**Supplementary Information**

**Effective Properties of Semi-transparent Radiative Cooling Materials With Spectrally Variable Properties**

Ravi Anant Kishore+,\*, Chuck Booten+,\*, and Sajith Wijesuriya

National Renewable Energy Laboratory, 15013 Denver West Parkway, Golden, CO 80401.

+ These authors contributed equally to the work.

\* Corresponding authors: [ravi.kishore@nrel.gov](mailto:ravi.kishore@nrel.gov) (RAK) and [Chuck.Booten@nrel.gov](mailto:Chuck.Booten@nrel.gov) (CB)

**MATLAB Code for Iterative Solver**

fun=@EffectiveProperties;

x0=[0 0 0 0];

lb=[0 0 0 0];

ub=[1 1 1 1];

A=[];

b=[];

Aeq=[];

beq=[];

[x,fval] = fmincon(fun,x0,A,b,Aeq,beq,lb,ub);

function Error = EffectiveProperties(x)

Eff\_e1\_os=x(1); %Outer surface effective emissivity in solar region

Eff\_t1\_os=x(2); %Outer surface effective transmissivity in solar region

Eff\_e2\_os=x(3); %Outer surface effective emissivity in thermal region

Eff\_t2\_os=x(4); %Outer surface effective transmissivity in thermal region

Eff\_r1\_os=1-(Eff\_e1\_os+Eff\_t1\_os);

%Outer surface effective reflectivity in solar region

Eff\_r2\_os=1-(Eff\_e2\_os+Eff\_t2\_os);

%Outer surface effective reflectivity in thermal region

%% Constant Parameters

T\_solar=5778; %Sun temperature (K)

T\_surface=300; %Surface temperature (K)

T\_sky=250; %Sky temperature (K)

C1=3.74E+08; %Constant (W\*um^4/m^2)

C2=1.44E+04; %Constant (um\*K)

stephan\_boltzman=5.67E-08; %Constant (W\*K/m^2)

%% Input variables

lambda1=xlsread('Input','B3:B30');%Imported wavelength (um)

e1\_os=xlsread('Input','C3:C30'); %Imported emissivity of outer surface

t1\_os=xlsread('Input','E3:E30');

%Imported transmissivity of outer surface

r1\_os=xlsread('Input','G3:G30');

%Imported reflectivity of outer surface

e\_is=0.91; %Emissivity of inner surface

r\_is=1-0.91; %Reflectivity of inner surface

%% Interpolation variables

dl=0.1; %Wavelength resolution (um)

lambda\_max=100; %Maximum wavelength in this study (um)

lambda=dl:dl:lambda\_max; %Wavelength distribution (um)

e\_os=interp1(lambda1,e1\_os,lambda)';

%Spectral variable emissivity of outer surface

t\_os=interp1(lambda1,t1\_os,lambda)';

%Spectral variable transmissivity of outer surface

r\_os=interp1(lambda1,r1\_os,lambda)';

%Spectral variable reflectivity of outer surface

%% Blackbody Sun

Total\_solar=961; %Total solar power used for scaling W/m^2

I\_solar=C1./(lambda.^5.\*(exp(C2./lambda./T\_solar)-1))';

%Radiation intensity of the blackbody Sun (W/m^2/um)

P\_solar=I\_solar\*dl; %Solar irradiance (W/m^2)

Q\_solar=P\_solar\*Total\_solar/sum(P\_solar);

%Scaled solar irradiance (W/m^2)

%% Ambient/sky

lambda\_a=xlsread('Input','Atm','B2:B288');

%Imported sky wavelength (um)

e\_atm1=xlsread('Input','Atm','D2:D288');

%Imported sky emissivity

e\_atm=interp1(lambda\_a,e\_atm1,lambda)';

%Spectral variable sky emissivity

I\_sky=C1./(lambda.^5.\*(exp(C2./lambda./T\_sky)-1))';

%Blackbody Radiation intensity from sky (W/m^2/um)

%% Radiative Surface

I\_surface=C1./(lambda.^5.\*(exp(C2./lambda./T\_surface)-1))';

%Blackbody radiation intensity from surface W/m^2/um)

Q\_rad=(I\_surface-e\_atm.\*I\_sky)\*dl;

%Blackbody radiation exchange between surface and sky (W/m2)

%% Adding indirect radiations

count=lambda\_max/dl; %Counting variable

Transmit=zeros(1,count)'; %Transmittance

Qnet1\_os=zeros(1,count)';

%Net heat transfer on outer surface in solar region (<2.5 um)

Qnet2\_os=zeros(1,count)';

%Net heat transfer on outer surface in thermal region (>2.5 um)

Qnet1\_is=zeros(1,count)';

%Net heat transfer on inner surface in solar region (<2.5 um)

Qnet2\_is=zeros(1,count)';

%Net heat transfer on inner surface in thermal region (>2.5 um)

%%Net radiation in solar and thermal regions using spectral properties

i=0;

for i=1:1:count

Transmit(i)=t\_os(i)/(1-r\_os(i)\*r\_is);

if i<=2.5/dl

Qnet1\_os(i)=e\_os(i)\*(1+Transmit(i)\*r\_is)\*(Q\_solar(i)-Q\_rad(i));

Qnet1\_is(i)=e\_is\*Transmit(i)\*(Q\_solar(i)-Q\_rad(i));

else

Qnet2\_os(i)=e\_os(i)\*(1+Transmit(i)\*r\_is)\*(Q\_solar(i)-Q\_rad(i));

Qnet2\_is(i)=e\_is\*Transmit(i)\*(Q\_solar(i)-Q\_rad(i));

end

i=i+1;

end

%%Net radiation in solar and thermal regions using effective properties

Eff\_Qnet1\_os=zeros(1,count)';

Eff\_Qnet2\_os=zeros(1,count)';

Eff\_Qnet1\_is=zeros(1,count)';

Eff\_Qnet2\_is=zeros(1,count)';

i=0;

for i=1:1:count

if i<=2.5/dl

Eff\_transmit1=Eff\_t1\_os/(1-Eff\_r1\_os\*r\_is);

Eff\_Qnet1\_os(i)=Eff\_e1\_os\*(1+Eff\_transmit1\*r\_is)\*(Q\_solar(i)-Q\_rad(i));

Eff\_Qnet1\_is(i)=e\_is\*Eff\_transmit1\*(Q\_solar(i)-Q\_rad(i));

else

Eff\_transmit2=Eff\_t2\_os/(1-Eff\_r2\_os\*r\_is);

Eff\_Qnet2\_os(i)=Eff\_e2\_os\*(1+Eff\_transmit2\*r\_is)\*(Q\_solar(i)-Q\_rad(i));

Eff\_Qnet2\_is(i)=e\_is\*Eff\_transmit2\*(Q\_solar(i)-Q\_rad(i));

end

i=i+1;

end

SumQnet1\_os=sum(Qnet1\_os);

SumQnet2\_os=sum(Qnet2\_os);

SumQnet1\_is=sum(Qnet1\_is);

SumQnet2\_is=sum(Qnet2\_is);

SumEffQnet1\_os=sum(Eff\_Qnet1\_os);

SumEffQnet2\_os=sum(Eff\_Qnet2\_os);

SumEffQnet1\_is=sum(Eff\_Qnet1\_is);

SumEffQnet2\_is=sum(Eff\_Qnet2\_is);

%%Difference (Error) between the two methods

Error=abs(SumQnet1\_os-SumEffQnet1\_os)+abs(SumQnet2\_os-SumEffQnet2\_os)+abs(SumQnet1\_is-SumEffQnet1\_is)+abs(SumQnet2\_is-SumEffQnet2\_is);

end