```

Inisialisasikan konstanta-konstanta yang diperlukan dalam teori kuantum. Digunakanlah unit natural.

```
hbar =1%1.0545718e-34; konstanta planck tereduksi
me = 1%9.1093e-31; massa elektron
e =1%1.602e-19; muatan elektron
k = 1%9*1e9; konstanta coulomb
a0 =(hbar^2)/(me*k*e^2*Z);%jari-jari Bohr
```

kemudian untuk mencari koefisien normalisasi digunakan persamaan (27) perlu didefinisikan terlebih dahulu integran-integran yang diperlukan

```
Integran_Azimuth = @(x) ((abs(Harmonicc(m,x)))^2);
Integran_Orbital = @(x) ((Legendre(l,m,cos(x)))^2*sin(x));
Integran_Radial = @(x) (exp(-(x/(n*a0)))*(2*x/(n*a0))^(l)*Laguerre(n-l-1,2*l+1,(2*x)/(n*a0))^2*x^2);
```

Kemudian menghitung integralnya masing masing dengan menggunakan metode simpson 3/8 yang telah dibuat sebelumnya dengan batas seperti pada persamaan (27).

```
Integral_Azimuth = simpson8_3_mod(Integran_Azimuth,0,2*pi)
Integral_Orbital = simpson8_3_mod(Integran_Orbital,0,pi)
Integral_Radial = simpson8_3_mod(Integran_Radial,0,100)
```

Sehingga didapatkan nilai koefisien normalisasinya.

```
Koefisien_normalisasi = sqrt(1/(Integral_Radial*Integral_Azimuth*Integral_Orbital));
```

Kemudian untuk melakukan plotting probabilitas dan fungsi gelombang radial sebagai fungsi (r/a0) didefinisikan nilai r/a0, anggap dari 0 s/d 10.

```
h = 1e-4;
r = 0:h:10;
N = length(r);
```

Probabilitas di setiap titik (P(r)) adalah mengikuti persamaan (28).

```
for i = 1:N;
    probabilitas(i) = Integran_Radial(r(i))*Koefisien_normalisasi^2*Integral_Orbital*Integral_Azimuth;
end
```

Dengan energi kuantisasi

```
ev = -13.6*Z^2/n^2;
```

Kemudian mulai dilakukan plotting. Untuk bagian pertama adalah melakukan ploting Radial Probability.

```
subplot(2,2,1)
plot(r,probabilitas);
title(['Distribusi Probabilitas Elektron pada Kulit ke-
',num2str(n),' orbit ke- ',num2str(l),' dengan orientasi-
',num2str(m)])
xlabel('Jarak dari inti : r/a_{0}','fontsize',12)
ylabel('Probabilitas Radial :
C_{nlm}R_{n,l}^{2}r^{2}','fontsize',12)
annotation('textbox','string',['Kuantisasi Energi =
',num2str(ev),'eV'],'FitBoxToText','on')
```

Kemudian untuk membentuk fungsi gelombang Azimuthal, didefinisikan nilai $\varphi\ dari\ 0 sampai 2\pi$

```
fi = 0:h:2;
F = length(fi);
```

Didapat fungsi gelombang azimuthal.

```
for i = 1:F;
    Azimuth(i) = Harmonicc(m,fi(i)*pi);
end
```

Plotting yang kedua

```
subplot(2,2,2)
plot(fi,Azimuth);title(['Fungsi Gelombang Azimuthal Pada Momen
Magnet - ',num2str(m)])
xlabel('Sudut Azimuthal 0 \leq \phi \leq 2\pi','fontsize',12)
ylabel('Fungsi Gelombang Azimuthal = \Phi_{m}(\phi)
','fontsize',12)
```

Kemudian membentuk fungsi gelombang orbital. Didefinisikan nilai dari θ dari 0 hingga π.

```
thetha = 0:h:1;
T = length(thetha);
```

Didapat fungsi gelombang orbital

```
for i = 1:T;
    Orbital(i) = Legendre(l,m,cos(thetha(i)*pi));
end
```

Plotting yang ketiga

```
subplot(2,2,3)
plot(thetha,Orbital);
title(['Fungsi Gelombang Orbital Pada Orbit ke- ',num2str(l)])
xlabel('Sudut Orbital 0 \leq \theta \leq \pi','fontsize',12)
ylabel('Fungsi Gelombang Orbital =
\Theta_{lm}(\theta)','fontsize',12)
```

Kemudian untuk fungsi radial digunakan nilai yang sama seperti plotting radial Probability, sehingga didapat fungsi gelombang radial.

```
for i = 1:N;
    Radial(i) = exp(-
(r(i)/(n*a0)))*(2*r(i)/(n*a0))^(l)*Laguerre(n-l-
1,2*l+1,(2*r(i))/(a0*n));
```

```
end
```

Plotting yang keempat

```
subplot(2,2,4)
plot(r,Radial);
title(['Fungsi Gelombang Radial Pada Kulit ke-',num2str(n)])
xlabel('Jarak dari inti r/a_{0}','fontsize',12)
ylabel('Fungsi Gelombang Radial = R_{nl}(r)','fontsize',12)
```

Kemudian dengan membuka gambar baru, dipanggil fungsi harmonik bola untuk l dan m yang ditentukan

```
figure
harmonikbola(l,m)
```

Keseluruhan skrip MATLABnya menjadi

```
function Probabilitas = Kuantum(n,l,m,A,Z)
%----inisialisasi konstanta----
format long
hbar =1;%1.0545718e-34; konstanta planck tereduksi
me = 1;%9.1093e-31; massa elektron
e =1;%1.602e-19; muatan elektron
k = 1;%9*1e9; konstanta coulomb
a0 =(hbar^2)/(me*k*e^2*Z);%jari-jari Bohr
%---inisialisasi integran normalisasi----
Integran_Azimuth = @(x)  ((abs(Harmonicc(m,x)))^2);
Integran_Orbital = @(x)  ((Legendre(l,m,cos(x)))^2*sin(x));
Integran_Radial                    =                 @(x)              (exp(-
(x/(n*a0)))*(2*x/(n*a0))^(l)*Laguerre(n-l-
1,2*l+1,(2*x)/(n*a0))^2*x^2);
%----hasil integral normalisasi----
Integral_Azimuth = simpson8_3_mod(Integran_Azimuth,0,2*pi);
Integral_Orbital = simpson8_3_mod(Integran_Orbital,0,pi);
Integral_Radial = simpson8_3_mod(Integran_Radial,0,100);
%----dapatkan koefisien normalisasi----
Koefisien_normalisasi                                                    =
sqrt(1/(Integral_Radial*Integral_Azimuth*Integral_Orbital));
%----dapatkan radial probability----
h = 1e-4;
r = 0:h:10;
N = length(r);
for i = 1:N;
    probabilitas(i)                                                      =
Integran_Radial(r(i))*Koefisien_normalisasi^2*Integral_Orbital*Int
egral_Azimuth;
end
ev = -13.6*Z^2/n^2;
%----plotting radial probability----
subplot(2,2,1)
plot(r,probabilitas);
title(['Distribusi   Probabilitas   Elektron   pada   Kulit   ke-
',num2str(n),'   orbit   ke-   ',num2str(l),'   dengan   orientasi-
',num2str(m)])
xlabel('Jarak dari inti : r/a_{0}','fontsize',12)
ylabel('Probabilitas                    Radial                    :
C_{nlm}R_{n,l}^{2}r^{2}','fontsize',12)
annotation('textbox','string',['Kuantisasi         Energi         =
',num2str(ev),'eV'],'FitBoxToText','on')
```

```
%----plotting fungsi gelombang azimuthal----
fi = 0:h:2;
F = length(fi);
for i = 1:F;
    Azimuth(i) = Harmonicc(m,fi(i)*pi);
end
subplot(2,2,2)
plot(fi,Azimuth);title(['Fungsi  Gelombang  Azimuthal  Pada  Momen
Magnet - ',num2str(m)])
xlabel('Sudut Azimuthal 0 \leq \phi \leq 2\pi','fontsize',12)
ylabel('Fungsi     Gelombang     Azimuthal     =     \Phi_{m}(\phi)
','fontsize',12)
%----plotting fungsi gelombang orbital----
thetha = 0:h:1;
T = length(thetha);
for i = 1:T;
    Orbital(i) = Legendre(l,m,cos(thetha(i)*pi));
end
subplot(2,2,3)
plot(thetha,Orbital);
title(['Fungsi Gelombang Orbital Pada Orbit ke- ',num2str(l)])
xlabel('Sudut Orbital 0 \leq \theta \leq \pi','fontsize',12)
ylabel('Fungsi            Gelombang            Orbital            =
\Theta_{lm}(\theta)','fontsize',12)
%----plotting fungsi gelombang radial----
for i = 1:N;
    Radial(i)                        =                        exp(-
(r(i)/(n*a0)))*(2*r(i)/(n*a0))^(l)*Laguerre(n-l-
1,2*l+1,(2*r(i))/(a0*n));
end
subplot(2,2,4)
plot(r,Radial);
title(['Fungsi Gelombang Radial Pada Kulit ke-',num2str(n)])
xlabel('Jarak dari inti r/a_{0}','fontsize',12)
ylabel('Fungsi Gelombang Radial = R_{nl}(r)','fontsize',12)
%----harmonik bola----
figure
harmonikbola(l,m)
```

BAB III

METODOLOGI PENELITIAN

3.1 Jenis Penelitian

Jenis penelitian yang digunakan dalam penelitian ini adalah komputasi dan metoda numerik, yaitu untuk mengetahui bentuk fungsi gelombang atom berelektron tunggal, distribusi probabilitas menemukan elektron pada kulit, dan bentuk harmonik bola fungsi gelombang anguler secara numerik.

3.2 Peralatan yang Digunakan

Dalam melakukan penelitian ini digunakan software berupa MATLAB dan Microsoft Word.

3.3 Lokasi dan Waktu Penelitian

Lokasi penelitian adalah di Departemen Fisika ITS dan di kontrakan. Waktu penelitian dimulai pada November 2018.

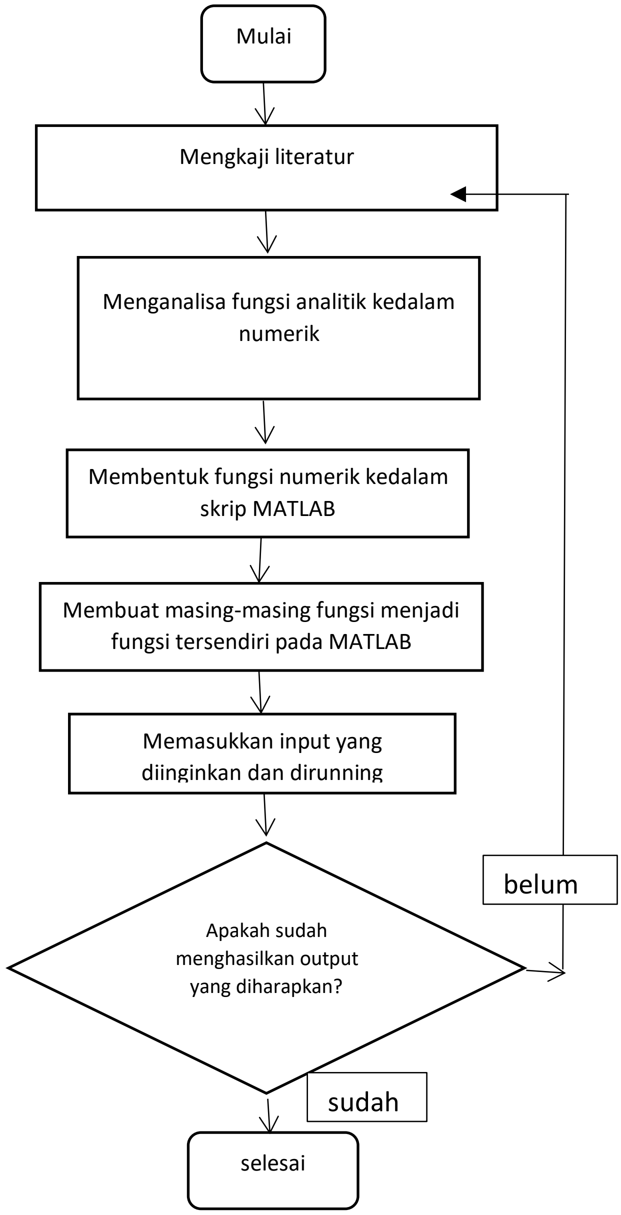
3.4 Prosedur Penelitian

Adapun langkah-langkah yang akan digunakan peneliti dalam mencapai tujuan penelitian ini adalah:

1. Mencari dan mengumpulkan literatur untuk dikaji dan disitasi.
2. Membuat fungsi-fungsi analitik ke dalam metode numerik.
3. Membuat skrip matlab dari metoda yang perlu digunakan dalam analisis. Dalam penelitian ini digunakan turunan numerik orde tinggi, interpolasi Lagrange, Integral Simpson 3/8.
4. Membuat fungsi-fungsi dalam matlab secara terpisah agar dapat digunakan terus menerus hanya dengan mengubah input yang diinginkan.
5. Melakukan running fungsi utama yang telah dibuat sebelumnya.
6. Didapatkan hasil berupa plot distribusi probabilitas, fungsi gelombang dan harmonik bola.

3.5 Diagram Alir

Berikut adalah diagram alir dalam melakukan penelitian:

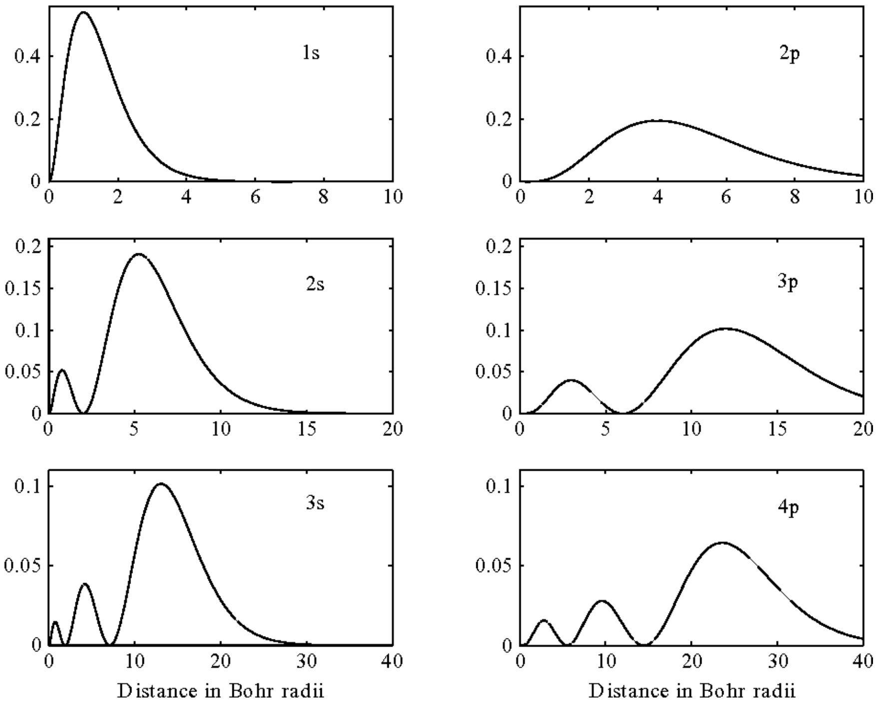


BAB IV

HASIL DAN PEMBAHASAN

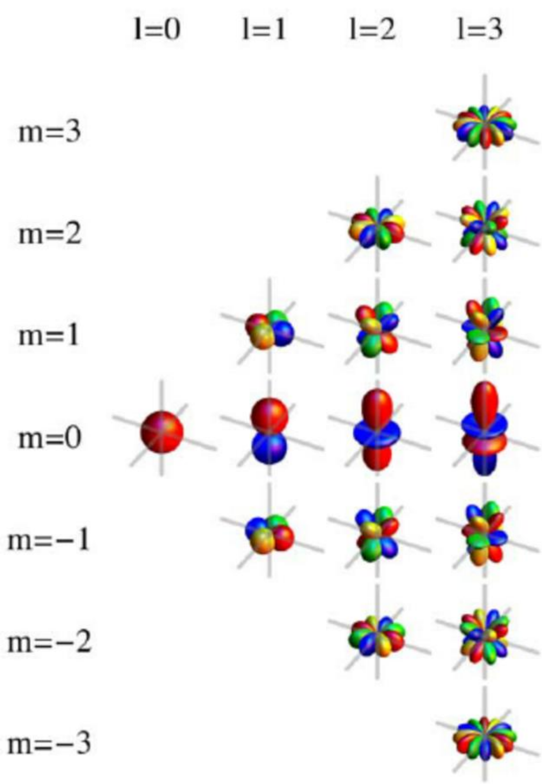
4.1 Hasil

Berikut adalah bentuk radial probability untuk atom hydrogen dari referensi.



Gambar 4.1 Distribusi radial probability untuk atom hidrogen menurut referensi

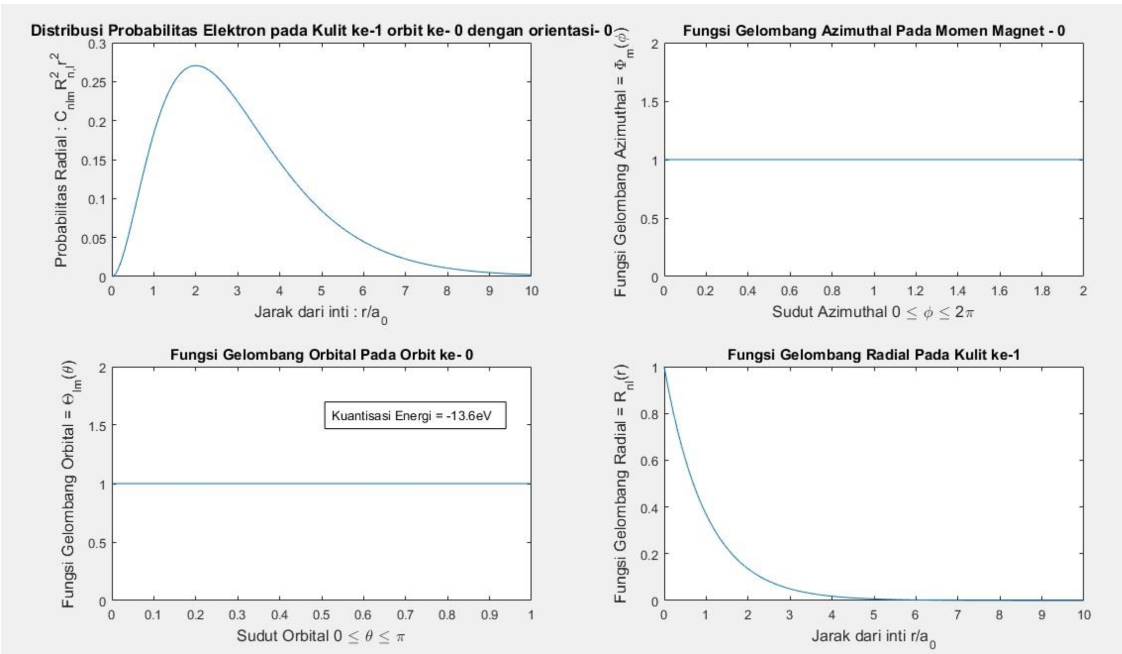
Berikut adalah harmonik bola dari referensi



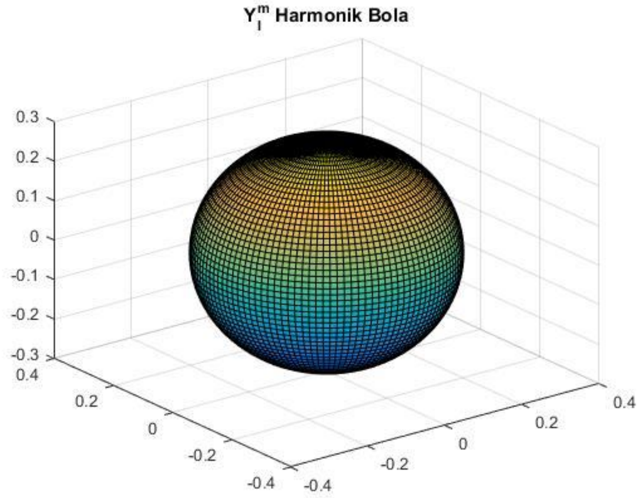
Gambar 4.2 Harmonik bola kompleks menurut referensi

Sedangkan dari hasil analisis dengan skrip MATLAB yang telah dibuat, untuk atom hidrogen ${}_{1}^{1}H$ didapat:

1. Untuk kulit pertama (n=1), orbit s (l = 0), dan bilangan magnetik 0, atau $1s^1$ memiliki bentuk:
   Kuantum(1,0,0,1,1)

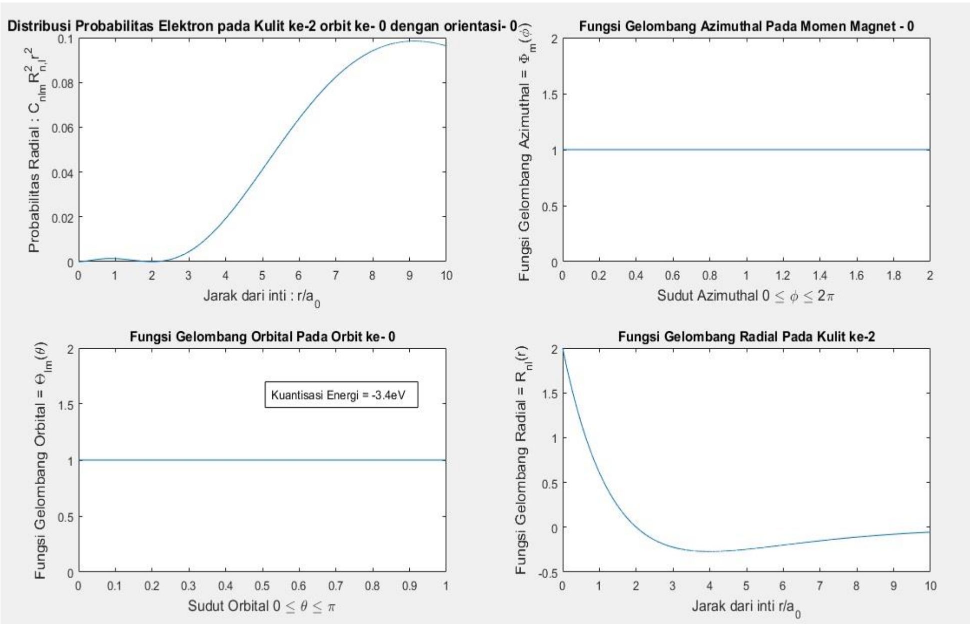


Gambar 4.3 Hasil dari MATLAB untuk ${}_{1}^{1}H\ 1s$

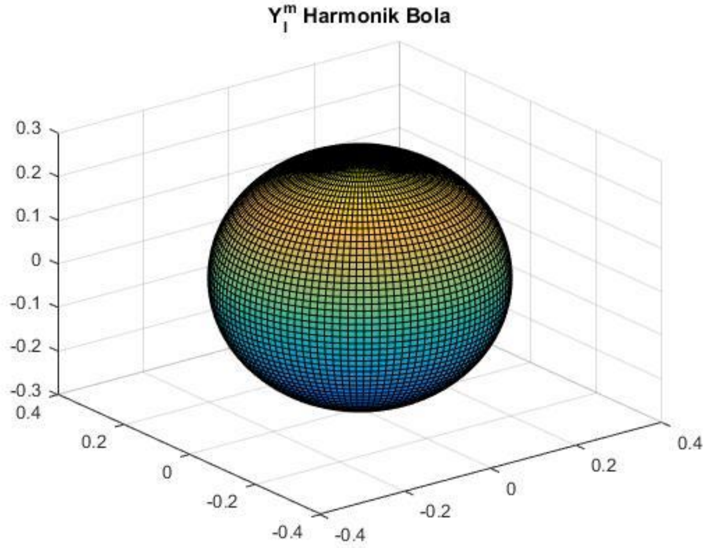


Gambar 4.4 Harmonik Bola Y_0^0

2. Untuk kulit kedua (n=2), orbit s (l = 0), dan bilangan magnetik 0, atau $2s^1$ memiliki bentuk:

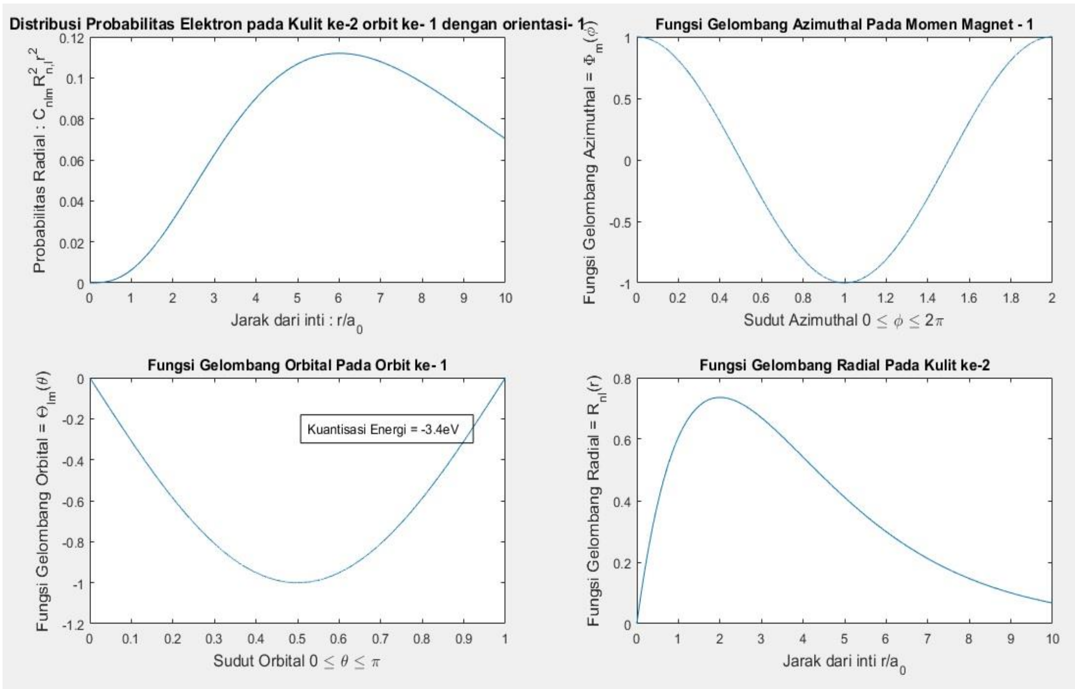


Gambar 4.5 Hasil dari MATLAB untuk ${}_{1}^{1}H\ 2s$

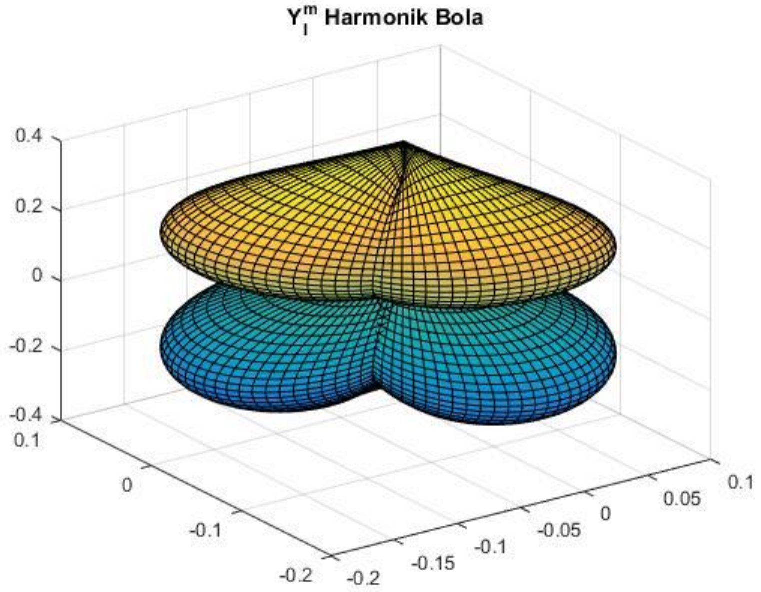


Gambar 4.6 Harmonik bola Y_0^0

3. Untuk kulit kedua (n=2), orbit p (l = 1), dan bilangan magnetik 1, atau $2p^3$ memiliki bentuk:

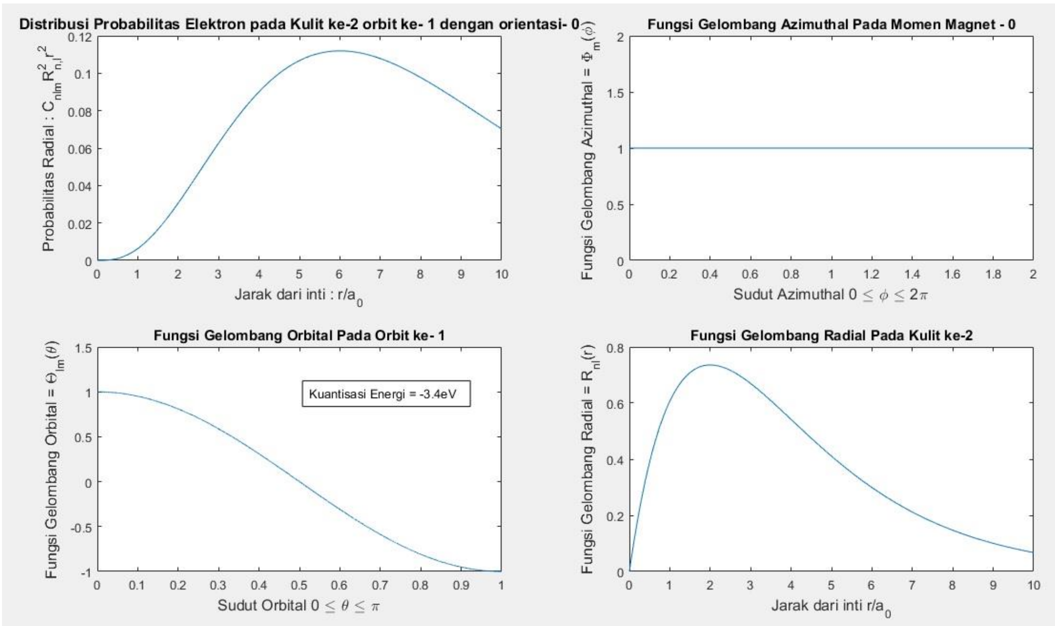


Gambar 4.7 Hasil dari MATLAB untuk ${}^1_1H\ 2p^3$

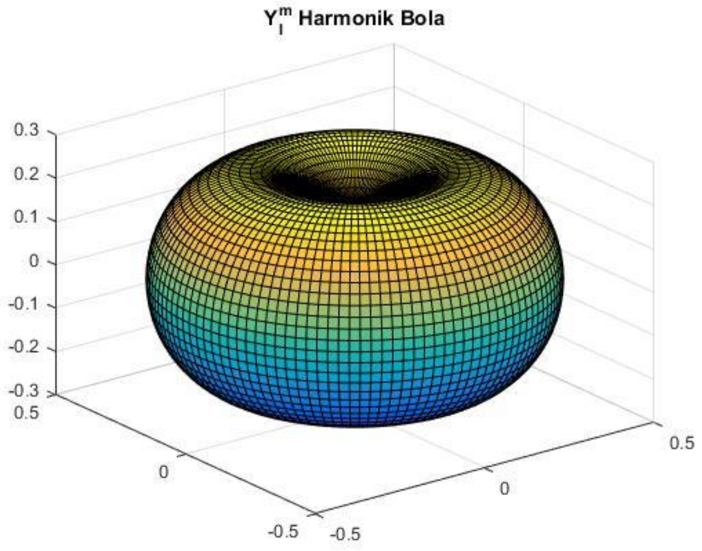


Gambar 4.8 Harmonik bola Y_1^1

4. Untuk kulit kedua (n=2), orbit p (l = 1), dan bilangan magnetik 0, atau $2p^2$ memiliki bentuk:

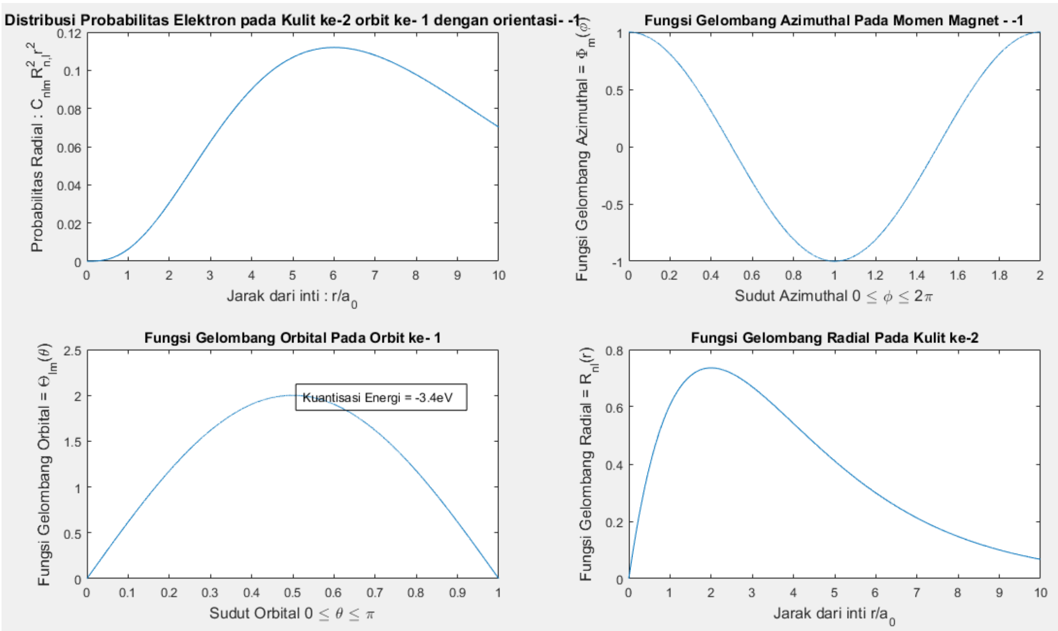


Gambar 4.9 Hasil dari MATLAB ${}^1_1H$ $2p^2$

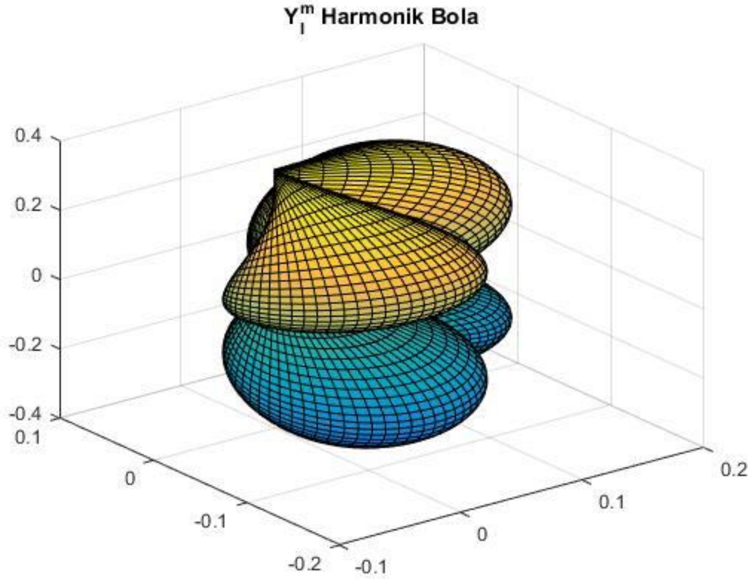


Gambar 4.10 Harmonik Bola Y_1^0

1. Untuk kulit kedua (n=2), orbit p (l = 1), dan bilangan magnetik -1, atau $2p^1$ memiliki bentuk:



Gambar 4.11 Hasil dari MATLAB ${}_1^1H\ 2p^1$



Gambar 4.12 Harmonik bola Y_1^{-1}

4.2 Pembahasan

Dalam perumusan metoda numerik untuk turunan orde tinggi digunakan nilai h yang digunakan berubah-ubah terhadap nilai orde turunan yang diinginkan. Hal ini dilakukan agar tidak perlu memasukkan input berupa nilai h. Selain itu juga untuk mencegah hasil turunan meledak. Nilai h minimum untuk turunan orde tinggi (5 keatas) adalah 10^{-17}. Namun untuk turunan orde rendah (1 atau 2) nilai h minimum adalah sekitaran orde 10^{-13}. Hal ini dapat dilihat bahwa untuk turunan orde tinggi, maka nilai penyebutnya yang merupakan h berpangkat orde turunannya akan sangat kecil, sehingga akan meledak. Misalnya untuk h sama dengan 10^{-4} untuk turunan orde 5 akan menghasilkan penyebut $\frac{1}{10^{-20}}$ sehingga akan mengalikan hasil jumlahan dengan 10^{20}, nilai yang sangat besar sehingga akan membuat hasil meledak. Apabila nilai h dibuat sangat kecil (dalam orde 10^{-1}) maka hasil akan tidak akurat. Maka h yang cocok digunakan adalah berada pada sekitaran batas minimum untuk nilai turunan orde rendah dan orde tinggi, serta dibuat berubah-ubah terhadap orde turunan yang diinginkan.

Penggunaan Interpolasi pada polinomial Laguerre Terasosiasi digunakan untuk kasus untuk x = 0, sehingga menurut persamaan (15) bahwa suku x^{-k} akan bernilai tak hingga. Hal ini akan berbeda apabila suku turunan pada persamaan (15) dikerjakan secara eksak akan menghilangkan x^{k}. Namun hal ini dapat memberikan informasi dimana walaupun terdapat suku x^{k}, ketika dikerjakan numerik pun limitnya tidak akan tak hingga pada x = 0, karena akan ada suku x^{k} dalam turunan. Dalam hal ini, untuk mencari nilai asli fungsi Laguerre Terasosiasi pada x = 0 digunakan interpolasi Lagrange dari data-data interpolasi pada x<0 dan x>0.

Nilai integral_radial yang merupakan subskrip pada fungsi Kuantum seharusnya dilakukan dari 0 hingga ∞. Namun pada kali ini diintegralkan dari 0 hingga 100 saja. Hal ini dilakukan karena selain untuk mempersingkat waktu running, namun juga memperhatikan bahwa solusi fungsi gelombang radial adalah seksponensial sehingga untuk x = 100, nilainya sudah sangat mendekati nol, sehingga sudah cukup untuk mewakili.

Apabila dilihat dari bentuknya, radial probability apabila dibandingkan dengan hasil dari referensi cukup memiliki bentuk yang serupa. Hanya saja nilai sebenarnya memiliki perbedaan yang cukup besar.

Bentuk harmonik bola dari referensi dan pembuatan secara numerik memiliki banyak perbedaan pada nilai nilai tertentu.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Simulasi fungsi gelombang atom, probabilitas radial dan harmonik bola ini dapat digunakan untuk melihat distribusi dan bentuk teori atom mekanika kuantum.

5.2 Saran

Saran yang ada yaitu dilakukan analisis numerik dengan error yang diminimalkan sedemikian sehingga hasil perhitungan dan simulasi secara numerik tidak berbeda terlalu jauh dengan hasil akhirnya

DAFTAR PUSTAKA

Philips, A.C. 2003. *"Introduction to Quantum Mechanics"*. England: John Wiley & Sons.

Wang, Quan, d.k.k. 2011. *" Spherical Harmonics Coefficient for Ligand-Based Virtual Screening of Cyclooxygenase Inhibitor"*. PloS ONE

Purwanto, Agus. 2005. "*Fisika Kuantum*". Yogyakarta: Penerbit Gava Media.

Griffith, David J.. 1995. "*Introduction to Quantum Mechanics"*. New Jersey: Prentice Hall, Inc.

Gasiorowich, Steven. 2003. "*Quantum Physics"*. United States of America: John Wiley & Sons.

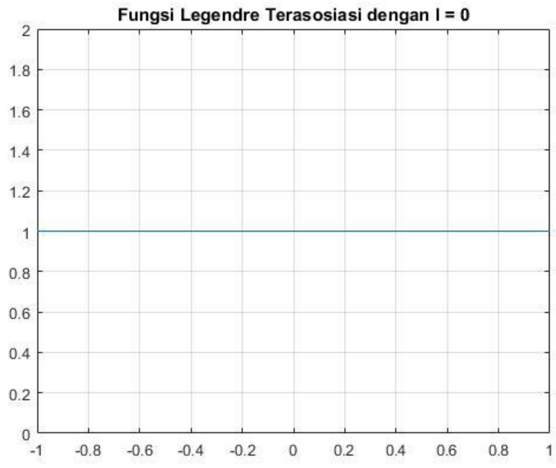
Chapra, Steven C. dan Canale, Raymond P.. 2010."*Numerical Methods for Engineers, Sixth Edition"*. New York: McGraw-Hill Company, Inc.

Kiusalaas, Jaan. 2005. "*Numerical Methods in Engineering with MATLAB"*. United Kingdom: Cambridge University Press.

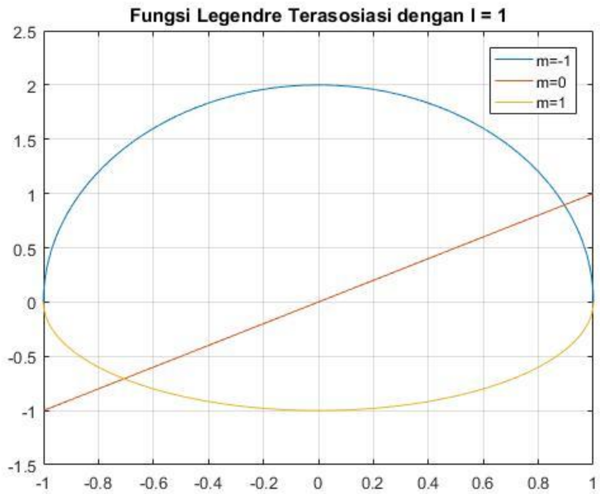
A. LAMPIRAN

1. Fungsi Legendre Terasosiasi

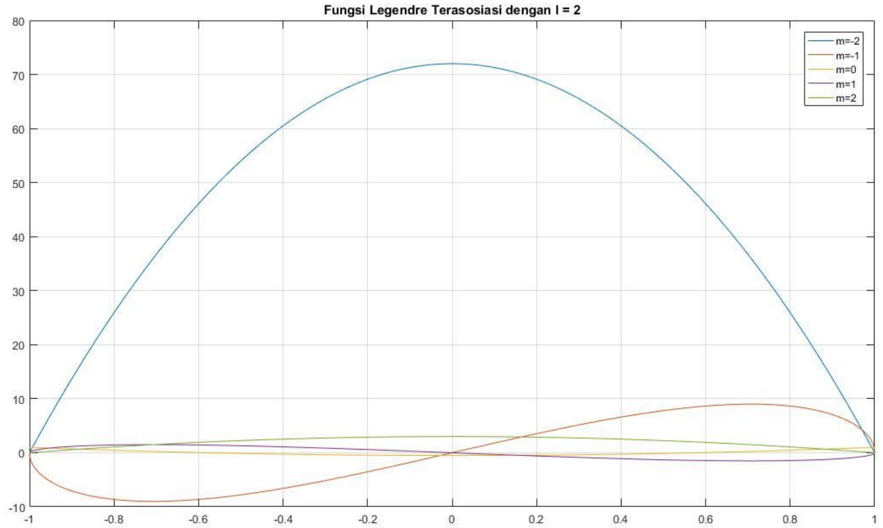
Berikut adalah beberapa hasil fungsi Legendre terasosiasi yang telah dibuat pada subbab 2.8.4



Gambar 1.1 Fungsi Legendre Terasosiasi dengan l = 0



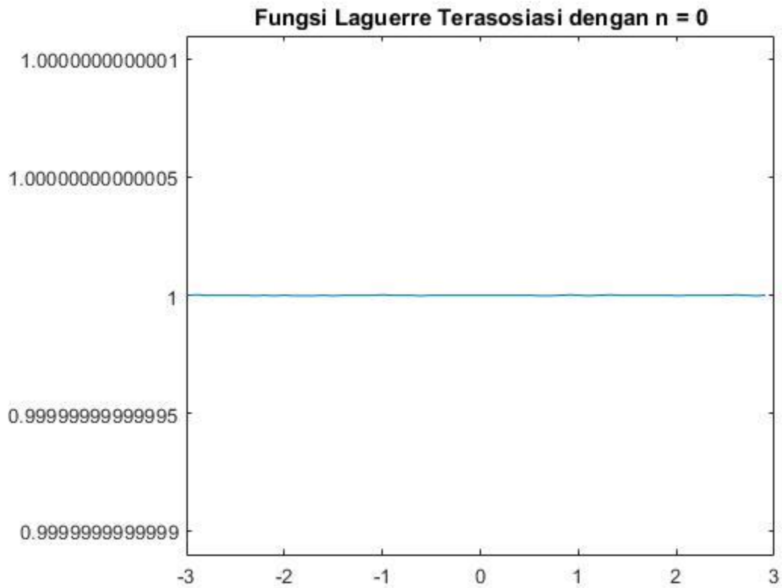
Gambar 1.2 Fungsi Legendre Terasosiasi dengan l = 1



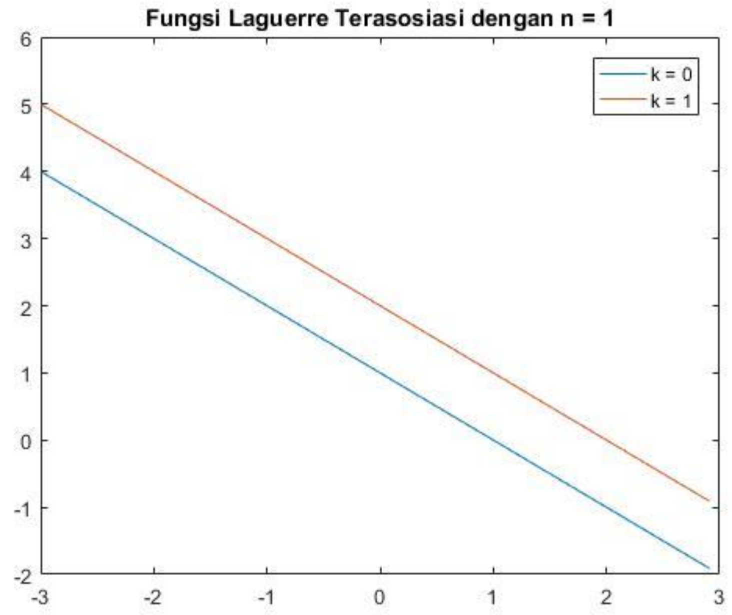
Gambar 1.3Fungsi Legendre dengan l=2

2. Fungsi Laguerre Terasosiasi

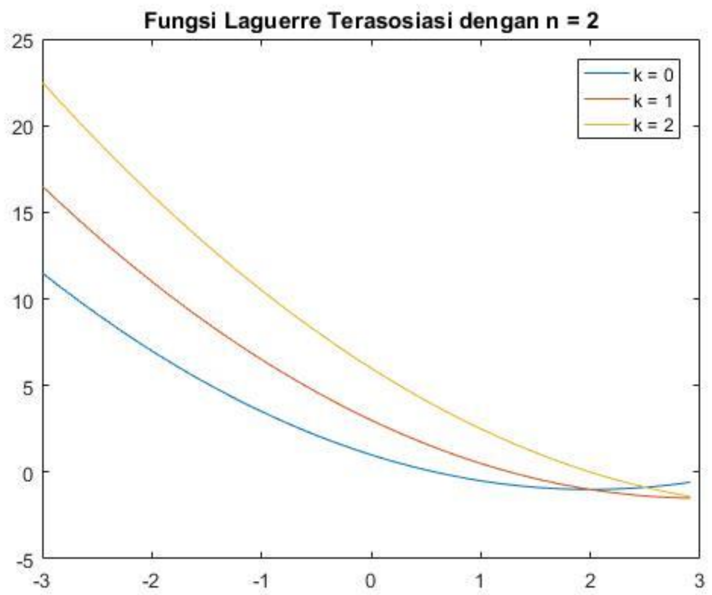
Berikut adalah beberapa hasil untuk Fungsi Laguerre Terasosiasi



Gambar 1.4 Fungsi Laguerre Terasosiasi dengan n = 0’



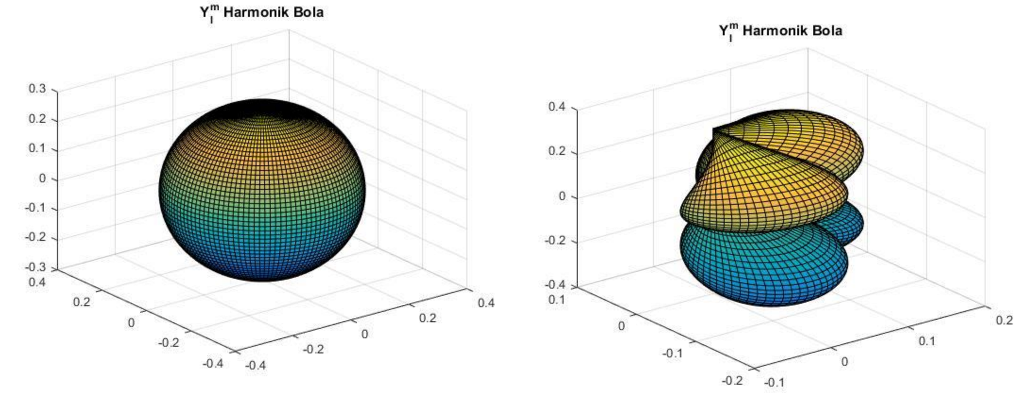
Gambar 1.5 Fungsi Laguerre Terasosiasi dengan n = 1



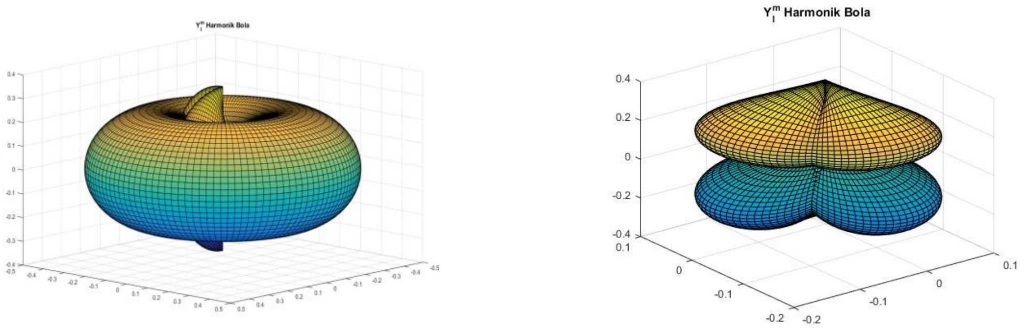
Gambar 1.6 Fungsi Laguerre Terasosiasi dengan n = 2

3. Harmonik Bola

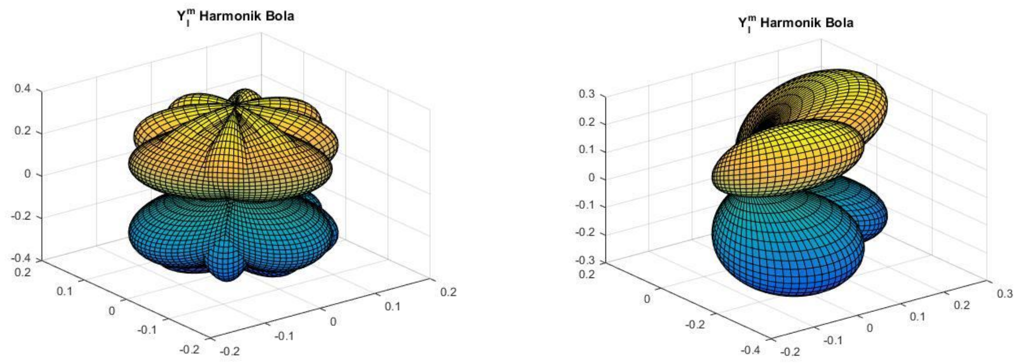
Berikut adalah beberapa hasil untuk harmonik bola.



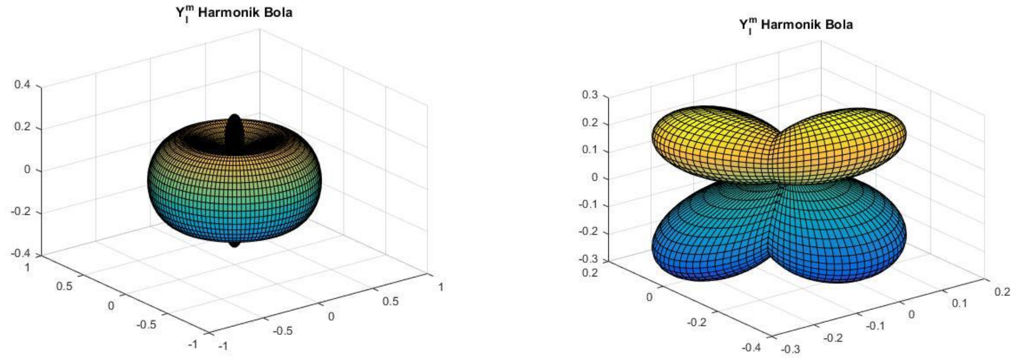
Gambar 1.7 $Y_0^0(kiri), Y_1^{-1}(kanan)$



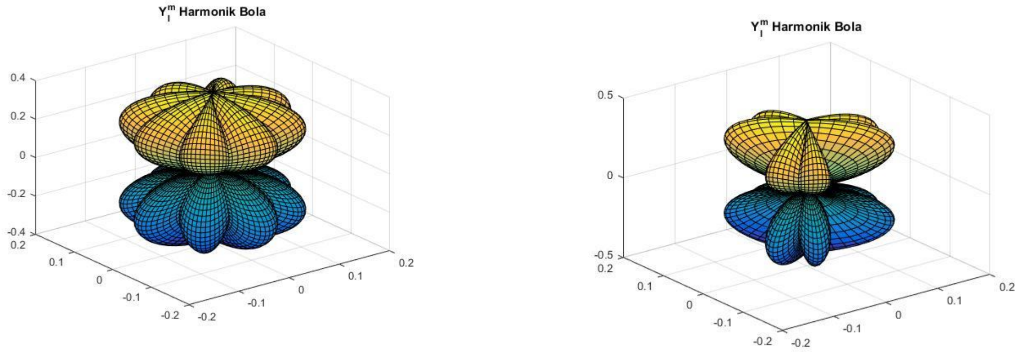
Gambar 1.8 $Y_1^0(kiri), Y_1^1(kanan)$



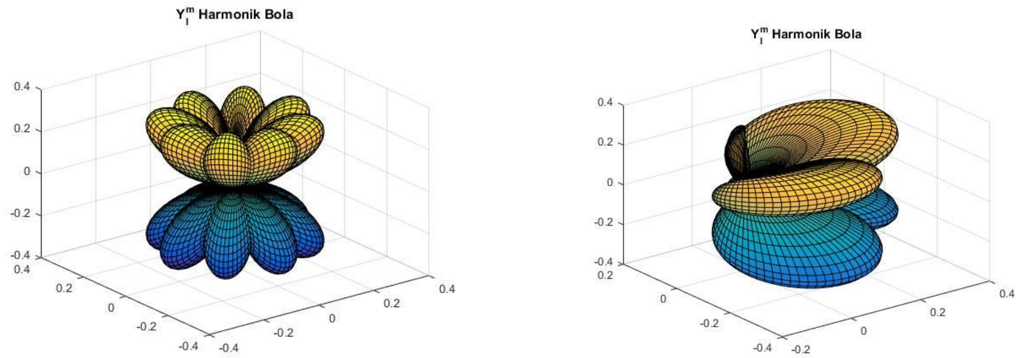
Gambar 1.9 $Y_2^{-2}(kiri), Y_2^{-1}(kanan)$



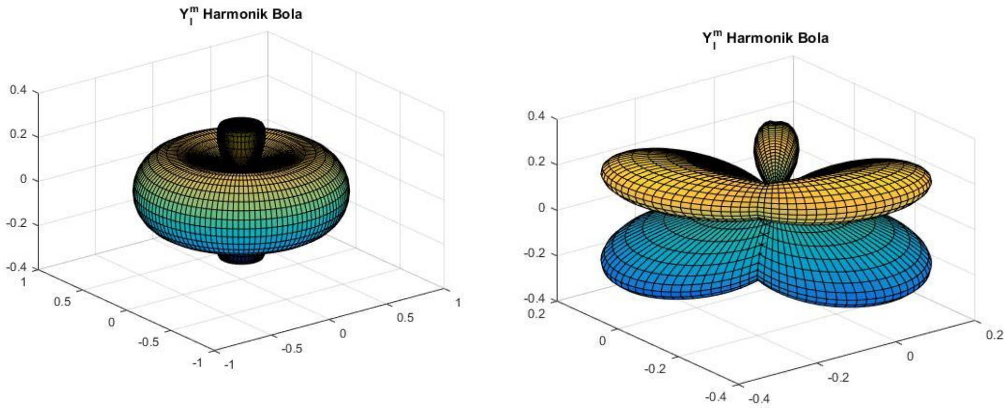
Gambar 1.10 $Y_2^0(kiri), Y_2^1(kanan)$



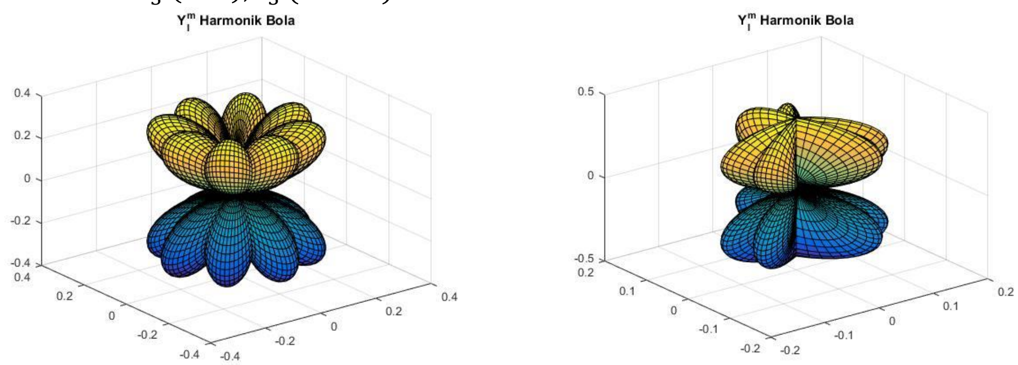
Gambar 1.11 $Y_2^2(kiri), Y_3^{-3}(kanan)$



Gambar 1.12 $Y_3^{-2}(kiri), Y_3^{-1}(kanan)$



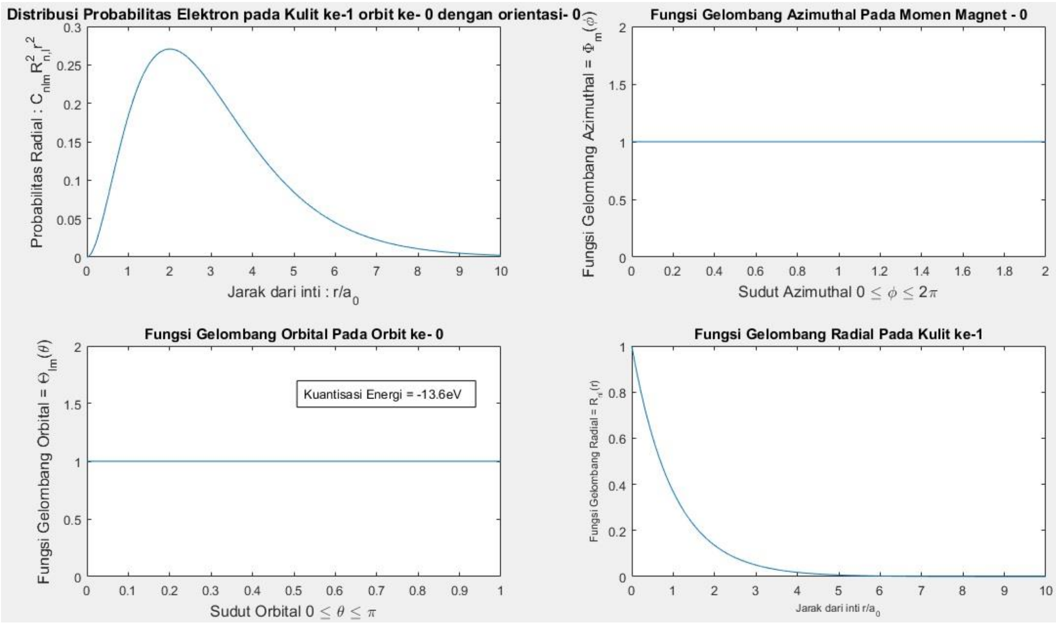
Gambar 1.13 $Y_3^0(kiri), Y_3^1(kanan)$



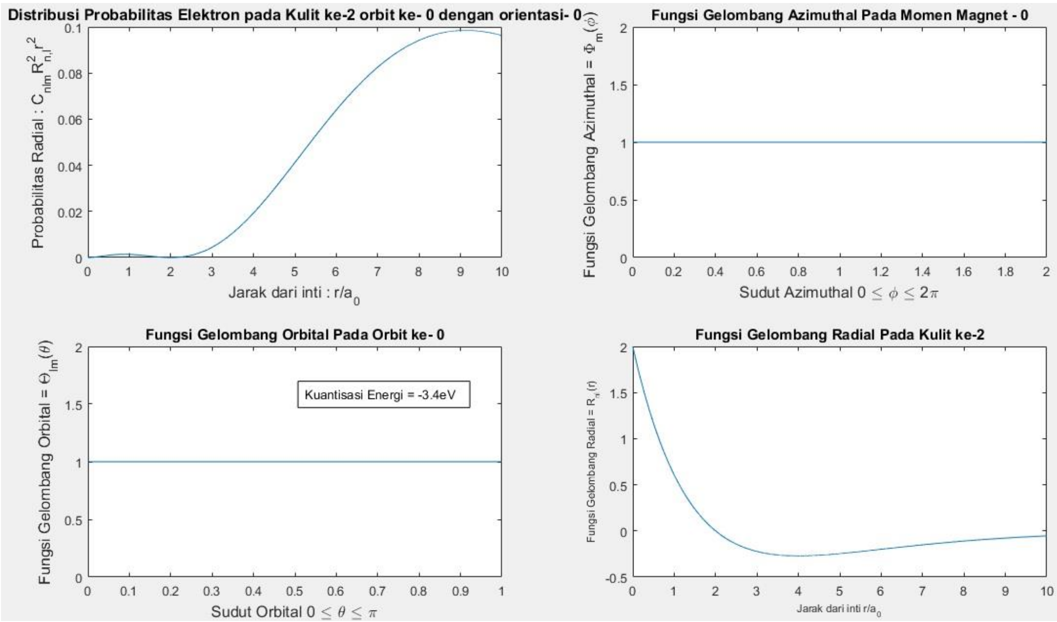
Gambar 1.14 $Y_3^2(kiri), Y_3^3(kanan)$

4. Isotop Atom Hidrogen Deuterium

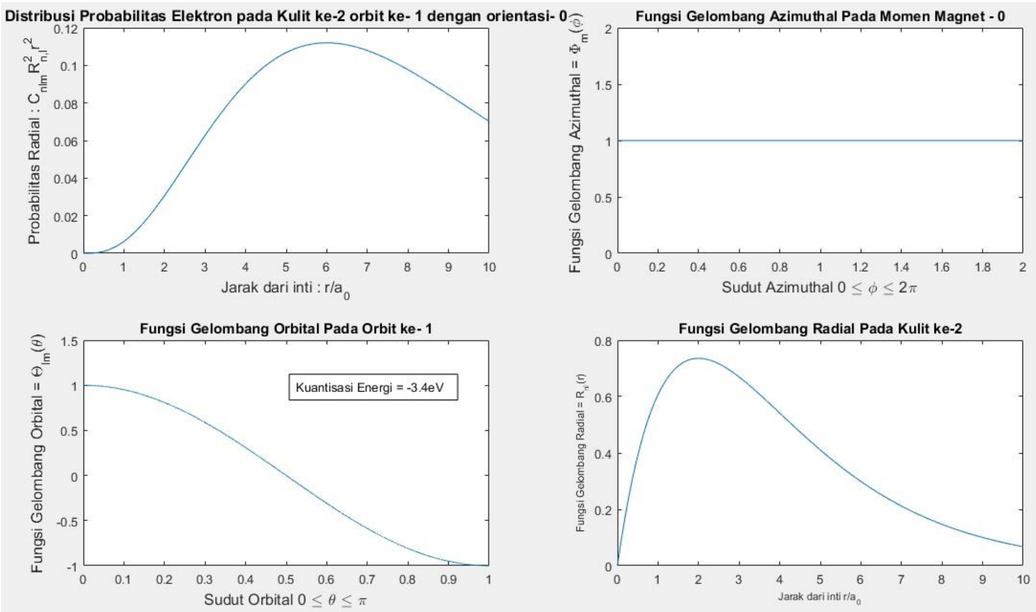
Berikut adalah distribusi fungsi gelombang dan radial probabilitas untuk atom Deuterium



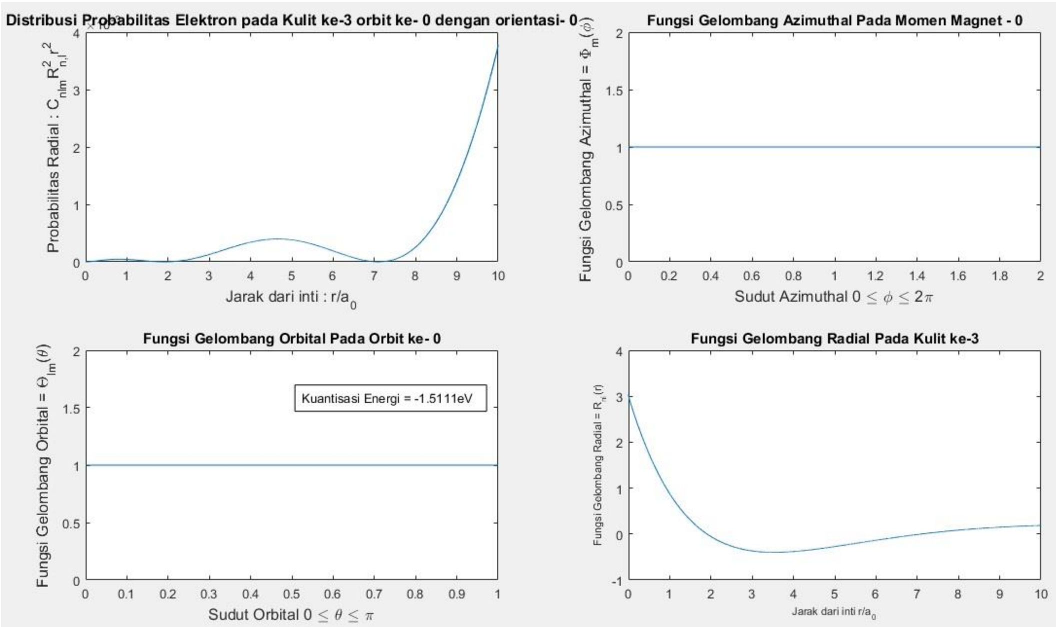
Gambar 1.15 Distribusi fungsi gelombang dan radial probability pada Deuterium ${}_1^2H\ 1s$



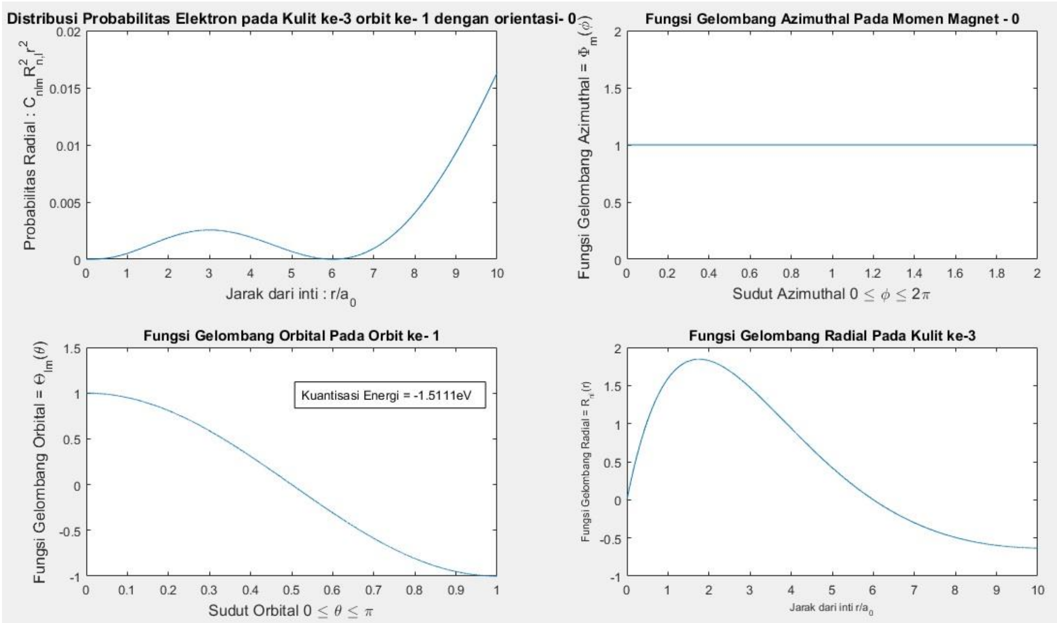
Gambar 1.16 Distribusi fungsi gelombang dan radial probability pada Deuterium ${}_1^2H\ 2s$



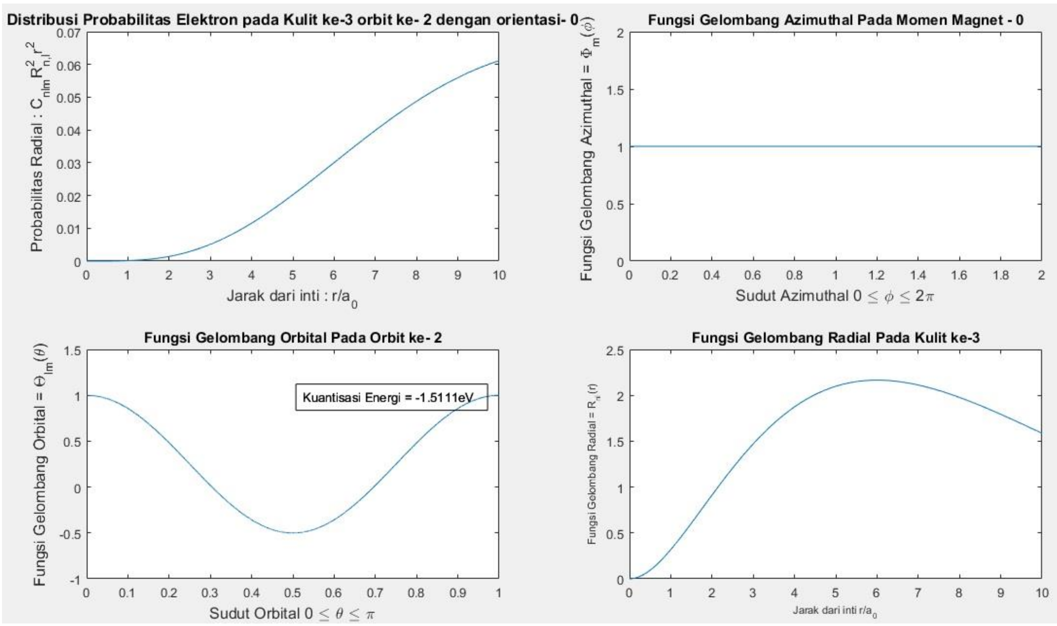
Gambar 1.17 Distribusi fungsi gelombang dan radial probability pada Deuterium ${}_{1}^{2}H$ $2p$



Gambar 1.18 Distribusi fungsi gelombang dan radial probability pada Deuterium ${}_{1}^{2}H$ $3s$



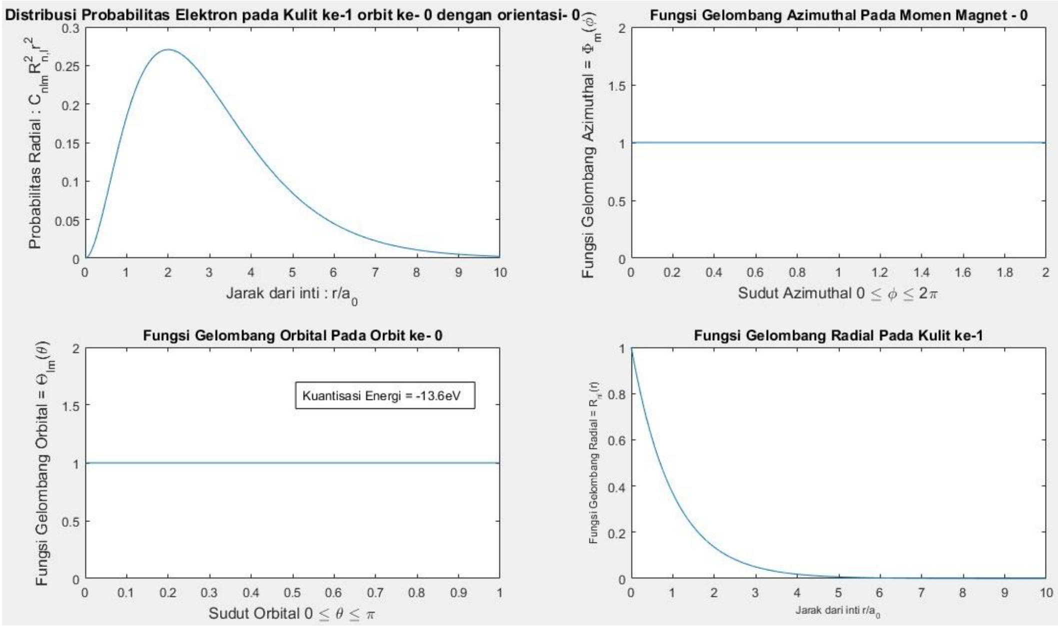
Gambar 1.19 Distribusi fungsi gelombang dan radial probability pada Deuterium ${}_{1}^{2}H\ 3p$



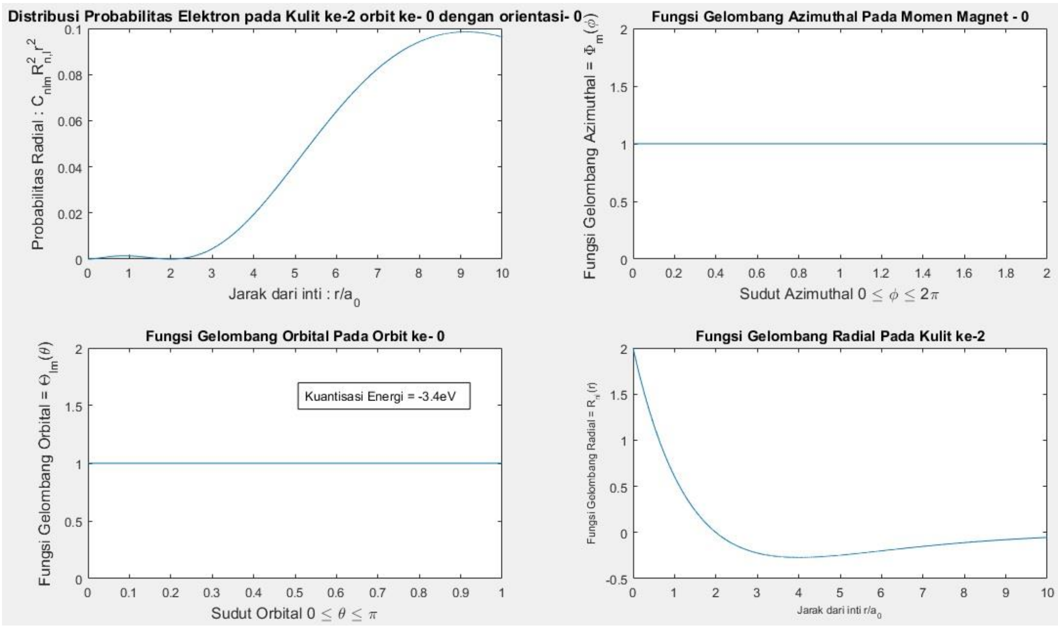
Gambar 1.20 Distribusi fungsi gelombang dan radial probability pada Deuterium ${}_{1}^{3}H\ 3d$

5. Isotop Atom Hidrogen Tritium

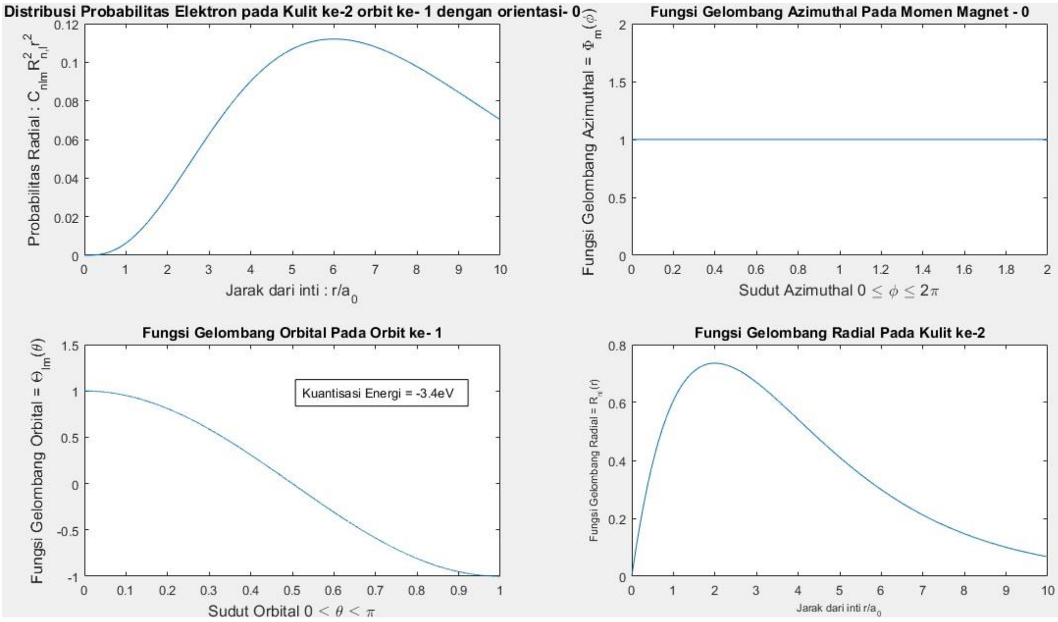
Berikut adalah distribusi fungsi gelombang dan radial probability untuk atom Tritium.



Gambar 1.21 Distribusi fungsi gelombang dan radial probability pada Tritium ${}^3_1H$ $1s$



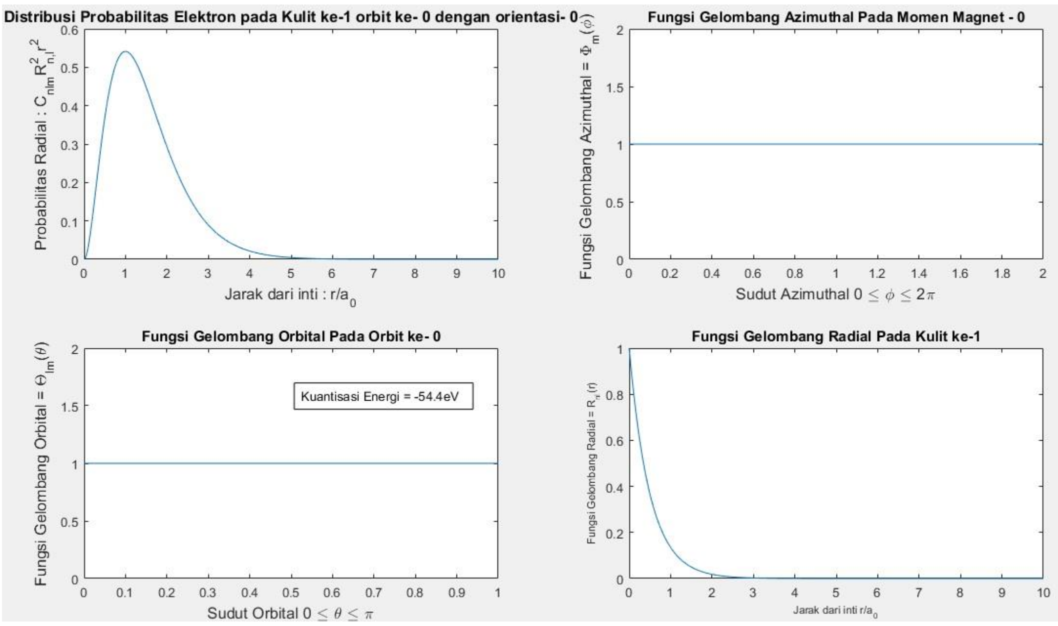
Gambar 1.22 Distribusi fungsi gelombang dan radial probability pada Tritium ${}^3_1H$ $2s$



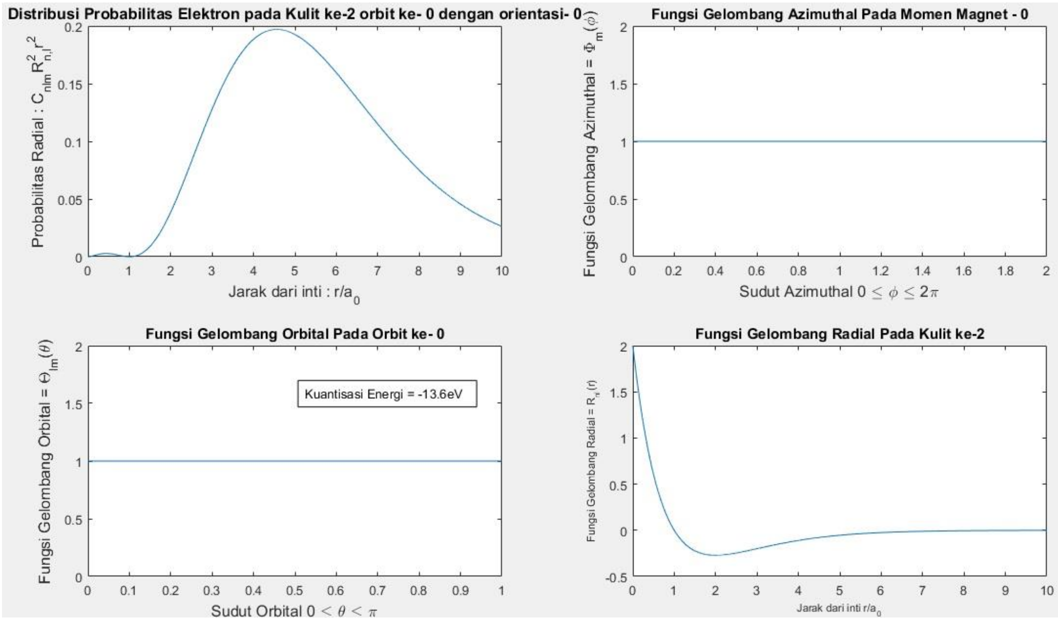
Gambar 1.23 Distribusi fungsi gelombang dan radial probability pada Tritium ${}^{3}_{1}H\ 2p$

6. Atom Helium

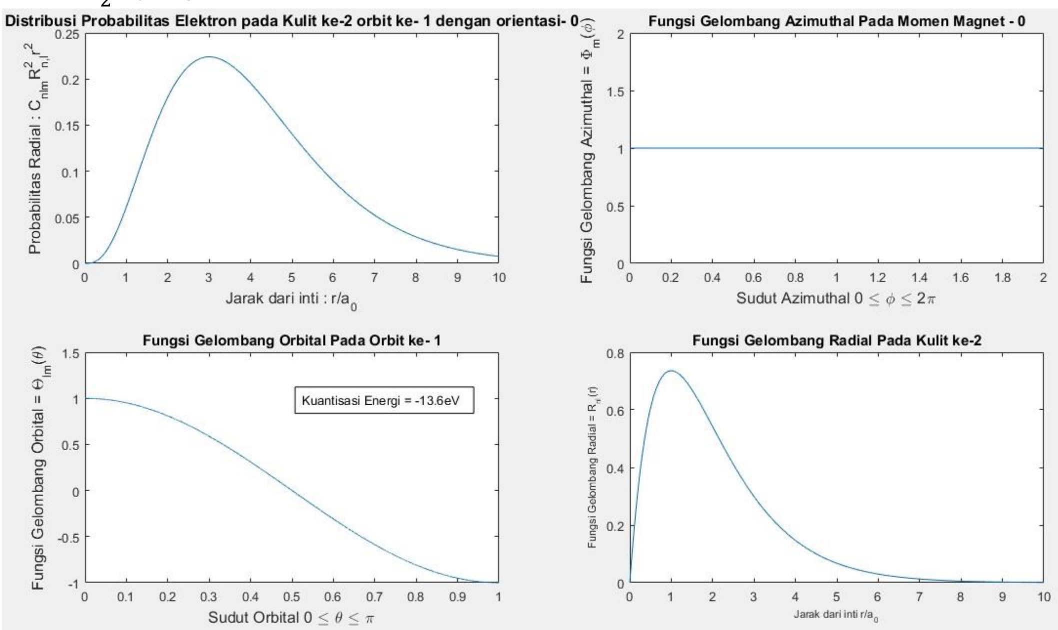
Berikut adalah distribusi fungsi gelombang dan radial probability untuk ion Helium.



Gambar 1.24 Distribusi fungsi gelombang dan radial probability pada helium ${}^{4}_{2}He^{+}1s$



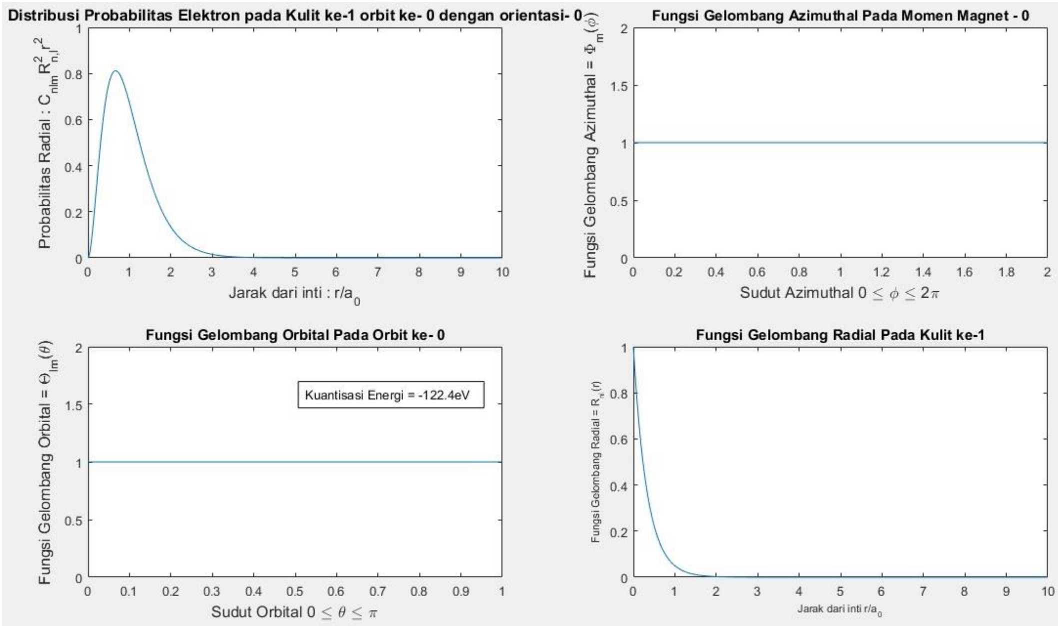
Gambar 1.25 Distribusi fungsi gelombang dan radial probability pada helium ${}^{4}_{2}He^{+}2s$



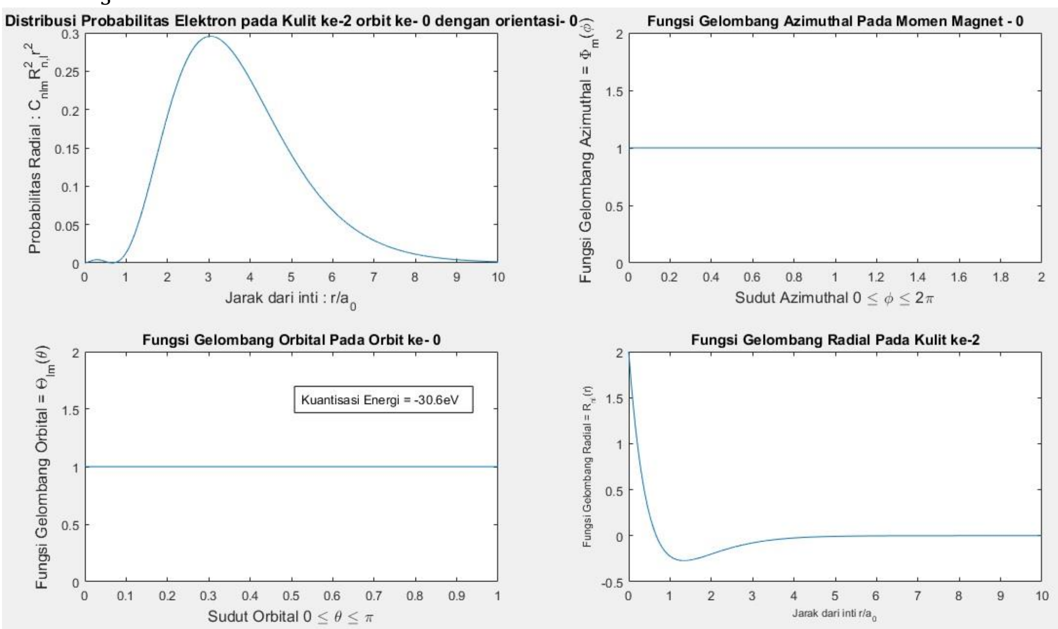
Gambar 1.25 Distribusi fungsi gelombang dan radial probability pada helium ${}^{4}_{2}He^{+}2p$

7. Atom Lithium

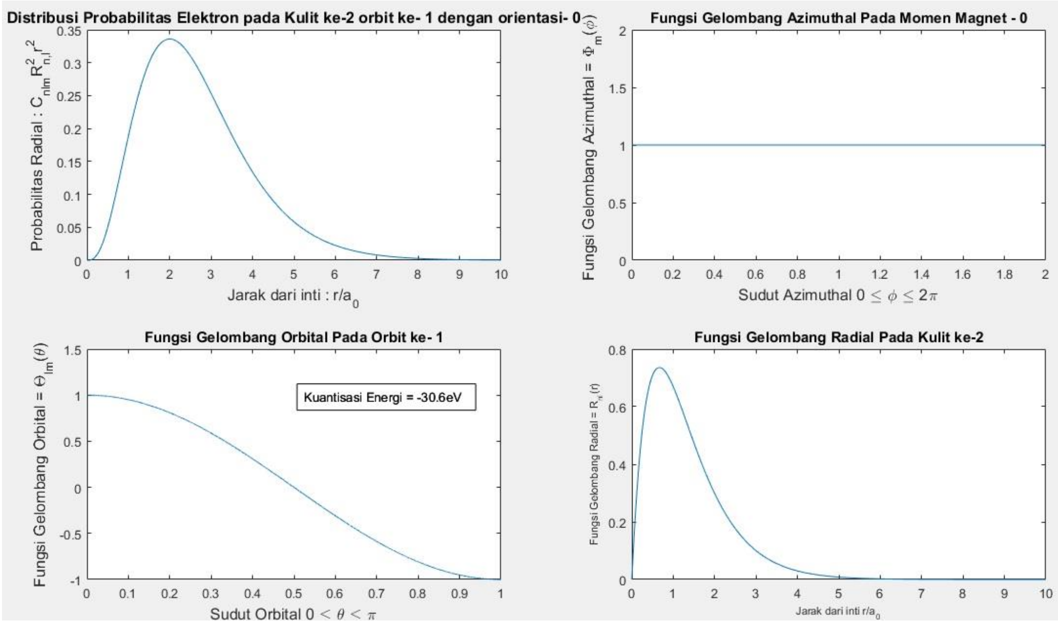
Berikut adalah distribusi fungsi gelombang dan radial probability untuk ion Lithium.



Gambar 1.26 Distribusi fungsi gelombang dan radial probability pada Lithium ${}^{7}_{3}Li^{2+}1s$



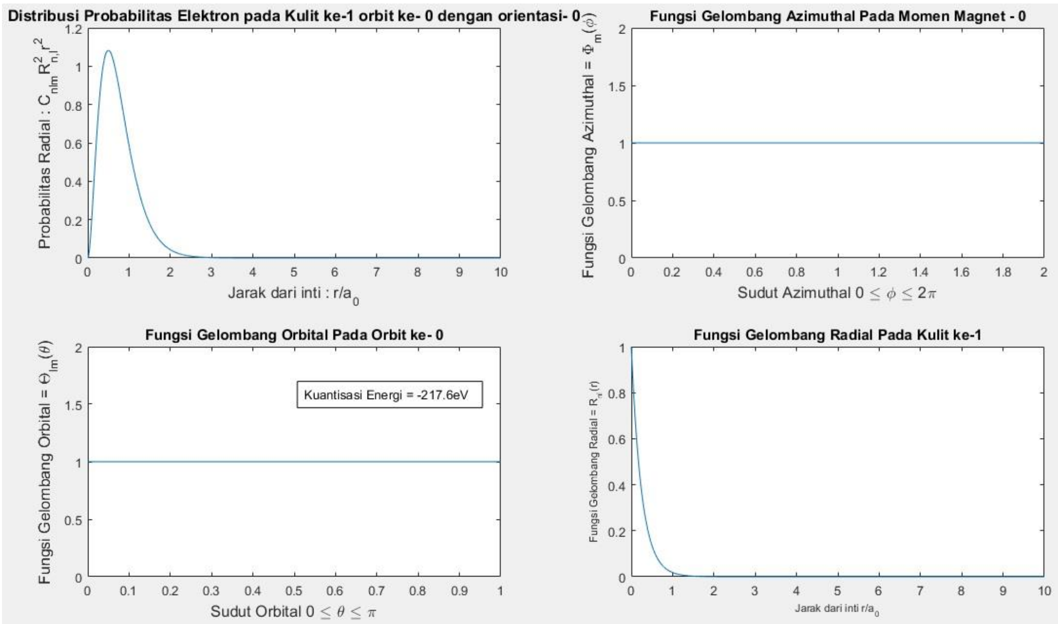
Gambar 1.27 Distribusi fungsi gelombang dan radial probability pada Lithium ${}^{7}_{3}Li^{2+}2s$



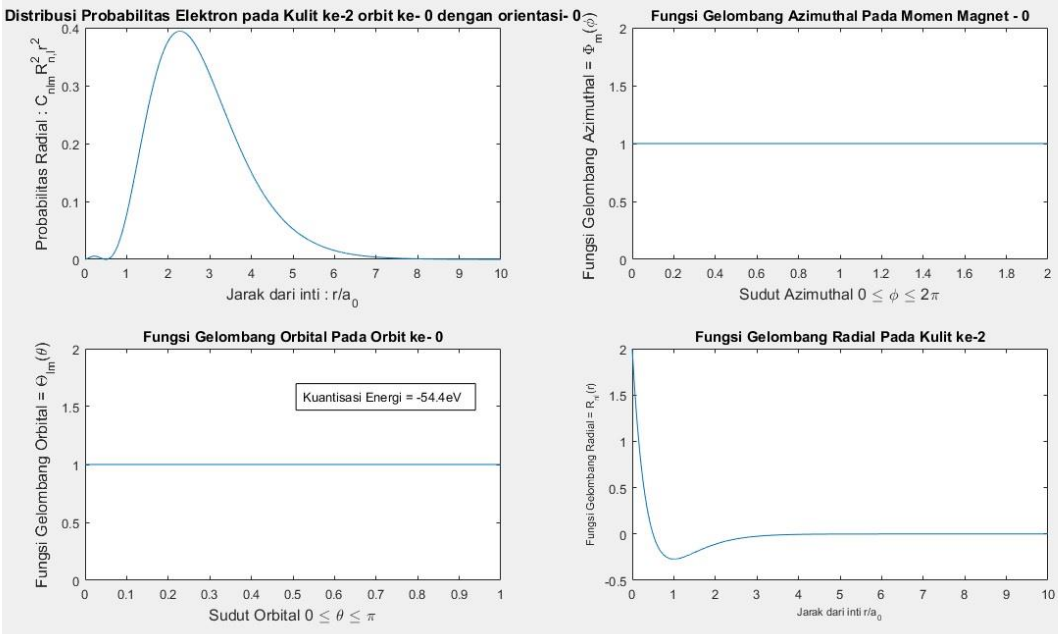
Gambar 1.28 Distribusi fungsi gelombang dan radial probability pada Lithium ${}^{7}_{3}Li^{2+}2p$

8. Atom Berilium

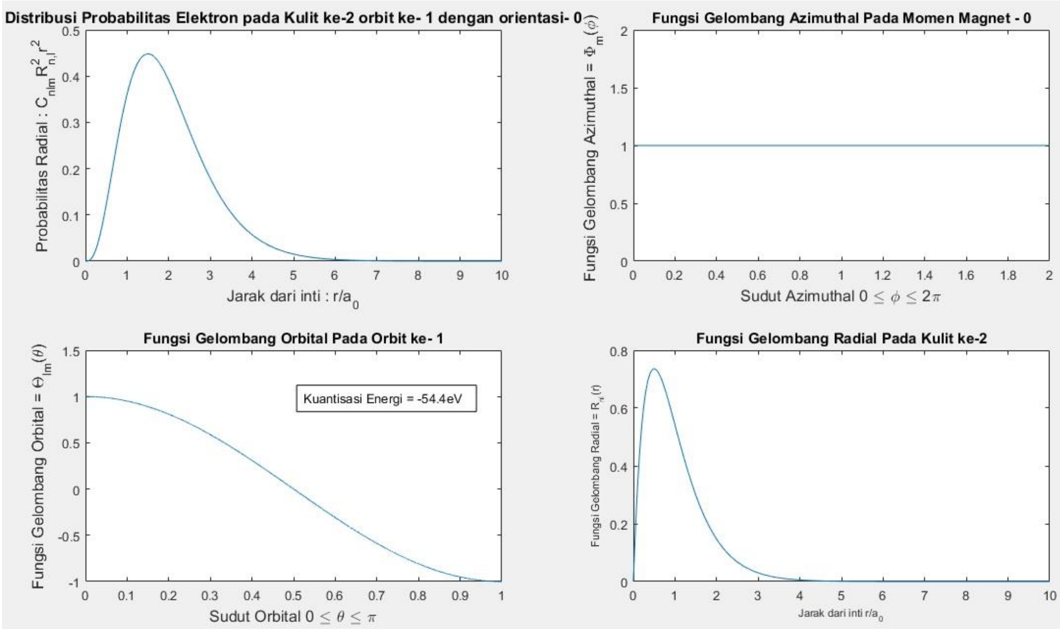
Berikut adalah distribusi fungsi gelombang dan radial probability untuk ion Berilium.



Gambar 1.28 Distribusi fungsi gelombang dan radial probability pada ion Berilium ${}^{9}_{4}Be^{3+}1s$



Gambar 1.29 Distribusi fungsi gelombang dan radial probability pada ion Berilium ${}^{9}_{4}Be^{3+}2s$



Gambar 1.30 Distribusi fungsi gelombang dan radial probability pada ion Berilium ${}^{9}_{4}Be^{3+}2p$