Supplementary Materials

The MATLAB codes for solving the boundary value problem are as follows:

```
function adiskvelo

format long g

%Define all parameters
global A1 Af Ahnf C1 Pr beta phi phi1 phi2 rho1 rho2
rhof rhohnf k1 S A B k2 kf cp1 cp2 cpf B1 khnf muhnf muf
k

%Boundary layer thickness & stepsize
etaMin = 0;
etaMax1 = 10;
etaMax2 = etaMax1;
stepsize1 = etaMax1;
stepsize2 = etaMax2;

%Input for the parameters
phi1 = 0.02;
phi2 = 0.02;
rho1 = 8933; %Cu
rho2 = 3970; %Al2O3
rhof = 997.1;
Pr = 6.2;
k1 = 400;
k2 = 40;
kf = 0.613;
cp1 = 385;
cp2 = 765;
cpf = 4179;
S = -0.1;
lambda = -0.9;
phi = phi1+phi2;
rhohnf=((1-phi)*rhof)+(phi1*rho1)+(phi2*rho2);
khnf=kf*((((phi1*k1+phi2*k2)/phi)+2*kf-2*phi*(kf-
((phi1*k1+phi2*k2)/phi)))/(((phi1*k1+phi2*k2)/phi)+2*kf+
phi*(kf-((phi1*k1+phi2*k2)/phi))));
k=khnf*(kf^(-1));
Af=rhof*cpf;
Ahnf=(1-phi)*Af+phi1*rho1*cp1+phi2*rho2*cp2;
A=1/((1-phi1-phi2)^2.5); %muhnf/muf
B=rhohnf*(rhof^(-1));
A1=Ahnf*(Af^(-1));
B1=B*(A^(-1));
C1=A1*(k^(-1));

%%%%%%%%%%%%%%%%%%%%%% first
solution %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

options = bvpset( 'stats', 'off', 'RelTol',1e-10);
```

```
solinit = bvpinit (linspace (etaMin, etaMax1,
stepsize1), @OdeInit1);
sol = bvp4c (@OdeBVP, @OdeBC, solinit, options);
eta = linspace (etaMin, etaMax1, stepsize1);
y = deval (sol, eta);

figure(1)
plot(sol.x,sol.y(2,:),'-.r')
xlabel('\eta')
ylabel( 'f(\eta)')
hold on

figure(2)
plot(sol.x,sol.y(1,:),'-.r')
xlabel('\eta')
ylabel( 'h(\eta)')
hold on
figure(3)
plot(sol.x,sol.y(4,:),'-.r')
xlabel('\eta')
ylabel( 'g(\eta)')
hold on

figure(4)
plot(sol.x,sol.y(6,:),'-.r')
xlabel('\eta')
ylabel('\theta(\eta)')
hold on

%Saving the output in txt file for first solution
descris = [sol.x; sol.y];
save  'k0_first.txt' descris -ascii

%Displaying the output for first solution
fprintf('\nFirst solution:\n');
fprintf( 'Cf
= %7.9f\n',(A*((y(3)*y(3)+y(5)*y(5))^(1/2))));
fprintf( 'Nu = %7.9f\n',(-(khnf/kf)*(y(7))));
fprintf( 'f = %7.9f\n',(y(3)));
fprintf( 'g = %7.9f\n',(y(5)));
fprintf('\n');


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%

%Define the ODE function
function ff = OdeBVP (x, y,C1, B1, Pr)

global C1 B1 Pr

ff = [-2*y(2)
y(3)
B1*(y(1)*y(3)+y(2)*y(2)-y(4)*y(4))
y(5)
```

```
B1*(y(1)*y(5)+2*y(2)*y(4))
y(7)
C1*Pr*(y(1)*y(7))];

%Define the boundary condition
function res = OdeBC (ya, yb, beta, S)
global beta S

res = [ya(1)-S
ya(2)-beta
ya(4)-1
ya(6)-1
yb(2)
yb(4)
yb(6)];
```