Table of Contents

Embarrassing Parallel GPU Greens Function Linear Super Position

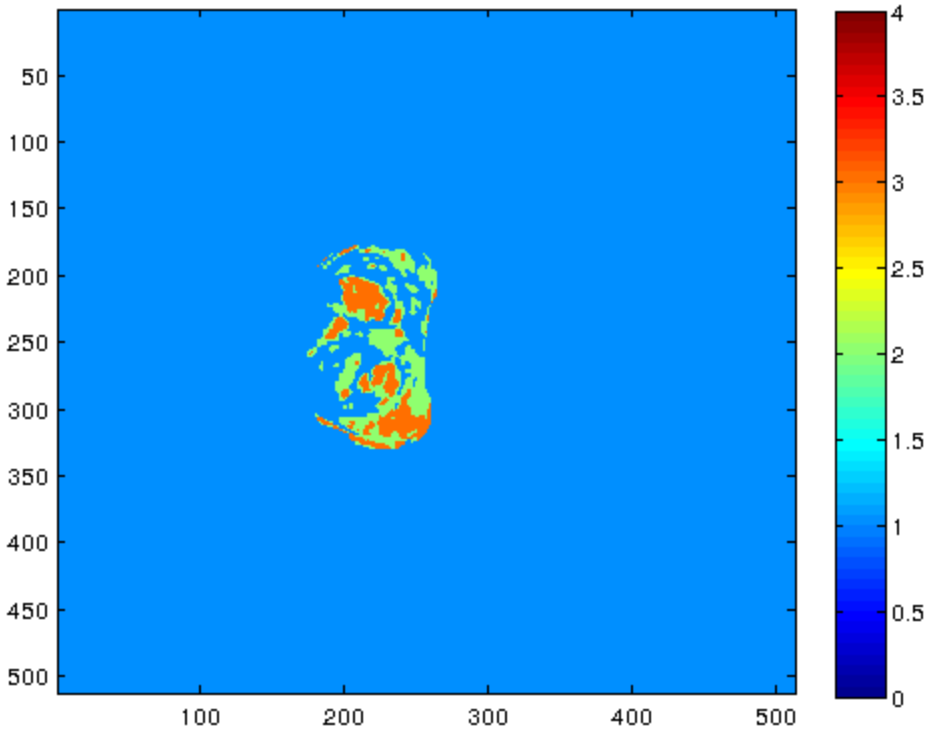
```
clear all
close all
format shortg
```

load tmp data structure

```
prostatedata =load_untouch_nii('prostateAxT2.nii.gz');
```

GMM tissue types

```
prostatelabel =load_untouch_nii('prostateLabel.nii.gz');
materialID = int32(prostatelabel.img);
materialID(materialID == 0  ) = 1;
materialID(materialID == 4  ) = 1;
materialID(materialID == 5  ) = 1;

[npixelx, npixely, npixelz] = size(materialID);
spacingX = prostatelabel.hdr.dime.pixdim(2)*1.e-3;
spacingY = prostatelabel.hdr.dime.pixdim(3)*1.e-3;
spacingZ = prostatelabel.hdr.dime.pixdim(4)*1.e-3;
idslice = 11;
handle5 = figure(5);
imagesc(materialID(:,:,idslice ),[0 4])
colorbar
```



Query the device

GPU must be reset on out of bounds errors reset(gpuDevice(1))

```
deviceInfo = gpuDevice(1);
numSMs = deviceInfo.MultiprocessorCount;
```

mueff at each wave length

TODO - error check same length

```
mueffHHb   = [7.e02 , 4.e02];
mueffHbO2  = [5.e02 , 6.e02];
sigmamueff = 30;
```

Setup Material Parameters

```
ntissue = 4;
nsource    = 10;
idxpix =250; idypix =200;
xloc       = idxpix *spacingX+spacingX*linspace(1,nsource ,nsource )+1.e-3;
yloc       = idypix *spacingY+spacingY*linspace(1,nsource ,nsource )+1.e-3;
zloc       = idslice  *spacingZ+spacingZ*linspace(1,nsource ,nsource )+1.e-3;
```

```
zloc       = idslice  *spacingZ*ones(1,10)+1.e-3;
power      = 10.;
```

initialize data arrays

initialize on host and perform ONE transfer from host to device

```
h_pasource     = zeros(npixelx,npixely,npixelz);
d_pasource      = gpuArray( h_pasource  );
```

Compile and setup thread grid

grid stride loop design pattern, 1-d grid http://devblogs.nvidia.com/parallelforall/cuda-pro-tip-write-flexible-kernels-grid-stride-loops/

```
ssptx = parallel.gpu.CUDAKernel('sdaFluenceModel.ptx', 'sdaFluenceModel.cu');
threadsPerBlock = 256;
ssptx.ThreadBlockSize=[threadsPerBlock  1];
ssptx.GridSize=[numSMs*32               1];
```

Run on GPU

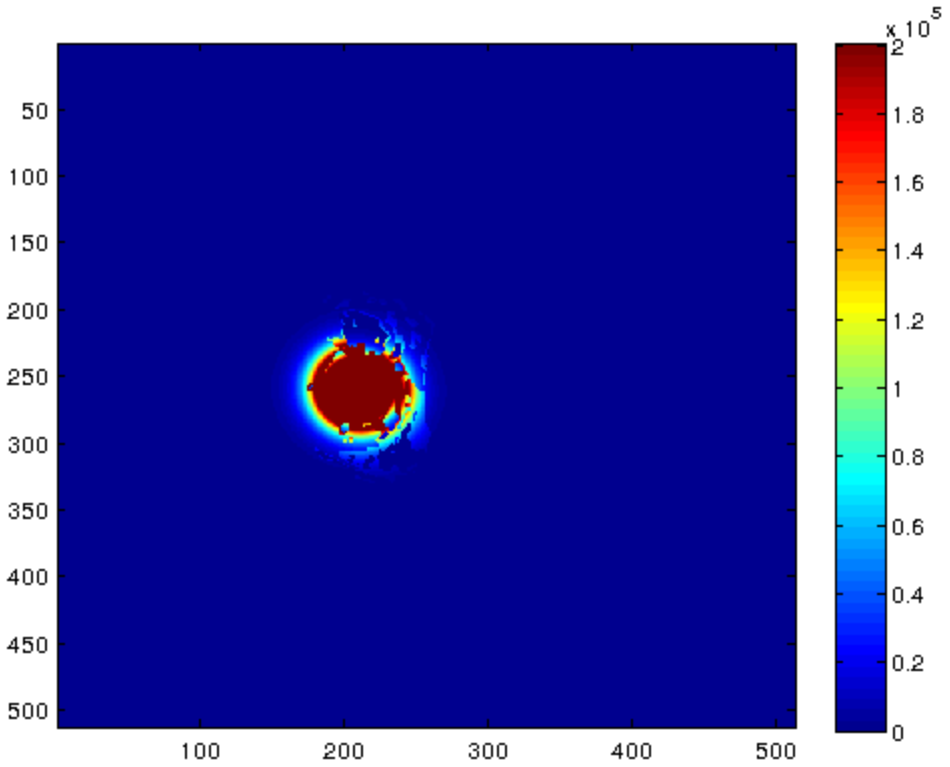
```
mueff      = [5.e02 , 4.e02 , 3.e02];
[d_pasource ] = feval(ssptx,ntissue,materialID, mueff, nsource, power ,xloc,yloc,z
```

transfer device to host

```
h_pasource  = gather( d_pasource  );
```

plot pasource

```
handle6 = figure(6);
imagesc(h_pasource(:,:,idslice), [0 2.e5]);
colormap default
colorbar
```



UQ grid

```
mixinggrid = [.2, .4, .6, .8]
```

mixinggrid =

0.2 0.4 0.6 0.8

global search and plot exhaustive search

```
tic
statfigurenames = {'meanpasourcelo','varpasourcelo','meanpasourcehi','varpasourceh
for idLambda = 1:numel(mueffHbO2)

%for idLambda = 1:1
```

UQ grid

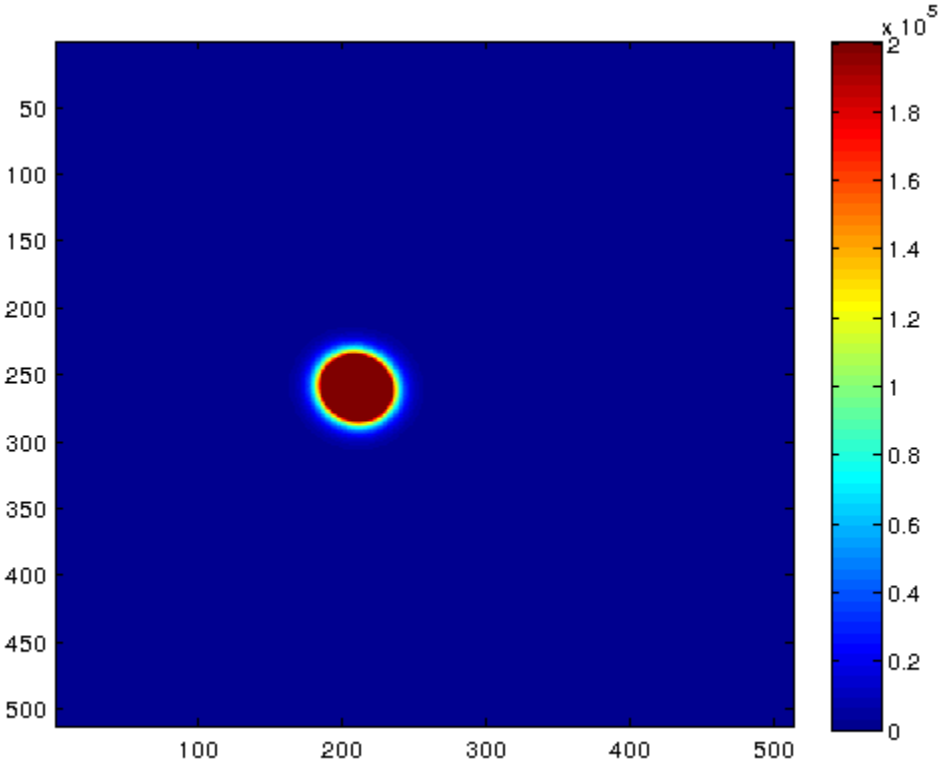
```
  HHbgrid  =  sigmamueff * [-2,-1,0,1,2] + mueffHHb(idLambda)
  HbO2grid =  sigmamueff * [-2,-1,0,1,2] + mueffHbO2(idLambda)
  TotalIter = numel(mixinggrid) * numel(mixinggrid) * numel(mixinggrid) * numel(HH
  IterCount = 0;
  % mean
```

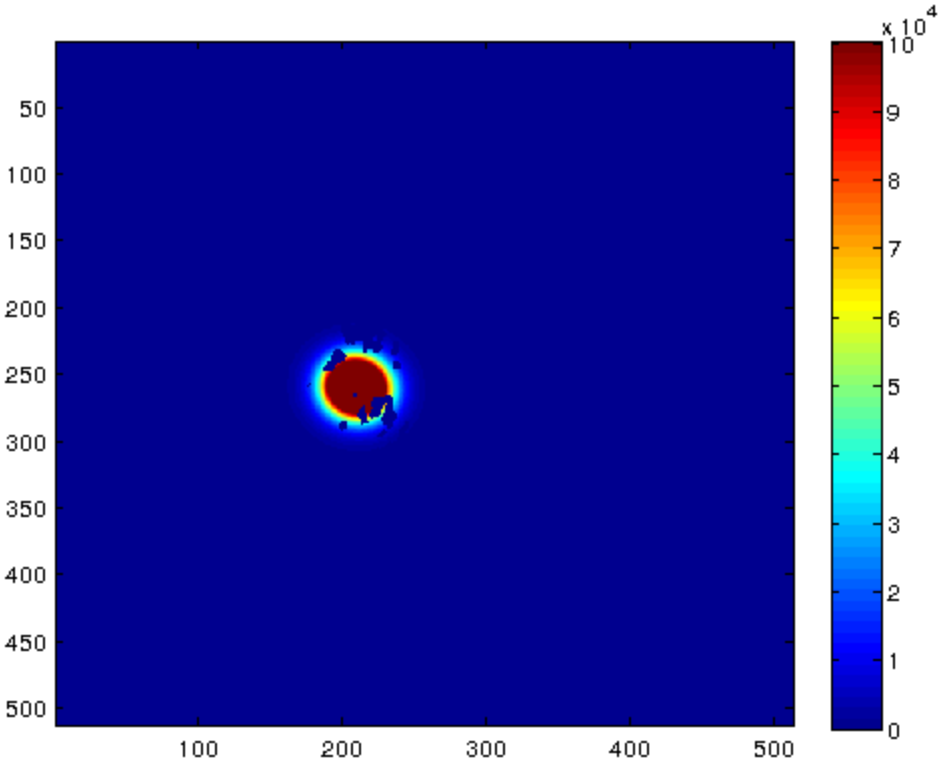
```
 d_pasourceMean  = gpuArray( zeros(npixelx,npixely,npixelz)  );
 for idOne = 1: numel(mixinggrid)
   % show iterations
   disp(sprintf('iter %d',idOne));
   for idTwo = 1: numel(mixinggrid)
     for idThree = 1: numel(mixinggrid)
       for idFour = 1: numel(HHbgrid)
         for idFive = 1: numel(HbO2grid)
            mueff = [   mixinggrid(idOne)   *HHbgrid( idFour)+ (1-mixinggrid(idOn
                        mixinggrid(idTwo)   *HHbgrid( idFour)+ (1-mixinggrid(idTw
                        mixinggrid(idThree) *HHbgrid( idFour)+ (1-mixinggrid(idTh
            IterCount =  IterCount +1;
            if mod(IterCount,100 )==0
              disp(sprintf(' %d %f %f %f ', IterCount, mueff));
            end
            [p_pasource] = feval(ssptx,ntissue,materialID, mueff, nsource, power
            d_pasourceMean  = d_pasourceMean  + 1./TotalIter * p_pasource;
 end
 end
 end
 end
 end
 IterCount = 0;
 % variance
 d_pasourceVar  = gpuArray( zeros(npixelx,npixely,npixelz)  );
 for idOne = 1: numel(mixinggrid)
   % show iterations
   disp(sprintf('iter %d',idOne));
   for idTwo = 1: numel(mixinggrid)
     for idThree = 1: numel(mixinggrid)
       for idFour = 1: numel(HHbgrid)
         for idFive = 1: numel(HbO2grid)
            mueff = [   mixinggrid(idOne)   *HHbgrid( idFour)+ (1-mixinggrid(idOn
                        mixinggrid(idTwo)   *HHbgrid( idFour)+ (1-mixinggrid(idTw
                        mixinggrid(idThree) *HHbgrid( idFour)+ (1-mixinggrid(idTh
            IterCount =  IterCount +1;
            if mod(IterCount,100 )==0
              disp(sprintf(' %d %f %f %f ', IterCount, mueff));
            end
            [p_pasource] = feval(ssptx,ntissue,materialID, mueff, nsource, power
            d_pasourceVar= d_pasourceVar + 1./(TotalIter-1) * (p_pasource - d_pas
end
end
end
end
end
% gather
host_Mean = gather(d_pasourceMean );
host_Std  = sqrt(gather(d_pasourceVar  ));

% plot
idplot  =  2*(idLambda-1) +1;
handleid = figure( idplot   );
imagesc(host_Mean(:,:,idslice), [0 2.e+05]);
```

```
colormap default
colorbar
saveas(handleid,statfigurenames{idplot },'png')
prostatedata.img =host_Mean; prostatedata.hdr.dime.datatype=64;
save_untouch_nii(prostatedata, [statfigurenames{idplot },'.nii.gz']);

% plot
idplot =  2*idLambda;
handleid = figure(  idplot );
% gather
imagesc(host_Std(:,:,idslice), [0 1.e5]);
colormap default
colorbar
saveas(handleid,statfigurenames{idplot },'png')
prostatedata.img =host_Std; prostatedata.hdr.dime.datatype=64;
save_untouch_nii(prostatedata, [statfigurenames{idplot },'.nii.gz']);
```

```
HHbgrid =

   640   670   700   730   760


HbO2grid =

   440   470   500   530   560

iter 1
 100 600.000000 600.000000 720.000000
 200 600.000000 640.000000 720.000000
 300 600.000000 680.000000 720.000000
 400 600.000000 720.000000 720.000000
iter 2
 500 640.000000 600.000000 720.000000
 600 640.000000 640.000000 720.000000
 700 640.000000 680.000000 720.000000
 800 640.000000 720.000000 720.000000
iter 3
 900 680.000000 600.000000 720.000000
 1000 680.000000 640.000000 720.000000
 1100 680.000000 680.000000 720.000000
 1200 680.000000 720.000000 720.000000
iter 4
 1300 720.000000 600.000000 720.000000
 1400 720.000000 640.000000 720.000000
 1500 720.000000 680.000000 720.000000
 1600 720.000000 720.000000 720.000000
iter 1
 100 600.000000 600.000000 720.000000
 200 600.000000 640.000000 720.000000
 300 600.000000 680.000000 720.000000
 400 600.000000 720.000000 720.000000
iter 2
 500 640.000000 600.000000 720.000000
```

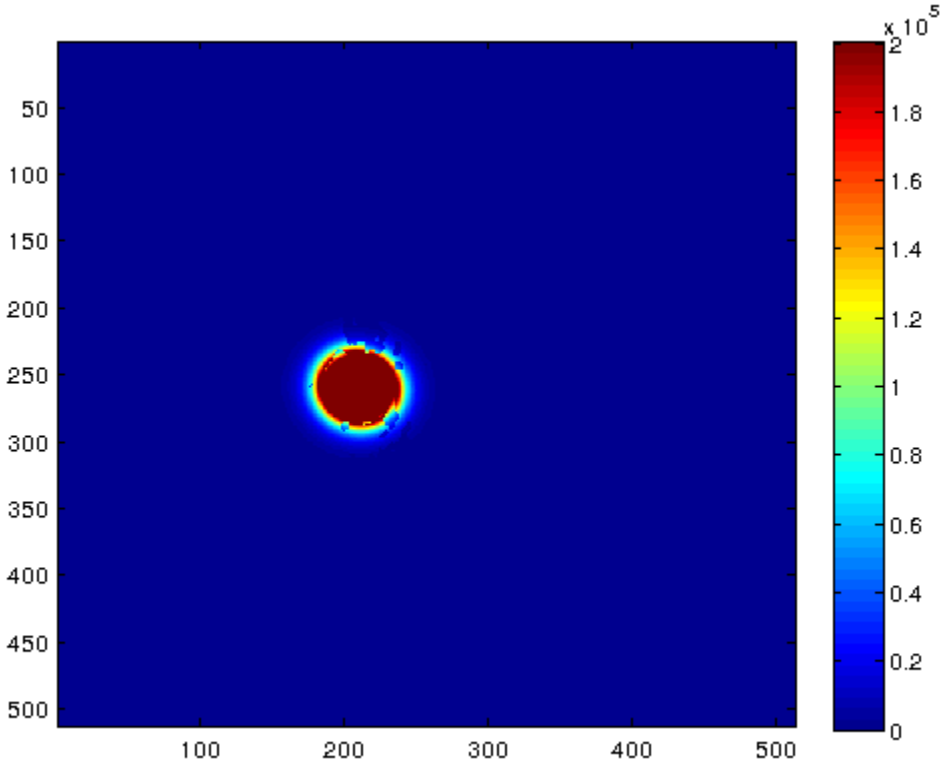
```
 600 640.000000 640.000000 720.000000

 700 640.000000 680.000000 720.000000
 800 640.000000 720.000000 720.000000
iter 3
 900 680.000000 600.000000 720.000000
 1000 680.000000 640.000000 720.000000
 1100 680.000000 680.000000 720.000000
 1200 680.000000 720.000000 720.000000
iter 4
 1300 720.000000 600.000000 720.000000
 1400 720.000000 640.000000 720.000000
 1500 720.000000 680.000000 720.000000
 1600 720.000000 720.000000 720.000000
```

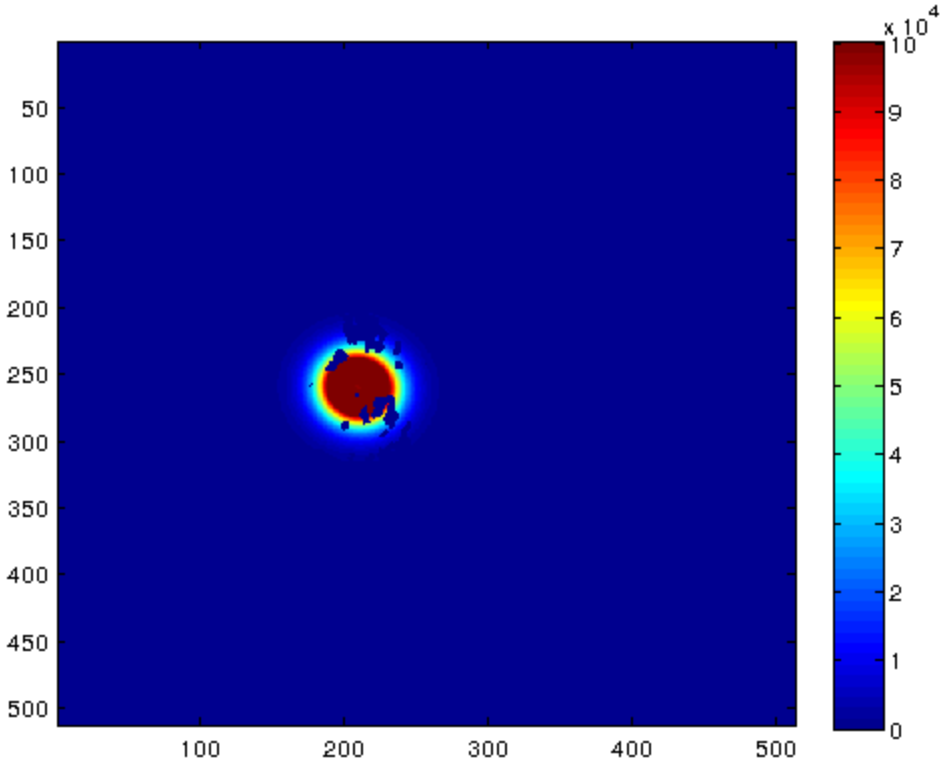





```
HHbgrid =

   340   370   400   430   460


HbO2grid =

   540   570   600   630   660

iter 1
 100 620.000000 620.000000 500.000000
 200 620.000000 580.000000 500.000000
 300 620.000000 540.000000 500.000000
 400 620.000000 500.000000 500.000000
iter 2
 500 580.000000 620.000000 500.000000
 600 580.000000 580.000000 500.000000
 700 580.000000 540.000000 500.000000
 800 580.000000 500.000000 500.000000
iter 3
 900 540.000000 620.000000 500.000000
 1000 540.000000 580.000000 500.000000
 1100 540.000000 540.000000 500.000000
 1200 540.000000 500.000000 500.000000
iter 4
```

```
 1300 500.000000 620.000000 500.000000

 1400 500.000000 580.000000 500.000000
 1500 500.000000 540.000000 500.000000
 1600 500.000000 500.000000 500.000000
iter 1
 100 620.000000 620.000000 500.000000
 200 620.000000 580.000000 500.000000
 300 620.000000 540.000000 500.000000
 400 620.000000 500.000000 500.000000
iter 2
 500 580.000000 620.000000 500.000000
 600 580.000000 580.000000 500.000000
 700 580.000000 540.000000 500.000000
 800 580.000000 500.000000 500.000000
iter 3
 900 540.000000 620.000000 500.000000
 1000 540.000000 580.000000 500.000000
 1100 540.000000 540.000000 500.000000
 1200 540.000000 500.000000 500.000000
iter 4
 1300 500.000000 620.000000 500.000000
 1400 500.000000 580.000000 500.000000
 1500 500.000000 540.000000 500.000000
 1600 500.000000 500.000000 500.000000
```






```
end
toc

saveas(handle5,'material','png')
saveas(handle6,'pasource','png')
```

```
statfigurenames =

    'meanpasourcelo'    'varpasourcelo'    'meanpasourcehi'    'varpasourc

Elapsed time is 645.561159 seconds.
```

Published with MATLAB® R2014a