Radiation Balance Between Parallel Plates

Assignment is completed by Mikkel Jaedicke (mijae12) & Anders Bæk (anbae12)

The radiation balance is covered by equations 1-3.

$$\begin{aligned} u\left(x\right) &= \varepsilon_1 \sigma T_1^4 + \left(1-\varepsilon_1\right) \int_{-\frac{1}{2}w}^{\frac{1}{2}w} F\left(x,y,d\right) \cdot v\left(y\right) dy \\ v\left(y\right) &= \varepsilon_2 \sigma T_2^4 + \left(1-\varepsilon_2\right) \int_{-\frac{1}{2}w}^{\frac{1}{2}w} F\left(x,y,d\right) \cdot u\left(x\right) dx \end{aligned} \tag{1}$$

$$F\left(x,y,d\right) = \frac{1}{2} \frac{d^2}{\left(d^2 + \left(x-y\right)^2\right)^{\frac{3}{2}}} \tag{2}$$

$$\begin{aligned} I_1\left(x\right) &= \int_{-\frac{1}{2}w}^{\frac{1}{2}w} F\left(x,y,d\right) \cdot v\left(y\right) dy \\ I_2\left(y\right) &= \int_{-\frac{1}{2}w}^{\frac{1}{2}w} F\left(x,y,d\right) \cdot v\left(x\right) dx \end{aligned} \tag{3}$$

DATA: $T_1 = 1000$, $T_2 = 500$, $\varepsilon_1 = 0.80$, $\varepsilon_2 = 0.60$, $\sigma = 1.7212 \cdot 10^{-9}$, $d = 1.0$, $w = 1.0$

Question 1

Equation 1 is rearranged to give equation 4.

$$\begin{aligned} u\left(x\right) - \left(1-\varepsilon_1\right) \int_{-\frac{1}{2}w}^{\frac{1}{2}w} F\left(x,y,d\right) \cdot v\left(y\right) dy &= \varepsilon_1 \sigma T_1^4 \\ v\left(y\right) - \left(1-\varepsilon_2\right) \int_{-\frac{1}{2}w}^{\frac{1}{2}w} F\left(x,y,d\right) \cdot u\left(x\right) dx &= \varepsilon_2 \sigma T_2^4 \end{aligned} \tag{4}$$

The integral needs to be discretized in order to make it into a system of linear equations (SLE). The trapez-method is used to obtain equation 5.

$$\begin{aligned} u\left(x\right) - \left(1-\varepsilon_1\right) + h \cdot \left(\frac{1}{2} \cdot F(x,y_0,d) \cdot v(y_0) + \frac{1}{2} \cdot F(x,y_N,d) \cdot v(y_N) + \sum_{i=1}^{N-1} F(x,y_i,d) \cdot v(y_i)\right) &= \varepsilon_1 \sigma T_1^4 \\ v\left(x\right) - \left(1-\varepsilon_2\right) + h \cdot \left(\frac{1}{2} \cdot F(x_0,y,d) \cdot u(x_0) + \frac{1}{2} \cdot F(x_N,y,d) \cdot u(x_N) + \sum_{i=1}^{N-1} F(x_i,y,d) \cdot u(x_i)\right) &= \varepsilon_2 \sigma T_2^4 \end{aligned} \tag{5}$$

From equation 5 a SLE in the form $A \cdot z = b$ can be made.
Equations 6-8 illustrates a SLE for $N = 4$. The C++ implementation of equation 6 and equation 8 is scaleable (see enclosed main.cpp, line 39-84, line 93-104).
Equation 7 display the returned `VecDoub` by SVD solve function.

$$
A = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & -\frac{1}{2}\cdot\beta_1\cdot f(x_1,y_0,d) & -\beta_1\cdot f(x_1,y_1,d) & -\beta_1\cdot f(x_1,y_2,d) & -\beta_1\cdot f(x_1,y_3,d) & -\frac{1}{2}\cdot\beta_1\cdot f(x_1,y_4,d) \\
0 & 0 & 1 & 0 & 0 & -\frac{1}{2}\cdot\beta_1\cdot f(x_2,y_0,d) & -\beta_1\cdot f(x_2,y_1,d) & -\beta_1\cdot f(x_2,y_2,d) & -\beta_1\cdot f(x_2,y_3,d) & -\frac{1}{2}\cdot\beta_1\cdot f(x_2,y_4,d) \\
0 & 0 & 0 & 1 & 0 & -\frac{1}{2}\cdot\beta_1\cdot f(x_3,y_0,d) & -\beta_1\cdot f(x_3,y_1,d) & -\beta_1\cdot f(x_3,y_2,d) & -\beta_1\cdot f(x_3,y_3,d) & -\frac{1}{2}\cdot\beta_1\cdot f(x_3,y_4,d) \\
0 & 0 & 0 & 0 & 1 & -\frac{1}{2}\cdot\beta_1\cdot f(x_4,y_0,d) & -\beta_1\cdot f(x_4,y_1,d) & -\beta_1\cdot f(x_4,y_2,d) & -\beta_1\cdot f(x_4,y_3,d) & -\frac{1}{2}\cdot\beta_1\cdot f(x_4,y_4,d) \\
-\frac{1}{2}\cdot\beta_2\cdot f(x_0,y0,d) & -\beta_2\cdot f(x_1,y_0,d) & -\beta_2\cdot f(x_2,y_0,d) & -\beta_2\cdot f(x_4,y_0,d) & -\frac{1}{2}\cdot\beta_2\cdot f(x_4,y_0,d) & 1 & 0 & 0 & 0 & 0 \\
-\frac{1}{2}\cdot\beta_2\cdot f(x_0,y_1,d) & -\beta_2\cdot f(x_1,y_1,d) & -\beta_2\cdot f(x_2,y_1,d) & -\beta_2\cdot f(x_4,y_1,d) & -\frac{1}{2}\cdot\beta_2\cdot f(x_4,y_1,d) & 0 & 1 & 0 & 0 & 0 \\
-\frac{1}{2}\cdot\beta_2\cdot f(x_0,y_2,d) & -\beta_2\cdot f(x_1,y_2,d) & -\beta_2\cdot f(x_2,y_2,d) & -\beta_2\cdot f(x_4,y_2,d) & -\frac{1}{2}\cdot\beta_2\cdot f(x_4,y_2,d) & 0 & 0 & 1 & 0 & 0 \\
-\frac{1}{2}\cdot\beta_2\cdot f(x_0,y_3,d) & -\beta_2\cdot f(x_1,y_3,d) & -\beta_2\cdot f(x_2,y_3,d) & -\beta_2\cdot f(x_4,y_3,d) & -\frac{1}{2}\cdot\beta_2\cdot f(x_4,y_3,d) & 0 & 0 & 0 & 1 & 0 \\
-\frac{1}{2}\cdot\beta_2\cdot f(x_0,y0_4,d) & -\beta_2\cdot f(x_1,y_4,d) & -\beta_2\cdot f(x_2,y_4,d) & -\beta_2\cdot f(x_4,y_4,d) & -\frac{1}{2}\cdot\beta_2\cdot f(x_4,y_4,d) & 0 & 0 & 0 & 0 & 1
\end{bmatrix} \tag{6}
$$

where: $\beta_1 = (1-\varepsilon_1)\cdot h$, $\beta_2 = (1-\varepsilon_2)\cdot h$ and $h = \frac{\left(\frac{1}{2}w-\left(-\frac{1}{2}w\right)\right)}{N}$

$$
z = \begin{bmatrix} u_0 & u_1 & u_2 & u_3 & u_4 & v_0 & v_1 & v_2 & v_3 & v_4 \end{bmatrix}^T \tag{7}
$$

$$
b = \begin{bmatrix} \varepsilon_1\sigma T_1^4 & \varepsilon_1\sigma T_1^4 & \varepsilon_1\sigma T_1^4 & \varepsilon_1\sigma T_1^4 & \varepsilon_1\sigma T_1^4 & \varepsilon_2\sigma T_2^4 & \varepsilon_2\sigma T_2^4 & \varepsilon_2\sigma T_2^4 & \varepsilon_2\sigma T_2^4 & \varepsilon_2\sigma T_2^4 \end{bmatrix}^T \tag{8}
$$

Question 2

SVD is used to solve the SLE from question 1. Results are shown in table 1. Q_1 and Q_2 needs to be calculated. A rearrangement of the equation for Q_1 is made in equation 9. A similar rearrangement is made for the equation for Q_2.

The constants c_{11} and c_{12} belongs to the equation for Q_1 and c_{21} and c_{22} belongs to the equation for Q_2.

$$
\begin{aligned}
Q_1 &= \int_{-\frac{1}{2}w}^{\frac{1}{2}w} \left(u\left(x\right) - I\left(x\right)\right) dx \\
u\left(x\right) &= c_{11} + c_{12} \cdot I_1 \\
I_1 &= \frac{u\left(x\right) - c_{11}}{c_{12}} \\
Q_1 &= \int_{-\frac{1}{2}w}^{\frac{1}{2}w} \left(u\left(x\right) - \frac{u\left(x\right) - c_{11}}{c_{12}}\right) dx
\end{aligned} \tag{9}
$$

$$
\begin{aligned}
c_{11} &= \varepsilon_1 \cdot \sigma \cdot T_1^4 \\
c_{12} &= \left(1 - \varepsilon_1\right) \\
c_{21} &= \varepsilon_2 \cdot \sigma \cdot T_2^4 \\
c_{22} &= \left(1 - \varepsilon_2\right)
\end{aligned} \tag{10}
$$

The integrals in equation 9 need to be discretized. This is done with the trapez-method and the resulting equations can be seen in equation 11. The results for Q_1 and Q_2 can be seen in table 2.

$$
\begin{aligned}
Q_1 &= \frac{1}{2} \cdot h \cdot \left(u\left(x_0\right) - \left(\frac{u\left(x_0\right) - c_{11}}{c_{12}}\right) + u\left(x_N\right) - \left(\frac{u\left(x_N\right) - c_{11}}{c_{12}}\right)\right) + h \sum_{i=1}^{N-1} u\left(x_i\right) - \left(\frac{u\left(x_i\right) - c_{11}}{c_{12}}\right) \\
Q_2 &= \frac{1}{2} \cdot h \cdot \left(v\left(y_0\right) - \left(\frac{v\left(y_0\right) - c_{21}}{c_{22}}\right) + v\left(y_N\right) - \left(\frac{v\left(y_N\right) - c_{21}}{c_{22}}\right)\right) + h \sum_{i=1}^{N-1} v\left(y_i\right) - \left(\frac{v\left(y_i\right) - c_{21}}{c_{22}}\right)
\end{aligned} \tag{11}
$$

The values for $u\left(x\right)$ and $v\left(y\right)$, $x = y = \pm 0.5$, $x = y = \pm 0.25$ and $x = y = 0$ for a given N is showed in table 1.

N	$u(-0.50)$	$u(-0.25)$	$u(0.0)$	$u(0.25)$	$u(0.50)$	$v(-0.50)$	$v(-0.25)$	$v(0.0)$	$v(0.25)$	$v(0.50)$
4	1.39015295e+03	1.39407857e+03	1.39407857e+03	1.39407857e+03	1.39015295e+03	2.60504838e+02	2.97020995e+02	3.11005259e+02	2.97020995e+02	2.60504838e+02
8	1.39040598e+03	1.39560589e+03	1.39444651e+03	1.39444651e+03	1.39040598e+03	2.61150040e+02	2.82020745e+02	3.12971204e+02	3.09290227e+02	2.61150040e+02
16	1.39046959e+03	1.39600813e+03	1.39453871e+03	1.39453871e+03	1.39046959e+03	2.61311714e+02	2.72227782e+02	3.13460057e+02	3.12528101e+02	2.61311714e+02
32	1.39048552e+03	1.39610999e+03	1.39456177e+03	1.39456177e+03	1.39048552e+03	2.61352155e+02	2.66906623e+02	3.13582105e+02	3.13348382e+02	2.61352155e+02
64	1.39048950e+03	1.39613554e+03	1.39456754e+03	1.39456754e+03	1.39048950e+03	2.61362267e+02	2.64160578e+02	3.13612607e+02	3.13554131e+02	2.61362267e+02
128	1.39049050e+03	1.39614193e+03	1.39456898e+03	1.39456898e+03	1.39049050e+03	2.61364795e+02	2.62768839e+02	3.13620232e+02	3.13605610e+02	2.61364795e+02
256	1.39049075e+03	1.39614353e+03	1.39456934e+03	1.39456934e+03	1.39049075e+03	2.61365427e+02	2.62068623e+02	3.13622138e+02	3.13618483e+02	2.61365427e+02
512	1.39049081e+03	1.39614393e+03	1.39456943e+03	1.39456943e+03	1.39049081e+03	2.61365585e+02	2.61717471e+02	3.13622615e+02	3.13621701e+02	2.61365585e+02
1024	1.39049083e+03	1.39614403e+03	1.39456945e+03	1.39456945e+03	1.39049083e+03	2.61365624e+02	2.61541638e+02	3.13622734e+02	3.13622505e+02	2.61365624e+02

Table 1: Values for $u\left(x\right)$ and $v\left(y\right)$, $x = y = \pm 0.5$, $x = y = \pm 0.25$ and $x = y = 0$ for a given N.

Question 3

The Richardson α^k estimate and Richardson error estimate is shown in 12. The α^k estimate is seen in table 2 to converge to 4. Thus $\alpha^k = 4$ is used in the Richarson error estimate.

$$\begin{aligned} \alpha^k &= \frac{S(h_1) - S(h_2)}{S(h_2) - S(h_3)} \\ EE &= \frac{S(h_2) - S(h_1)}{\alpha^k - 1} \end{aligned} \tag{12}$$

Q_1, Q_2, EE and α^k is calculated for a given N, and the results can be seen in table 2. It is clearly seen that the estimated error decreases when N is increased.

N	Q_1	α_1^k	EE_1	Q_2	α_2^k	EE_2
4	1.27409460e+03	-0.00000000e+00	4.24698201e+02	-2.76582033e+02	0.00000000e+00	-9.21940109e+01
8	1.27208978e+03	-6.35513621e+02	-6.68275530e-01	-2.80870375e+02	6.44962508e+01	-1.42944760e+00
16	1.27158320e+03	3.95762143e+00	-1.68857871e-01	-2.81952494e+02	3.96291264e+00	-3.60706311e-01
32	1.27145623e+03	3.98967651e+00	-4.23237000e-02	-2.82223632e+02	3.99103702e+00	-9.03790941e-02
64	1.27142447e+03	3.99743475e+00	-1.05877150e-02	-2.82291454e+02	3.99777714e+00	-2.26073368e-02
128	1.27141653e+03	3.99935965e+00	-2.64735257e-03	-2.82308412e+02	3.99944538e+00	-5.65261796e-03
256	1.27141454e+03	3.99983995e+00	-6.61864625e-04	-2.82312651e+02	3.99986141e+00	-1.41320345e-03
512	1.27141405e+03	3.99996020e+00	-1.65467803e-04	-2.82313711e+02	3.99996540e+00	-3.53303919e-04
1024	1.27141392e+03	3.99999013e+00	-4.13670527e-05	-2.82313976e+02	3.99999125e+00	-8.83261731e-05

Table 2: Q_1, Q_2, EE and α^k is calculated for a given N.

The code[1] for solving the SLE, calculating Q_1 and Q_2 and calculating the Richardson α^k and error estimate is shown in code snippet 1.

[1]The complete code implementation can by found inside the zipped file.

```
    //Solve with SVD
    SVD obj(AA);
    VecDoub z(NN);
    obj.solve(bb, z);

    //Calculate Q1 and Q2
    VecDoub z_u(NN/2),z_v(NN/2);
    for(int i = 0; i<NN; i++){
        if(i<NN/2){
            z_u[i] = z[i];
        }
        else{
            z_v[i-NN/2] = z[i];
        }
    }
    Q1_lastlast = Q1_last;
    Q2_lastlast = Q2_last;
    Q1_last = Q1;
    Q2_last = Q2;

    //Calculate Q1 and Q2
    Q1 = Q(z_u,const1_1,const1_2,h);
    Q2 = Q(z_v,const2_1,const2_2,h);

    //Richardson alpha_k estimate
    alpha_k1 = (Q1_lastlast - Q1_last)/(Q1_last-Q1);
    alpha_k2 = (Q2_lastlast - Q2_last)/(Q2_last-Q2);


    // It is seen that both alpha_k converges to 4!
    alpha_k = 4;

    //Richardson error estimate
    error1 = (Q1 - Q1_last)/(alpha_k-1);
    error2 = (Q2 - Q2_last)/(alpha_k-1);
```

Code Snippet 1: Code snippet of main.cpp, line 139-173.