REFERENCES

Al-Sanea, S. A. (July, 2002). Thermal performance of building roof elements. *Building and Environment* , 665-675.

BF5 - Sunshine Sensor. (2011). Retrieved 10 12, 2011, from Delta-T Devices: http://www.delta-t.co.uk/product-display.asp?id=BF5%20Product&div=Meteorology%20and%20Solar

Cengel, Y. A. (2007). *Heat and Mass Transfer.* New York: McGraw-Hill Companies, Inc.

Clarke, J. A., & Maver, T. W. (1991). Advanced Design Tools for Energy Conscious Building Design:Development and Dissemination. *Building and Environment,Issue1* , pp. 25-34.

DL2e - Data Logger. (2011). Retrieved 10 12, 2011, from Delta-T Devices: http://www.delta-t.co.uk/product-display.asp?id=564&div=Data%20Loggers

Duffie, J. A., & Beckman, W. A. (2006). *Solar Engineering of Thermal Processes.* New Jersey: John Wiley & Sons Inc.

Elseragy, A. A., & Gadi, M. B. (2003). Computer Simulation of Solar Radiation Recieved By Curved Roof in Hot-Arid Regions. *Eighth International IBPSA Conference*, (pp. 283-290). Eindhoven.

Faghih, A. K., & Bahadori, M. N. (2011). Thermal performance evaluation of domed roofs. *Energy and Buildings* , 1254–1263.

Fisher, D., & Pedersen, C. (1997). Convective Heat tranfer in Building Energy and Thermal Load calculatons. *ASHRAE Transactions*, (p. Vol 103).

Fitzgerald, W. B., Fahmya, M., Smith, I. J., Carruthers, M. A., Carsona, B. R., Sun, Z., et al. (2011). An assessment of roof space solar gains in a temperate maritime climate. *Energy and Buildings* , 1580–1588.

Ghoshdastidar, P. (2005). *Heat Transfer.* New Delhi: Oxford University Press.

Grag, H. P., & Prakash, J. (2006). *Solar Energy Fundermentals and Applications.* New Delhi: Tata McGraw-Hill Publishing Company Limited.

Hashimoto, Y., & Yoneda, H. (2009). Numerical Study on the influence of Ceiling Height for Displacement Ventillation. *Eleventh International IBPSA Conference*, (pp. 1045-1052). Glasgow.

Howell, J. R. (2010). *A Catalog of Radiation Heat Transfer Configuration Factors* . Retrieved March 03, 2011, from http://www.engr.uky.edu/rtl/Catalog/tablecon.html

Huang, J., Hanford, J., & Yang, F. (1999). *Residential heating and cooling loads component analysis.* Berkeley: University of California.

Incropera, F. P., & Dewitt, D. P. (2009). *Fundermantals of Heat and Mass Transfer.* New Delhi: Wiley India(P.) Ltd.

Infrared Services Inc. (2000). Retrieved January 10, 2011, from http://www.infrared-thermography.com/material.htm

Jayasinhe, M. T. (2003). *Energy Efficient Houses for Tropical Climates.* Colombo: McBolon Polymer Pvt Ltd.

Mahan, J. (2002). *Radiation heat transfer : a statistical approach.* New York: John Wiley & Sons.

Modest, M. F. (2003). *Radiative Heat Transfer.* San Diego, California: Acadamic Press.

onset HOBO Data loggers. (2011). Retrieved 10 12, 2011, from HOBO U12 Data Loggers: http://www.onsetcomp.com/products/data-loggers/u12-001

Papadakisa, G., Frangoudakisa, A., & Kyritsisa, S. (1992). Mixed, forced and free convection heat transfer at the greenhouse cover. *Journal of Agricultural Engineering Research* , 191-205.

Parker, D. S., McIlvaine, J. E., Barkasz, S. F., Beal, D. J., & Anello, M. T. (2000). *Laboratory Testing of the Reflectance Properties of Roofing Material.* Cocoa, FL: Florida Solar Energy Center.

Shahmohamadi, P., Che-Ani, A., Ramly, A., Maulud, K., & Mohd-Nor, M. (2010). Reducing urban heat island effects: A systematic review to achieve energy consumption balance. *International Journal of Physical Sciences* , 626-636.

Shao, J., Liu, J., Zhao, J., Zhang, W., Sun, D., & Fu, Z. (2009). A novel method for full-scale measurement of the external convective heat transfer coefficient for building horizontal roof. *Energy and Buildings* , 840-847.

Sjosten, J., Olofsson, T., & M, G. (2003). Heating Energy Use Simulation For Residential Buildings. *Eighth International IBPSA Conference*, (pp. 1221-1226). Eindhoven.

Sri Lanka Sustainable Energy Authority. (2009). *Code of practice for energy efficient buildings Sri Lanka - 2008.* Colombo: Design Systems (Pvt) Ltd. , Colombo 10.

Sri Lanka Sustainable Energy Authority. (2010). *Sri Lanka Energy Balance 2007,.* Colombo: Sri Lanka Energy Managers Association.

Srivastava, S. K., Gaur, A., Singh, O. P., & Tiwari, R. N. (1995). Comparison of Methods for Estimating Daily and Hourly Diffuse Solar Radiation. *Applied Energy* , 119-123.

Tang, R., Meir, I. A., & Wu, T. (2006). Thermal performance of non air-conditioned buildings with vaulted roofs in comparison with flat roofs. *Building and Environment* , 268–276.

U.S. Department of Energy. (2011, March). *Chapter1. 2010 Buildings Energy Data Book*. Retrieved July 24, 2011, from Buildings Energy Data Book: http://buildingsdatabook.eren.doe.gov/ChapterIntro1.aspx

Vueghs, P., Koning H, P., Pin, O., & Beckers, P. (2008). Random Hemisphere Method for Radiation Ray Tracing Computations . *5th European Thermal-Sciences Conference,*. Netherlands.

Vujic, M. R., Lavery, N. P., & Brown, S. R. (2006). Numerical sensitivity and view factor calculation using the Monte Carlo method. *Journal of Mechanical Engineering Science* , 697-702.

Zmrhal, V., Hensen, J., & Drkal, F. (2003). Modelling and Simulation of a Room with a Radient Cooling Ceiling. *Eighth International IBPSA Conference*, (pp. 1491-1496). Eindhoven.

APPENDIX A: APPROACH TO THE RADIATION HEAT EXCHANGE

Radiation

A body which is at above zero absolute temperature emits radiation. The body emits radiation in to all directions over a broad range of wave lengths. Since emitted radiation depends on various factors an idealized body is defined and it is referred as a black body.

Black body

A black body is defined as a perfect emitter and absorber of radiation. At a specified temperature and wave length, no surface can emit more energy than a black body. A black body absorbs all incident radiation, regardless of wave length and direction. Also a black body emits radiation energy uniformly in all directions per unit area normal to direction of emission (Cengel, 2007).

According to the Stefan–Boltzmann law, the rate at which radiation energy emitted by a blackbody per unit surface area is expressed as,

$$E_b = \sigma T^4$$

E_b *(W/m^2)* is known as black body emissive power. σ is Stefan–Boltzmann constant and T is the absolute temperature of the surface.

Intensity of emitted radiation

Level of radiation is frequently expressed by radiation intensity and it is an indication of radiation energy received or emitted by a surface. Emission is the origin of the radiation. Spectral intensity of emitted radiation *($I_{\lambda,e}$)* can be define as the rate at which radiant energy is emitted at the wave length λ in the *(θ,□)* direction, per unit area of the emitting surface normal to this direction, per unit solid angle about this direction, and per unit wave length interval $d\lambda$ about λ (Incropera & Dewitt, 2009).

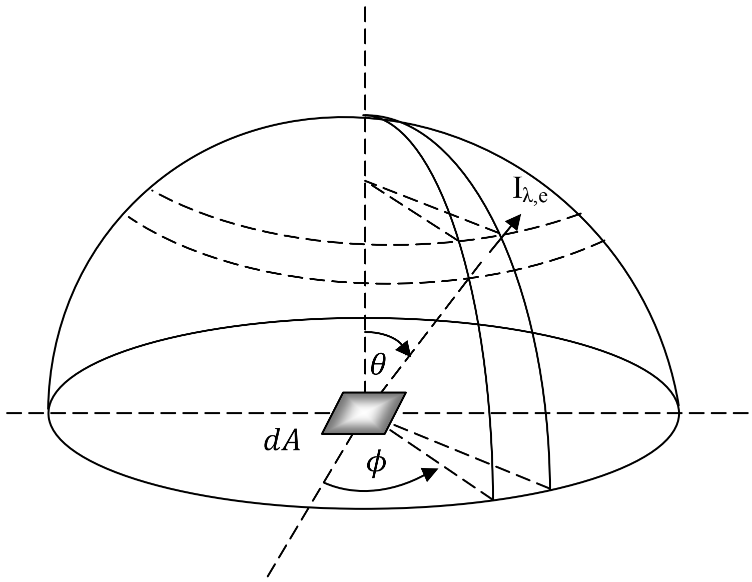

Figure 1: Emission of radiation from a differential area *dA* into a hemispherical space

According to the definition of the radiation intensity,

$$I_{\lambda,e}(\lambda,\theta,\phi) = \frac{dq}{dA\cos\theta \; d\Omega \; d\lambda}$$

Where, $d\Omega(solid\ angle) = \sin\theta \;\; d\theta d\phi$ and $(dq/d\lambda) = dq_{\lambda}$.

$$I_{\lambda,e}(\lambda,\theta,\phi) = \frac{dq_{\lambda}}{dA\cos\theta \;\; \sin\theta \;\; d\theta \; d\phi}$$

Emissivity

Emissivity represents the ratio between radiation emitted by a surface at a given temperature and radiation emitted by a black body at the same temperature. The emissivity of a real surface may vary with the temperature, direction and the wavelength of emitted radiation. Due to this reason different emissivities are defined depending on the considered effect. Thus, spectral hemispherical emissivity is defined as,

$$\varepsilon_\lambda(\lambda,T) = \frac{E_\lambda(\lambda,T)}{E_{\lambda,b}(\lambda,T)}$$

Total hemispherical emissivity which indicates an average over all possible wave lengths and directions is defined as follows.

$$\varepsilon(T) = \frac{E(T)}{E_b(T)}$$

Irradiance

Radiation energy incident on a surface may originate from emission and reflection occurring at other surfaces. By taking it into consideration irradiance (also known as irradiance) G *(W/m²)* is expressed as the rate at which radiation incident per unit area from all directions and all wave lengths.

Absorption, reflection and transmission of radiation

Radiation incident on a surface is subjected to absorption, reflection and transmission.

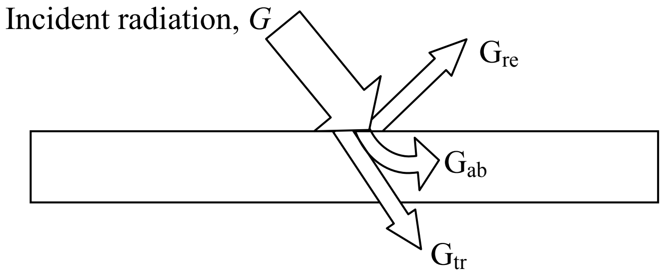

Figure 2: Reflection, absorption and transmittance of incident radiation by a semitransparent material

As in the case of emission, above three incidents may also be characterized by both spectral and directional dependence.

Spectral hemispherical absorptivity, α is defined as the fraction of irradiance absorbed by the surface,

$$\alpha_{\lambda(\lambda)} = \frac{G_{\lambda,abs(\lambda)}}{G_{\lambda(\lambda)}}$$

Spectral hemispherical refectivity, ρ is defined as the fraction of irradiance reflected by the surface,

$$\rho_{\lambda(\lambda)} = \frac{G_{\lambda,ref(\lambda)}}{G_{\lambda(\lambda)}}$$

Spectral hemispherical transmitivity, τ is defined as the fraction of irradiance transmitted by the surface,

$$\tau_{\lambda(\lambda)} = \frac{G_{\lambda,tr(\lambda)}}{G_{\lambda(\lambda)}}$$

Total hemispherical quantities which represent an intergraded average over wave length and direction can be defined based on above equations.

According to above definitions it is clear that for all wavelengths $\alpha_\lambda + \rho_\lambda + \tau_\lambda = 1$. It is considered as $\tau = 0$ for opaque surfaces.

As given by Kirchhoff's law, the total hemispherical emissivity of a surface at temperature T is equal to its total hemispherical absorptivity for radiation coming from a blackbody at the same temperature (Cengel, 2007).

$$\varepsilon(T) = \alpha(T)$$

View factor

View factor is useful in order to compute the magnitudes of radiation exchange between surfaces. The view factor is defined as the fraction of the radiation leaving surface i, which is directly intercepted by surface j and denoted by F_{ij}.

$$F_{ij} = \frac{q_{i \to j}}{A_i J_i}$$

View factor is a purely geometric quantity and also known as shape factor, configuration factor and angle factor.

View factor relations

According to the view factor integral following relationship can be obtained.

$$A_i F_{ij} = A_j F_{ji}$$

Above relationship is referred to as reciprocity rule. Another important relation which is referred to as summation rule, applies to an enclosure with N number of surfaces. From the definition of the view factor it is clear that,

$$\sum_{j=1}^{N} F_{ij} = 1$$

$F_{ii} = 0,$ If the surface i is plane or convex and $F_{ii} \neq 0,$ If the surface i is concave (Ghoshdastidar, 2005)

A relationship for the additive nature of view factors can be obtained as follows (Figure 3).

$$A_j = \sum_{k=1}^{N} A_k$$

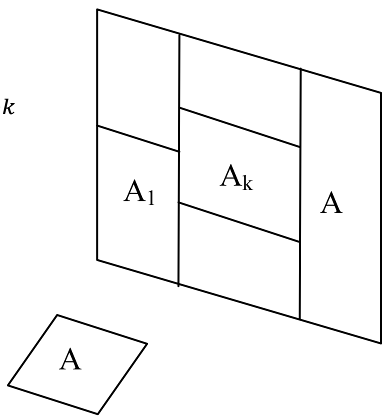

Figure 3: View factor between A_i and a composite area

Dividing the above equation by A_iJ_i following relationship is obtained.

$$F_{i(j)} = \sum_{k=1}^{N} F_{ik}$$

Where parenthesis around a subscript indicate that it is a composite surfaces, in which case j is equivalent to (1, 2,…., k,…N)

In addition to the above relationships, symmetry of the enclosure can be considered in order to get relationships for view factors.

APPENDIX B: A SAMPLE VIEW FACTOR CODE

This is one of the three view factor codes developed for the simulation tool. Dissimilarities present among these codes are mainly due to different shapes of roof surfaces. In addition to the main file, there are13 other MATLAB files which should be run in order to work ROTSIM.

```
function [ F] = VF_F4(n)
%Function for View factor from surface 4

% Defining initial variables
disp('');
%tic
i= [1 0 0];
j= [0 1 0];

% Defining the coner points of the roof

cord=coordinate; %Importing coordinates of the geometry

p0 = cord(:,1);

p1 = cord(:,2);

p2 = cord(:,3);

p3 = cord(:,4);

%p4 = cord(:,5);

p5 = cord(:,6);


%obtaining the inclination angles of the roof surface

%defining the projections on YZ plane and ZX plane

p1YZ = [ 0 p1(2) p1(3) ];
```

```
p1ZX = [ p1(1) 0 p1(3) ];

% A= alpha, B= beta
A = (acos((dot(p1YZ,j)/(norm(p1YZ)))));
B = (acos((dot(p1ZX,i)/(norm(p1ZX)))));

%Defining the conversion matrices-m1 martx for surface 1

m1 = [1 0 0 0; 0 cos(A) (-sin(A)) 0; 0 sin(A) cos(A) 0; 0 0 0 1];

m2r= [cos(pi+B)     0    sin(pi+B) 0
    0               1    0          0
    (-sin(pi+B)) 0    cos(pi+B) 0
    0               0    0          1 ]; % rotation matrix

m2m= [1 0 0 norm(p3-p0); 0 1 0 0; 0 0 1 0; 0 0 0 1]; %moving along

m2 = m2m*m2r;

m4 = [ cos(2*pi-B)     0 sin(2*pi-B) 0
    0               1 0           0
    (-sin(2*pi-B)) 0 cos(2*pi-B) 0
    0               0 0           1];

%starting the loop

%converting base coordinate into triangular surface's coordinates
p14Tr = m4\p1;

%converting base coordinate into triangular surface's coordinates(surface 1)
p11Re = m1\p1;
p21Re = m1\p2;

p11Tr = p11Re;

%converting base coordinate into surface 2 coordinate
p12Tr = m2\p2;

%defining the ratio of sub elements of the rectangle of surface1. c-ratio
```

```
c1x=1;
c1y=1;
if ((p21Re(1,1)-p11Re(1,1))>(p11Re(2,1)))
    c1x=round((p21Re(1,1)-p11Re(1,1))/(p11Re(2,1))); %x/y
else
    c1y=round((p11Re(2,1))/(p21Re(1,1)-p11Re(1,1))); %y/x
end


%Writing the programme for right angled triangle relative to iso scale triangle

count=0;
count2=0;
F41Trh1=0;
F41Trh2=0;
F41Reh=0;
F42Tr=0;

r=1;
for c= 0: (n-2)

    for d= r: (2*n-1-r)

        q=1;

        for b= 0: (n-2)

            for a= q: (n-1)

                %defining the coordinates of triangular surface's elements

                x11 = p11Tr(1,1);%point 1 reletive to surface 1 co: sys: triangular section
                y11 = p11Tr(2,1);

                p1bTrh = m1*[x11*a/n; y11*b/n; 0; 1];%element trangular half
                p2bTrh = m1*[x11*(a+1)/n; y11*b/n; 0; 1];%element
                p4bTrh = m1*[x11*a/n; y11*(b+1)/n; 0; 1];%element

                x14 = p14Tr(1,1);
                y24 = p5(2);
```

```
        p1bTr = m4*[x14*c/n; y24*d/(2*n); 0; 1];%element triangle isoscale
        p2bTr = m4*[x14*c/n; y24*(d+1)/(2*n); 0; 1];%element
        p4bTr = m4*[x14*(c+1)/n; y24*d/(2*n); 0; 1];%element


        n1jHh = m1*[0; 0; 1; 1]-m1*[0; 0; 0; 1]; %n2j homogenious and getting the normal
        n1jh = [n1jHh(1,1) n1jHh(2,1) n1jHh(3,1)];

        n4jH = m4*[0; 0; 1; 1]-m4*[0; 0; 0; 1]; %n2j homogenious and getting the normal
        n4j = [n4jH(1,1) n4jH(2,1) n4jH(3,1)];


        dAjh = norm(p2bTrh-p1bTrh)*norm(p4bTrh-p1bTrh);

        xijh = ((p1bTrh(1,1)+p2bTrh(1,1))/2)-(p1bTr(1,1)+ p4bTr(1,1))/2;
        yijh = ((p1bTrh(2,1)+p4bTrh(2,1))/2)-(p1bTr(2,1)+ p2bTr(2,1))/2;
        zijh = ((p1bTrh(3,1)+p4bTrh(3,1))/2)-(p1bTr(3,1)+ p4bTr(3,1))/2;

        sijh = [xijh yijh zijh];%vector from surface 4 to surface 1
        sjih = (-sijh);


        dFijh = ((dot(-n4j,sijh))*(dot(-n1jh,sjih))*dAjh)/(pi*(norm(sijh).^4));

        F41Trh1 = F41Trh1 + dFijh;

        count=count+1;

    end
    q=q+1;

end

%witing the loop for triangular half 2 of  surface 1
q=1;

for b= 0: (n-2)

    for a= 0: (n-1-q)

        x11 = p11Tr(1,1);%point 1 reletive to surface 1 co: sys: triangular section
```

```
        y11 = p11Tr(2,1);
        x21= p21Re(1,1);

        p1bTrh = m1*[x21+x11*a/n; y11*b/n; 0; 1];%element trangular half
        p2bTrh = m1*[x21+x11*(a+1)/n; y11*b/n; 0; 1];%element
        p4bTrh = m1*[x21+x11*a/n; y11*(b+1)/n; 0; 1];%element

        x14 = p14Tr(1,1);
        y24 = p5(2);

        p1bTr = m4*[x14*c/n; y24*d/(2*n); 0; 1];%element triangle isoscale
        p2bTr = m4*[x14*c/n; y24*(d+1)/(2*n); 0; 1];%element
        p4bTr = m4*[x14*(c+1)/n; y24*d/(2*n); 0; 1];%element



        n1jHh = m1*[0; 0; 1; 1]-m1*[0; 0; 0; 1]; %n2j homogenious and getting the normal
        n1jh = [n1jHh(1,1) n1jHh(2,1) n1jHh(3,1)];

        n4jH = m4*[0; 0; 1; 1]-m4*[0; 0; 0; 1]; %n2j homogenious and getting the normal
        n4j = [n4jH(1,1) n4jH(2,1) n4jH(3,1)];

        dAjh = norm(p2bTrh-p1bTrh)*norm(p4bTrh-p1bTrh);

        xijh = ((p1bTrh(1,1)+p2bTrh(1,1))/2)-(p1bTr(1,1)+ p4bTr(1,1))/2;
        yijh = ((p1bTrh(2,1)+p4bTrh(2,1))/2)-(p1bTr(2,1)+ p2bTr(2,1))/2;
        zijh = ((p1bTrh(3,1)+p4bTrh(3,1))/2)-(p1bTr(3,1)+ p4bTr(3,1))/2;

        sijh = [xijh yijh zijh];%vector from surface 4 to surface 1
        sjih = (-sijh);



        dFijh2 = ((dot(-n4j,sijh))*(dot(-n1jh,sjih))*dAjh)/(pi*(norm(sijh).^4));

        F41Trh2 = F41Trh2 + dFijh2;

        count2=count2+1;

    end
    q=q+1;

end
```

```
%writing the loop for rectangular half of surface 1
n1x=c1x*n;
n1y=c1y*n;
count3=0;

if p1==p2
    F41Reh=0;
else

    for b= 0 : (n1y-1)

        for a= 0 : (n1x-1)

            p1bRe = m1*[(p11Re(1,1)+(p21Re(1,1)-p11Re(1,1))*a/n1x); p11Re(2,1)*b/n1y; 0; 1];%element
            p2bRe = m1*[(p11Re(1,1)+(p21Re(1,1)-p11Re(1,1))*(a+1)/n1x); p11Re(2,1)*b/n1y; 0; 1];%element
            p4bRe = m1*[(p11Re(1,1)+(p21Re(1,1)-p11Re(1,1))*a/n1x); p11Re(2,1)*(b+1)/n1y; 0; 1];%element

            x14 = p14Tr(1,1);
            y24 = p5(2);

            p1bTr = m4*[x14*c/n; y24*d/(2*n); 0; 1];%element triangle isoscale
            p2bTr = m4*[x14*c/n; y24*(d+1)/(2*n); 0; 1];%element
            p4bTr = m4*[x14*(c+1)/n; y24*d/(2*n); 0; 1];%element
            n1jHh = m1*[0; 0; 1; 1]-m1*[0; 0; 0; 1]; %n2j homogenious and getting the normal- normal is same
everywhere
            n1jh = [n1jHh(1,1) n1jHh(2,1) n1jHh(3,1)];


            n4jH = m4*[0; 0; 1; 1]-m4*[0; 0; 0; 1]; %n2j homogenious and getting the normal
            n4j = [n4jH(1,1) n4jH(2,1) n4jH(3,1)];


            xijh=((p1bRe(1,1)+p2bRe(1,1))/2)-(p1bTr(1,1)+ p4bTr(1,1))/2;
            yijh=((p1bRe(2,1)+p4bRe(2,1))/2)-(p1bTr(2,1)+ p2bTr(2,1))/2;
            zijh=((p1bRe(3,1)+p4bRe(3,1))/2)-(p1bTr(3,1)+ p4bTr(3,1))/2;
            sijh=[xijh yijh zijh];
            sjih=(-sijh);

            dAjh=norm(p2bRe-p1bRe)*norm(p4bRe-p1bRe);
```

```
            dFijr = ((dot(-n4j,sijh))*(dot(-n1jh,sjih))*dAjh)/(pi*(norm(sijh).^4));


            F41Reh =  F41Reh + dFijr;

            count3=count3+1;

        end
    end
end

%witing the loop for triangul  surface 2
q=1;

for b= 0: (n-2)

    for a= q: (2*n-1-q)

        x12 = p12Tr(1,1);% surface 4 and surface 2 become the same when the local coorninates are considered
        y12 = p12Tr(2,1);

        p1bTr2 = m2*[x12*b/n; y12*a/n; 0; 1];%element triangle isoscale2
        p2bTr2 = m2*[x12*b/n; y12*(a+1)/n; 0; 1];%element
        p4bTr2 = m2*[x12*(b+1)/n; y12*a/n; 0; 1];%element

        x14 = p14Tr(1,1);
        y24 = p5(2);

        p1bTr = m4*[x14*c/n; y24*d/(2*n); 0; 1];%element triangle isoscale
        p2bTr = m4*[x14*c/n; y24*(d+1)/(2*n); 0; 1];%element
        p4bTr = m4*[x14*(c+1)/n; y24*d/(2*n); 0; 1];%element

        n2jH = m2*[0; 0; 1; 1]-m2*[0; 0; 0; 1]; %n2j homogenious and getting the normal
        n2j = [n2jH(1,1) n2jH(2,1) n2jH(3,1)];
        n4jH = m4*[0; 0; 1; 1]-m4*[0; 0; 0; 1]; %n2j homogenious and getting the normal
        n4j = [n4jH(1,1) n4jH(2,1) n4jH(3,1)];

        dAjh = norm(p2bTr2-p1bTr2)*norm(p4bTr2-p1bTr2);

        xijh = ((p1bTr2(1,1)+p4bTr2(1,1))/2)-(p1bTr(1,1)+ p4bTr(1,1))/2;
```

```
            yijh = ((p1bTr2(2,1)+p2bTr2(2,1))/2)-(p1bTr(2,1)+ p2bTr(2,1))/2;
            zijh = ((p1bTr2(3,1)+p4bTr2(3,1))/2)-(p1bTr(3,1)+ p4bTr(3,1))/2;

            sijh = [xijh yijh zijh];%vector from surface 4 to surface 1
            sjih = (-sijh);


            dFij2 = ((dot(-n4j,sijh))*(dot(n2j,sjih))*dAjh)/(pi*(norm(sijh).^4));

            F42Tr= F42Tr + dFij2;

            count3=count3+1;

          end
          q=q+1;

        end


    end
    r=r+1;

end

F41Tr1 = F41Trh1/(n*n-n);%triangle1 of surface 1 reletive to surf 4
F41Tr2 = F41Trh2/(n*n-n);
F41Re  = F41Reh/(n*n-n);%Rectangle of surface 1 reletive to surf 4
F42      =F42Tr/(n*n-n);
F41      =F41Tr1+F41Tr2+F41Re;

F=[F41 F42];

disp('');
end
```

APPENDIX C: MEASURING EQUIPMENTS

Hobo U12 Data logger-Onset, USA

Measurement range: -20° to 70°C

Accuracy: ± 0.35°C from 0° to 50°C

Resolution: 0.03°C at 25°C

Operating temperature:

Logging: -20° to 70°C

Sample Rate: 1 second to 18 hours

Figure 4: HOBO U12

Source: onset HOBO Data loggers (2011)

DL2e data logger-Delta-T devices, UK

Voltage range: ±4mV, ±32mV, ±262mV, ±2.097V

Accuracy: ±0.2% (-20 to 60°C)

Resolution: 1mV, 8mV, 64mV, 0.5m

Resistance ranges: 1kW, 10kW, 100kW, 1MW

Accuracy: ±0.1% (±0.6% to 50°C, on lowest range)

Resolution: 0.01W (lowest range)

Figure 5: DL2e data logger

Source: DL2e - Data Logger (2011)

<u>BF5 sunshine sensor-Delta-T devices, UK</u>

Overall accuracy: Total: ±5 $W.m^{-2}$ ±12%

Overall accuracy: Diffuse: ±20 $W.m^{-2}$ ±15%

Resolution: 0.3 $W.m^{-2}$

Range: 0 - 1250 $W.m^{-2}$

Output sensitivity: 1mV = 0.5 $W.m^{-2}$

Output range: 0 – 2500 mV

Figure 6: BF5 Sunshine sensor

Source: BF5 - Sunshine Sensor(2011)

APPENDIX D: DEVELOPED SOFTWARE TOOL (ROTSIM)

Appendix D is included in the CD, which is attached with the thesis.