September 10 - 12, 2007
Pilsen, Czech Republic

FORCES AND TORQUES FOR LINEAR, CYLINDRICAL AND SPHERICAL MOTORS - ANALYTICAL CALCULATIONS

D. SPAŁEK, K. WALECZEK

Abstract: *The paper deals with the problem of electromagnetic field analysis for linear, cylindrical and spherical electromechanical converters. The electromagnetic field is determined analytically with the help of variables separation method. The forces and torques in electromagnetic field are evaluated by Maxwell and Lorentz methods analytically.*

Key words: *Analytical solutions, linear, cylindrical, spherical motor*

INTRODUCTION

Modern technologies enable to construct electromechanical converters at different geometry and material parameters. Particularly, electromechanical converters (e.g. induction motors) – linear, cylindrical and spherically shaped could contain magnetically anisotropic parts. The intention of this paper is to present the analytical solutions of electromagnetic field equations for linear, cylindrical and spherical induction motor that will be used for electromagnetic force/torque calculations.

1 MAIN EQUATIONS

The first pair of Maxwell equations take the well-known form

$$\mathrm{curl}\vec{E} = -\dot{\vec{B}} \quad \wedge \quad \mathrm{div}\vec{B} = 0 . \qquad (1)$$

The second pair of Maxwell equations can be presented in vector notation as follows

$$\mathrm{div}\vec{D} = \rho \quad \wedge \quad \mathrm{curl}\vec{H} = \vec{j} + \dot{\vec{D}} . \qquad (2)$$

Constitutive relations for electromagnetic field vectors for non-hysteresis medium [3, 4, 8, 11, 13] are

$$H_u = \nu_{uv} B_v , \qquad (3)$$

$$D_u = \varepsilon_{uv} E_v , \qquad (4)$$

where ε_{uv} denote dielectric permittivity, ν_{uv} are magnetic reluctivity, *u,v* mean number of curvilinear system co-ordinate (summation due to twice appearing indices is accepted). For the three considered cases of electromechanical converters (linear, cylindrical and spherical - see figures below) only one component of magnetic vector potential does not vanish, it was denoted as third component [1, 4, 19] i.e.

for linear problem (Cartesian co-ordinate system 1-*x*, 2-*y*, 3-z, , L_x=L_y=L_z=1)

$$\vec{A} = \vec{A}_z = A_z \vec{i}_z = A \vec{i}_z , \qquad (5)$$

for cylindrical problem (cylindrical co-ordinate system 1-r, 2-α, 3- z, , L_r=1, L_α=*r*, L_z=1)

$$\vec{A} = \vec{A}_z = A_z \vec{i}_z = A \vec{i}_z , \qquad (6)$$

for spherical problem (spherical co-ordinate system 1-*r*, 2-*φ*, 3-*θ*, L_r=1, $L_\varphi = r\sin\theta$, L_θ=*r*)

$$\vec{A} = \vec{A}_\theta = A_\theta \vec{i}_\theta = A \vec{i}_\theta . \qquad (7)$$

1.1 Co-ordinates and notation

Lame coefficients for Cartesian, cylindrical and spherical co-ordinate systems are grouped in Table 1.

Co-ordinate System	L_1	L_2	L_3
Cartesian (x_1=x, x_2=y, x_3=z)	1	1	1
Cylindrical (x_1=r, x_2=α, x_3=z)	1	r	1
Spherical (x_1=r, x_2=φ, x_3=θ)	1	rsinθ	r

The Lame coefficients are grouped in Table 1. The assumed magnetic vector potential placement is due to the shape of magnetomotive force pattern and adequate co-ordinate system placement. The accuracy of such assumption for magnetic vector potential symmetry results from the technical construction of each mechanical converter i.e. linear, cylindrical and spherical.

2 SOLUTION OF PROBLEMS LINEAR, CYLINDRICAL AND SPHERICAL

The magnetic field flux density can be written in unified form with the help of Lame coefficients as follows

$$\vec{B} = \frac{\vec{i}_1}{L_2 L_3}\frac{\partial A L_3}{\partial x_2} - \frac{\vec{i}_2}{L_1 L_3}\frac{\partial A L_3}{\partial x_1}. \quad (8)$$

This notation simplifies the analysis for three cases considered. The magnetic field strength components for the anisotropic region can be shown in the form of

$$H_1 = \nu_{11} B_1 + \nu_{12} B_2, \quad (9)$$

$$H_2 = \nu_{21} B_1 + \nu_{22} B_2, \quad (10)$$

because the third component of magnetic field strength disappears B_3=0 due to equation (8).

3 ELECTROMAGNETIC FIELD DISTRIBUTION

For homogeneous region magnetic reluctivities are spatially constant, and due to Eqns (1) - (11) the main equation for magnetic vector potential takes the given below form

$$\nu_{21}\frac{\partial^2 A}{\partial x_1 \partial x_2} - \nu_{22}\frac{\partial}{\partial x_1}\frac{L_2}{L_3}\frac{\partial A L_3}{\partial x_1} - \nu_{11}\frac{\partial}{\partial x_2}\frac{\partial A}{L_2 \partial x_2} + \nu_{12}\frac{\partial}{\partial x_2}\frac{\partial A L_3}{L_3 \partial x_1} = -L_2 \gamma \dot{A} \quad (11)$$

The equation (11) leads to the three relations for the linear, cylindrical and spherical electromechanical converters, respectively.

At complex analysis the time-partial derivative of A is presented as multiplication of the operand iω (i means imaginary unit) and the complex magnetic potential A at the steady state for time-sinusoidal varying fields [1, 2, 8, 9] as follows

$$\dot{\vec{A}} \to i\omega \vec{A}, \quad (12)$$

where ω is field pulsation. The magnetic vector potential since now is complex number.

The separation of variables method is defined in form given below

$$A = A_3(x_1, x_2, x_3) = R(x_1, x_3)F(x_2) = R \cdot F. \quad (13)$$

For the function $F(x_2)$ it is assumed that the separation constant equals to p^2 for the first mmf space harmonic h = 1 i.e.

$$\frac{1}{F}\frac{\partial^2 F}{\partial x_2^2} = -p^2, \quad (14)$$

and for mmf higher space harmonics p is replaced by ph.

For the linear converter (Cartesian co-ordinate system, see Fig1)

$$\frac{d^2 R}{dx^2} + ik\frac{\nu_{xy} + \nu_{yx}}{\nu_{yy}}\frac{dR}{dx} - a_0 R = 0. \quad (15)$$

where

$$a_0 = k^2 \frac{\nu_{xx}}{\nu_{yy}} + \frac{i\omega\gamma}{\nu_{yy}}, \quad a_1 = \frac{\nu_{xy} + \nu_{yx}}{2\nu_{yy}} ki, \quad (16a,b)$$

$$\lambda_{1,2} = -a_1 \pm \sqrt{a_1^2 + a_0}, \quad (17)$$

with the solutions in the form of

$$R(x) = a_a \exp(\lambda_1 x) + b_a \exp(\lambda_2 x). \quad (18)$$

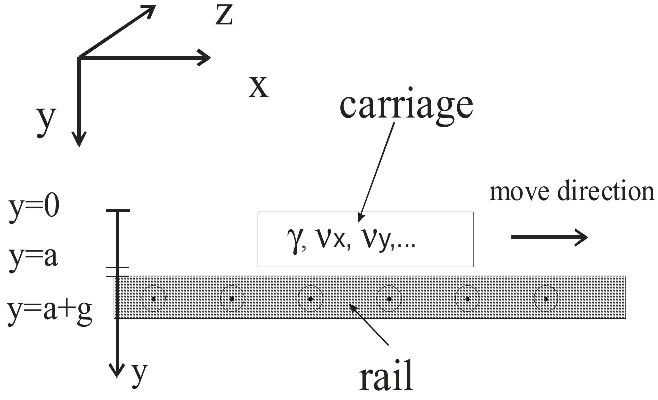

Fig. 1. Linear motor

For the cylindrical converter (cylindrical co-ordinate system)

$$\frac{d^2 R}{dr^2} + \frac{(1-2c)}{r}\frac{dR}{dr} - \left[\frac{\nu_{rr} p^2}{\nu_{\alpha\alpha} r^2} + \beta^2\right] R = 0, \quad (19)$$

where

$$c = -pi(\nu_{r\alpha} + \nu_{\alpha r})/2\nu_{\alpha\alpha},$$

$$\beta = \sqrt{i\omega\gamma/\nu_{\alpha\alpha}},$$

$$p_B = \sqrt{c^2 + p^2 \nu_{rr}/\nu_{\alpha\alpha}}, \quad (20a,b,c)$$

with the solution in the form [7, p.362]

$$R(\beta r) = a_a (\beta r)^c I_{pB}(\beta r) + b_a (\beta r)^c K_{pB}(\beta r). \quad (21)$$

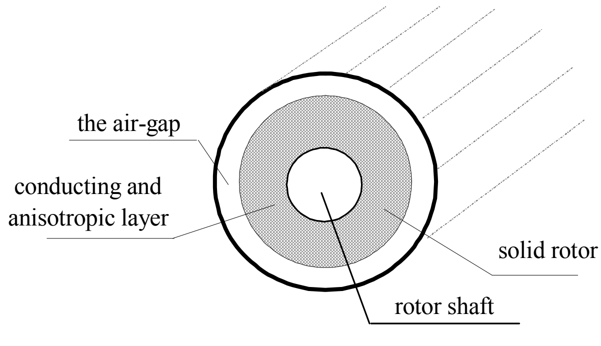

Fig.2: Cylindrical induction motor model

The solution for spherical symmetry for a unidirectional rotating field leads to equation in the form of

$$\frac{\partial^2 R}{\partial r^2} + 2(1 - h(\theta))\frac{\partial R}{\partial r} + R\left(-\beta^2 - \frac{p^2 \nu_{rr} - ip\sin\theta\,\nu_{r\varphi}}{r^2 \nu_{\varphi\varphi}\sin^2\theta}\right) = 0,$$

(22)

with the following analytical solution for an anisotropic region [7] (p. 363 Eqn B110(3)), as well as it could be checked by putting in):

$$R(r,\theta) = (\beta r)^{\lambda(\theta)-\frac{1}{2}}\left(C_1 I_{\lambda(\theta)}(\beta r) + C_2 K_{\lambda(\theta)}(\beta r)\right) \quad (23)$$

where

$$\lambda(\theta) = \pm\sqrt{(h(\theta)-\tfrac{1}{2})^2 + \frac{p^2 \nu_{rr} - ip\sin\theta\,\nu_{r\varphi}}{r^2 \nu_{\varphi\varphi}\sin^2\theta}},$$

$$h(\theta) = -\frac{\nu_{r\varphi}+\nu_{\varphi r}}{2\nu_{\varphi\varphi}\sin\theta}ip, \quad \beta^2 = \frac{i\gamma\omega}{\nu_{\varphi\varphi}}. \qquad (24a,b,c)$$

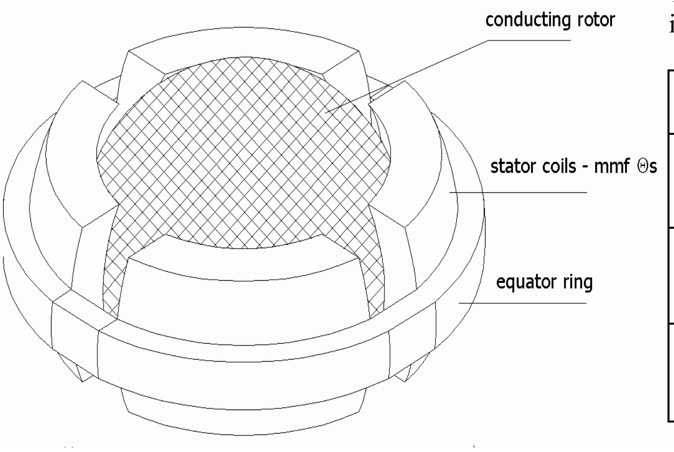

Fig.3. Spherical electromechanical converter

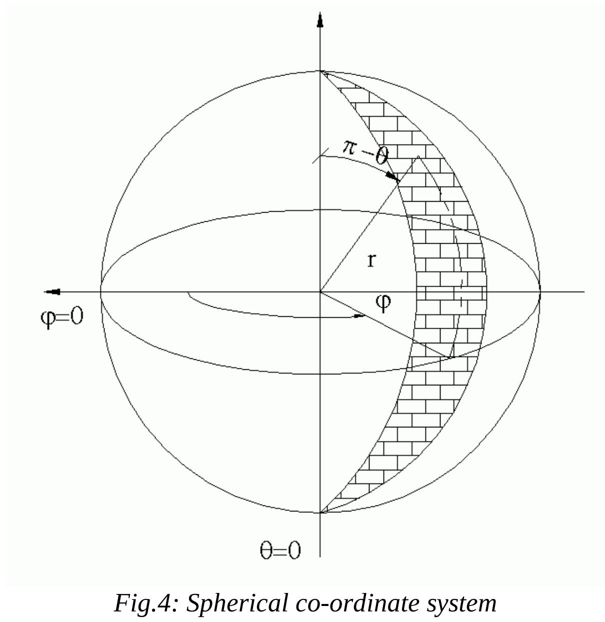

Fig.4: Spherical co-ordinate system

4 SOLUTIONS AND BOUNDARY CONDITIONS

There are four conditions defined for the electromagnetic field vectors. They enable to calculate the four unknown constants a_a, b_a, a_δ, b_δ. The boundary conditions result from physical principles:

a) The magnetic field strength disappears at the inner layer surface (r=R-a) as a consequence of the fact that magnetic reluctivity of the rotor core is assumed to be zero.

b) The continuity of the normal (radial) magnetic flux density (r=R).

c) The longitudinal (tangential) component of the magnetic field strength (r=R).

d) The magnetomotive force of the electromechanical converter stator currents leads to the following condition for the longitudinal (tangential) component of magnetic field strength at the stator surface (r=R+g).

The analysis of electromagnetic field is the background for electromechanical converter force/torque analysis. The obtained solution for magnetic field vector potential is used for force/torque calculation.

Co-ordinate System	$A=A(x_1, x_2, x_3)$	Function
Cartesian (x_1=x, x_2=y, x_3=z)	A=X(x)Y(y)	$Y(y) = \exp(-iky)$
Cylindrical (x_1=r, x_2=α, x_3=z)	A=R(r)S(α)	$S(\alpha) = \exp(-ip\alpha)$
Spherical (x_1=r, x_2=φ, x_3=θ)	A=R(r,θ)F(φ)	$F(\varphi) = \exp(-ip\varphi)$

Tab. 1: Separation of variables method applied for linear, cylindrical and spherical converters

5 ELECTROMAGNETIC FORCE AND TORQUE

While evaluating the magnetic field potential distribution both the magnetic flux density and the electromagnetic torque can be evaluated, analytically. Exemplary, the Maxwell stress tensor leads to the total electromagnetic torque by means of the formula for cylindrical case as follows [5, 6, 14, 15, 19]

$$T_e = \nu_o r \int_{\partial V} B_2 B_1 dS. \qquad (25)$$

The electromagnetic torque has been calculated with the help of Lorentz formula for cylindrical case

$$T_{eL} = \int_V r j_z B_r dV. \qquad (26)$$

The both torques are equal. Analogously for linear converter (for forces) and for spherical converter (torque around z-axis) are calculated.

The chosen result of calculation are shown in Figs. 5, 6. The calculations has been provided with C++ program that main form is shown in Fig. 7 (see Appendix). The program enables to evaluate:

- forces/torques (Maxwell, Lorentz and coenergy method),
- power losses by Poynting vector and Joule formula,

- curves for synchormous/induction electro-mechanical converters versus power angle/ speed,
- variation of some parameters e.g. air-gap width,
- influence of higher mmf space harmonics - Fig7.

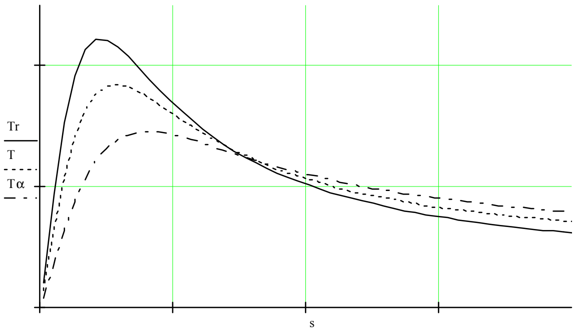

Fig.5: Cylindrical induction motor torque - speed curve

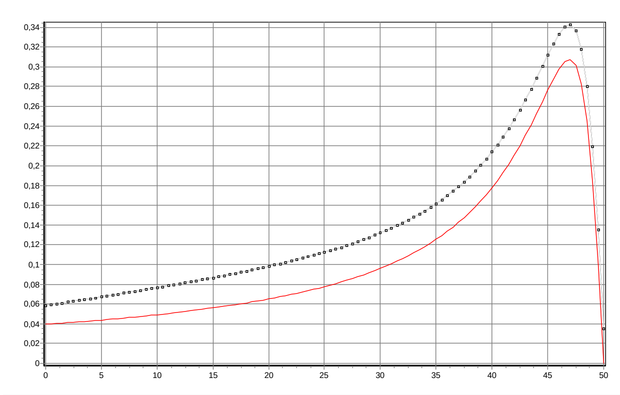

Fig.6: Spherical motor torque - speed curves for first mmf space harmonic - solid line, and for all considered mmf harmonics - dash-doted line, (SI units)

CONCLUSIONS

The paper presents electromagnetic force/torque calculation described in analytical way.

The electromagnetic field distribution is evaluated with the help of variables separation method for linear, cylindrically and spherically shaped fields.

The forces/torques are calculated by both Maxwell and Lorentz methods.

The presented analyses can be used as benchmark tasks for numerical methods.

The C++ designed program is available at
http://www.elektr.polsl.pl/dspalek

6 REFERENCES

[1] Adkins B., Harley P.G.: The general theory of alternating current machines. Chapman and Hall, London 1978.

[2] Bolte E., Hahlweg C.: Analysis of steady-state of high speed induction motors with exterior rotor and conductive layer on the slotted stator. Proceedings of XV ICEM , Belgium, 2002.

[3] Workshop TEAM http://ics.ec-lyon.fr/team.html (2005)

[4] BOLTE E., HAHLWEG C.: 'Analysis of steady-state of high speed induction motors with exterior rotor and conductive layer on the slotted stator' *Proceedings of International Conference on Electrical Machine ICEM*, 2002 p.19

[5] DAVEY K., VACHTSEVANOS G., POWERS R.: The analysis of fields and torques in spherical induction motors. IEEE Trans. Magn., Vol. 23, No. 1, pp. 273-282 (1987).

[6] LEE K.-M., ROTH R., ZHOU Z.: Dynamic modeling and control of a ball-joint-like variable reluctance spherical motor. ASME J. Dynam. Syst. Meas. Cont., Vol. 118, No. 1, pp. 29–40 (1996).

[7] GRADSZTAJN I. S., RYZIK I. M.: 'Tables of integrals, sums, series and terms' (Tablicy intjegralow, sum, rjadow i proizwjedjenij in Russian), Moscow, 1962

[8] BINNS K.J, LAWRENSON P.J., TROWBRIDGE C.W.: The analytical and numerical solution of electric and magnetic fields. John Wiley & Sons, 1994.

[9] TAVNER P.J.: 'Cross-magnetisation effects in electrical machines'. *IEE Proc. Electr. Power Appl.*, 2004, **151**, (3), pp. 249-259

[10] NETHE A., SCHOLZ T., STAHLMANN H-D.: An analytical solution method for magnetic fields using the Fourier analysis and its application of ferrofluid driven electric machines. Proc. ISTET 2003, Warsaw, Poland, Vol. II, pp. 421–424 (2003).

[11] VANDE SANDE H., HAMEYER K.: 'Comparison of neutral network and polynomial models for the approximation of nonlinear and anisotropic ferromagnetic materials'. *IEE Proc. Sci. Meas. Technol.*, 2002, **149**, (5), pp. 214-217

[12] DE GERSEM H., HAMAYER K.: 'Harmonic boundary conditions for circular inclusions and their coupling to electrical circuits'. *IEE Proc. Sci. Meas. Technol.*, 2001, **148**, (6), pp. 257-262

[13] LINDELL I.V., HÄNNINGEN J.J., NIKOSKINEN K.I.: 'Electrostatic image theory for an anisotropic boundary'. *IEE Proc. Sci. Meas. Technol.*, 2004, **153**, (3), pp. 188-194

[14] BENHAMA A., WILIAMSON A.C., REECE A.B.J.: 'Virtual work approach the computation of magnetic force distribution from finite element field solutions'. *IEE Proc. Electr. Power Appl.*, 2000, **147**, (6), pp. 437-442

[15] VANDEVELDE L., MELKEBEEK J.A.A.: 'Computation of deformation of ferromagnetic material'. *IEE Proc. Sci. Meas. Technol.*, 2002, **149**, (5), pp. 222-226

[16] HARRISON D.J.: 'Characterization of cylindrical eddy-current probes in terms of their spatial frequency spectra'. *IEE Proc. Sci. Meas. Technol.*, 2001, **148**, (4), pp. 183-186

[17] ZHU Z.Q., NG K., SCHOFIELD N., HOWE D.: 'Improved analytical modeling rotor eddy current loss in brushless machines equipped with surface-mounted permanent magnets'. *IEE Proc. Electr. Power Appl.*, 2004, **151**, (6), pp. 641-650

[18] SAGEDHI S.H.H, SALEMI A.H.: 'Electromagnetic field distributions around conductive slabs, produced by eddy-current probes with arbitrary shape current-carrying excitation loops'. *IEE Proc. Sci. Meas. Technol.*, 2001, **148**, (4), pp. 187-192

[19] SPAŁEK D.: 'Spherical induction motor with anisotropic rotor - analytical solutions for electromagnetic field distribution, electromagnetic torques and power losses'. *International Compumag Society*. Testing Electromagnetic Analysis Methods (T.E.A.M.) - problem no. 34. http://www.compumag.co.uk/team.html (2007)

Dariusz Spałek
Electrical Engineering Faculty,
Silesian University of Technology, Poland
ul. Akademicka 10, 44-100 Gliwice
Dariusz.Spalek@polsl.pl
http://www.elektr.polsl.pl/dspalek/

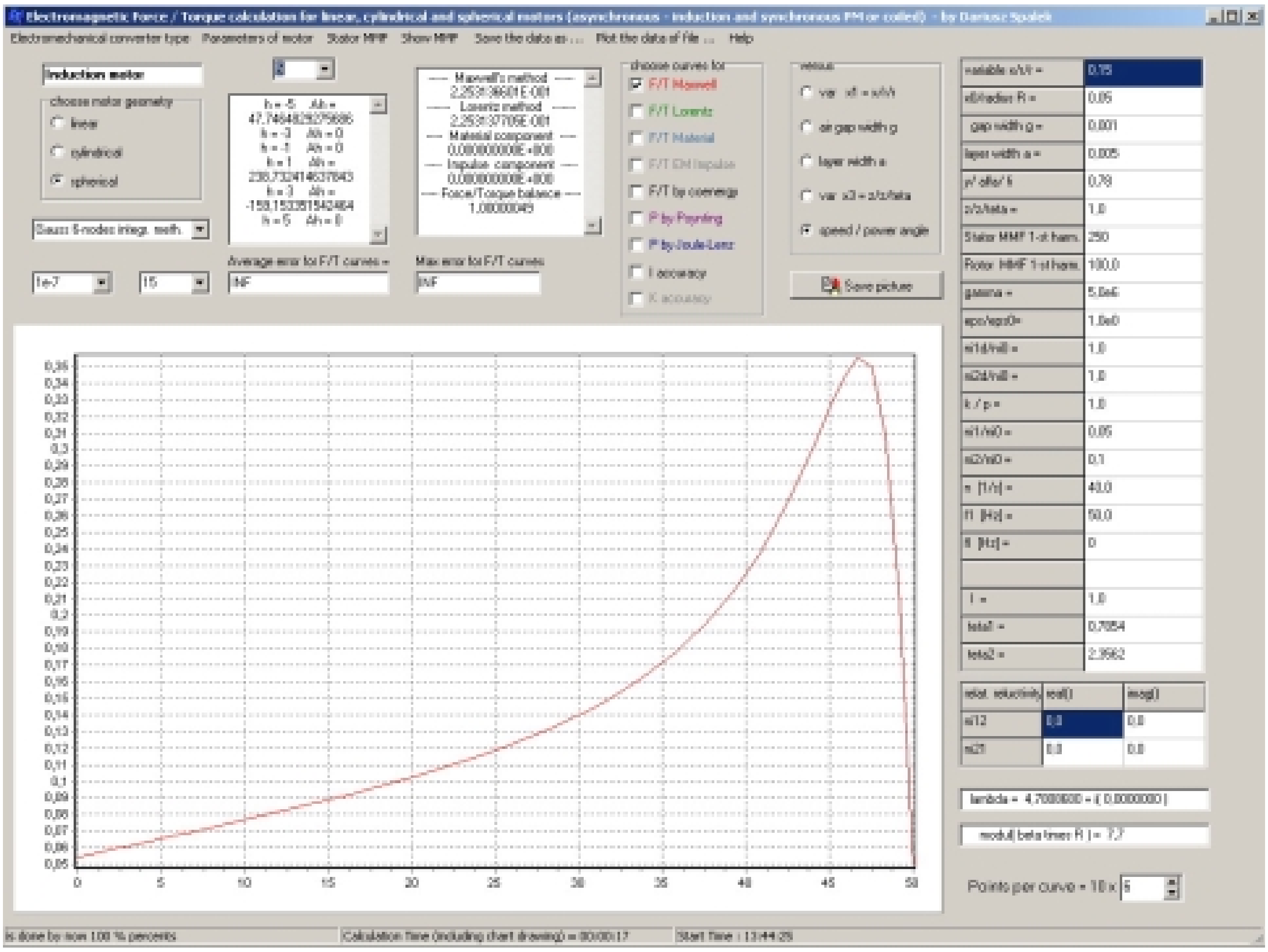

Fig. 7: The program C++ for linear, cylindrical and spherical motor analysis

APPENDIX

Extract of C++ program code - class for linear, cylindrical and spherical motor functions definitions

```
template <typename typ, typename typC, typename typFun, typename typFunC>
class SilnikAsynchroniczny:  public KlasaSilnik< typ, typC, typ_clFunkcja, typ_clFunkcjaC>
{private:
 public:
  typC f, df , dg , Ia, s, Sn, h, A, b, lambda1, lambda2; typ kd, kappa1, kappa2, Ra;
```

```
typC S(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ st_1; s._M_re=fI; s._M_im=2.0*M_PI*fr; h=-(ni12+ni21)*k*i/ni2/2.0; kd=k*sqrt(ni1d/ni2d);
  if (Motor==linear) { f=(0,0); dg=0.0; st_1=1.0; R=a; Ra=R-a; b=i1; A=((k*k*ni1+gamma*s+eps*s*s)/ni2); lambda1=h-sqrt(h*h+A); lambda2=h+sqrt(h*h+A); };
  if (Motor==cylin ) { f=h; dg=1.0; st_1=1.0; Ra=R-a; b=sqrt(s*(gamma+eps*s)/ni2); lambda1=sqrt((h*h)+(k*k*ni1)/ni2); lambda2=lambda1; };
  if (Motor==spher ) { h=h/sin(teta); f=h-0.5; dg=1.0; st_1=1.0/sin(teta); Ra=R-a; b=sqrt(s*(gamma+eps*s)/ni2); lambda1=sqrt((f*f)+(ni1*k*k/sin(teta)/sin(teta)-k*i*ni12/sin(teta))/ni2); lambda2=lambda1;
        kappa1=-0.5+sqrt( 0.25+(ni1d/ni2d)*(k*k/sin(teta)/sin(teta)) ); kappa2=-1.0-kappa1; };
     Form1->Edit2->Text=" lambda = "+FloatToStrF(real(lambda1),ffFixed,7,7)+" + i( "+FloatToStrF(imag(lambda1),ffFixed,7,7)+ " )";
     Form1->Edit3->Text=" modul( beta times R ) = "+FloatToStrF(abs(b*R),ffFixed,7,1);
 if (Motor==linear){ return (ni2*dF1(f+dg,lambda1,b*Ra)+ni21*k*i*F1(f,lambda1,b*Ra))/(ni2*dF2(f+dg,lambda2,b*Ra)+ni21*k*i*F2(f,lambda2,b*Ra)); };
 if (Motor==cylin) { return (ni2*b*Ra*dF1(f,lambda1,b*Ra)+ni21*k*i*F1(f,lambda1,b*Ra))/(ni2*b*Ra*dF2(f,lambda2,b*Ra)+ni21*k*i*F2(f,lambda2,b*Ra)); };
 if (Motor==spher) { return (ni2*dF1(f+dg,lambda1,b*Ra)+ni21*k*i*(F1(f,lambda1,b*Ra)*st_1))/( ni2*dF2(f+dg,lambda2,b*Ra)+ ni21*k*i*(F2(f,lambda2,b*Ra)*st_1) ); }; };// S
 typC Q(typ k, typ x, typ g, typ a, typ teta, typ fr) { Sn=S(k,x,g,a,teta,fr); return F1(f,lambda1,b*R)-S(k,x,g,a,teta,fr)*F2(f,lambda2,b*R); }; // Q
 typC P(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ r; Sn=S(k,x,g,a,teta,fr);
     if (Motor==linear) {r=1.0;}; if (Motor==cylin ) {r=R;};
  if (Motor!=spher) { return (ni2/ni2d/kd)*b*r*(dF1(f,lambda1,b*R)-Sn*dF2(f,lambda2,b*R))+(ni21*i*k/ni2d/kd)*Q(k,x,g,a,teta,fr); };
  if (Motor==spher) { return (ni2/ni2d)* (dF1(f+1.0,lambda1,b*R)-Sn*dF2(f+1.0,lambda2,b*R))+(ni21/ni2d)*k*i*(Q(k,x,g,a,teta,fr)/sin(teta)); }; }; //P
 typC U(typ k, typ x, typ g, typ a, typ teta, typ fr) { Sn=S(k,x,g,a,teta,fr);
  if (Motor!=spher) { return 0.5*(P(k,x,g,a,teta,fr)+Q(k,x,g,a,teta,fr))*G(-kd,R); };
  if (Motor==spher) { return ( Q(k,x,g,a,teta,fr)*(kappa2+1.0)-P(k,x,g,a,teta,fr) )*G(-kappa1,R)/(kappa2-kappa1); }; }; // U
 typC W(typ k, typ x, typ g, typ a, typ teta, typ fr) { Sn=S(k,x,g,a,teta,fr);
  if (Motor!=spher) { return 0.5*(-P(k,x,g,a,teta,fr)+Q(k,x,g,a,teta,fr))*G(+kd,R); };
  if (Motor==spher) { return ( P(k,x,g,a,teta,fr)-Q(k,x,g,a,teta,fr)*(kappa1+1.0))*G(-kappa2,R)/(kappa2-kappa1); }; }; // W
 typC aa1(typ k, typ x, typ g, typ a, typ teta, typ fr) { Sn=S(k,x,g,a,teta,fr);
  if (Motor!=spher) { return k/ni2d/kd/(U(k,x,g,a,teta,fr)*G(kd,R+g)-W(k,x,g,a,teta,fr)*G(-kd,R+g)); };
  if (Motor==spher) { return k/ni2d/sin(teta)/(U(k,x,g,a,teta,fr)*(1.0+kappa1)*G(kappa1,R+g)+(1.0+kappa2)*W(k,x,g,a,teta,fr)*G(kappa2,R+g)); }; }; // aa1
 typC Aa(typ k, typ x, typ g, typ a, typ teta, typ fr) { Sn=S(k,x,g,a,teta,fr);
     return aa1(k,x,g,a,teta,fr)*(F1(f,lambda1,b*x)-Sn*F2(f,lambda2,b*x)); }; // Aa
 typ Aa2(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ CM; Sn=S(k,x,g,a,teta,fr);
  if (Motor==linear) {CM=l*M_PI;}; if (Motor==cylin ) { CM=l*k*x*M_PI;}; if (Motor==spher ) {CM=k*x*x*M_PI*sin(teta);}; return CM*abs(Aa(k,x,g,a,teta,fr))*abs(Aa(k,x,g,a,teta,fr));}; // Aa2
 typ INTEGRAL_Aa2(typ k, typ x, typ g, typ a, typ teta, typ fr) { return INTEGRAL(Aa2,k,x,g,a,teta,fr,1,R-a,R); }; // INTEGRAL_Aa2

 typ FL(typ k, typ x, typ g, typ a, typ teta, typ n) { typ F=0.0, fr, h;
  for (int c=(Cmax-C)/2; c<=(Cmax+C)/2; c++) { h=2*c-Cmax; if (h==0) {fr=0;} else {fr=-n+fsyn/h;}; if ( (fabs(fr/fsyn) >= frmin) && (h!=3*floor(h/3.0)) && (TetaS[c]!=0) ) { /*Lorentz*/ Sn=S(k*h,x,g,a,teta,fr);
  if (Motor!=spher) { F=F+imag(s)*gamma*TetaS[c]*TetaS[c]*INTEGRAL_Aa2(k*h,x,g,a,teta,fr); };
  if (Motor==spher) { F=F+imag(s)*gamma*TetaS[c]*TetaS[c]*INTEGRAL(INTEGRAL_Aa2,k*h,x,g,a,teta,fr,0,teta1,teta2); }; /*Lor*/ };
            }; return F; };
               /*Poynting*/
 typ FP(typ k, typ x, typ g, typ a, typ teta, typ n) { typ F=0.0, fr, h;
  for (int c=(Cmax-C)/2; c<=(Cmax+C)/2; c++) { h=2*c-Cmax; if (h==0) {fr=0;} else {fr=-n+fsyn/h;}; if ( (fabs(fr/fsyn) >= frmin) && (h!=3*floor(h/3.0)) && (TetaS[c]!=0) ) { /*Lorentz*/ Sn=S(k*h,x,g,a,teta,fr);
  if (Motor!=spher) { F+=2.0*imag(s)*real(s)*eps*TetaS[c]*TetaS[c]*INTEGRAL_Aa2(k*h,x,g,a,teta,fr); };
```

```
    if (Motor==spher) {
F+=2.0*imag(s)*real(s)*eps*TetaS[c]*TetaS[c]*INTEGRAL(INTEGRAL_Aa2,k*h,x,g,a,teta,fr,0,teta1,teta2); };
/*Lor*/ };
                    };    return F;   };   /*Poynting*/
  typ   P_Integr(typ k, typ x, typ g, typ a, typ teta, typ n) { typ F=0.0, h, fr;
//    for (int c=(Cmax-C)/2; c<=(Cmax+C)/2; c++)  { if (2*c-Cmax==0) {fr=0;} else {fr=-n+fsyn/(2.0*c-Cmax);}; if (
(fabs(fr/fsyn) >= frmin) && (2.0*c!=Cmax) && (TetaS[c]!=0) )  { /*Lorentz*/ Sn=S(k,x,g,a,teta,fr);
//  dla 1-szej harmonicznej tylko ponizsza linia
    for (int c=(Cmax-1)/2; c<=(Cmax+1)/2; c++)   { h=2*c-Cmax; if (h==0) {fr=0;} else {fr=-n+fsyn/h;};  if (
(fabs(fr/fsyn) >= frmin) && (h!=3*floor(h/3.0)) && (TetaS[c]!=0) )  { /*Lorentz*/ Sn=S(k*h,x,g,a,teta,fr);
    if (Motor!=spher)  {  F+=abs(s*s)*gamma*TetaS[c]*TetaS[c]*INTEGRAL_Aa2(k*h,x,g,a,teta,fr)/(k*h); };
    if (Motor==spher) {
F+=abs(s*s)*gamma*TetaS[c]*TetaS[c]*INTEGRAL(INTEGRAL_Aa2,k*h,x,g,a,teta,fr,0,teta1,teta2)/(k*h);  };    /*
Power losses by Joule integral*/ };
                    };    return F;   };
  typC Ad(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ L2; Sn=S(k,x,g,a,teta,fr);   // aby pobrane zostaly dane
       if (Motor==linear) {L2=1.0;};  if (Motor==cylin ) {L2=x;};  if (Motor==spher ) {L2=x*sin(teta);};
    if (Motor!=spher)  { return  aa1(k,x,g,a,teta,fr)*(U(k,x,g,a,teta,fr)*G(+kd,x)+W(k,x,g,a,teta,fr)*G(-kd,x));  };
    if (Motor==spher)  { return  aa1(k,x,g,a,teta,fr)*(U(k,x,g,a,teta,fr)*G(kappa1,x)+W(k,x,g,a,teta,fr)*G(kappa2,x));
}; };//Ad
  typC B1d(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ L2; Sn=S(k,x,g,a,teta,fr);   // aby pobrane zostaly dane
       if (Motor==linear) {L2=1.0;};  if (Motor==cylin ) {L2=x;};  if (Motor==spher ) {L2=x*sin(teta);};
    if (Motor!=spher)  { return  -(i*k/L2)*aa1(k,x,g,a,teta,fr)*(U(k,x,g,a,teta,fr)*G(+kd,x)   +W(k,x,g,a,teta,fr)*G(-
kd,x));  };
    if (Motor==spher) { return -
(i*k/L2)*aa1(k,x,g,a,teta,fr)*(U(k,x,g,a,teta,fr)*G(kappa1,x)+W(k,x,g,a,teta,fr)*G(kappa2,x)); }; };//B1d
  typC H2d(typ k, typ x, typ g, typ a, typ teta, typ fr) { Sn=S(k,x,g,a,teta,fr);    // aby pobrane zostaly dane
    if (Motor!=spher)  { return  -  ni2d  *aa1(k,x,g,a,teta,fr)*(U(k,x,g,a,teta,fr)*dG(+kd,x)+W(k,x,g,a,teta,fr)*dG(-
kd,x));  };
    if (Motor==spher) { return -
(ni2d/x)*aa1(k,x,g,a,teta,fr)*(U(k,x,g,a,teta,fr)*dG(kappa1+1.0,x)+W(k,x,g,a,teta,fr)*dG(kappa2+1.0,x));  };  };//H2d
  typ  c_real_H2dconjB1d(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ Cx; Sn=S(k,x,g,a,teta,fr);   // aby pobrane
zostaly dane
    if  (Motor==linear)  { Cx=l*M_PI/k; };   if (Motor==cylin )  {  Cx=l*M_PI*x*x;};   if (Motor==spher )
{Cx=M_PI*x*x*x*x*sin(teta)*sin(teta);};
               return  Cx*real(H2d(k,x,g,a,teta,fr)*conj(B1d(k,x,g,a,teta,fr)));  }; //c_real_H2dconjB1d
  typ  FM (typ k, typ x, typ g, typ a, typ teta, typ n)  { typ F=0.0, h, fr;  // Cmax - zawsze liczba nieparzysta
  for (int c=(Cmax-C)/2; c<=(Cmax+C)/2; c++)
   { h=2*c-Cmax; if (h==0) {fr=0;} else {fr=-n+fsyn/h;}; if ( (fabs(fr/fsyn) >= frmin) && (h!=3*floor(h/3.0)) &&
(TetaS[c]!=0) )

   { /*Maxwell*/ Sn=S(k*h,x,g,a,teta,fr);
    if (Motor!=spher)  {  F+=TetaS[c]*TetaS[c]*c_real_H2dconjB1d(k*h,x,g,a,teta,fr); };
    if  (Motor==spher)   {  F+=TetaS[c]*TetaS[c]*INTEGRAL(c_real_H2dconjB1d,k*h,x,g,a,teta,fr,0,teta1,teta2); };
/*Maxwell*/  };
                         }; return F; };
  typ  c_real_sAdconjH2d(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ Cx; Sn=S(k,x,g,a,teta,fr);   // aby pobrane
zostaly dane
    if   (Motor==linear)  {  Cx=l*M_PI/k;  };   if  (Motor==cylin )  {  Cx=l*M_PI*x;};   if  (Motor==spher )
{Cx=M_PI*x*x*x*sin(teta);};
                return  -Cx*real(s*Ad(k,x,g,a,teta,fr)*conj(H2d(k,x,g,a,teta,fr)));  }; //c_real_sAdconjH2d
  typ  P_S (typ k, typ x, typ g, typ a, typ teta, typ n)       {typ F=0.0, h, fr;  // Cmax - zawsze liczba nieparzysta
  for (int c=(Cmax-C)/2; c<=(Cmax+C)/2; c++)  {  h=2*c-Cmax;  if (h==0) {fr=0;} else {fr=-n+fsyn/h;};   if (
(fabs(fr/fsyn) >= frmin) && (h!=3*floor(h/3.0)) && (TetaS[c]!=0) )  { /*P_Poynting*/ Sn=S(k*h,x,g,a,teta,fr);
    if (Motor!=spher)  {  F+=TetaS[c]*TetaS[c]*c_real_sAdconjH2d(k*h,x,g,a,teta,fr); };
    if  (Motor==spher)   {  F+=TetaS[c]*TetaS[c]*INTEGRAL(c_real_sAdconjH2d,k*h,x,g,a,teta,fr,0,teta1,teta2); };
/* Power losses by Poynting */  };
                          }; return F; };
  typC B1a(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ L2; Sn=S(k,x,g,a,teta,fr);
   if (Motor==linear) {L2=1.0;};  if (Motor==cylin) { L2=x;};  if (Motor==spher) {L2=x*sin(teta);};
        return -i*(k/L2)*aa1(k,x,g,a,teta,fr)*(F1(f,lambda1,b*x)-S(k,x,g,a,teta,fr)*F2(f,lambda2,b*x)); };  // B1a
  typC B2a(typ k, typ x, typ g, typ a, typ teta, typ fr) { typC cc; typ df; Sn=S(k,x,g,a,teta,fr);
   if (Motor==linear) {cc=1.0;};  if (Motor==cylin) { cc=b; df=0.0;};   if (Motor==spher) {cc=1.0/x; df=1.0;};
```

```
        return -cc*aa1(k,x,g,a,teta,fr)*(dF1(f+df,lambda1,b*x)-S(k,x,g,a,teta,fr)*dF2(f+df,lambda2,b*x)); };  // B2a
  typ  B12a(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ CM; Sn=S(k,x,g,a,teta,fr);
    if  (Motor==linear)  {CM=l*M_PI;};   if  (Motor==cylin)  {  CM=l*k*x*M_PI;};   if  (Motor==spher)
{CM=k*x*x*M_PI*sin(teta);};
        return  imag((ni21-conj(ni12))*CM*B1a(k,x,g,a,teta,fr)*conj(B2a(k,x,g,a,teta,fr)));};   // B12a
  typ INTEGRAL_B12a(typ k, typ x, typ g, typ a, typ teta, typ fr) { return INTEGRAL(B12a,k,x,g,a,teta,fr,1,R-a,R); };
// INTEGRAL_B12a

  typ   FFe(typ k, typ x, typ g, typ a, typ teta, typ n)      { typ F=0.0, h, fr; typC dniH=ni21-conj(ni12);
  for  (int  c=(Cmax-C)/2;  c<=(Cmax+C)/2;  c++)   {  h=2*c-Cmax;  if  (h==0)  {fr=0;}  else  {fr=-n+fsyn/h;};  if  (
(fabs(fr/fsyn)  >=  frmin)  &&  (h!=3*floor(h/3.0))  &&  (abs(dniH)!=0)  &&  (TetaS[c]!=0)  )   {  /*Material*/
Sn=S(k*h,x,g,a,teta,fr);
    if  (Motor!=spher)  {  F=F+TetaS[c]*TetaS[c]*INTEGRAL_B12a(k*h,x,g,a,teta,fr); };
    if  (Motor==spher)   {  F=F+TetaS[c]*TetaS[c]*INTEGRAL(INTEGRAL_B12a,k*h,x,g,a,teta,fr,0,teta1,teta2); };
/*Material*/ };
                               }; return F; };
  typ jA_HB(typ k, typ x, typ g, typ a, typ teta, typ fr) { typ Cx; Sn=S(k,x,g,a,teta,fr);
     if   (Motor==linear)   {Cx=l*M_PI;};    if   (Motor==cylin)   {   Cx=l*k*x*M_PI;};    if   (Motor==spher)
{Cx=k*x*x*M_PI*sin(teta);};
       return  Cx*real( gamma*s*Aa(k,x,g,a,teta,fr)*conj(i*Aa(k,x,g,a,teta,fr))
       +i*conj(ni1*B1a(k,x,g,a,teta,fr))*B1a(k,x,g,a,teta,fr)+i*conj(ni12*B2a(k,x,g,a,teta,fr))*B1a(k,x,g,a,teta,fr)
       +i*conj(ni21*B1a(k,x,g,a,teta,fr))*B2a(k,x,g,a,teta,fr)+i*conj(ni2*B2a(k,x,g,a,teta,fr))*B2a(k,x,g,a,teta,fr)      );
};
  typ INTEGRAL_jA_HB(typ k, typ x, typ g, typ a, typ teta, typ fr) { return INTEGRAL(jA_HB,k,x,g,a,teta,fr,1,R-
a,R); };

  typ   Fcoenergy(typ k, typ x, typ g, typ a, typ teta, typ n)   { typ F=0.0, h, fr;
  for  (int  c=(Cmax-C)/2;  c<=(Cmax+C)/2;  c++)   {  h=2*c-Cmax;  if  (h==0)  {fr=0;}  else  {fr=-n+fsyn/h;};  if  (
(fabs(fr/fsyn) >= frmin) &&   (h!=3*floor(h/3.0)) && (TetaS[c]!=0) )  { /*Material*/ Sn=S(k*h,x,g,a,teta,fr);
    if  (Motor!=spher)  {  F+=TetaS[c]*TetaS[c]*INTEGRAL(jA_HB,k*h,x,g,a,teta,fr,1,R-a,R); };
    if  (Motor==spher)   {  F+=TetaS[c]*TetaS[c]*INTEGRAL(INTEGRAL_jA_HB,k*h,x,g,a,teta,fr,0,teta1,teta2); };
/*Coenergy*/ };
                                 }; return F; };
  typ   I_Acc(typ k, typ x, typ g, typ a, typ teta, typ n)   { typC F;  typ fr=fsyn-n;
    if (fabs(fr/fsyn) >= frmin) { Sn=S(k,x,g,a,teta,fr);
    if  (Motor==linear)  {  F=( lambda1*dF1(f,lambda1,x)+(-2.0*h)*dF1(f,lambda1,x))/( A*F1(f,lambda1,x) );  };
    if      (Motor==cylin)       {      F=(     b*b*d2I(f,lambda1,b*x)+(1.0-2.0*h)*b*dI(f,lambda1,b*x)/x      )/(
(b*b+k*k*ni1/ni2/x/x)*I(f,lambda1,b*x) );  };
    if     (Motor==spher)       {     F=(   b*b*d2I(f,lambda1,b*x)+2.0*(1.0-h)*b*dI(f,lambda1,b*x)/x    )/(    (b*b+(
ni1*k*k/sin(teta)/sin(teta)-k*i*ni12/sin(teta))/ni2/x/x )*I(f,lambda1,b*x) );  };
                        };   return abs(F);    };
  typ   K_Acc(typ k, typ x, typ g, typ a, typ teta, typ n)   { typC F; typ fr=fsyn-n;
    if (fabs(fr/fsyn) >= frmin) { Sn=S(k,x,g,a,teta,fr);
    if  (Motor==linear)  {  F=( lambda2*dF2(f,lambda2,x)+(-2.0*h)*dF2(f,lambda2,x))/( A*F2(f,lambda2,x) );  };
    if      (Motor==cylin)       {      F=(     b*b*d2K(f,lambda2,b*x)+(1.0-2.0*h)*b*dK(f,lambda2,b*x)/x      )/(
(b*b+k*k*ni1/ni2/x/x)*K(f,lambda2,b*x) ) ;  };
    if     (Motor==spher)       {     F=(   b*b*d2K(f,lambda1,b*x)+2.0*(1.0-h)*b*dK(f,lambda1,b*x)/x    )/(    (b*b+(
ni1*k*k/sin(teta)/sin(teta)-k*i*ni12/sin(teta))/ni2/x/x )*K(f,lambda1,b*x) );;  };
                        };   return abs(F);    };

   SilnikAsynchroniczny(void) {};
//   SilnikAsynchroniczny(typ x,typFun F);
  ~SilnikAsynchroniczny() {};
 };
```

Dariusz Spałek
Electrical Engineering Faculty,
Silesian University of Technology, Poland
ul. Akademicka 10, 44-100 Gliwice
Dariusz.Spalek@polsl.pl
http://www.elektr.polsl.pl/dspalek/