BOUNDARY ELEMENT METHOD FOR 3D FOURIER-KIRCHHOFF HEAT TRANSFER EQUATION

***Ewa Majchrzak*[1, 2], *Łukasz Turchan*[1]**

[1]*Department of Strength of Materials and Computational Mechanics*
Silesian University of Technology, Poland
[2]*Institute of Mathematics, Czestochowa University of Technology, Poland*
ewa.majchrzak@polsl.pl, lukasz.turchan@polsl.pl

Abstract. The 3D heat transfer problem (steady state) is considered. The equation describing the thermal processes contains the convective term (substantial derivative). The problem is solved by means of the boundary element method. The numerical model for constant boundary elements and constant internal cells is presented. In the final part of the paper the examples of computations are shown. The numerical results obtained by means of the BEM are compared with analytical solution and the very good compatibility can be observed.

1. Formulation of the problem

The following Fourier-Kirchhoff equation (steady state, 3D problem) is considered

$$x \in \Omega: \quad \lambda \nabla^2 T(x) - c\mathbf{u} \cdot \nabla T(x) + Q(x) = 0 \tag{1}$$

where λ is the thermal conductivity and c is the volumetric specific heat, respectively $\mathbf{u} = [u_1, u_2, u_3]$ is the velocity, $Q(x)$ is the source function, T denotes the temperature and $x = [x_1, x_2, x_3]$ are the spatial co-ordinates.

For $\mathbf{u} = [u_1, 0, 0]$ the equation (1) takes a form

$$x \in \Omega: \quad \lambda \nabla^2 T(x) - c u_1 \frac{\partial T(x)}{\partial x_1} + Q(x) = 0 \tag{2}$$

The boundary conditions in the form

$$\begin{aligned} x \in \Gamma_1: &\quad T(x) = T_b \\ x \in \Gamma_2: &\quad q(x) = -\lambda \frac{\partial T(x)}{\partial n} = q_b \end{aligned} \tag{3}$$

are also known, at the same time $\partial T/\partial n$ denotes the normal derivative, $\mathbf{n} = [\cos \alpha_1, \cos \alpha_2, \cos \alpha_3]$ is the normal outward vector, T_b, q_b are the known boundary temperature and boundary heat flux, respectively.

To solve the problem (2), (3) the boundary element method is proposed.

2. Boundary element method

At first, the weighted residual criterion [1, 2] for equation (2) is formulated

$$\int_{\Omega}\left[\lambda\nabla^2 T(x) - cu_1\frac{\partial T(x)}{\partial x_1} + Q(x)\right]T^*(\xi, x)\,\mathrm{d}\Omega = 0 \tag{4}$$

where ξ is the observation point and $T^*(\xi, x)$ is the fundamental solution.

The integral (4) is substituted by a sum of two integrals, while the first of them is transformed using the 2nd Green formula

$$\begin{gathered}\int_{\Omega}\lambda\nabla^2 T(x)T^*(\xi, x)\,\mathrm{d}\Omega = \int_{\Omega}\lambda\nabla T^*(\xi, x)T(x)\,\mathrm{d}\Omega + \\ \int_{\Gamma}\left[\lambda T^*(\xi, x)\frac{\partial T(x)}{\partial n} - \lambda\frac{\partial T^*(\xi, x)}{\partial n}T(x)\right]\mathrm{d}\Gamma\end{gathered} \tag{5}$$

The next integral in formula (4) is integrated by parts

$$\begin{gathered}\int_{\Omega}cu_1\frac{\partial T(x)}{\partial x_1}T^*(\xi, x)\,\mathrm{d}\Omega = \int_{\Gamma}cu_1 T^*(\xi, x)T(x)\cos\alpha_1\,\mathrm{d}\Gamma - \\ \int_{\Omega}cu_1\frac{\partial T^*(\xi, x)}{\partial x_1}T(x)\,\mathrm{d}\Omega\end{gathered} \tag{6}$$

Introducing (5), (6)into (4) one has

$$\begin{gathered}\int_{\Omega}\left[\lambda\nabla^2 T^*(\xi, x) + cu_1\frac{\partial T^*(\xi, x)}{\partial x_1}\right]T(x)\,\mathrm{d}\Omega + \\ \int_{\Gamma}\left[\lambda T^*(\xi, x)\frac{\partial T(x)}{\partial n} - \lambda\frac{\partial T^*(\xi, x)}{\partial n}T(x) - cu_1 T^*(\xi, x)T(x)\cos\alpha_1\right]\mathrm{d}\Gamma + \\ \int_{\Omega}Q(x)T^*(\xi, x)\,\mathrm{d}\Omega = 0\end{gathered} \tag{7}$$

Fundamental solution $T^*(\xi, x)$ should fulfil the following equation

$$x \in \Omega: \quad \lambda\nabla^2 T^*(\xi, x) + cu_1\frac{\partial T^*(\xi, x)}{\partial x_1} = -\delta(\xi, x) \tag{8}$$

where $\delta(\xi, x)$ is the Dirac function.

Taking into account the property (8) the equation (7) takes a form

$$
\begin{gathered}
T(\xi)+\int_{\Gamma} T^{*}(\xi, x) q(x) \mathrm{d}\Gamma= \\
\int_{\Gamma}\left[q^{*}(\xi, x)-c u_{1} T^{*}(\xi, x) \cos \alpha_{1}\right] T(x) \mathrm{d}\Gamma+\int_{\Omega} Q(x) T^{*}(\xi, x) \mathrm{d}\Omega
\end{gathered} \tag{9}
$$

where

$$
q(x)=-\lambda \frac{\partial T(x)}{\partial n}, \quad q^{*}(\xi, x)=-\lambda \frac{\partial T^{*}(\xi, x)}{\partial n} \tag{10}
$$

For $\xi \in \Gamma$ the boundary integral equation is obtained

$$
\begin{gathered}
B(\xi) T(\xi)+\int_{\Gamma} T^{*}(\xi, x) q(x) \mathrm{d}\Gamma= \\
\int_{\Gamma}\left[q^{*}(\xi, x)-c u_{1} T^{*}(\xi, x) \cos \alpha_{1}\right] T(x) \mathrm{d}\Gamma+\int_{\Omega} Q(x) T^{*}(\xi, x) \mathrm{d}\Omega
\end{gathered} \tag{11}
$$

where $B(\xi) \in (0, 1)$ is the coefficient connected with the location of point ξ on the boundary Γ.

For the problem considered the fundamental solution is the following [3]

$$
T^{*}(\xi, x)=\frac{1}{4 \pi \lambda r} \exp \left(-\frac{c u_{1}}{2 \lambda}\left[r+\left(x_{1}-\xi_{1}\right)\right]\right) \tag{12}
$$

where r is the distance between the points $\xi = (\xi_1, \xi_2, \xi_3)$ and $x = (x_1, x_2, x_3)$.

Using formula (12) the heat flux $q^*(\xi, x)$ resulting from fundamental solution can be calculated

$$
\begin{gathered}
q^{*}(\xi, x)=\frac{1}{4 \pi r^{3}} \exp \left(-\frac{c u_{1}}{2 \lambda}\left[r+\left(x_{1}-\xi_{1}\right)\right]\right) \cdot \\
{\left[d\left(1+\frac{c u_{1}}{2 \lambda} r\right)+\frac{c u_{1}}{2 \lambda} r^{2} \cos \alpha_{1}\right]}
\end{gathered} \tag{13}
$$

where

$$
d=\left(x_{1}-\xi_{1}\right) \cos \alpha_{1}+\left(x_{2}-\xi_{2}\right) \cos \alpha_{2}+\left(x_{3}-\xi_{3}\right) \cos \alpha_{3} \tag{14}
$$

3. Numerical realization

To solve equation (11) the boundary is divided into N boundary elements and the interior is divided into L internals cells. Next, the integrals appearing in (11) are substituted by the sums of integrals.

So, for optional boundary point $\xi^i \in \Gamma$ one has

$$
\begin{gathered}
B\left(\xi^i\right)T\left(\xi^i\right)+\sum_{j=1}^{N}\int\limits_{\Gamma_j} T^*\left(\xi^i,x\right)q\left(x\right)\mathrm{d}\Gamma_j = \\
\sum_{j=1}^{N}\int\limits_{\Gamma_j}\left[q^*\left(\xi^i,x\right)-cu_1T^*\left(\xi^i,x\right)\cos\alpha_1\right]T\left(x\right)\mathrm{d}\Gamma_j+\sum_{l=1}^{L}\int\limits_{\Omega_l}Q\left(x^l\right)T^*\left(\xi^i,x\right)\mathrm{d}\Omega_l
\end{gathered} \tag{15}
$$

When the constant boundary elements and constant internal cells are used, this means

$$
x\in\Gamma_j: \quad \begin{cases} T\left(x\right)=T\left(x^j\right)=T_j \\ q\left(x\right)=q\left(x^j\right)=q_j \end{cases} \tag{16}
$$

and

$$
x\in\Omega_l: \quad Q\left(x\right)=Q\left(x^l\right)=Q_l \tag{17}
$$

then equation (15) takes form

$$
\begin{gathered}
\frac{1}{2}T_i+\sum_{j=1}^{N}q_j\int\limits_{\Gamma_j}T^*\left(\xi^i,x\right)\mathrm{d}\Gamma_j = \\
\sum_{j=1}^{N}T_j\int\limits_{\Gamma_j}\left[q^*\left(\xi^i,x\right)-cu_1T^*\left(\xi^i,x\right)\cos\alpha_1\right]\mathrm{d}\Gamma_j+\sum_{l=1}^{L}Q_l\int\limits_{\Omega_l}T^*\left(\xi^i,x\right)\mathrm{d}\Omega_l
\end{gathered} \tag{18}
$$

or

$$
\sum_{j=1}^{N}G_{i,j}q_j=\sum_{j=1}^{N}\left(H_{i,j}-U_{i,j}\right)T_j+\sum_{l=1}^{L}P_{i,l}Q_l, \quad i=1,2,\ldots,N \tag{19}
$$

where

$$
G_{i,j}=\int\limits_{\Gamma_j}T^*\left(\xi^i,x\right)\mathrm{d}\Gamma_j \tag{20}
$$

and

$$
H_{i,j}=\begin{cases} \displaystyle\int\limits_{\Gamma_j}q^*\left(\xi^i,x\right)\mathrm{d}\Gamma_j, & i\neq j \\ \displaystyle\int\limits_{\Gamma_j}q^*\left(\xi^i,x\right)\mathrm{d}\Gamma_j-\frac{1}{2}, & i=j \end{cases} \tag{21}
$$

while

$$
U_{i,j}=c\,u_1\int\limits_{\Gamma_j}T^*\left(\xi^i,x\right)\cos\alpha_1\mathrm{d}\Gamma_j \tag{22}
$$

and

$$P_{i,l} = \int_{\Omega_l} T^*\left(\xi^i, x\right) \mathrm{d}\Omega_l \tag{23}$$

The system of equations (19) allows one to determine the "missing" boundary values T_j and q_j. Next, the temperatures at the internal points ξ^i, $i = N + 1, N + 2, ..., N + L$ can be calculated using the formula

$$T_i = \sum_{j=1}^{N} \left(H_{i,j} - U_{i,j}\right) T_j - \sum_{j=1}^{N} G_{i,j} q_j + \sum_{l=1}^{L} P_{i,l} Q_l \tag{24}$$

4. Discretization of boundary and interior of cuboid

The cuboid of dimension l_1 x l_2 x l_3 is considered (Fig. 1). The boundary is divided into N constant boundary elements. It is assumed that $hs = l_1 / n_1$, $n_2 = l_2 / hs$, $n_3 = l_3 / hs$, and then $N = 2n_1n_2 + 2n_1n_3 + 2n_2n_3$. Below the fragment of Delphi code in which the boundary nodes coordinates are defined is shown.

```
{przod}
    for i:=1 to n1*n3 do y[i]:=0;
    for k:=0 to n3-1 do
          for i:=1 to n1 do
                begin
                x[i+n1*k]:=i*hs-hs/2.0;
                z[i+n1*k]:=k*hs+hs/2.0;
                end;
{prawa}
    for i:=n1*n3+1 to n1*n3+n2*n3 do x[i]:=n1*hs;
    for k:=0 to n3-1 do
          for j:=1 to n2 do
                begin
                y[j+n2*k+n1*n3]:=j*hs-hs/2.0;
                z[j+n2*k+n1*n3]:=k*hs+hs/2.0;
                end;
{tyl}
    for i:=n1*n3+n2*n3+1 to n1*n3+2*n2*n3 do y[i]:=n2*hs;
    for k:=0 to n3-1 do
          for i:=1 to n1 do
                begin
                x[i+n1*k+n1*n3+n2*n3]:=n1*hs-(i*hs-hs/2.0);
                z[i+n1*k+n1*n3+n2*n3]:=k*hs+hs/2.0;
                end;
{lewa}
    for i:=2*n1*n3+n2*n3+1 to 2*n1*n3+2*n2*n3 do x[i]:=0.0;
    for k:=0 to n3-1 do
          for j:=1 to n2 do
             begin
             y[j+n2*k+2*n1*n3+n2*n3]:=n2*hs-(j*hs-hs/2.0);
```

```
            z[j+n2*k+2*n1*n3+n2*n3]:=k*hs+hs/2.0;
            end;
{dol}
    for  i:=2*n1*n3+2*n2*n3+1  to  2*n1*n3+2*n2*n3+n1*n2  do
          z[i]:=0.0;
    for j:=0 to n2-1 do
          for i:=1 to n1 do
                begin
                x[i+n1*j+2*n1*n3+2*n2*n3]:=i*hs-hs/2.0;
                y[i+n1*j+2*n1*n3+2*n2*n3]:=j*hs+hs/2.0;
                end;
{gora}
    for i:=2*n1*n3+2*n2*n3+n1*n2+1 to N do z[i]:=n3*hs;
    for j:=0 to n2-1 do

          for i:=1 to n1 do
             begin
             x[i+n1*j+2*n1*n3+2*n2*n3+n1*n2]:=i*hs-hs/2.0;
             y[i+n1*j+2*n1*n3+2*n2*n3+n1*n2]:=j*hs+hs/2.0;
             end;
```

The coordinates of internal nodes can be determined as follows:

```
{wewnetrzne}
    for k:=0 to n3-1 do
    for j:=0 to n2-1 do
          for i:=1 to n1 do
                begin
                x[N+i+n1*j+n1*n2*k]:=i*hs-hs/2.0;
                y[N+i+n1*j+n1*n2*k]:=j*hs+hs/2.0;
                z[N+i+n1*j+n1*n2*k]:=k*hs+hs/2.0;
                end;
```

To calculate the integrals (20), (21) and (22) the Gaussian cubatures method is used [1, 2]. In this method the coordinates of vertexes of each boundary element (quadrilateral) should be known. If the vertexes of quadrilateral Γ_j one denotes by (x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3), (x_4, y_4, z_4) then on the basis of boundary node coordinates `x[j]`, `y[j]`, `z[j]` they can be defined as follows:

```
if j <= n1*n3 then begin         {przod}
    x1:=x[j]-hs/2;    x2:=x[j]+hs/2;    x3:=x[j]+hs/2;
    x4:=x[j]-hs/2;
    y1:=0.0;    y2:=0.0;          y3:=0.0;          y4:=0.0;
    z1:=z[j]-hs/2;    z2:=z[j]-hs/2;    z3:=z[j]+hs/2;
    z4:=z[j]+hs/2;
    end
else
if j <= n1*n3+n2*n3 then begin         {prawa}
    x1:=l1;     x2:=l1;           x3:=l1;           x4:=l1;
    y1:=y[j]-hs/2;    y2:=y[j]+hs/2;    y3:=y[j]+hs/2;
    y4:=y[j]-hs/2;
```

```
      z1:=z[j]-hs/2;    z2:=z[j]-hs/2;    z3:=z[j]+hs/2;
      z4:=z[j]+hs/2;
      end
   else
   if j <= 2*n1*n3+n2*n3 then begin {tyl}
      x1:=x[j]+hs/2;    x2:=x[j]-hs/2;    x3:=x[j]-hs/2;
      x4:=x[j]+hs/2;
      y1:=l2;     y2:=l2;           y3:=l2;           y4:=l2;
      z1:=z[j]-hs/2;    z2:=z[j]-hs/2;    z3:=z[j]+hs/2;
      z4:=z[j]+hs/2;
      end
   else
   if j <= 2*n1*n3+2*n2*n3 then begin     {lewa}
      x1:=0.0;    x2:=0.0;          x3:=0.0;          x4:=0.0;
      y1:=y[j]+hs/2;    y2:=y[j]+hs/2;    y3:=y[j]-hs/2;
      y4:=y[j]-hs/2;
      z1:=z[j]+hs/2;    z2:=z[j]-hs/2;    z3:=z[j]-hs/2;
      z4:=z[j]+hs/2;
      end
   else
   if j <= 2*n1*n3+2*n2*n3+n1*n2 then begin     {dol}
      x1:=x[j]-hs/2;    x2:=x[j]-hs/2;    x3:=x[j]+hs/2;
      x4:=x[j]+hs/2;
      y1:=y[j]-hs/2;    y2:=y[j]+hs/2;    y3:=y[j]+hs/2;
      y4:=y[j]-hs/2;
      z1:=0.0;    z2:=0.0;          z3:=0.0;          z4:=0.0;
      end
   else
   if j <= N then begin {gora}\
      x1:=x[j]-hs/2;    x2:=x[j]+hs/2;    x3:=x[j]+hs/2;
      x4:=x[j]-hs/2;
      y1:=y[j]-hs/2;    y2:=y[j]-hs/2;    y3:=y[j]+hs/2;
      y4:=y[j]+hs/2;
      z1:=l3;     z2:=l3;           z3:=l3;           z4:=l3;
      end;
```

After determining the integrals $G_{i,j}$, $H_{i,j}$, $U_{i,j}$, $P_{i,l}$ and taking into account the boundary conditions (3), the system of equations (19) can be solved by means of the Gaussian elimination method [2].

5. Example of computations

The following input data are introduced: thermal conductivity $\lambda = 10$ W/(m·K), volumetric specific heat $c = 10^6$ J/(m^3·K), velocity $u_1 = 0.0001$ m/s, source function $Q = 0$ W/m^3.

The cuboid of dimension 0.05 x 0.05 x 0.025 m^3 is considered as shown in Figure 1. It's assumed that $n_1 = n_2 = 10$, $n_3 = 5$, so $N = 400$ boundary elements have been distinguished. It should be pointed out that in the case $Q = 0$ only the boundary should be discretized (c.f. equations (9) and (18)).

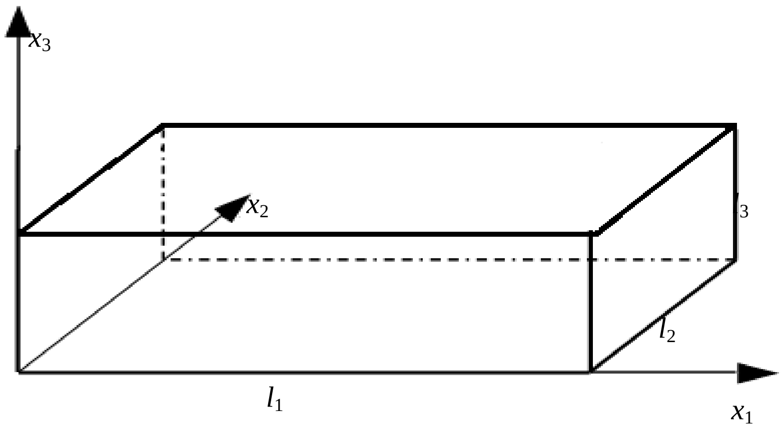

Fig 1. Domain considered

The following boundary conditions are accepted:

$$
\begin{array}{llll}
x_1 = 0, & 0 < x_2 < l_2, & 0 < x_3 < l_3: & T(x) = 50 \\
x_1 = l_1, & 0 < x_2 < l_2, & 0 < x_3 < l_3: & T(x) = 100 \\
0 < x_1 < l_1, & x_2 = 0, & 0 < x_3 < l_3: & q(x) = 0 \\
0 < x_1 < l_1, & x_2 = l_2, & 0 < x_3 < l_3: & q(x) = 0 \\
0 < x_1 < l_1, & 0 < x_2 < l_2, & x_3 = 0: & q(x) = 0 \\
0 < x_1 < l_1, & 0 < x_2 < l_2, & x_3 = l_3: & q(x) = 0
\end{array}
\tag{25}
$$

In Figure 2 the temperature distribution in the plane $x_2 = l_2 / 2$ is shown. Taking into account the assumed boundary conditions, the results are the same for each plane $x_2 = s$, where $s \in [0, l_2]$.

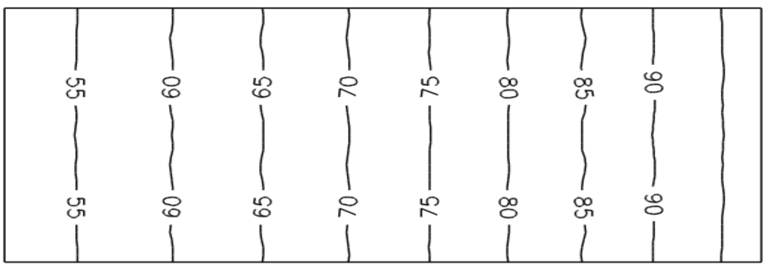

Fig. 2. Temperature distribution (plane $x_2 = l_2 / 2$)

The solution obtained can be compared with analytical solution for 1D problem

$$
T_a(x_1) = C_1 + C_2 \exp\left(\frac{u_1 c}{\lambda} x_1\right)
\tag{26}
$$

where C_1, C_2 are the integral constants determined from boundary conditions $T_a(0) = 50°C$ and $T_a(l_1) = 100°C$.

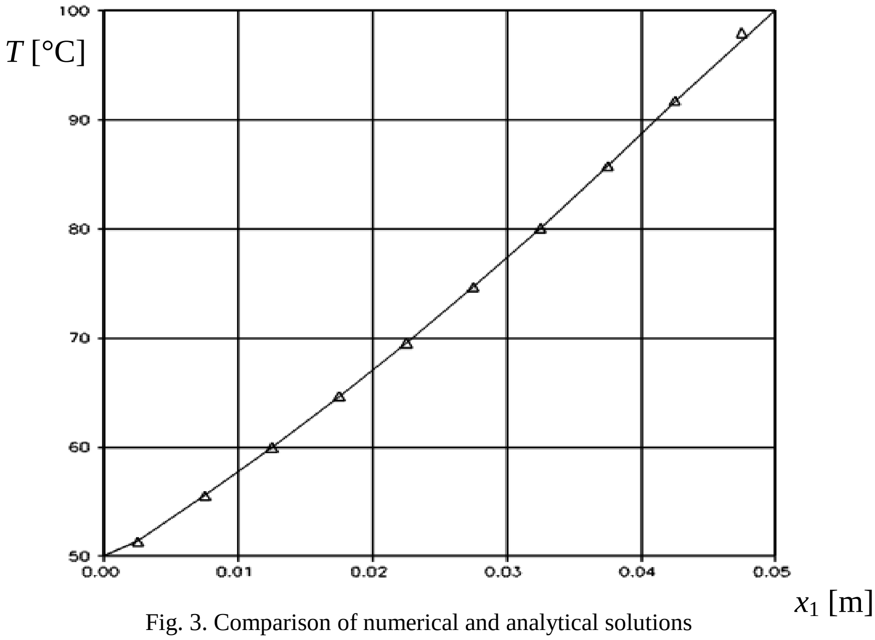

Fig. 3. Comparison of numerical and analytical solutions

The results are shown in Figure 3, where the solid line presents the analytical solution, while the symbols illustrate the numerical one. It is visible, that both solutions are practically the same.

Conclusions

Application of the boundary element method for numerical solution of 3D Fourier-Kirchhoff equation is presented. Both the theoretical and practical aspects of the problem solution are discussed. Among others, the way of discretization for cuboid using constant boundary elements is shown. Numerical results compared with analytical solution confirm the exactness and effectiveness of the algorithm proposed.

The method presented can be applied for numerical modelling of heat transfer proceeding in domain of porous media, in particular the bioheat transfer equation basing on the theory of porous media [4] can be considered.

Acknowledgement

This paper is a part of Grant No N N501 3667 34.

References

[1] Brebbia C.A., Dominguez J., Boundary elements, an introductory course, Computational Mechanics Publications, McGraw-Hill Book Company, London 1992.

[2] Majchrzak E., Metoda elementów brzegowych w przepływie ciepła, Wyd. Pol. Częstochowskiej, Częstochowa 2001.

[3] Klekot J., Boundary element method for elliptic equation - determination of fundamental solution. Scientific Research of the Institute of Mathematics and Computer Science, Czestochowa University of Technology 2010, 1(9), 61-70

[4] Nakayama A., Kuwanara F., A general bioheat transfer model based on the theory of porous media, Int. J. of Heat and Mass Transfer 2008, 51, 3190-3199.