TISSUE OPTICS 2011

# Photon Diffusion

*Computer Exercise*

Tomas Svensson
Erik Alerstam
Stefan Andersson-Engels

Department of Physics, Lund

March 16, 2011

This document contains instructions for the computer exercise on light diffusion. We welcome comments that may improve this document. Contact us at:

`erik.alerstam@fysik.lth.se`
`stefan.andersson-engels@fysik.lth.se`

## Contents

# 1 Introduction

The aim of this exercise is to make you familiar with some computer programs developed to model light transport in turbid media. The programs might be helpful in your project. Thus, in preparing yourselves for the exercise and during the exercise itself, please consider how you could use the programs in the project, and make sure you understand how the programs could work for your specific problem.

The time resolved diffusion equation is a very useful model for describing light propagation in tissue. We recall the equation and the conditions that apply to it:

$$\frac{1}{c'}\frac{d}{dt}\Phi(\bar{\boldsymbol{r}},t) - \nabla D(\bar{\boldsymbol{r}})\nabla\Phi(\bar{\boldsymbol{r}},t) + \mu_{\mathrm{a}}\Phi(\bar{\boldsymbol{r}},t) = S(\bar{\boldsymbol{r}},t) \tag{1}$$

where $c'$ is the speed of light in the tissue, $D(\bar{\boldsymbol{r}})$ is the diffusion coefficient (which we allow to vary in space), $\mu_{\mathrm{a}}$ is the absorption coefficient, $\Phi(\bar{\boldsymbol{r}},t)$ is the light fluence rate, and $S(\bar{\boldsymbol{r}},t)$ represents a light source. It is valid under the following conditions:

- light fluence rate must be calculated far away from the source
- absorption must be much smaller than the scattering, i.e. $\mu_{\mathrm{a}} \ll \mu_{\mathrm{s}}'$

The diffusion equation can be solved analytically for a few simple cases, such as an infinite homogeneous space (the simplest case), a semi-infinite homogeneous half-space, or a homogeneous slab. These cases are obviously not very useful, other than for pedagogical reasons (which is why we will study them in this exercise!). For more complex geometries, such as those encountered in real tissue, such approximations can only be used a first rough estimate, and numerical methods should be used for improved accuracy. The diffusion equation is a very general equation in physics, used in various forms to describe for example gas (or fluid) diffusion, propagation of sound, and the propagation of neutrons in nuclear reactors. Because of this, different numerical methods have been developed to solve it. A commonly used method is the Finite Element Method (FEM), which is very flexible because it allows great freedom in choosing the geometry.

# 2 Analytic Solutions

## 2.1 Infinite Homogeneous Medium

Essentially, we want to solve a partial differential equation. Let us begin with the simplest case: a single light pulse in the middle of a huge homogeneous medium. The source is then given by

$$S(\bar{\boldsymbol{r}},t) = \delta(\bar{\boldsymbol{r}})\delta(t) \tag{2}$$

(we assume that the source strength is 1). According to basic theory on partial differential equations, the solution to the diffusion equation can now be expressed by a Green's function as (note the dimension of this expression!):

$$\Phi(r,t) = c'\,(4\pi Dc't)^{-3/2}\,\exp(-\mu_{\mathrm{a}}c't)\,\exp\left(\frac{-r^2}{4Dc't}\right) \tag{3}$$

## 2.2 Semi-infinite Homogeneous Half-space

Next, we are interested in the case of a semi-infinite half-space, where the medium is defined by (for example) the positive $z$-axis, and the $x,y$-plane acts as the boundary. Now we run into a few obstacles. The light pulse is assumed to enter the medium as a beam along the $z$-axis. So, where do we put the source? The best approximation is to place the source in $z = z_0 = 1/(\mu_a + \mu_s')$, i.e., one mean free path length into the medium. It is now possible to use a mirror-image approach, similar to those used in electro-magnetic field theory, to accomplish a solution for the half-space geometry. We place a negative mirror-image source in $z = -z_0$. When doing this, we make sure that the fluence rate on the boundary is zero, i.e., we set the boundary condition to be "zero fluence on the boundary". In a first approximation, this may seem to be a nice and simple way to take care of the boundary conditions, but when you think about it in a little more physical perspective, it is in fact not a very good idea.

In a boundary between two media with different refractive indices, e.g., air with $n = 1.0$, and tissue with $n = 1.4$, there will be a Fresnel reflection. This means that just below the surface of the tissue, the fluence rate will not be exactly zero, but a little higher. To deal with this, we introduce an *extrapolated boundary* at a distance $z_e$ outside the real boundary. Now we say that the fluence rate is zero on the extrapolated boundary. This means that the fluence rate on the real boundary gets to be non-zero, which is just what we wanted (see figure 1). The value of $z_e$ is a function of the difference in refractive indices between the two media. For a tissue-air boundary, the value is $z_e \simeq 2z_0$ (typically some millimeter). Figure 2 shows the positions of the positive source, the extrapolated boundary and the negative image source.

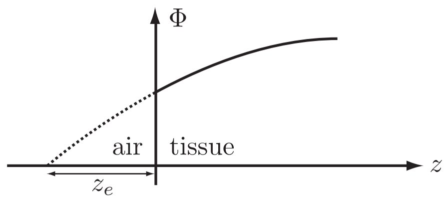


Figure 1: The principle of the extrapolated boundary method for treating the boundary conditions. The fluence rate $\Phi$ is non-zero on the physical boundary because of Fresnel reflection, so in the calculations an extrapolated boundary at a distance of $z_e$ outside the physical boundary is assumed, where the condition of zero photon density can be applied.

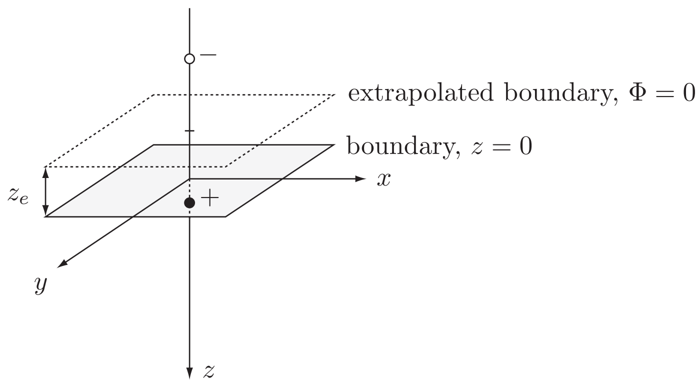


Figure 2: The geometry for the semi-infinite (half-space) case. The physical boundary is at $z = 0$ (grey plane). The positive source is put at $z = z_0$ (filled black dot). The extrapolated (virtual) boundary is at $z = -z_e$, on which a negative source mirrored with respect to that boundary, and the positive source, ensures zero fluence. Thus, the negative source is found at $z = -z_0 - 2z_e$ (open dot).

### 2.3 Homogeneous Slab

To cope with the case of a slab with thickness $d$, all we have to do is to extend the mirrored source concept. We have to introduce an infinite series of positive and negative image sources along the $z$-axis, but in practice only the first few will have significant contributions to the solution. This means that we easily can truncate the series at the 5th or 6th image source, and not lose any valuable information (at least not until the 10th or so decimal).

## 3 Home Assignment

Calculate the fluence rate at a depth of 2.0 cm, 1 ns after a short laser pulse of energy $E_0 = 1$ $\mu$J irradiates a small spot at the surface of a semi-infinite homogenous medium. The optical properties are $n = 1.4$, $\mu_s' = 50$ cm$^{-1}$ and $\mu_a = 0.03$ cm$^{-1}$.

## 4 Instructions

During this exercise you will use MATLAB functions to illustrate analytical solutions to the diffusion equation. To run the MATLAB functions, open MATLAB and call them from your own MATLAB scripts (or the command window directly).

Before starting, you need to get the files used in the exercise. Create a working directory (folder), and download the files from the course homepage or directly from:

`www.atomic.physics.lu.se/fileadmin/atomfysik/Biophotonics/Education/diffusion.zip`

> IMPORTANT: It is highly recommended to write your MATLAB code in scripts instead of just typing them in the command prompt. For example, make a script for each part of the exercise, each making all the necessary computations and plots for the current assignments. This way your code may easily be reused and/or modified. If you don't know how to do this, please ask your instructor.

### 4.1 Analytical Solutions

For analytical solutions, a few different MATLAB functions are available.

#### 4.1.1 infinite.m

This function calculates the solution to the time-resolved diffusion equation in an infinite medium (without any boundaries). The geometry is indicated in figure 3.

Choose the input parameters as follows:

| | |
|---|---|
| Scattering Coefficient, $\mu_s'$ [1/m] | 5000 |
| Absorption Coefficient, $\mu_a$ [1/m] | 3 |
| Refractive index, $n$ | 1.4 |
| Source/detector separation [mm] | 10 |
| Start time [ns] | 0 |
| Time step [ns] | 0.01 |
| Stop time [ns] | 10 |

Thus, we will now consider the fluence rate (as a function of time, $\Phi = \Phi(t)$) at a distance 10 mm from an isotropic pulsed source.

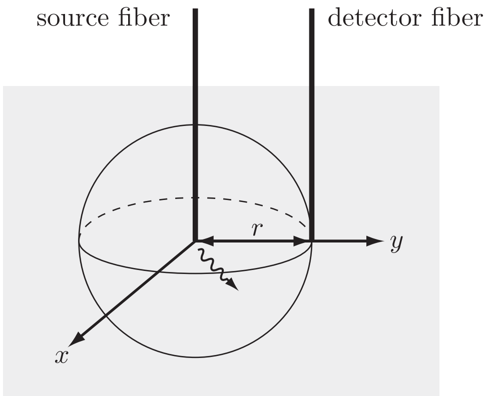


Figure 3: The infinite homogenous geometry. One optical fiber delivers light into the interior of some turbid media. A second fiber collects diffuse light (at a distance $r$).

**Assignments:**

1. Vary the absorption coefficient around the initial value in steps of a factor of two and try to explain the alterations in the shape. Plot all curves in the same figure to enable a good comparison. Also try to normalize all curves, and to change the $y$-axis to a logarithmic scale.
2. What are the curves representing? Units?
3. Both the intensity and shape of the curves are altered. Try to explain both effects in mathematical and physical terms.
4. Produce corresponding plots by instead altering the scattering coefficient. Explain!
5. How will the curve alter if you keep the optical properties constant, and instead vary the inter-fibre distance (distance between source and detector).
6. Discuss your observations and answers with your instructor.

#### 4.1.2 semi.m

All the effects you have observed so far depend on the optical properties and inter fibre distance only. In the following simulations we will also study the effect of the boundary conditions.

The MATLAB function `semi.m` computes the solution to the time-resolved diffusion equation in a semi-infinite medium (with one boundary). See figure 4! Because of the cylindrical symmetry in half-space geometries, this function uses cylindrical coordinates. The detector position, i.e., the position where we want to calculate the fluence rate, must thus be provided as a $\rho$-coordinate (the radial distance from the source) and a $z$-coordinate (the depth into the medium). This means that, for example, if you want to calculate the fluence in a point directly below the source, set $\rho = 0$. The source is always in $(\rho, z) = (0, 0)$, while the detector should be within the medium, $(z > 0, )$.

**Assignments:**

1. Use the same optical properties as you used earlier. First, probe $z = 10$ mm and $\rho = 0$. Compare the curve with what you get for infinite.m (with the same inter-fibre distance!).

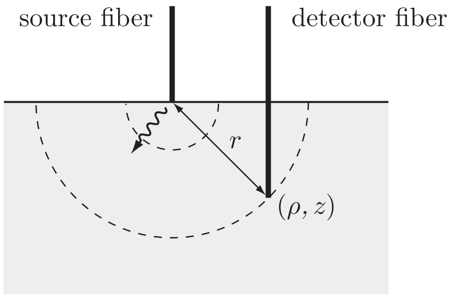


Figure 4: The semi-infinite geometry. One optical fiber delivers light onto the boundary of some turbid media. A second fiber probes diffuse light within the material, at a distance $r$). Note the cylindrical symmetry.

2. Now, move the detector closer and closer to the surface of the scattering volume, while keeping the inter-fibre distance, $r = \sqrt{\rho^2 + z^2}$, constant at 10 mm (move along the dotted line in figure 4). Explain your observations, again both in mathematical and physical terms.

3. Discuss your observations and answers with your instructor.

#### 4.1.3 slab.m

This function computes the solution to the time-resolved diffusion equation in a slab geometry. The only physical difference to `semi.m` is that we have added another boundary.

**Assignments:**

1. Plot the fluence rate in at $\rho = 0$ mm, $z = 10$ mm for different slab thicknesses (for example $d = 11$, 13 and 15 mm) and explain the differences. Also, compare with an infinitely thick (semi-infinite) slab. Discuss your observations and answers with your instructor.

#### 4.1.4 semi_ all.m and slab_ all.m

These scripts use the same geometry as `semi.m` and `slab.m`, respectively. The difference is that these MATLAB functions computes the fluence rate distribution in a plane perpendicular to the surface (and through the source) for as a function of time. Hence, these functions generate three dimensional datasets.

**Assignments:**

1. Write a script that generates movie sequences that illustrates the fluence evolution in the two cases. Such a script may look like this:

```
fluence=semi_all(...)
for i=1:nt
   imagesc((fluence(:,:,i))
   superdupermovie(i)=getframe
end
```

Play the movie by writing `movie(superdupermovie)`in the MATLAB Command Window.

2. Compare the differences in fluence rate for the two cases.

3. As stated earlier, the diffusion equation is not valid near the source. Nevertheless, the output from `semi_ all.m` and `slab_ all.m` shows results near the source as well. Think about what this means, and try to figure out how the fluence should look near the source in a real case.

4. Discuss your observations and answers with your instructor.

#### 4.1.5 slab_ out.m

This program is very similar to `slab.m`, but instead of giving the fluence rate somewhere inside the medium, it gives the flux across the boundary as given by Fick's law. This is the property that we would actually measure with a detector located outside the medium. Depending on whether the detector is placed on the same side as the source or the opposite side, we will get the reflected flux or the transmitted. Note that `slab_ out.m` can only provide reflectance data.

**Assignments:**

1. Calculate the reflectance for a number of different absorption and scattering coefficients, as you did for the fluence rate at an earlier stage in this exercise. Make a similar comparison. Explain the results.

2. Discuss your observations and answers with your instructor.

Finally, consider whether, and in that case how, you can utilize these tools for your project.

# 5 Appendix - MatLab Tools

This appendix provide information on how to use the MATLAB function described in section 2. Help on these tools are available also from MatLab! An example: type `help slab_all` for help on `slab_all.m`.

**infinite.m**

Calculates photon fluence as a function of time at a position $r$ in a infinite medium geometry (spherical symmetry) after an infinitely short input pulse.

**Syntax:**

```
fluence=infinite(mus,mua,n,r,t)
```

**Arguments:**

| | |
|---|---|
| `mus` | Reduced scattering coefficient [1/m] |
| `mua` | Absorption coefficient [1/m] |
| `n` | Refractive index |
| `r` | Radius where photon fluence are to be calculated [m] |
| `t` | Time vector [s] |

**Output:**

| | |
|---|---|
| `fluence` | Photon fluence as a function of time at radius `r` |

**Examples:**

Trying different absorptions...

```
>> t=(0:0.01:10)*1e-9;
>> f1=infinite(1500,10,1.4,0.02,t);
>> f2=infinite(1500,5,1.4,0.02,t);
>> f3=infinite(1500,1,1.4,0.02,t);
```

Plotting normalized versions...

```
>> plot(t*1e9,[f1./max(f3);f2./max(f3);f3./max(f3)])
```

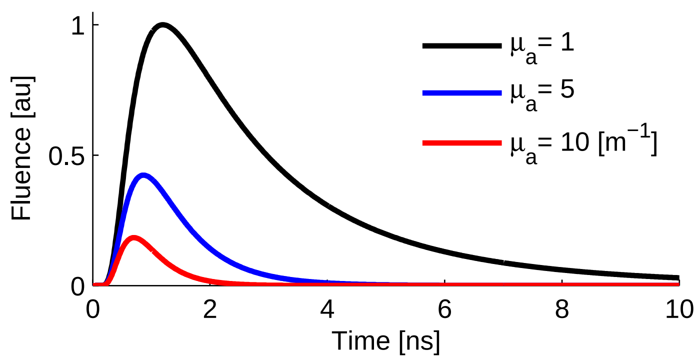


Figure 5: Fluence impulse responses for three different absorption levels in an homogenous infinite media.

## semi.m

Calculates photon fluence (impulse response) at a certain position $(\rho, z)$ as a function of time inside a semi-infinite geometry (cylindrical symmetry)

**Syntax:**

```
fluence=semi(mus,mua,n,z,r,t)
```

**Arguments:**

| | |
|---|---|
| `mus` | Reduced scattering coefficient [1/m] |
| `mua` | Absorption coefficient [1/m] |
| `n` | Refractive index |
| `r` | Detector radial location ($\rho$) [m] |
| `z` | Detector depth [m] |
| `t` | Time vector [s] |

**Output:**

| | |
|---|---|
| `fluence` | Photon fluence as a function of time at location (`r`,`z`) |

**Examples:**

A comparison with the infinite geometry

```
>> t=(0:0.01:4)*1e-9;
>> f_i=infinite(1500,10,1.4,0.02,t);
>> f_si=semi(1500,10,1.4,0.003,sqrt(0.02^2-0.003^2),t);
>> plot(t*1e9,[f_i./max(f_i);f_si./max(f_si)],'LineWidth',2)
```

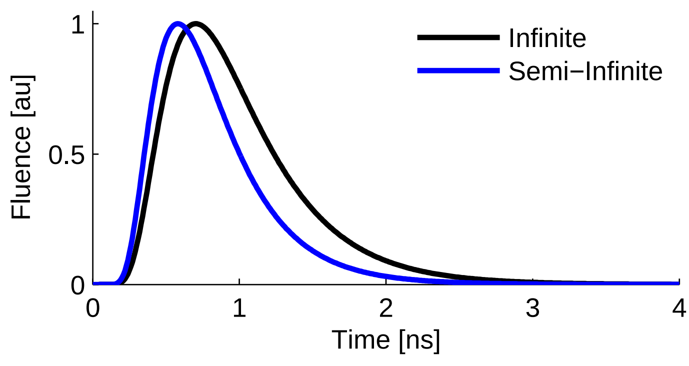


Figure 6: The influence of a boundary (according to the extrapolated boundary method).

## semi_all.m

Calculates photon fluence (impulse response) as a function of position $(\rho, z)$ and time inside a semi-infinite homogenous medium.

**Syntax:**

```
fluence=semi_all(mus,mua,n,z,r,t)
```

**Arguments:**

| | |
|---|---|
| `mus` | Reduced scattering coefficient [1/m] |
| `mua` | Absorption coefficient [1/m] |
| `n` | Refractive index |
| `z` | Detector depth range [m] |
| `r` | Detector radial range ($\rho$) [m] |
| `t` | Time vector [s] |

**Output:**

| | |
|---|---|
| `fluence` | A three dimensional array containing photon fluence at all times given by `t`, and at all mm positions within the range specified by `r` and `z`. The dimensions are, in order: $(z, r, t)$ |

**Examples:**

```
>> fluence=semi_all(1000,10,1.4,0.006,0.008,(0:0.01:1)*1e-9);
>> size(fluence)
ans =     7     9   301
```

Note that the data corresponds to the spatial coordinates ranges $z = 0 : 6$ (7 samples) and $r = 0 : 8$ (9 samples). Use the `squeeze`-function to extract time resolved data.

```
>> imagesc(fluence(:,:,2));set(gca,'DataAspectRatio',[1 1 1])
```

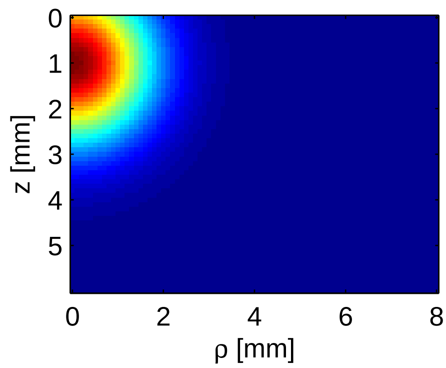


Figure 7: Cylindrically symmetric fluence in a semi-infinite medium at a certain time ($t = 0.01$ ns). Here, the resolution is 0.1 mm - not 1 mm as in the implementation you are using. To increase resolution, change the parameter `rf` in `semi_all.m`

## slab.m

Calculates photon fluence (impulse response) at a certain position $(\rho, z)$ as a function of time inside a slab (layer).

**Syntax:**

```
fluence=slab(d,mus,mua,n,z,r,t)
```

**Arguments:**

| | |
|---|---|
| `d` | Slab thickness [m] |
| `mus` | Reduced scattering coefficient [1/m] |
| `mua` | Absorption coefficient [1/m] |
| `n` | Refractive index |
| `r` | Detector radial location ($\rho$) [m] |
| `z` | Detector depth [m] |
| `t` | Time vector [s] |

**Output:**

| | |
|---|---|
| `fluence` | Photon fluence as a function of time at location (`r,z`) |

**Examples:**

A comparison with the infinite geometry

```
>> t=(0:0.01:2)*1e-9;
>> f_slab=slab(0.012,5000,2,1.4,0.01,0,t);
>> f_semi=semi(5000,2,1.4,0.01,0,t);
>> plot(t*1e9,[f_SLAB./max(f_SLAB);f_semi./max(f_semi)])
```

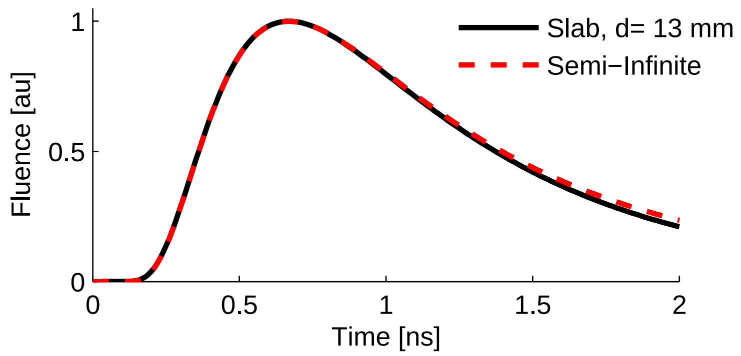


Figure 8: The influence of a second boundary - the slab geometry. The detector is located 10 mm beneath the source

## slab_all.m

Calculates photon fluence (impulse response) as a function of position $(\rho, z)$ and time inside a slab.

**Syntax:**

```
fluence=slab_all(d,mus,mua,n,r,t)
```

**Arguments:**

| | |
|---|---|
| `d` | Slab thickness [m] |
| `mus` | Reduced scattering coefficient [1/m] |
| `mua` | Absorption coefficient [1/m] |
| `n` | Refractive index |
| `r` | Detector radial range ($\rho$) [m] |
| `t` | Time vector [s] |

**Output:**

| | |
|---|---|
| `fluence` | A three dimensional array containing photon fluence at all times given by `t`, and at all mm positions within the range specified by `r`. When it comes to depth ($z$) every mm position within the slab is sampled (including borders $z = 0$ and $z = d$. The dimensions are, in order: $(z, r, t)$ |

**Examples:**

```
>> t=(0:0.01:0.4)*1e-9;f1=slab_all(0.01,1500,50,1.4,0.020,t);
>> size(f)
ans =     11     21     51
```

Compare with `slab.m`...

```
>> f2=slab(0.01,1500,50,1.4,0.006,0.007,t);
>> hold on;plot(t*1e9,squeeze(f1(7,8,:));plot(t*1e9,f2,'*r')
```

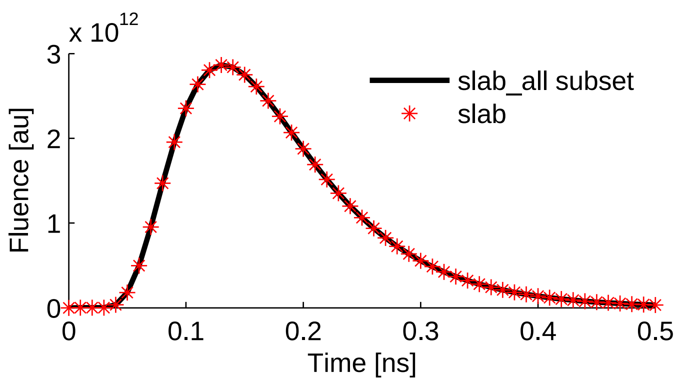


Figure 9: The output of `slab_all.m` contains time resolved impulse responses at several combinations of $\rho$ and $z$.

## slab_out.m

Calculates the reflectance (impulse response) at a certain radial location $\rho$ as a function of time in a slab geometry.

**Syntax:**

```
R=slab_out(d,mus,mua,n,r,t)
```

**Arguments:**

| | |
|---|---|
| `d` | Slab thickness [m] |
| `mus` | Reduced scattering coefficient [1/m] |
| `mua` | Absorption coefficient [1/m] |
| `n` | Refractive index |
| `r` | Detector radial location ($\rho$) [m] |
| `t` | Time vector [s] |

**Output:**

| | |
|---|---|
| `R` | Reflectance as a function of time at location $(\rho, z)$=(`r`,0) |

**Examples:**

```
R1=slab_out(0.05,1500,10,1.4,0.02,(0:0.01:5)*1e-9);
R2=slab_out(0.05,1500,10,1.4,0.025,(0:0.01:5)*1e-9);
R3=slab_out(0.05,1500,10,1.4,0.030,(0:0.01:5)*1e-9);
plot((0:0.01:5),[R1./max(R1);R2./max(R2);R3./max(R3)],'LineWidth',2)
```

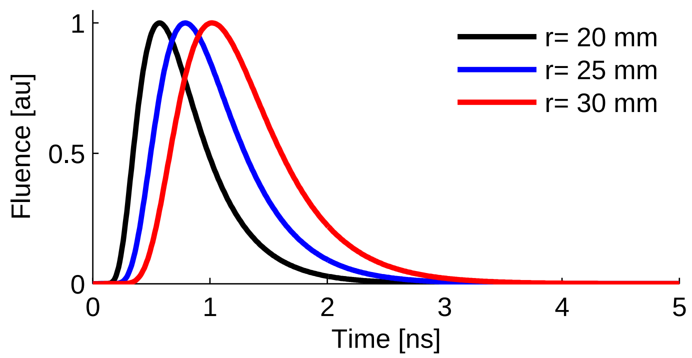


Figure 10: Diffuse reflectance and the influence of source-detector separation on impulse response. Note, that an increasing fibre separation will significantly lower signal levels (this is a normalized plot).