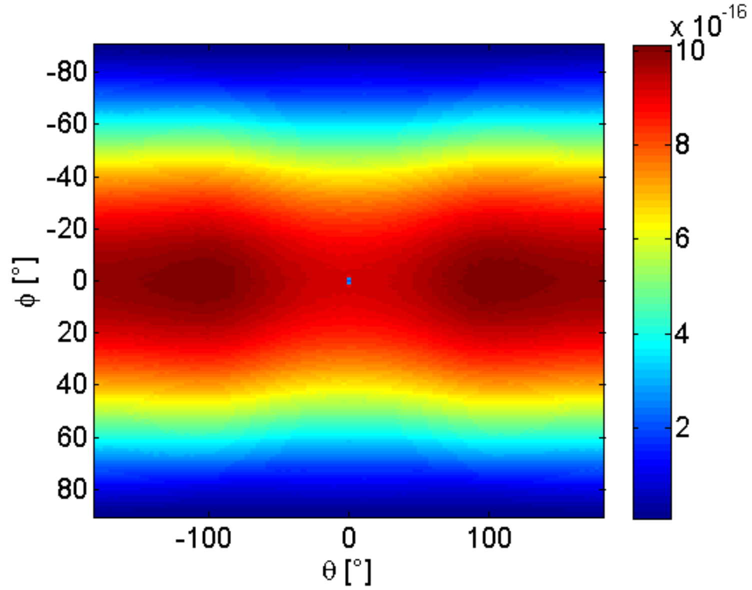
University of Copenhagen

Multiple scattering of neutrons in vanadium



Bachelor thesis by Oliver Hammer Boda
2$^{\text{nd}}$ of April 2014

Supervised by:
Kim Lefmann (lefmann@fys.ku.dk)
Mads Berthelsen (mads.bertelsen@gmail.com)

Abstract

This report investigates how the number of times a neutron has scattered determines where it will reach the surrounding detector. It is accomplished by simulating realistic multiple scattering in a vanadium sample and catching neutrons that have scattered 1,2,...,n times with separate detectors. The sample used is a solid vanadium cylinder with radius 2 cm and height 1.5 cm. We have found that the neutrons scattered more than 2 times appears uniformly on the detector, while the single scattered neutrons tend not to exit the sample on the opposing side. The intensity decreases exponentially for each extra scattering.

Contents

1 Introduction 3

2 Neutron Scattering 3

2.1 Neutrons over X-rays . . . 3

3 General neutron scattering theory 4

3.1 Absorption . . . 4

3.2 Elastic- and inelastic scattering . . . 4

3.3 Coherent and Incoherent scattering . . . 5

4 Simulating neutron scattering 5

4.1 Monte Carlo-simulations . . . 5

4.2 The McStas project . . . 5

4.3 Focusing . . . 6

4.4 Weight . . . 6

5 Experimental setup 6

5.1 Arm . . . 6

5.2 Source . . . 7

5.3 Collimator . . . 7

5.4 Sample . . . 7

5.5 Beamstop . . . 7

5.6 Detector . . . 8

6 Implementation 8

6.1 Does it scatter? . . . 8

6.2 Where does it scatter? . . . 9

6.3 Choosing a random direction . . . 9

6.4 No more scattering . . . 10

6.5 When to absorb? . . . 10

6.6 Comparing two component outputs . . . 10

7 Results 10

7.1 Comparing with original sample . . . 11

7.2 Single and multiple scattering . . . 11

8 Conclusions and future work 11

1 Introduction

A vanadium sample is commonly used to calibrate instruments for neutron scattering experiments. Vanadium with the atomic number 23 has properties that makes it suitable for calibration, the share of elastic incoherent scattering is big, so the scattered neutrons leaving the sample have random directions but with well defined energy.

When doing simulations of neutron scattering in a vanadium sample it's common to use techniques to speed up the process, some of these are 'unrealistic' in the sense that the method they use is different than real life neutron scattering. Some information for instance about how many times the neutrons scatter before they leave the sample is lost and might be useful knowledge when making a precise calibration.

We investigate the multiple scattered neutrons by making a realistic sample, where we can keep track on each neutron and how many times it has scattered. By looking at the simulation results for neutrons that have scattered a specific number of times, it's possible to see any patterns related to the number of times the neutrons have scattered.

2 Neutron Scattering

Neutron scattering is widely used to investigate properties of materials where X-rays can't be used. The method is costly so doing a simulation is much preferred whenever possible. Neutrons interact with nuclei via the strong nuclear force and magnetic moments via the electromagnetic force. These two properties causes scattering and absorption in nuclei of atoms inside materials.

2.1 Neutrons over X-rays

X-rays are still more used than neutrons for particle scattering, but for multiple reasons neutrons scattering. Reasons for using neutrons instead of x-rays[4]:

Energy and Wavelength

Neutrons have a wavelength comparable to inter-atomic distances and elementary excitation levels of solids, so both of these properties can be investigated by neutrons.

Isotopes and light elements

The neutron scattering cross section varies varies freely from element to element and isotopes as well. This makes it easier to study some of the lighter elements, compared to X-rays. With X-ray scattering the cross section increases with atomic number.

Quantitative experiments

Neutrons penetrate matter relatively easily since the interaction is weak. This means that it is possible not only to investigate the surface of the sample. Scattering simulations with neutrons gives relatively precise results since other interactions happen less frequently.

Magnetism

When using spin-polarized neutrons for a simulation, you can gain information about the atomic magnetic moment.

Transparency

Since neutrons easily penetrate through matter it is possible to investigate samples of great thickness. Tens of cm can be used as sample depending on the material. It's possible to build a sample environment without disturbing the experiment.

3 General neutron scattering theory

3.1 Absorption

The neutrons can be absorbed by the nuclei of the sample atoms. The probability of this happening, $P_a(z)$ depends on the absorption cross section, $\Sigma_a = \frac{N \cdot \sigma_a}{V}$, of the sample isotope and the distance z travelled in the sample[4]:

$$\boxed{P_a(z) = \exp(-\Sigma_a \cdot z) \qquad (1)}$$

3.2 Elastic- and inelastic scattering

Both elastic- and inelastic scattering takes place in a sample. When the scattering is elastic the energy is conserved.

3.3 Coherent and Incoherent scattering

When a neutron scatters, the new direction will be random if the scattering is incoherent. We have chosen only to implement elastic incoherent scattering in the component, since the contribution from inelastic and coherent scattering is small in a vanadium sample.

4 Simulating neutron scattering

We want to test experimental non-idealities and to get a better understanding. It is possible to simplify the algorithm in order to make it faster is by neglecting multiple scattering. This report will investigate the behaviour of the neutrons scattered 1, 2, 3,...,n times.
When simulating neutron scattering you can increase the speed of the simulation with several methods that are different than how it happens in reality. Some of these methods are commonly used techniques that don't affect the correctness, others may compromise the accuracy of the result.

4.1 Monte Carlo-simulations

When problems are too complex for an analytic solution *Monte Carlo*[1] techniques offers an estimate of the result. In some problems in nature *Monte Carlo* is more than just an estimate, since many problems in nature are probabilistic. If there for each neutron is a chance that it will perform each of the presented interactions, then for many neutrons this will give a good estimate of where the neutrons would actually end up going. When implementing *Monte Carlo* you express the different scenarios in terms of probability and use a *random number generator* to determine the outcome. *Monte Carlo* is well known and well documented technique that does not contribute to any systematic error.

4.2 The McStas project

The software used for setting up the test environment and performing the simulations is called *McStas*. The *McStas* project[2] offers open-source software to Monte Carlo-simulate neutron scattering. The user sets up the testing environment by placing components in an instrument file. The user can use components offered by *McStas* or write his/her own components in a

C-language-like style. This report's starting point is the *McStas*-component 'V_sample.comp'. It represents a Vanadium-sample that does incoherent elastic single scattering. I have modified it to do incoherent elastic multiple scattering.
This report will not include a tutorial on how to perform simulations etc. with the McStas software, only information that is specific for the setup used to produce the data presented in this report.

4.3 Focusing

Focusing is implemented in the original component, but in the modified component it is left out, since we were aiming for a realistic component. When choosing a direction for a scattered neutron you choose a direction where the neutron will reach the detector. The share of neutrons will then be increased. In this report the detector is surrounding the sample, so there would be no gain from implementing this method.

4.4 Weight

Not to be confused with mass. The *weight* of a particle determines how many particles it represents. If neutrons travels a long way through the sample, then instead of 'loosing' neutrons by absorbing them, you can lower the weight of them, and raise the weight of the neutrons travelling less likely to be absorbed. That way you end up with more data from the detectors.

5 Experimental setup

This section will describe what my setup consists of. Figure 1 shows a 2D-sketch of the 3D-setup, there is a legend in the right side. Since the detectors are 20 meters in diameter, it will not be included on the sketch.

5.1 Arm

The arm is a fictional component. It is only used as a reference point for the other components. For ease my 'Arm' is placed in $(X, Y, Z) = (0, 0, 0)$.

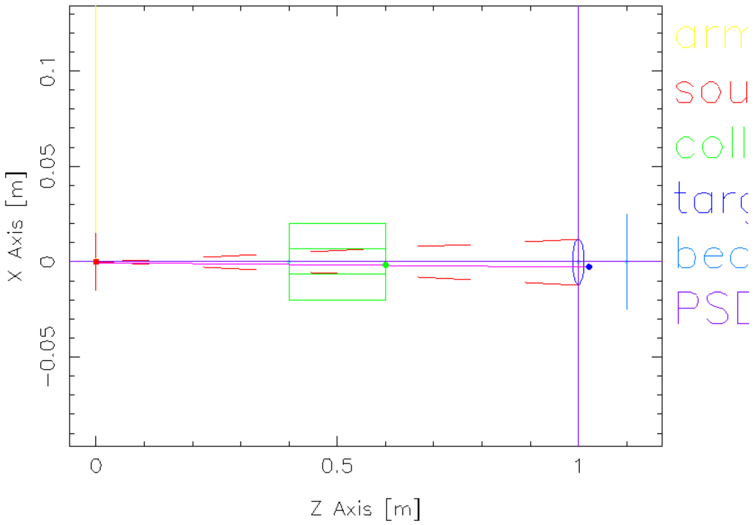


Figure 1: This is the setup where we perform simulations.

5.2 Source

This is where all the neutrons are generated. The source is 2 mm in diameter at its opening and points directly at the center of the sample.

5.3 Collimator

The collimator is placed so that all neutrons go through it on their way to the source. It makes sure, that all the neutrons have almost the same direction.

5.4 Sample

This is the vanadium sample where everything related to scattering and absorption takes place. The sample is shaped as a solid cylinder with radius 2 cm and 1.5 cm height along the y-axis (perpendicular to the beam, see Figure 1). My contribution to the McStas-project can be found in this component.

5.5 Beamstop

The *beamstop* catches all the neutrons that are not scattered. It is placed 10 cm behind the sample so it catches the direct beam exiting the sample.

It makes it easier to analyse output data. Note that some of the neutrons scattered can hit the *beamstop*, this makes the data in this region unreliable, but it's only a small area (see for instance the centered dot in Figure 2).

5.6 Detector

Only neutrons that hits the detector will be part of the result. In this report we have chosen to use the '4PI'-detector, which is surrounding the sample, so all neutrons that aren't absorbed (and don't hit the *beamstop*) will eventually reach the detector. It is shaped as a sphere and is 20 meters in diameter. The sample is placed in the center.
It is possible to set up multiple detectors that detects only the neutrons that satisfy certain conditions. Some of the plots included in this report contain only the neutrons that has scattered a specific number of times. This enables us to investigate the behaviour of the single scattered neutrons and compare it to the behaviour of the ones that have scattered multiple times.

6 Implementation

In this section I will describe the code that runs for each neutron as long as the neutron is not moving away from the sample. The code can be found in the *TRACE*-section of the component, see Appendix 9. By looking at the times of intersecting with the sample boarders we can determine where it is and where it's heading.

6.1 Does it scatter?

The probability of the neutron scattering is dependent on the scattering cross section Σ_s and the distance travelled z_0: The probability of scattering P_s can be expressed by[4]:

$$\boxed{P_s(z) = 1 - \exp(-\Sigma_s \cdot z_0) \qquad (2)}$$

6.2 Where does it scatter?

If Equation 2 determines that the neutron does scatter, then we should find out where it scatters. The accumulative probability for scattering $\frac{I(z)}{I_0}$ can be expressed in terms of length travelled z and the scattering cross section Σ_s [4]:

$$\frac{I(z)}{I_0} = \exp(-\Sigma_s z)$$

The probability distribution for scattering $P(z)$ is therefore:

$$\frac{d}{dz}\frac{I(z)}{I_0} = \Sigma_s \cdot \exp(-\Sigma_s z)$$

The probability of scattering in the interval $0 < z < z_0$:

$$P(z) = \int_0^{z_0} \Sigma_s \cdot \exp(-\Sigma_s z) dz = \frac{1 - \exp(-\Sigma_s z)}{1 - \exp(-\Sigma_s z_0)}$$

$$P(z) \cdot (1 - \exp(-\Sigma_s z_0)) = 1 - \exp(\Sigma_s z)$$

$$\boxed{z = \frac{\ln(1 - (1 - \exp(-\Sigma_s z_0)) \cdot P(z))}{-\Sigma_s} \qquad (3)}$$

Where $P(z)$ is implemented as a random number generator returning values $0 \geq P(z) \geq$. Now the neutron is propagated a distance z to where it scatters.

6.3 Choosing a random direction

Spherical coordinates are easier to use when choosing a new direction for the scattered neutrons. We need to choose a random direction in the unit circle and scale it with the speed. Conversion from spherical to Cartesian coordinates is done the usual way. To generate a random direction we will need to choose a $0 < \theta < 2\pi$ and $0 < \phi < \pi$. θ represents the angle in the horizontal plan and should be chosen randomly. ϕ is the polar angle. For ϕ we need to generate a random number from a sinus-distribution, this is because of the non-linear correlation between ϕ and $\sin(\phi)$.

Generating a random number from a sinus-distribution can be done by generating a random number m from a uniform distribution and applying $\sin^{-1}(m)$.

6.4 No more scattering

No matter if the neutron has scattered or not, before the neutron continues to the detector we evaluate the probability of absorption, see Equation 1.

6.5 When to absorb?

Absorption is the process that drops the neutron. To avoid doing unnecessary calculations we want to absorb as early as possible. From a coding perspective the easiest time to determine if the neutron gets absorbed is when it exits the sample, because at that time we know how far the neutron has travelled in the sample, so we only calculate a probability of absorption once. The downside is, if it turns out that the neutron should be absorbed end up possibly doing unneeded calculations.

6.6 Comparing two component outputs

The sample described in this report is evolved from *McStas*'s *V_sample* component. Since we know that the original component (only doing single scattering) is producing a correct result, we want to confirm that the single scattering my component does is equivalent of the original one's. We let M_r be a matrix containing the relative difference between two matrices M_a and M_b, each containing $x \times y$ elements. The elements of M_r are set to:

$$\boxed{M_r[i][j] = \frac{M_a[i][j] - M_b[i][j]}{M_a[i][j]}, i \in x, j \in y \qquad (4)}$$

This will not only be used when comparing the new sample with the original, but also when investigating how the output changes for neutrons scattered a different number of times.

7 Results

The results comes from simulating $3e10$ neutrons into two samples respectively. One sample is the original "V_sample" from *McStas*, this sample is only capable of single scattering. The other sample is build upon the first-mentioned and it does realistic multiple scattering. Some of the detectors catch only neutrons that have scattered a specific number of times. This is useful when investigating certain properties.

7.1 Comparing with original sample

Firstly we want to examine if the single scattering the original sample produces is comparable to the modified version's. Raw plots can be seen in Figure 2 and 3. They look alike, but the relative difference between them seen in Figure 4 shows us, that the two components produce different outputs.

To try to narrow down the difference. When absorption is disabled in both components, the single scattering detectors picks up following intensities:

Sample name	Intensity $[1/(sec \cdot pixels)]$
V_sample_original	$7.707(2)e-12$
V_sample_modified	$7.709(7)e-12$

The percentage deviation in intensity D between the original vanadium sample and then one I provide is:

$$D = \frac{7.7072 \cdot 10^{12} - 7.7097 \cdot 10^{12}}{7.7072 \cdot 10^{12}} \cdot 100\% \approx 0.032\% \tag{5}$$

7.2 Single and multiple scattering

Figure 6 helps determining how the single scattering model differs from the multiple scattered.

8 Conclusions and future work

From Figure 4 we see that the two components do not produce the same data. We only compare single scattering by single scattering and both of them only do incoherent elastic scattering, but we still get different results. We found that the total intensities from their two detectors only vary by 0.032%4, so the difference is not related to the probability of scattering. It might be related to choosing a random direction or chances of absorption.
We found that the neutrons that have only scattered once are less likely to exit the sample through the opposing side of the sample, the reasons for this

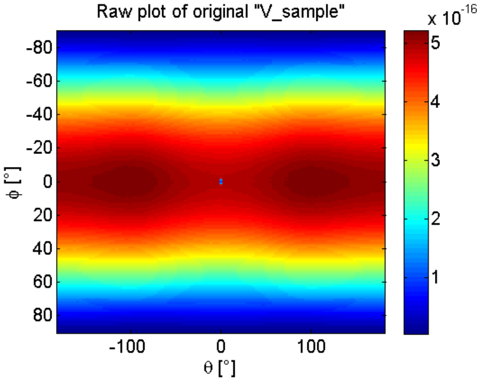


Figure 2: Plot of the original *V_sample*. There is clearly a symmetry about $\theta = 0°$ and $\phi = 0°$, which can be explained by the symmetry in the setup. The intensity greatly vary with ϕ, more neutrons reach the detector around $\phi = 0$ than around $|\phi| = 80°$. There is also a dependency on θ, the neutrons are more likely to go around $|\theta| =\sim 110°$ which corresponds to a bit back towards the neutron source in both left and right direction. In the center of the plot we see a small 'bump', that is the *beamstop* placed in $\theta = \phi = 0°$ right behind the sample. The intensity is measured in $1/(sec \cdot pixel)$.

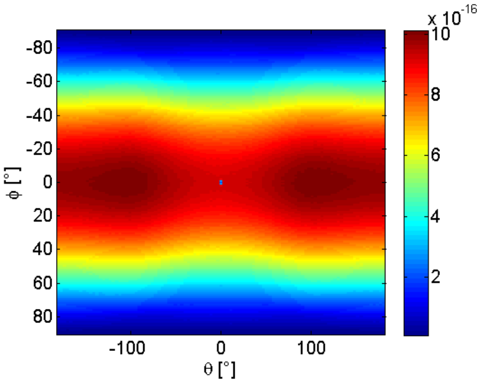


Figure 3: Plot of the single scattered neutrons coming from the modified version *V_sample*. There are several similarities between this plot and the one in Figure 2. Geometrically they *seem* almost identical. One difference to notice is, that the intensity varies by a factor of ~ 2.5 (see magnitude of intensity in the colorbar for the two figures). See Figure 4 for a better comparison. The intensity is measured in $1/(sec \cdot pixel)$.

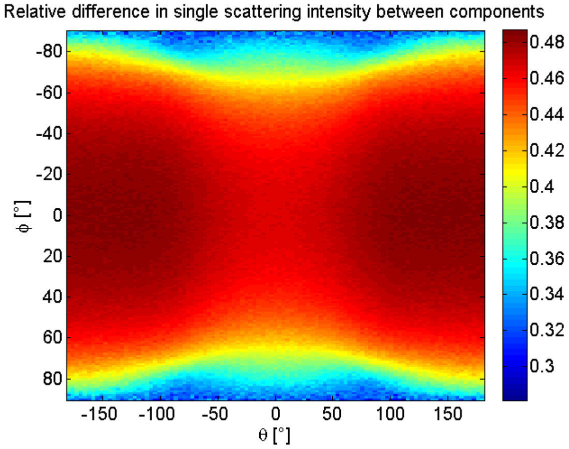


Figure 4: Plot of the relative difference between the two components as described in equation 4. Both detectors only detect single scattered neutrons, see Figure 2 and 3. If the two samples gave the same result, we would plot values around 0. For all values of ϕ and θ we see that the relative difference is above 0, meaning that the detector in the original component was hit by more neutrons in total and for each pixel. There is a lower concentration of neutrons around $|\phi| = 80^\circ$ compared to the rest of the plot.

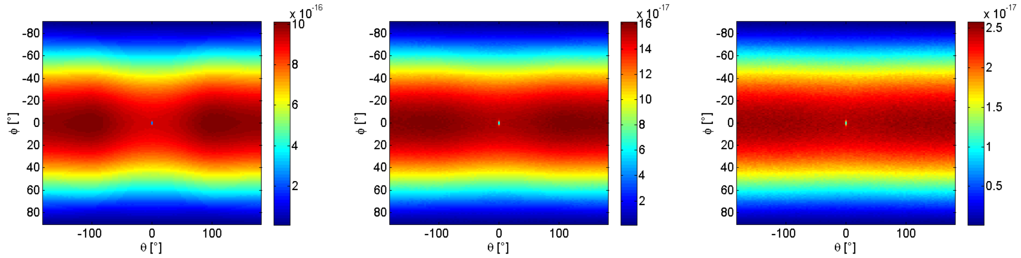


Figure 5: From left to right; we see intensities from the respectively 1-, 2- and 3-times scattered neutrons. The distortion increases hence a drop in intensity for each extra scattering. As earlier described, we see that more of the 1-time scattered neutrons reach the detector around $(\phi, |\theta|) = (0^\circ, \sim 110^\circ)$ than $(\phi, \theta) = (0^\circ, 0^\circ)$. This pattern is less obvious for the 2-times scattered neutrons and for neutrons scattered 3 times, it is almost not existing.

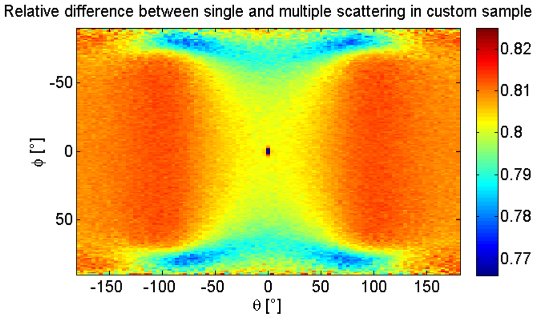


Figure 6: Relative difference in single and multiple scattering done in my component. Here we are comparing the neutrons scattered 1 time to all the rest.

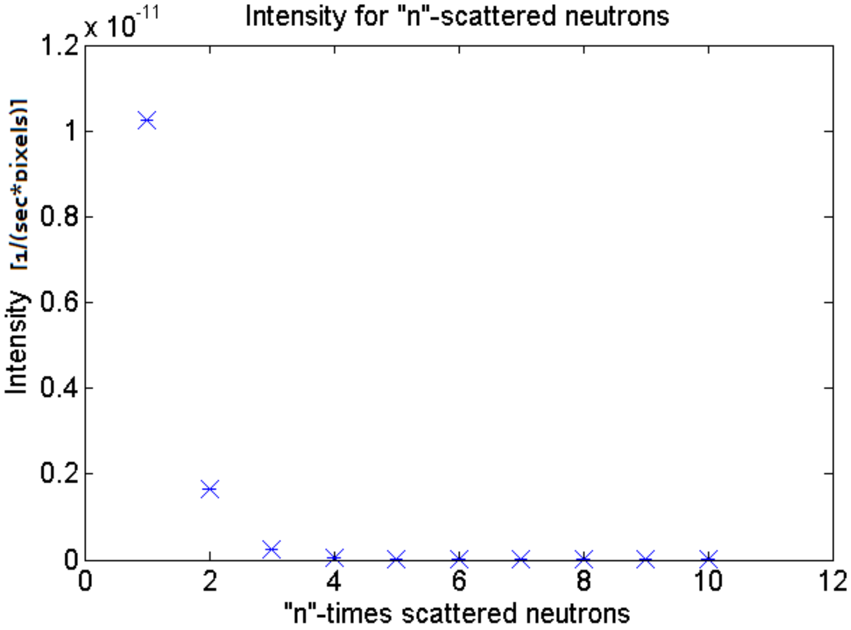


Figure 7: Here we see a relation between the intensity and the number of times the neutrons have scattered, errorbars are included but small. Since the neutrons are captured by the same detector, this is comparable to number of neutrons scattered n times. See raw data in Appendix A. It could look like the intensity is exponentially decreased.

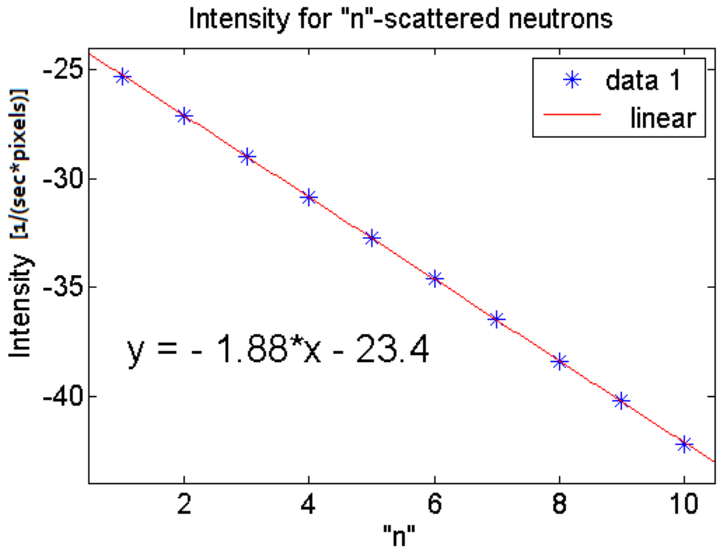


Figure 8: Now we see a clear linear relation. Which means that the intensity exponentially decreases for each extra time the neutrons have scattered.

are more absorption and more scattering since the travelled distance is longer. The double scattered neutrons have the same 'pattern' but less distinct, and the neutrons scattered more than 2 times appears uniformly distributed on the detector.

We have found that there is an exponential decrease in intensity for each time the neutrons have scattered. From equation 2 we know that when evaluating the probability of scattering it is only dependent on the distance ahead of it, when the medium consists of only one material like vanadium. When the neutrons enter the sample and have not yet scattered, they have the whole sample diameter ahead of them, making them more likely to scatter or get absorbed before they reach the opposing side. For the neutrons that have already scattered the first time, their positions seems random when and if they scatter a second time. There is a fixed percentage of the neutrons that either scatter or get absorbed, this witnesses about no general tendency in position and direction when scattering.

More effort needs to be put into finding the bug that makes the sample I provide produce a different single scattering output than the original vanadium sample. For now we do not know if the conclusions made in this report are

valid and apply to experimental data.
The exponential decrease in intensity might be a product of no general tendency for the neutrons in the sample. But on Figure 5 we see that the neutrons scattered 2 times do not appear uniformly like the neutrons scattered 3 or more times. It could be interesting to investigate this with a smaller detector, that way we would see where the neutrons exit the sample independently from their direction.
To get a more realistic picture of where the scattered neutrons hit the sphere-detector, the data could be plotted on to a 3D-sphere. Plotting the ratio 'intensity'/'surface area' would circumvent the possible confusion about smaller bins on the detector getting hit by a fewer neutrons[1]. It would also be interesting to perform the same tests on other sizes of sample. Different sample shapes could also be tested, but small changes would have to be made to the sample.

References

[1] Dieter W. Heerman, *Computer Simulation Methods in Theoretical Physics*, 1990.

[2] *McStas - A neutron simulation package*, `www.mcstas.org` DTU Physics, NBI KU, PSI, ILL.

[3] Peter Kjær Willendrup, Erik Knudsen, Kim Lefmann, Emmanuel Farhi, *McStas Component Manual*, `www.mcstas.org/download/components/doc/manuals/mcstas-components.pdf`, Risø DTU, 2013.

[4] *VNT - Virtual Neutrons for Teaching*, `vnt.nmi3.org/wiki/`, University of Copenhagen.

[5] William H. Press, Brian P. Flannery, Saul A. Teukolsky, William T. Vetterling, *Numerical Recipes in Fortran 77: 'The Art of Scientific Computing'*, 1992.

[6] Mads Bertelsen, *Optimizing neutron guides using the minimalist principle and guide_bot*, University of Copenhagen, 2014.

[7] John R. Taylor, *An introduction to error analysis - The study of uncertainties in physical measurements*, 2. edition, 1996.

[8] Albert-José Dianoux, Gerry Lander, *Neutron Data Booklet*, 2. edition, 2003.

Appendix A

The intensities from all the detectors surrounding my sample, each neutron is only detected once:

Times scattered	Intensity $[1/(sec \cdot pixel)]$
1	$1.02441 \pm 0.00001 \times 10^{-11}$
2	$1.66313 \pm 0.00005 \times 10^{-12}$
3	$2.6123 \pm 0.0002 \times 10^{-13}$
4	$4.0370 \pm 0.0009 \times 10-14$
5	$6.184 \pm 0.003 \times 10^{-15}$
6	$9.39 \pm 0.01 \times 10^{-16}$
7	$1.428 \pm 0.005 \times 10^{-16}$
8	$2.13 \pm 0.02 \times 10^{-17}$
9	$3.33 \pm 0.08 \times 10^{-18}$
10	$4.6 \pm 0.3 \times 10^{-19}$
11	0

The data plotted in Figure 8. The standard deviation is included and adjusted[7]:

Times scattered	log(Intensity $[1/(sec \cdot pixel)]$)
1	-25.30432 ± 0.00005
2	-27.12232 ± 0.00007
3	-28.9734 ± 0.0002
4	-30.8407 ± 0.0005
5	-32.717 ± 0.001
6	-34.601 ± 0.003
7	-36.485 ± 0.008
8	-38.39 ± 0.02
9	-40.24 ± 0.05
10	-42.2 ± 0.1

Appendix B

```
TRACE
%{
double t0, t3;                  /* Entry/exit time for outer cylinder */
double t1, t2;                  /* Entry/exit time for inner cylinder */
double v;                       /* Neutron velocity */
double dt0, dt1, dt2, dt;       /* Flight times through sample */
double l_full;                  /* Flight path length for non-scattered ↩
    neutron */
double l_accum=0;                 /* Accumulated flight path length */
double l_i, l_o=0;              /* Flight path lenght in/out for ↩
    scattered neutron */
double my_a=0;                    /* Velocity-dependent attenuation ↩
    factor */
double solid_angle=0;           /* Solid angle of target as seen from ↩
    scattering point */
double aim_x=0, aim_y=0, aim_z=1;   /* Position of target relative to ↩
    scattering point */
double v_i, v_f, E_i, E_f; /* initial and final energies and ↩
    velocities */
double dE;                 /* Energy transfer */
double t_scatter;
double phi,theta;
double scatter_distance;
int    intersect=0;
double sin_phi;
int scenario=0;

const double kVanadium_Sigma = 6117.76/8.45903e-26*5.08e-28;
// xs-absorbtion = xs-scattering
//vandium_density/vanadium_atomic_mass*vanadium_sigma_inc;
double vanadium_Sigma;

scatter_count = 0;
v = sqrt(vx*vx + vy*vy + vz*vz);

if (v) {
  vanadium_Sigma = kVanadium_Sigma*(2200/v);
} else {
  printf("warning: v = 0\n");
  vanadium_Sigma = kVanadium_Sigma;
}

while (scenario != 2) {

  if (VarsV.shapetyp == 2)
    intersect = sphere_intersect(
        &t0, &t1, x, y, z, vx, vy, vz, rad_sphere
    );
  else
    if (VarsV.shapetyp == 1)
    intersect = box_intersect(
        &t0, &t1, x, y, z, vx, vy, vz, xwidth, yheight, zthick
    );
else
  intersect = cylinder_intersect(
      &t0, &t1, x, y, z, vx, vy, vz, radius_o, h
  );

if (t0 > 0) { // haven't reached sample yet
  scenario = 0;
} else {
if (t0 <= 0 && t1 > 0) {  // inside of sample
  scenario = 1;
} else {
if (intersect==0 || t1 <= 0) { // we are not going to reach sample
  scenario = 2;
} } }

if (scenario == 0) { // travel to the sample
  PROP_DT( min(t0,t1) );
  t1 -= t0;
}

if (scenario <= 1) { // travel in the sample and maybe do scattering
  l_full = v * t1; // Length of path through sample

  // will it scatter?
   if (exp(-vanadium_Sigma*l_full) < rand01()) {
    scatter_count++;

    // - It scatters! But where?
    scatter_distance =
        -log(1-(1-exp(-vanadium_Sigma*l_full))*rand01())/↩
            vanadium_Sigma;
    l_accum += scatter_distance;
    dt = scatter_distance/v;
    PROP_DT(dt);    /* Point of scattering */

    // Random direction:
    phi=sinrand();
    theta=2*PI*rand01();
    sin_phi=sin(phi); // only calculate sin(phi) once and store the ↩
        result
    vx = v*cos(theta)*sin_phi;
    vy = v*sin(theta)*sin_phi;
    vz = v*cos(phi);

    SCATTER;

  } else { // it didn't scatter, traveling out of the sample
    PROP_DT( t1 + 0.000005);
    l_accum += l_full;
  }
}
```

```
  if (scenario == 2) {
    PROP_DT(0.000005); // travel 1cm at 2km/s
  }
}
// Finally check if the neutron was absorbed during the flight
if (exp(-vanadium_Sigma*l_accum) < rand01()) {
  ABSORB;
}
%}
```

Appendix C

```
/*↩
    *********************************************************************↩
    */
/* This is the instrument-file
DECLARE
%{
  double coll_div = 60;
  double t1,t2;
  int flag;
%}

/* This code only runs once */
INITIALIZE
%{
  t1=clock();
%}

TRACE

COMPONENT arm = Arm() AT (0,0,0) ABSOLUTE

COMPONENT source = Source_simple(radius = 0.001, dist = 1,
  focus_xw=0.001, focus_yh=0.001, E0=5, dE=0.001)
  AT (0,0,0) RELATIVE arm

COMPONENT target = V_sample_multiple(radius = 0.002,
  yheight = 0.015, focus_r = 0, pack = 1,
  target_x = 0, target_y = 0, target_z = 1)
  AT (0,0,1) RELATIVE arm
  EXTEND
  %{
    flag = scatter_count;
  %}

COMPONENT beamstop = Beamstop(
    xwidth = 0.005, yheight = 0.005)
  AT (0, 0, 0.1) RELATIVE target

//catches everything coming out of the sample
COMPONENT PSD_4pi = PSD_monitor_4PI(radius=10, nx=101, ny=180,
  filename="vanadium.psd", restore_neutron=1)
  AT (0,0,1) RELATIVE arm

// multiple-scattered neutrons
COMPONENT PSD_4pi_multiple = PSD_monitor_4PI(radius=10, nx=101, ny↩
    =180,
  filename="vanadium_multiple.psd", restore_neutron=1)
  WHEN (flag > 1)
  AT (0,0,1) RELATIVE arm

// single-scattered neutrons
COMPONENT PSD_4pi_single = PSD_monitor_4PI(radius=10, nx=101, ny=180,
  filename="vanadium_1.psd", restore_neutron=1)
  WHEN (flag == 1)
  AT (0,0,1) RELATIVE arm

// Scattered 2 times
COMPONENT PSD_4pi_2 = PSD_monitor_4PI(radius=10, nx=101, ny=180,
  filename="vanadium_2.psd", restore_neutron=1)
  WHEN (flag == 2)
  AT (0,0,1) RELATIVE arm

/*********************************/
/*                               */
/* AND ALL THE OTHER DETECTORS   */
/* FROM 1 TO 11                  */
/*                               */
/*********************************/

// Scattered 11 times
COMPONENT PSD_4pi_11 = PSD_monitor_4PI(radius=10, nx=101, ny=180,
  filename="vanadium_11.psd", restore_neutron=1)
  WHEN (flag == 11)
  AT (0,0,1) RELATIVE arm

/* This code is only run once */
FINALLY
%{
  t2=clock();
  printf("time elapsed: %.fms\n",(t2-t1)/CLOCKS_PER_SEC*1000);
%}

END
```