Supplement of

A Consistent Prescription of Stratospheric Aerosol for Both Radiation and Chemistry in the Community Earth System Model (CESM1)

R. R. Neely III et al.

Correspondence to: R. R. Neely III (r.neely@leeds.ac.uk)

The copyright of individual parts of the supplement might differ from the CC-BY 3.0 licence.

S1. Mie Calculations for Optical Look up Table Code used in CESM1(CAM4)

```
function mielookupgenerationcam4simplebands
%Written using Matlab 2014b and utilizing the % Mie code of Matzler [2002]
load('volcsulfrefind75-25.mat'); %Load volcanic aerosol refractive indices from the GEISA spectroscopic database
r= .1E-9:.1E-9:2000E-9; %Specify the range of radii
lambda(1,:)= [200E-9,245E-9,265E-9,275E-9,285E-9,295E-9,305E-9,350E-9,640E-9,...
    2630E-9,4160E-9,4160E-9]; %Specify the wavenumber of each band's upper limit
lambda(3,:)= [245E-9,265E-9,275E-9,285E-9,295E-9,305E-9,350E-9,640E-9,700E-9,...
    2860E-9,4550E-9,4550E-9]; %Specify the wavenumber of each band's lower limit
lambda(2,:)= (lambda(1,:)+lambda(3,:))./2; %Calculate the midpoint each band;
massfactor=3./(4.*1.7.*1E3.*r); %Mass conversion factor for each radii
for range=1:3;% For loop for the upper mid and lower wavelengths.
    for i=1:length(lambda); %For loop for each band within the upper lower and mid point ranges.
        x=(2*pi./(squeeze(lambda(range,i))))).*r; %Calculate the size parameter.
        Realindex=feval(realindfittedmodel,lambda(range,i)./1E-6); %Convert the real index of refraction to correct units.
        Imagindex=feval(imindfittedmodel,lambda(range,i)./1E-6); %Convert the imaginary index of refraction to correct units.
        m=complex(Realindex,Imagindex); %Use the real and imaginary indices of refraction to create the complex index of refraction
        % Below is copied from Matzler [2002] to use the information above to calculate the Mie parameters
        j=i;
        for i=1:length(x)
            nmax=round(2+x(i)+4*x(i)^(1/3))
            n1=nmax-1;
            n=(1:nmax);
            cn=2*n+1; c1n=n.*(n+2)./(n+1); c2n=cn./n./(n+1);
            x2=x(i)*x(i);
            f=Mie_abcd(m,x(i));
            anp=(real(f(1,:)));
            anpp=(imag(f(1,:)));
            bnp=(real(f(2,:)));
            bnpp=(imag(f(2,:)));
            g1(1:4,nmax)=[0; 0; 0; 0]; % displaced numbers used for
            g1(1,1:n1)=anp(2:nmax); g1(2,1:n1)=anpp(2:nmax); g1(3,1:n1)=bnp(2:nmax); g1(4,1:n1)=bnpp(2:nmax);
            dn=cn.*(anp+bnpp);
            q=sum(dn);
            qext=2*q/x2;
            en=cn.*(anp.*anp+anpp.*anpp+bnp.*bnp+bnpp.*bnpp);
            q=sum(en);
            qsca=2*q/x2;
            qabs=qext-qsca;
            fn=(f(1,:)-f(2,:)).*cn;
            gn=(-1).^n;
            f(3,:)=fn.*gn;
            q=sum(f(3,:));
            qb=q*/x2;
            asy1=c1n.*(anp.*g1(1,:)+anpp.*g1(2,:)+bnp.*g1(3,:)+bnpp.*g1(4,:)); asy2=c2n.*(anp.*bnp+anpp.*bnpp);
            asy=4/x2*sum(asy1+asy2)/qsca;
            %Individual calculated parameters are saved to an array.
            qratio=qb/qsca;
            Qb(range,i,j)=qb;
            Qext(range,i,j)=qext;
            Qabs(range,i,j)=qabs;
            Qsca(range,i,j)=qsca;
            X(range,i,j)=x(i);
            G(range,i,j)=asy;
            indref(range,i,j)=m;
        end
        clearvars -except Qb r X m Qext Qsca Qabs asy lambda massfactor G range indref realindfittedmodel imindfittedmodel
    end
end
w=Qsca./Qext;
% Array is saved to mat format file.
savefilename=sprintf('/Users/rneely/Dropbox/camRTmieparameters1-9and17-19.mat');
savefile=[savefilename];
save(savefile,'Qb','r','X','indref','Qext','Qsca','Qabs','G','lambda','w','massfactor')
```

S2. Mie Calculations for Optical Look up Table Code CESM1(CAM5)

```
function mielookupgenerationrrtmg
%Written using Matlab 2014b and utilizing the % Mie code of Matzler [2002]
load('volcsulfrefind75-25.mat'); %Load volcanic aerosol refractive indices from the GEISA spectroscopic database
r=.1E-9:.1E-9:2000E-9; %Specify the range of radii
%Shortwave Bands
%comment out next 3 lines calculate long wave parameter file
% Specify the wavenumber of each bands upper limit
lambdawn(1,:)= [ 2600, 3250, 4000, 4650, 5150, 6150, 7700, 8050,12850,16000,22650,29000,38000, 820];
%Specify the wavenumber bands lower limit
lambdawn(3,:)= [ 3250, 4000, 4650, 5150, 6150, 7700, 8050,12850,16000,22650,29000,38000,50000, 2600];
% Calculate the midpoint of the each band;
lambdawn(2,:)=(lambdawn(1,:)+lambdawn(3,:))./2;
%lw
% Uncomment out next 3 lines calculate long wave parameter file
% Specify the wavenumber of each bands upper limit
% lambdawn(1,:)= [10, 350, 500, 630, 700, 820, 980, 1080,1180, 1390, 1480, 1800, 2080, 2250, 2390, 2600];
%Specify the wavenumber bands lower limit
% lambdawn(3,:)= [350, 500, 630, 700, 820, 980, 1080, 1180,1390, 1480, 1800, 2080, 2250, 2390, 2600, 3250];
% Calculate the midpoint of the each band;
% lambdawn(2,:)=(lambdawn(1,:)+lambdawn(3,:))./2
lambda=1e-2./lambdawn; % Convert wavenumber to wavelength
massfactor=3./(4.*1.75.*1E3.*r);
for range=1:3; % For loop for the upper mid and lower wavelengths.
for i=1:length(lambda); % For loop for each band within the upper lower and mid point ranges.
x=(2*pi./(squeeze(lambda(range,i))))).*r; Calculate the size parameter.
% Convert the imaginary index of refraction to correct units.
Realindex=realind(find(realmicron.*1E-6<=lambda(range,i),1,'first'));
Imagindex=imind(find(immicron.*1E-6<=lambda(range,i),1,'first'));
if isempty(Imagindex)==1
    Imagindex=4e-6;
end
if isempty(Realindex)==1
    Realindex=1.45;
end
lambda(range,i);
% Use the real and imaginary indices of refraction to create the complex index of refraction
m=complex(Realindex,Imagindex);
% Below is copied from Matzler [2002] to use the information above to calculate the Mie parameters
j=i;
for i=1:length(x)
nmax=round(2+x(i)+4*x(i)^(1/3))
n1=nmax-1;
n=(1:nmax);
cn=2*n+1; c1n=n.*(n+2)./(n+1); c2n=cn./n./(n+1);
x2=x(i)*x(i);
f=Mie_abcd(m,x(i));
anp=(real(f(1,:)));
anpp=(imag(f(1,:)));
bnp=(real(f(2,:)));
bnpp=(imag(f(2,:)));
g1(1:4,nmax)=[0; 0; 0; 0]; % displaced numbers used for
g1(1,1:n1)=anp(2:nmax); g1(2,1:n1)=anpp(2:nmax); g1(3,1:n1)=bnp(2:nmax); g1(4,1:n1)=bnpp(2:nmax);
dn=cn.*(anp+bnp);
q=sum(dn);
qext=2*q/x2;
en=cn.*(anp.*anp+anpp.*anpp+bnp.*bnp+bnpp.*bnpp);
q=sum(en);
qsca=2*q/x2;
qabs=qext-qsca;
fn=(f(1,:)-f(2,:)).*cn;
gn=(-1).^n;
f(3,:)=fn.*gn;
q=sum(f(3,:));
qb=q.*q/x2;
asy1=c1n.*(anp.*g1(1,:)+anpp.*g1(2,:)+bnp.*g1(3,:)+bnpp.*g1(4,:)); asy2=c2n.*(anp.*bnp+anpp.*bnpp);
asy=4/x2*sum(asy1+asy2)/qsca;
qratio=qb/qsca;
%Individual calculated parameters are saved to an array.
Qb(range,i,j)=qb;
Qext(range,i,j)=qext;%*massfactor(i);
Qabs(range,i,j)=qabs;%*massfactor(i);
Qsca(range,i,j)=qsca;%*massfactor(i);
X(range,i,j)=x(i);
G(range,i,j)=asy;
indref(range,i,j)=m;
end
clearvars -except Qb r X m Qext Qsca Qabs asy lambda massfactor G range indref realmicron immicron realind imind
end
end
w=Qsca./Qext;
% Array is saved to mat format file.
savefilename=sprintf('/Users/rneely/Dropbox/camRRTMGsw.mat'); %change sw to lw to save long wave parameter file
savefile=[savefilename];
save(savefile,'Qb','r','X','indref','Qext','Qsca','Qabs','G','lambda','w','massfactor')
```