

DEUTERONSIMULERING



29 maj 2019

Gustaf Sjösten
cid: gussjos
19950128-4575

Simon Jacobsson
cid: simjac
19970417-0456

1 Inledning & Teori

Deuteronen är en atomkärna bestående enbart av en proton och en neutron. Det är den enklaste sammansatta kärnan och finns i atomen deuterium. Enligt nya simuleringar (2016, [2]) baserade på en atomär excitation av myoniskt deuterium har man hittat att deuteronens radie är flera standardavvikelser mindre än det tidigare accepterade värdet [1].

I den här laborationen bestämmer vi ett numeriskt värde på rms-radien r_{rms} för deuteronen genom att första lösa Schrödingerekvationen (S.E.)

$$\mathcal{H}\psi(\vec{x}) = (\mathcal{T} + \mathcal{V})\psi(\vec{x}) \quad (1)$$

för deuteronen numeriskt med en så kallad *Malfliet Tjon*-potential

$$\mathcal{V} = V_{\text{MT}} := \sum_{i=1}^3 V_i \frac{e^{-a_i}}{r}. \quad (2)$$

Potentialen är sfäriskt symmetrisk, så vi löser endast för den radiella delen av vågfunktionen $\psi(\vec{r}) \propto R(r) =: u(r)/r$, varefter vi kan bestämma $r_{\text{rms}} = \sqrt{\langle \psi | (r/2)^2 | \psi \rangle}$. I labb-pm [1] ges randvillkor som t.ex. att $V(r)$ avtar asymptotiskt samt att $u(r) \rightarrow 0$ då $r \rightarrow 0$ samt då $r \rightarrow \infty$.

Vidare är vi också intresserade av grundtillståndsennergien E_d för deuteronen (vilken också kommer påverka vår uppskattning av r_{rms}), varför vi kan sätta banrörelsemängdsmomentet $l = 0$ och skriva om S.E. på formen

$$\frac{d^2}{dr^2}u(r) = F(r)u(r), \quad (3)$$

där $F(r) = K(V(r) - E_d)$ med $K = 2\mu/\hbar^2$, där $\mu = \frac{m_p m_n}{m_p + m_n}$ är den reducerade massan för deuteronen och m_p och m_n är massan för protonen respektive neutronen. Differentialekvationen (3) kan lösas numeriskt med *Numerous metod* genom en Taylorutveckling av $u(r)$ kring r . Numeriska värden på vågfunktionen ges till slut via differensrelationen

$$u_1 = \frac{u_0(2 + \frac{5}{6}h^2 F_0) - u_{-1}(1 - \frac{1}{12}h^2 F_{-1})}{1 - \frac{1}{12}h^2 F_1}, \quad (4)$$

där u_0 motsvarar $u(r)$ och u_1, u_{-1} motsvarar $u(r+h), u(r-h)$ för en fix steglängd h .

För att möta randvillkoren i båda ändpunkter med Numerovs algoritm kan vi generera $u(r)$ med (4) både från r_{min} och r_{max} och avsluta båda iterationer i en matchningspunkt r_* som brukar väljas till den radie där potentialen har sitt minimum [1]. I denna punkt har vi ett nytt randvillkor, dvs att derivatan för $u(r)$ skall vara kontinuerlig. Taylorutveckling kring r_* ger då att skillnaden i derivata för vågfunktionens lösningar på båda sidor om r_* ges av differensekvationen (se vårt resultat för härledning av denna)

$$u^{(1),<}(r_*) - u^{(1),>}(r_*) = \frac{u^{>}(r_* + h) + u^{<}(r_* - h) - u(r_*)(2 + h^2 F(r_*))}{h} + \mathcal{O}(h^2), \quad (5)$$

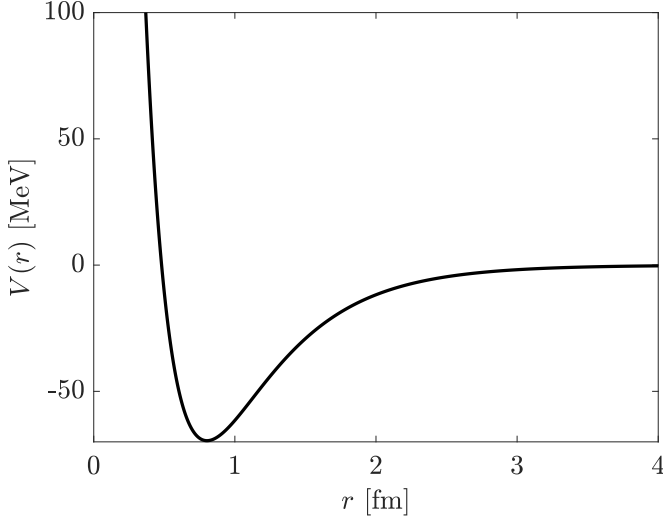
där $u^{(1),<}(r_*)$ svarar mot derivatan till vänster om r_* och vice versa. Genom att ansätta ett tolerensvillkor $\Delta\epsilon$ för accepterad avvikelse i derivatorna samt en första gissning på E_d kan vi lösa $u(r)$ numeriskt med (4). Detta gör vi genom att stega oss från båda ändpunkterna och sedan studera lösningarnas skillnad i derivata i r_* . Om skillnaden är större än $\Delta\epsilon$ uppdateras E_d enligt intervallhalveringsmetoden, se t.ex. [4], för att ge en ny uppskattning av skillnaden derivata i (5). När metoden har konvergerat enligt vårt tolerensvillkor har vi en uppskattning på E_d och kan även plotta $u(r)$.

2 Resultat & Diskussion

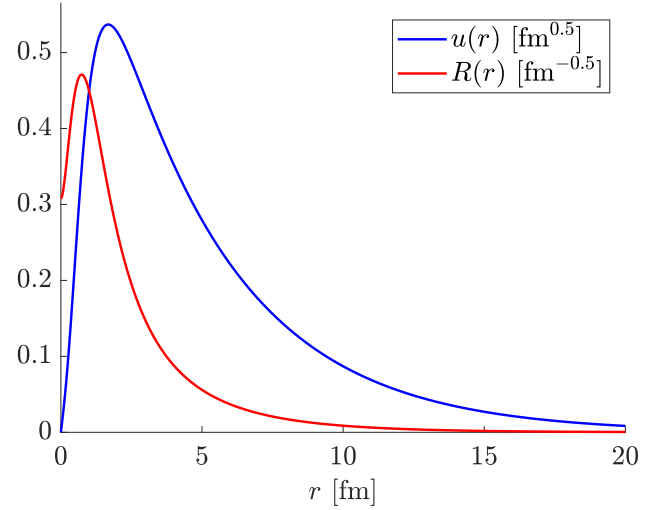
Vi använde följande konstanter [1, 3]

- $m_p = 938.272046 \text{ MeV}/c^2$
- $m_n = 939.5654133 \text{ MeV}/c^2$
- $\hbar c = 197.327 \text{ MeV fm}$

Vidare har vi i den här undersökningen använt matchningspunkten $r_\star = 1 \text{ fm}$.



(a) Malfliet-Tjon potentialen.



(b) Deuteronens vågfunktion, som vi antagit har radiell symmetri. $\psi(\vec{r}) = CR(r) = C\frac{u(r)}{r}$ där C är en normeringskonstant.

Figur 1: Plottar från simuleringen.

I figur 1a presenteras Malfliet-Tjon-potentialen som funktion av avståndet r mellan protonen och neutronen. I figur 1b presenteras deuteronens vågfunktion som funktion av r efter att de numeriska stegen i teorin hade utförts. Vid denna konvergerade vågfunktion uppmättes ett numeriskt värde på grundtillståndsenergin $E_d \approx -2.262 \text{ MeV}$. Vi ser att sannolikhetstätheten för att mäta de två partiklarna på samma ställe, $C^2R(r=0)^2$, är nollskild men att den största sannolikhetsmassan ligger kring r_{rms} . Detta är vad vi förväntar oss då potentialen har ett minimum där — alltså har systemet minst energi i denna punkt och är således mest stabilt då partiklarnas befinner sig på avståndet r_{rms} ifrån varandra.

Vidare noterar vi i figur 1a att potentialen har ett minimum nära 1 fm som vi förväntade oss enligt figur 1 i [1]. Dessutom avtar den asymptotiskt mot noll som vi också har använt som randvillkor. Vi noterar också i figur 1b att $u(r) \rightarrow 0$ då $r \rightarrow 0$ och då $r \rightarrow \infty$ vilket vi också använde som randvillkor. Dessutom satte vi också $r_\star = 1 \text{ fm}$ — i figuren ser vi att derivatan är kontinuerlig i r_\star som vi också hade satt som konvergensvillkor. Sammantaget verkar vågfunktionen vara konvergerad med avseende på alla våra kriterium.

Vi uppskattade värdet på grundtillståndsenergin till värdet $E_d = -2.262 \text{ MeV}$, vilket är lite mindre än det experimentellt bestämda värdet på $E_d = -2.224 \text{ MeV}$ som anges i labbpm [1]. Vi tror att det är den approximerade potentialen som har mest påverkan på värdets avvikelse från det experimentellt uppmätta och inte simuleringen. Detta baserar vi på att vi samma konvergerade värde på E_d för $N = 10\,000$ så väl som för $N = 100\,000$ samt att vi får samma konvergerade värde på E_d för $r_{\text{max}} = 100 \text{ fm}$ såväl som för $r_{\text{max}} = 1\,000 \text{ fm}$.

För r_{rms} erhåller vi värdet 1.96 fm, vilket är väldigt nära referensvärdet 1.97 fm från labb-pm [1]. Som vi nämnde i teorin beräknas rms-radien som

$$\begin{aligned} r_{\text{rms}}^2 &= \langle \psi | (r/2)^2 | \psi \rangle = \int_{\mathbb{R}^3} \psi(\vec{r})^* \left(\frac{r}{2}\right)^2 \psi(\vec{r}) dV \\ &= \int_0^\pi \sin(\theta) d\theta \int_0^{2\pi} |Y_0^0(\theta, \varphi)|^2 d\varphi \int_0^\infty R(r) \left(\frac{r}{2}\right)^2 R(r) r^2 dr \\ &= \left\{ Y_0^0 = \frac{1}{\sqrt{4\pi}}, \int_0^\pi \sin(\theta) d\theta \int_0^{2\pi} d\varphi = 4\pi \right\} \\ &= \int_0^\infty R(r) \left(\frac{r}{2}\right)^2 R(r) r^2 dr. \end{aligned}$$

Det sista uttrycket användes för att räkna ut r_{rms} med Matlabs `trapz`-funktion.

Om man använder (4) för att göra en beräkning men sätter $E \neq E_d$ får man en lösning som alltså inte är C^2 i r_* . En sådan lösning svarar då inte mot en lösning av S.E. och om vi beräknar dess energi genom $\int \psi H \psi d^3\vec{r}$ får vi en annan energi. Till exempel kan vi stoppa in $E = -1.26$ MeV i (4) och iterera oss fram till en vågfunktion $R(r)$, men när vi sedan beräknar väntevärdet av Hamiltonianen får vi $\int_0^\infty R(r) H R(r) r^2 dr = -2.09$ MeV $\neq E$. Denna diskreptans antas bero på att vågfunktionens andraderivatas singularitet vid $|\vec{r}| = r_*$ gör numerisk derivataberäkning olämplig. Vi kan därför ana att $H\psi = -1.26$ MeV $\cdot \psi$ inte är lösbar inom C^2 .

Härledning av differensrelation för skillnaden i derivata

För att härleda ekvation (5) börjar vi med att Taylorutveckla $u^>(r_* + h) + u^<(r_* - h)$ kring r_* :

$$\begin{aligned} u^>(r_* + h) + u^<(r_* - h) &= u^>(r_*) + u^<(r_*) + hu^{(1),>}(r_*) - hu^{(1),<}(r_*) + \frac{h^2}{2}u^{(2),>}(r_*) + \frac{h^2}{2}u^{(2),<}(r_*) + \mathcal{O}(h^3) \\ &= \left\{ u^>(r_*) = u^<(r_*) = u(r_*), u^{(2)}(r) = F(r)u(r) \right\} \\ &= 2u(r_*) + hu^{(1),>}(r_*) - hu^{(1),<}(r_*) + 2\frac{h^2}{2}F(r_*)u(r_*) + \mathcal{O}(h^3). \end{aligned}$$

som vi kan lösa för $u^{(1),>}(r_*) - u^{(1),<}(r_*)$:

$$u^{(1),>}(r_*) - u^{(1),<}(r_*) = \frac{u^>(r_* + h) + u^<(r_* - h) - u(r_*)(2 + h^2F(r_*))}{h} + \mathcal{O}(h^2).$$

Referenser

- [1] A. Ekström, *En realistisk vågfunktion för deutronen*, Institutionen för Fysik, Chalmers Tekniska Högskola, Göteborg, Sverige, 2019.
- [2] R. Pohl, F. Nez, L. M. P. Fernandes, F. D. Amaro, F. Biraben, J. M. R. Cardoso, D. S. Covita, A. Dax, S. Dhawan, M. Diepold, A. Giesen, A. L. Gouvea, T. Graf, T. W. Hänsch, P. Indelicato, L. Julien, P. Knowles, F. Kottmann, E.-O. Le Bigot, Y.-W. Liu, J. A. M. Lopes, L. Ludhova, C. M. B. Monteiro, F. Mulhauser, T. Nebel, P. Rabinowitz, J. M. F. dos Santos, L. A. Schaller, K. Schuhmann, C. Schwob, D. Taqqu, J. F. C. A. Veloso, and A. Antognini, *Laser spectroscopy of muonic deuterium*, Science **353** (2016), no. 6300, 669–673, available at <https://science.sciencemag.org/content/353/6300/669.full.pdf>.
- [3] C. Nordling och J. Österman, *Physics Handbook for Science and Engineering*, Studentlitteratur, 2015.
- [4] I. Gustafsson och K. Holmåker, *Numerisk analys*, Liber, 2016.

A Matlab-kod

```
1 %% Steg 1
2 clear, clc
3 %grid i fm
4 rmax=100;
5 %antal steg
6 N=100000;
7 %ekvidistant steglängd
8 h=(rmax)/(N);
9 fprintf('steglängd:%f\n',h)
10
11 %initialisera grid, och potentialen V(r)
12 %for varje r. Använd ej exakt noll.
13 r=linspace(1e-16,rmax,N+1);
14 u=zeros(1,N+1);
15 Fvec=zeros(1,N+1);
16 Vr=zeros(1,N+1);
17
18 V = @potential;
19
20 for i=1:N+1
21     Vr(i) = V(r(i));
22 end
23
24 fs = 17;
25
26 plot(r,Vr,'linewidth',2,'color',[0,0,0])
27 axis([0 4 -70 100])
28 xlabel('$r$ [fm]','fontsize',fs, 'Interpreter', 'latex')
29 ylabel('$V(r)$ [MeV]','fontsize',fs, 'Interpreter', 'latex')
30
31 %% Steg 2: Matchning
32 clc, clf
33 hold on
34
35 E_min = -2.224*4;
36 E_max = 0.0;
37
38 E = 0.5*(E_min + E_max);
39
40 %E = -2.000
41 max_iter = 1000;
42 tol_kontinuitet = 1e-5;
43
44 m_p = 938.272046; % MeV/c^2
45 m_n = 939.5654133; % MeV/c^2
46 mu = m_p*m_n/(m_p+m_n); % MeV/c^2
47
48 hbarc = 197.327; % MeV*fm
49 % hbar^2 c^2= 197.327^2 MeV^2fm^2
```

```

50
51 K = 2*mu/hbarc^2;
52
53 % iterera over energin E
54 counts = 0;
55 for iter = 1:max_iter
56     counts = counts + 1;
57     % initialisera Fvec(r) (dvs. F i ekv. 15),
58     % denna vektor beror pa valet av E
59
60     %valj matchningspunkt (motsv. grid-index)
61     rmp_i = ceil(1/rmax*N);
62
63
64     for i=1:N+1
65         F(i) = K*(Vr(i)-E);
66     end
67
68     %printf('Match r =%.16f \n',r(rmp_i));
69
70     %initialisera utat-integrerad vagfunktion
71     u(1)=0;
72     u(2)=h;
73
74     % Numerov utat
75     for i = 2:rmp_i-1
76         u(i+1) = (u(i)*(2+(5/6)*h^2*F(i))-u(i-1)*(1-(1/12)*h^2*F(
77             i-1))) ...
78             / (1-(1/12)*h^2*F(i+1));
79     end
80
81     u_out_mp = u(rmp_i);
82
83     %initialisera inat-integrerad vagfunktion
84     u(N+1) = 0;
85     u(N) = h;
86
87     % Numerov inat
88     for i=N:-1:rmp_i+1
89         u(i-1) = (u(i)*(2+(5/6)*h^2*F(i))-u(i+1)*(1-(1/12)*h^2*F(
90             i+1))) ...
91             / (1-(1/12)*h^2*F(i-1));
92     end
93
94     u_in_mp = u(rmp_i);
95
96     skalfaktor = u_out_mp/u_in_mp;
97
98     % matcha "hojden"
99     u(rmp_i:N) = skalfaktor*u(rmp_i:N);
100

```

```

99     % berakna diskontinuitet av derivatan i mp
100    matchning = (u(rmp_i-1)+u(rmp_i+1)-u(rmp_i)*(2+h^2*F(rmp_i)))/h;
101
102    if abs(matchning) < tol_kontinuitet
103        break;
104    end
105
106    %disp('u(rstar)')
107    %disp(u(rmp_i))
108
109    if u(rmp_i)*matchning > 0
110        E_max=E;
111    else
112        E_min=E;
113    end
114    E=0.5*(E_max+E_min);
115 end
116
117 disp(['E_d = ', num2str(E, 20)])
118 disp('nbrIterations = ')
119 disp(counts)
120 R = u./r;
121 u = u/sqrt(trapz(r,R.^2.*r.^2));
122 R = R/sqrt(trapz(r,R.^2.*r.^2));
123 plot(r,u,'linewidth',1.5,'color','b')
124 plot(r(2:end),R(2:end),'linewidth',1.5,'color','r')
125
126 xlabel('$r \sim [\textup{fm}]$', 'fontsize',fs, 'Interpreter', 'latex'
127 )
128 axis([0 20 0 1.2*max(R)])
129 leg1 = legend('$u(r) \sim [\textup{fm}]^{0.5}$', '$R(r) \sim [\textup{fm}]^{-0.5}$');
130 set(leg1,'Interpreter','latex');
131
132 r_rms = sqrt(trapz(r, R.*(r/2).^2.*R.*r.^2))
133 set(leg1,'FontSize',fs);
134
135
136 %% berakna <H>
137
138 dR = gradient(R)./gradient(r);
139 d2R = gradient(dR)./gradient(r);
140
141 expected_H = trapz(r,-r.^2.*R.*(hbar*c)^2./((2*mu).*((1./r.^2).*(2*
    r.*dR + r.^2.*d2R)) + r.^2.*R.*V(r).*R)

```

```

1 function V = potential(r)
2
3     l1 = -586.04;

```

```
4      12 = 1458.19;  
5      13 = -872.15;  
6  
7      mu1= 1.55;  
8      mu2= 3.11;  
9      mu3= 6.00;  
10  
11     V1 = 11*exp(-mu1*r);  
12     V2 = 12*exp(-mu2*r);  
13     V3 = 13*exp(-mu3*r);  
14  
15     V = (V1+V2+V3)./r;  
16 end
```