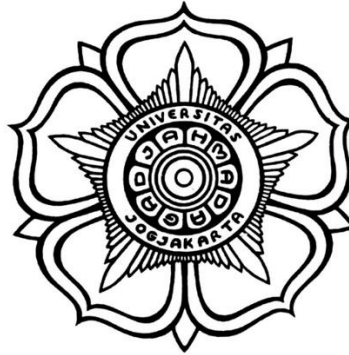


TUGAS P13

KOMPUTASI NUKLIR

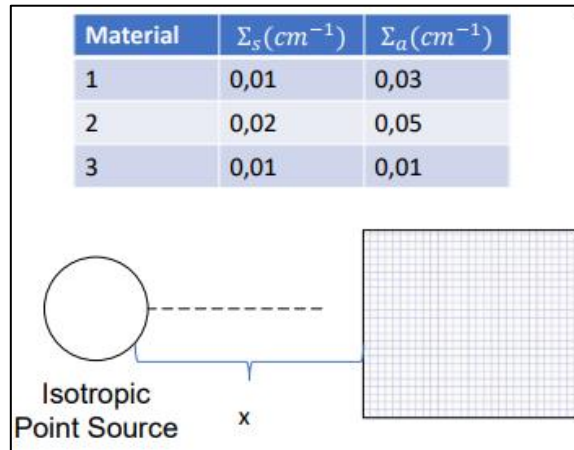


Yessika Natalia Chelsie (18/425243/TK/46938)

PROGRAM STUDI TEKNIK NUKLIR
DEPARTEMEN TEKNIK NUKLIR DAN TEKNIK FISIKA
FAKULTAS TEKNIK
UNIVERSITAS GADJAH MADA
YOGYAKARTA
2021

A. DESKRIPSI MASALAH

Terdapat sumber neutron bentuk titik yang memancarkan radiasi secara isotropik pada jarak $x=15$ cm dari sebuah *slab* dengan ukuran 100×100 cm yang terbagi ke dalam 100 region pada masing-masing sumbu. Diketahui material tersebut terbuat dari 3 macam elemen. Tampang lintang makroskopik serapan dan hamburan diberikan pada data di samping. $N=10^6$ Plot grafik fluence neutron yang jatuh pada setiap region. Hitung jumlah sampel yang berhasil mencapai *slab*, yang langsung menembus *slab*, mengalami serapan, terhambur ke depan *slab*, dan terhambur ke belakang *slab*.



B. IMPLEMENTASI PROGRAM

Program utama tugas.m

```
%Asumsi: Shielding calculation p.392
clear all
global transmission mencapai serapan scatter_depan scatter_belakang tidak_mencapai total
N=1E6;
Sig_s=0.04;
Sig_a=0.09;
thickness=100;
tugasakhir = coba_plot1(Sig_s,Sig_a,thickness,N);
transmission
mencapai
serapan
scatter_depan
scatter_belakang
tidak_mencapai
total
```

Subprogram coba_plot1.m

```
function tugasakhir = coba_plot1(Sig_s,Sig_a,thickness,N)
tugasakhir=0;
Sig_t=Sig_a+Sig_s; %sigma total
iSig_t=1/Sig_t;
Pos_collx=zeros(1,N);
Pos_colly=zeros(1,N);
jarak=15; %jarak 15 cm sebelum slab
```

```

global transmission mencapai serapan scatter_depan scatter_belakang tidak_mencapai total
transmission=0; %langsung melewati/menembus slab
mencapai=0;
serapan=0;
scatter_depan=0;
scatter_belakang=0;
tidak_mencapai=0;

for i=1:N

%   x = zeros(10000,1);
%   y = zeros(10000,1);
    x = 0; %posisi awal
    y = thickness/2;
    move_x = 0;
    move_y = 0;

    %arah random (360 derajat, asumsi sumber ditengah)
    theta = 360*rand;

    alive=1;

    %jumlah interaksi neutron %berapa kali ngewhile saat alive=1
    j = 1;

    if ((theta<atand((thickness/2)/jarak)) || (theta>(360-atand((thickness/2)/jarak))))
%mengarah slab
        mencapai = mencapai+1;
        alive=1;
    else
        tidak_mencapai = tidak_mencapai + 1;
        alive=0;
    end

    %tracking neutron
    while(alive)
        %get distance to collision, jarak random
        l=-log(1-rand)*iSig_t;

        %move particle
        %x
        move_x = l*cosd(theta);
        %y
        move_y = l*sind(theta);

        %posisi skrg
        x = x + move_x;
        y = y + move_y;
    end
end

```

```

    %still in the slab?
    if (j==1) && (x>(jarak+thickness)) && (theta<atand((thickness/2)/jarak))
%menembus slab
        transmission=transmission+1;
        alive=0;
    elseif (j==1) && (x<jarak) && (theta<atand((thickness/2)/jarak)) %tidak sampai slab
        tidak_mencapai = tidak_mencapai+1;
        alive=0;
    elseif (j>1)&&(x>(jarak+thickness)) %udah masuk slab, terus keluar makin ke kanan
        scatter_belakang=scatter_belakang+1;
        alive=0;
    elseif (j>1)&&(x<jarak) %udah masuk slab, terus keluar makin ke kiri
        scatter_depan=scatter_depan+1;
        alive=0;

    else %scatter or absorb
        if (rand<Sig_s*iSig_t)
            %scatter, pick new theta (arah terhambur yang baru)
            theta=360*rand;
            j = j+1;
        else %absorb
            serapan=serapan+1;
            alive=0;
        end
    end

end

    Pos_collx(i)=x;
    Pos_colly(i)=y;
end

total = transmission+scatter_depan+scatter_belakang+serapan+tidak_mencapai;

%batas sumbu
sx = [15:115];
sy = [0:100];
hitung = zeros(100,100);

%hitung interaksi
for i=1:N
    for ty=100:-1:1
        if(Pos_colly(i)>=sy(ty))&&(Pos_colly(i)<sy(ty+1)) %tepi kiri dan tepi kanan region 1x1
            for tx=1:100
                if(Pos_collx(i)>=sx(tx))&&(Pos_collx(i)<sx(tx+1))
                    hitung(tx,ty)=hitung(tx,ty)+1;
                end
            end
        end
    end
end
end

```

```

end

pltx = 1:1:100;
plty = 1:1:100;
imagesc(sx,sy,hitung');
set(gca,'YDir','normal');
grid on
grid minor
colorbar
colormap('spring')

end

```

Subprogram coba_scatter1.m

```

function tugasakhir=coba_scatter1(Sig_s,Sig_a,thickness,N)
Sig_t=Sig_a+Sig_s; %sigma total
iSig_t=1/Sig_t;
Pos_coll=zeros(1,N);

jarak=15; %jarak 15 cm sebelum slab

transmission=0; %langsung melewati/menembus slab
mencapai=0;
serapan=0;
scatter_depan=0;
scatter_belakang=0;

for i=1:N
    %x = zeros(10000,1);
    %y = zeros(10000,1);
    x(1) = 0; %posisi awal
    y(1) = thickness/2;
    move_x(1) = 0;
    move_y(1) = 0;

    %arah random (360 derajat, asumsi sumber ditengah)
    theta = 360*rand;

    alive=1;

    %jumlah interaksi neutron %berapa kali ngewhile saat alive=1
    j = 1;
    %parameter plot
    k = 1;

    %tracking neutron
    while(alive)
        if (theta<atand((thickness/2)/jarak)) %mencapai slab
            mencapai = mencapai+1;
            alive=1;

```

```

end

%get distance to collision, jarak random
l=-log(1-rand)*iSig_t;

%move particle
%x
move_x(j) = l*cosd(theta);
%y
move_y(j) = l*sind(theta);

%posisi skrg
x(j+1) = x(j) + move_x(j);
y(j+1) = y(j) + move_y(j);

%still in the slab?
if (x(j+1)>(jarak+thickness)) %langsung menembus slab
    transmission=transmission+1;
    alive=0;
elseif (x(j+1)<jarak) %tidak sampai slab
    alive=0;
elseif (x(j+1)>(jarak+thickness)) %udah masuk slab, terus keluar makin ke kanan
    scatter_belakang=scatter_belakang+1;
    alive=0;
elseif (x(j+1)<jarak) %udah masuk slab, terus keluar makin ke kiri
    scatter_depan=scatter_depan+1;
    alive=0;
else %scatter or absorb
    if (rand<Sig_s*iSig_t)
        %scatter, pick new theta (arah terhambur yang baru)
        theta=360*rand;
        j = j+1;
    else %absorb
        serapan=serapan+1;
        alive=0;
    end
end
end

end

%plot vektor atau tracking neutron
for k = 1:j
    u = x(k); %posisi sebelumnya
    v = y(k);
    hold on;
    quiver(u,v,move_x(k),move_y(k),0,"linewidth",2);
    axis equal;
    grid on;
    k = k+1;
end
end
end

```

```

    %Pos_coll(i)=x;

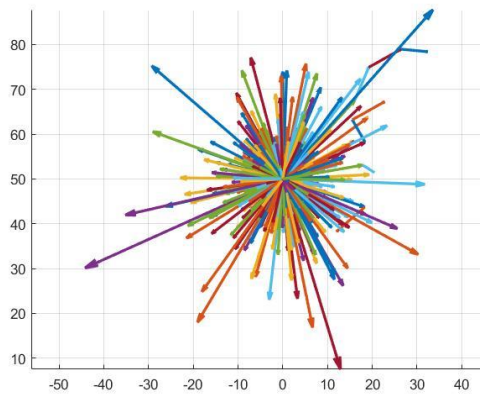
%plot([15 15], [0 100]) %jarak ke slab

%% Save the figure
%print('coba_scatter1.png','-dpng','-r0')

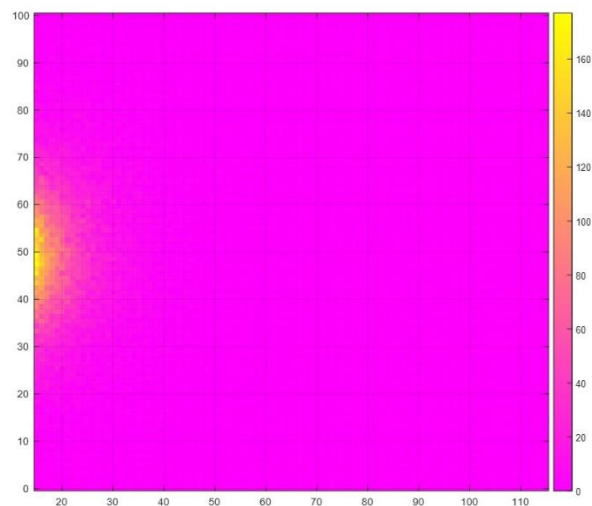
close all;
end

```

C. HASIL



Gambar 1. Output coba_scatter1



Gambar 2. Plot region slab

D. PEMBAHASAN

Pada tugas kali ini, penulis membuat 1 program utama dan 2 sub program. Sub program `coba_scatter1` dan sub program `coba_plot1` memiliki algoritma yang sama, namun berbeda

tujuan. Sub program `coba_scatter` digunakan untuk menghasilkan *output* berupa bagaimana melacak sejumlah neutron beserta interaksi yang terjadi dari selagi di udara sejauh 15m dan di *slab* dari suatu sumber neutron isotropik. Sedangkan sub program `coba_plot1` bertujuan untuk menunjukkan banyaknya neutron pada suatu region di *slab* yang dilaluinya.

Algoritma pengerjaan tugas dimulai dari mendefinisikan semua parameter yang digunakan. Pada tugas ini dibutuhkan data tampang lintang serapan dan tampang lintang hamburan dari material. Sebenarnya material terdiri dari 3 jenis elemen yang berbeda, namun saat pengerjaan penulis mengasumsikan bahwa material *slab* adalah campuran homogen. Oleh karena itu, asumsi terdapat tiga region yang berbeda berdasarkan perbedaan elemen ini diabaikan. Alhasil dalam memperhitungkan tampang lintang untuk *slab* mengikuti persamaan [1] :

$$\Sigma_{Mixture} = \sum_i \sigma_i \cdot N_i$$

$$\Sigma_g = 0,01 + 0,02 + 0,01 = 0,04$$

$$\Sigma_a = 0,03 + 0,05 + 0,01 = 0,09$$

Asumsi lainnya yang digunakan pada tugas kali ini adalah tidak diperhitungkannya tampang lintang di udara sehingga interaksi neutron dan udara akan diabaikan. Selanjutnya, untuk melakukan pelacakan neutron digunakan perintah pengulangan `for`. Jadi, dari neutron pertama sampai ke-N didefinisikan semua kemungkinan interaksi yang terjadi. Adapun nilai random yang digunakan pada tugas kali ini ditujukan pada sudut pancaran neutron dari sumber yang mana dikarenakan merupakan sumber isotropik, maka sudut hamburan akan dirandom sebesar 360° . Selain itu, ada juga parameter *alive* yang berarti 1 jika masih akan melanjutkan interaksi (energi belum habis) dan 0 jika tidak akan melanjutkan interaksi. Setelah mendefinisikan kedua parameter tersebut, selanjutnya penulis memperhitungkan berapa kemungkinan dari sumber yang mencapai *slab* maupun tidak mencapai *slab*. Jika sudut dibawah dari $\tan^{-1}(50/15)$ atau $360^\circ - \tan^{-1}(50/15)$ maka akan terdapat kemungkinan neutron mencapai *slab*. Hal ini dikarenakan asumsi penulis adalah sumber berada di tengah-tengah, jika diplot dalam kartesian berarti di $y=50$ dan $x=0$. Oleh sebab itu, sudut pancaran neutron hanya akan terbatas pada kedua sudut tersebut, jika melebihi maka akan tidak mengarah ke *slab*. Dan karena mencapai, maka *alive* akan bernilai 1 dan selain itu, maka akan tidak mencapai *slab* dan *alive* akan bernilai 0.

Selain sudut, jarak perjalanan yang ditempuh dari *slab* juga akan dibuat random. Adapun persamaan dari jarak yang akan ditempuh ini berdasarkan pada *mean free path* sebelum akhir tumbukkan. [2]

$$f(\lambda)d\lambda = e^{-\lambda} d\lambda \dots \text{untuk } \lambda > 0$$

$$\bar{\lambda} = \int_0^{\infty} \lambda e^{-\lambda} d\lambda = 1$$

$$\Sigma_t l = \lambda; d\lambda = \Sigma_t dl$$

$$f(l) = \Sigma_t e^{-\Sigma_t l}$$

$$l = \frac{-\log(1 - \theta)}{\Sigma_t}$$

Perjalanan/perpindahan partikel kemudian akan diproyeksikan ke sumbu x dan sumbu y berdasarkan hasil nilai random jarak yang akan ditempuh partikel. Selanjutnya, penulis mengasumsikan ada neutron yang langsung menembus *slab*, tidak mencapai dalam artian jarak tempuh partikel <15 (masih di udara) dan dalam sudut pancaran yang mengarah ke *slab*, hamburan ke belakang yang artinya sudah sempat masuk *slab*, tetapi akhirnya keluar *slab* ke arah x positif, dan hamburan ke depan yang artinya sudah sempat masuk *slab*, tetapi akhirnya keluar *slab* ke arah x negatif. Untuk permasalahan hamburan dan serapan, neutron dengan interaksi setelah interaksi pertama akan bergantung pada tampang lintangnya material untuk terjadi hamburan atau serapan. Jika terjadi hamburan, maka akan dibuat sudut hamburan baru dan *alive* akan tetap bernilai 1 dan sebaliknya pada serapan, maka akan bernilai 0. Selama *alive* bernilai 1 maka perintah *while* akan terus dijalankan untuk melacak posisi neutron di *slab*.

Untuk perhitungan akhir, sebanyak 1.000.000 neutron didapati hasil dari menjalankan program utama tugas.m dan sub program coba_plot1. sebagai berikut :

transmission = 0

mencapai = 407538

serapan = 164966

scatter_depan = 55418

scatter_belakang = 0

tidak_mencapai = 779616

total = 1000000

Dari hasil tersebut, total adalah jumlah mencapai dan tidak_mencapai. Selain itu, definisi lain dari total adalah jumlah yang langsung melewati *slab*, terhambur kedepan, terhambur kebelakang, terserap, dan tidak mencapai *slab*. Hasil ini sudah berhasil menjawab pertanyaan soal terkait berapa jumlah neutron yang berhasil mencapai *slab* = mencapai, yang langsung menembus = transmission, mengalami serapan = serapan, terhambur ke depan *slab* = scatter_depan, dan terhambur ke belakang *slab* = scatter_belakang. Adapun untuk menghitung interaksi di dalam *slab* digunakan *imagesc* untuk memplot berapa neutron yang terdapat pada suatu region dari *slab*.

Referensi :

- [1] "Macroscopic Cross-section | Definition & Examples | nuclear-power.net," *Nuclear Power*. <https://www.nuclear-power.net/nuclear-power/reactor-physics/nuclear-engineering-fundamentals/neutron-nuclear-reactions/macroscopic-cross-section/> (diakses Jun 22, 2021).
- [2] *Computational Nuclear Engineering and Radiological Science Using Python*. Elsevier, 2018. doi: 10.1016/C2016-0-03507-6.