Contents


```
clear all
clc
```

Code compiled by Roshan R Rao

relevant INPUT to the model

```
year=2022
month=03
date=20
hour=07                              % local clocktime
min=0                                % local clocktime
sec=0                                % local clocktime

timezone=5.5                         % time zone of the location
lat=13                               % latitude (decimal) positive is east
lon=77                               % latitude (decimal) positive is north

k=1.4                                % bulk thermal conductivity W/mK
pveff=0.14;                          % rated pv conversion efficiency (100% = 1)
sc_cond=1                            % short circuit condition sc_cond==1 for short circuit, 0 otherwise

beta=15;                             % tilt angle in degrees
azim_w=0;                            % wall azimuth angle (degrees)

velup=0.1;                     % air velocity above panel (m/s)
velbo=0.1;                     % air velocity below panel (m/s)
ghi_py=900                     % pyranometer measured GHI W/m2

Tground=26.85;                % ground (below PV) in deg. C
Tambup=28;                    % amb. air above panel in in deg. C
Tambbo=30;                    % amb. air below panel in in deg. C
```

```
year =

        2022


month =

     3


date =

    20


hour =

     7


min =

     0


sec =

     0


timezone =

    5.5000


lat =
```

```
    13


lon =

    77


k =

    1.4000


sc_cond =

     1


ghi_py =

   900
```

ATMOSPHERIC VARIABLES

```
P=924;                       % pressure in mb              %
W=2.23;                      % precipitable water vapor (cm)         %
ho=22;                       % height of maximum ozone concentration (km)    %
o3=0.3;                      % ozone amount atm-cm                           %
rg=0.3;                      % ground reflectivity for atm rad transfer      %
```

kelvin conversions

```
Tground=Tground+273.15;                 % in KELVIN
Tambup=Tambup+273.15;                  % above panel in KELVIN
Tambbo=Tambbo+273.15;                  % below panel in KELVIN
```

GEOMETRIC INFORMATION OF PV PANEL

```
t1=0.003;
t2=0.0005;
t3=1e-7;
t4=0.0004;
t5=0.0005;
t6=0.0001;
thickness=[t1;t2;t3;t4;t5;t6];            % thickness of layers (order : glass,eva,arc,si,eva,tedlar)
```

THERMAL CONDUCTIVITY OF PV PANEL (simplified to be bulk thermal conductivity = k)

```
k1=k;
k2=k;
k3=k;
k4=k;
k5=k;
k6=k;
therm_cond=[k1;k2;k3;k4;k5;k6];           % thermal conductivity of layers (order : glass,eva,arc,si,eva,tedlar)
```

RADIATIVE PROPERTIES OF TOP AND BOTTOM SURFACE OF PV PANEL

```
eps_F=0.9;
eps_B=0.9;
```

OPTICAL MODELING AND GHI CORRECTION

```
opt_prop=xlsread('opt_prop.xlsx');             % OPTICAL PROPErTIES READ THROUGH THIS FILE
wv=opt_prop(:,1) ;                             % wavelength (nm)
layer_i_n=opt_prop(:,2) ;                      % layer_i real part of refractive index
layer_i_k=opt_prop(:,3);                       % layer_i imaginary part of refractive index
layer_i_ac=(4.*pi.*layer_i_k)./(wv.*10^-9);    % layer_i absorption coefficient (4*pi*k/lamda)

atmos=xlsread('atmosphere.xlsx');    % ATMospheric data READ THROUGH THIS FILE

lam=atmos(:,1);              % wavelength in micrometre
sp_etr=atmos(:,2);           % spectral extra terrestrial irradiance (W/m2/um)
sp_wva=atmos(:,3);           % spectral water vapor absorption coefficient
sp_oza=atmos(:,4);           % spectral ozone absorption coefficient
sp_uga=atmos(:,5);           % spectral uniformly mixed gases absorption coefficient
AOD=atmos(:,6);              % spectral AOD


d=datetime(year,month,date,hour,min,sec);
doy = day(d,'dayofyear');                              % day of the year
clocktime=hour+(min./60)+(sec./3600);                  % local clock time in decimal (24 h format)
```

```
[thetta,decl,h_angle]=in_angle(doy,clocktime,timezone,lat,lon,beta,azim_w);   % incident angle between panel normal and sun
if abs(thetta)>90
    thetta=89.9999;           % sunset sunrise conditions
end


altitude=asind((sind(lat).*sind(decl)) + cosd(lat).*cosd(decl).*cosd(h_angle));
z=90-altitude;           % z is the apparent solar zenith angle
if abs(z)>90
    z=89.99999;                % sunset sunrise conditions
end
```

to adjust the spectrum based on pyranometer measurement

```
[~,ghi_m_per_um]=to_get_normalized_rbird_radiation(doy,clocktime,timezone,year,month,date,hour,min,sec,ghi_py,atmos,lat,lon,azim_w,beta,z,P,W,ho,c
ghi_m=ghi_m_per_um./1000;                 % Code needs W/m2/nm
```

SOLVING EQUATIONS

```
% initial values----------------
Tup=Tambup ;                   % in KELVIN
Tcell=Tambbo+2;                 % in KELVIN
Tbo=Tambbo;                    % in KELVIN

T_log(1,1)=Tup;
T_log(1,2)=Tcell;
T_log(1,3)=Tbo;
% initial values ----------------
iter=1;
stop_param=1;         % dummy starter
while stop_param>1e-10
[matrix,rhs]=newton_raphson_roshan(Tambup,Tambbo,Tup,Tbo,Tcell,Tground,velup,velbo,doy,clocktime,lat,lon,beta,azim_w,ghi_m,thickness,therm_cond,ep
rawsol=linsolve(matrix,rhs);

er1=rawsol(1,1);
er2=rawsol(2,1);
er3=rawsol(3,1);
errorlog(iter,1)=er1;
errorlog(iter,2)=er2;
errorlog(iter,3)=er3;

Tup=er1+Tup;
Tcell=er2+Tcell;
Tbo=er3+Tbo;

T_log(iter+1,1)=Tup;
T_log(iter+1,2)=Tcell;
T_log(iter+1,3)=Tbo;

iteration(iter,1)=iter;
iter=iter+1;
stop_param=max([abs(er1) abs(er1) abs(er1)])
errorlog;                    % in the order : (up,cell,bottom)
end
eup=errorlog(end,1);
ecell=errorlog(end,2);
ebot=errorlog(end,3);
```

```
stop_param =

   26.1947


stop_param =

    2.3428


stop_param =

    0.2710


stop_param =

    0.0284


stop_param =

    0.0029


stop_param =

   3.0354e-04


stop_param =
```

```
    3.1356e-05


stop_param =

    3.2391e-06


stop_param =

    3.3461e-07


stop_param =

    3.4565e-08


stop_param =

    3.5706e-09


stop_param =

    3.6888e-10


stop_param =

    3.8131e-11
```

celcius conversion back

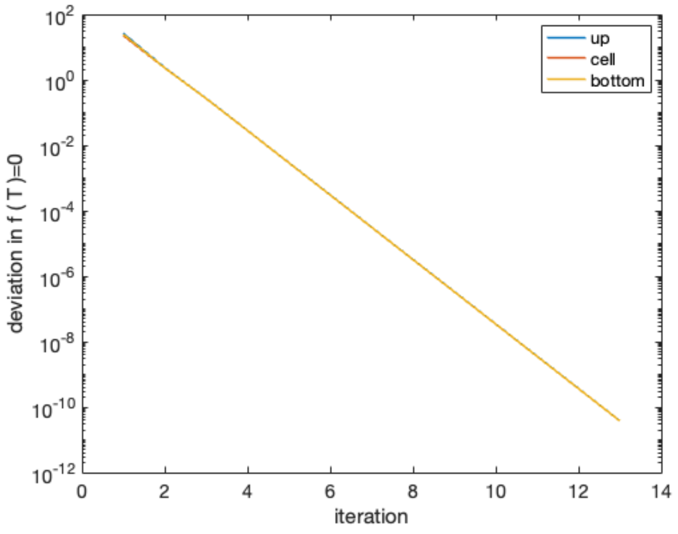
```
final=T_log(end,:)-273.15;
```

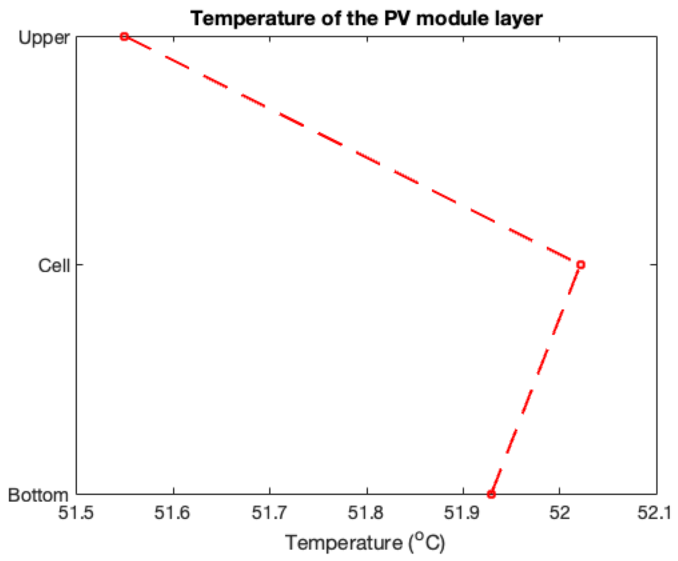
PLOTTING

```
aaa=figure
plot(iteration,abs(errorlog(:,1)),iteration,abs(errorlog(:,2)),iteration,abs(errorlog(:,3)),'LineWidth',1.5)
legend('up','cell','bottom')
set(gca, 'YScale', 'log')
xlabel('iteration')
ylabel('deviation in f ( T )=0')
set(gca,'FontSize',14)

aa=figure
plot(flipud(final'),(1:3),'--rs','LineWidth',2)
xlabel('Temperature (^oC)')
title('Temperature of the PV module layer')
yticks([1 2 3])
yticklabels({'Bottom','Cell','Upper'})
set(gca,'FontSize',14)
% [a1,a2,a3,a4,a5,a6,a7,a8,a9,a10,a11,a12]=energy_rates(Tambup,Tambbo,Tup,Tbo,Tcell,Tground,velup,velbo,doy,clocktime,lat,lon,beta,azim_w,ghi_m,th
%
% q_cell_glass=a1
% q_cell_tedlar=a2
% q_ele_gen=a3
% q_si_abs=a4
%
% q_conv_f=a5
% q_rad_f_sky=a6
% q_rad_f_gr=a7
% q_first3_abs=a8
%
% q_conv_b=a9
% q_rad_b_sky=a10
% q_rad_b_gr=a11
% q_last2_abs=a12
%
% rad_to_ground=abs(a11)+abs(a7)
% balance1=a1+a2+a3-a4
% balance2=a5+a6+a7-(a1+a8)
% balance3=a9+a10+a11-(a12+a2)
```

```
aaa =

  Figure (1) with properties:

      Number: 1
        Name: ''
       Color: [0.9400 0.9400 0.9400]
    Position: [440 278 560 420]
       Units: 'pixels'

  Use GET to show all properties
```

```
aa = 

  Figure (2) with properties:

      Number: 2
        Name: ''
       Color: [0.9400 0.9400 0.9400]
    Position: [440 278 560 420]
       Units: 'pixels'

  Use GET to show all properties
```





Published with MATLAB® R2023a