# THEORY OF THE PASSIVE COMPENSATION OF A THREE-PHASE NONLINEAR LOAD

*Daniel MAYER, **Petr KROPÍK
*Faculty of Electrical Engineering, University of West Bohemia Pilsen,
Sady Pětatřicátníků 14, 30614 Plzeň, Czech Republic, tel.: 377634633, E-mail: mayer@kte.zcu.cz
** Faculty of Electrical Engineering, University of West Bohemia Pilsen,
Sady Pětatřicátníků 14, 30614 Plzeň, Czech Republic, tel.: 377634639, E-mail: pkropik@kte.zcu.cz

**SUMMARY**

*An algorithm is described for defining passive parameters of two-poles RLC, with whose help it is possible to compensate a three-phase load. The method is valid for sinusoidal or nonsinusoidal, balanced or unbalanced three-phase power system with linear or nonlinear load. The problem is formed as an optimisation problem for minimizing losses in line. Given calculation method is illustrated by three numerical problems. It is possible to modify this method even for solution of other power engineering problems, e.g. for design of filters enabling increased quality of transmitted electric energy by suppressing unwanted higher harmonics in network.*

**Keywords:** *losses in three-phase line, MATLAB, objective function, optimisation*

## 1. INTRODUCTION

Let us describe the method for calculation of parameters of two-poles RLC for non-linear load of inductive character. Compensation two-pole is formed by series connection of static condenser with reactance coil that limits switching impulses of current as well as higher harmonics current going through compensation condenser. For non-sinusoidal currents the notion of reactive power is ambiguous (see e.g. [3]), so it will not be used, the focus will be on active power and calculation will be formed as optimization problem. The objective function will be losses in line and we will define such parameters *R, L, C* of compensation two-poles that minimize these losses. In comparison with our earlier work [1] we will not use the analytical solution and the calculation will be done numerically using standard program set *Optimization Toolbox*, which is part of a computation system *MATLAB*.

## 2. DEFINING THE SOLVED PROBLEM

We will deal with the following system configuration: three-phase non-linear load of inductive character is connected to balanced three-phase network, whose voltage are sinusoidal functions with period $T$, Fig. 1. The load draws currents $i_1(t)$, $i_2(t)$, $i_3(t)$ that are periodical, generally unbalanced and nonsinusoidal. To the load terminals shunt compensators are attached that contain two-poles RLC. The inductance of reactance coils is chosen so that resonance frequency $f_r$ of the two-poles is distanced from the frequency of higher harmonics generated by the nonlinear load, usually $f_r$ = 189 Hz or $f_r$ = 134 Hz.

The network is connected with the load through line with currents $i_{l1}(t)$, $i_{l2}(t)$, $i_{l3}(t)$. As for parameters *(R,L)* let us consider that the influence of voltage drop in line on the terminal voltage of load can be neglected so that network-voltage rigidity can be considered sufficient. Let us suppose that time course of voltage on load terminal is known. We define parameters *R, L, C of* compensation two-poles, for which the losses in line are minimal. Or, from mathematical point of view, we minimize the functional, which is objective function

$$F = \frac{1}{T}\int_0^T (i_{l1}^2 + i_{l2}^2 + i_{l3}^2)\,\mathrm{d}t \qquad (1)$$

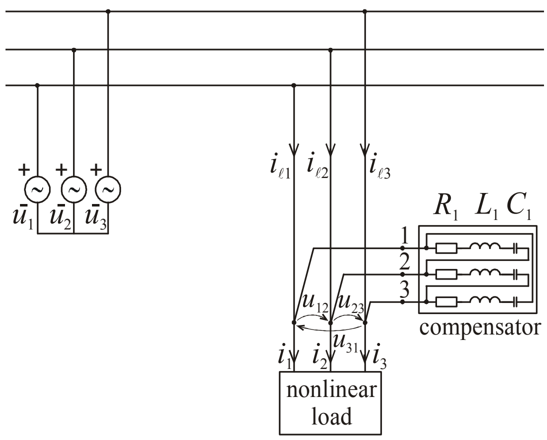


**Fig. 1** Three-phase circuit structure

## 3. DEFINING OF OPTIMIZATION PROBLEM

### 3.1. Algorithm of calculation of compensation of two-poles parameters

Instantaneous values of phase voltages and line voltages of a balanced network are

$$u_1 = U \sin \omega t$$
$$u_2 = U \sin(\omega t - 2\pi / 3)$$
$$u_2 = U \sin(\omega t + 2\pi / 3) \qquad (2)$$

and

$$u_{12} = \sqrt{3}U\sin(\omega t + \pi/6)$$
$$u_{23} = \sqrt{3}U\sin(\omega t - \pi/2) \qquad (3)$$
$$u_{12} = \sqrt{3}U\sin(\omega t + 5\pi/6)$$

Instantaneous values of currents consumed by the load are supposed to be given.

Currents in wye-connected compensation two-poles $R_i L_i C_i$ ($i$ = 1,2,3) are

$$\begin{aligned} i_{12} &= I_{12}\sin(\omega t + \pi/6 - \psi_1) \\ i_{23} &= I_{23}\sin(\omega t - \pi/2 - \psi_2) \\ i_{31} &= I_{31}\sin(\omega t + 5\pi/6 - \psi_3) \end{aligned} \qquad (4)$$

where

$$I_{ij} = \sqrt{3}U\left[R_i + (\omega L_i - 1/\omega C_i\right]^{-\frac{1}{2}} \qquad i,j = 1,2,3;\ i \neq j \qquad (5)$$

The reactance coil has inductance $L_i$, which is defined so that the two-pole has the chosen resonance frequency $f_r$, thus

$$L_i = \frac{1}{\omega_0^2 C_i}, \qquad \text{where} \quad \omega_0 = 2\pi f_r \qquad (6)$$

Let its resistance be k-multiple of inductive reactance, thus

$$R_i = k\,\omega L_i = \frac{k\,\omega}{\omega_0^2 C_i} \qquad (7)$$

Then phase angle $\psi_1 = \psi_2 = \psi_3 = \psi \in \langle 0, \pi/2 \rangle$, when

$$\tan\psi = \frac{1}{R_i}\left(\omega L_i - \frac{1}{\omega C_i}\right) = \frac{1}{k}\left(1 - \frac{\omega_0^2}{\omega^2}\right) \qquad (8)$$

it is possible to express equation (5) using equations (6) and (7) in the form

$$I_{ij} = \frac{C_i U_{ij}}{A} \qquad (9)$$

where

$$A^2 = \frac{\omega^2}{k^2\omega_0^4} + \left(\frac{\omega}{\omega_0^2} - \frac{1}{\omega}\right)^2 \qquad (10)$$

instantaneous currents in line are calculated from equations

$$\begin{aligned} i_{l1} &= i_1 + i_{12} - i_{31} \\ i_{l2} &= i_2 + i_{23} - i_{12} \\ i_{l3} &= i_3 + i_{31} - i_{23} \end{aligned} \qquad (11)$$

These currents are substituted to eq.(1), and so the optimization problem is formulated. The solutions are the parameters of compensation two-poles.

### 3.2. Special cases

If the load is linear, unbalanced and of inductance character, it draws currents

$$\begin{aligned} i_1 &= I_1\sin(\omega t - \varphi_1) \\ i_2 &= I_2\sin(\omega t - \varphi_2 - 2\pi/3) \\ i_3 &= I_3\sin(\omega t - \varphi_3 + 2\pi/3) \qquad & \varphi_1, \varphi_2, \varphi_3 \geq 0 \end{aligned} \qquad (12)$$

For compensation with static condensers only ($L_i = 0$, $R_i = 0$, then $\omega_0 \to \infty$), is $A = 1/\omega$ and $\psi = \pi/2$.

### 3.3. Numerical minimization of the objective function (1)

All above-mentioned formulas were implemented using programming language of computational system MATLAB and MATLAB Optimization Toolbox. At the beginning current amplitudes $I_{12}$, $I_{23}$ and $I_{31}$ had been computed – see equations (5). Constants definitions and auxiliary computations are not given here, as it is mentioned above. In the second step variables $A_1$, $A_2$ a $A_3$ were computed using equations (10)

```
A1=(0.1.^2).*((omg.^2)./(omg0.^4))+(((omg./
(omg0^2))-(1./omg)).^2);
A2=(0.1.^2).*((omg.^2)./(omg0.^4))+(((omg./
(omg0^2))-(1./omg)).^2);
A3=(0.1.^2).*((omg.^2)./(omg0.^4))+(((omg./
(omg0^2))-(1./omg)).^2);
```

Computation followed with calculating of relevant amplitudes according to equation (5)

```
I12=C1.*(U12./sqrt(A1));
I23=C2.*(U23./sqrt(A2));
I31=C3.*(U31./sqrt(A3));
```

and currents in compensation two-poles using equation (4)

```
i12 = I12.*sin(omg.*t+(pi./6)-psi1);
i23 = I23.*sin(omg.*t-(pi./2)-psi2);
i31 = I31.*sin(omg.*t+(5.*pi./6)-psi3);
```

Before current in the load were calculated, we had computed following auxiliary variables, which represents final angles. We need to calculate these angles to make program code more transparent and we need to know it in the next part of computation.

```
ang_i1 = omg.*t-(pi./3);
ang_i2 = omg.*t+((-52.*(2.*pi./360))-
((2./3).*pi));
ang_i3 = omg.*t+((-
68.*(2.*pi./360))+((2./3).*pi));
```

Calculation of current in the load according to equation (12)

```
i1=I1.*sin(ang_i1);
i2=I2.*sin(ang_i2);
i3=I3.*sin(ang_i3);
```

The part of program code shown above generated a course of currents in case of linear load. In case of non-linear load this course must been adjusted

```
i1 = ~((mod(ang_i1,pi) < angle_4_t) &
(mod(ang_i1,pi) > 0)).*i1;
```

```
i2 = ~((mod(ang_i2,pi) < angle_4_t) &
(mod(ang_i2,pi) > 0)).*i2;
i3 = ~((mod(ang_i3,pi) < angle_4_t) &
(mod(ang_i3,pi) > 0)).*i3;
```

Some parts of the currents $i_1$, $i_2$ and $i_3$ courses had been levelled with the zero by this part of program code, according to value of variable `angle_4_t`. This method produced required course of currents. At the end of computation the courses of currents $i_{l1}$, $i_{l2}$ and $i_{l3}$ were calculated with help of conditions (11)

```
il1=i1+i12-i31;
il2=i2+i23-i12;
il3=i3+i31-i23;
```

Final sum of squares of these currents was computed

```
y=(il1.^2)+(il2.^2)+(il3.^2);
```

The numerical integration was based on equation (1). A standard MATLAB functions `quad` and `quadl` can be used. These functions used recursive adaptive Simpson quadrature algorithm. Function quad(*fun*, *a*, *b*) approximates the integral of function *fun* from *a* to *b* within an error of $10^{-6}$. Function *fun* accepts vector *x* and returns vector *y*. Using form quad(*fun*, *a*, *b*, *tol*) uses an absolute error tolerance *tol* instead of the default ($10^{-6}$). In our calculations it was needed to set this tolerance usually between $10^{-7}$ and $10^{-9}$ to reach an adequate accuracy of integration – see course of integrated function in case of non-linear and unbalanced load (Fig. 5). For this reason we used function quadl instead of quad. The function quadl should be more efficient with high accuracies and smooth integrands.

Finally we used this function in the following form

```
quadl('fun',0,T,1e-8,[],C1,C2,C3) / T;
```

Additional arguments `C1`, `C2` and `C3` were passed directly to function `fun(t,C1,C2,C3)`.

Result of this integration represents our objective function. To solve optimization problem, we applied standard MATLAB functions *fminsearch*, *fminunc* and *fmincon* included in MATLAB Optimization Toolbox.

Function *fminsearch* is generally referred to as unconstrained non-linear optimization. We used it in form

```
options = optimset('fminsearch');
options.TolFun=1e-15;
options.TolX=1e-15;
options.MaxFunEvals=1000;
[min,fval,exitflag,output]=fminsearch(@object
ive_f,input,options);
```

A variable `options` represent set of initial parameters of this function. Useful parameters are

- `Display` – Level of display. `'off'` displays no output; `'iter'` displays output at each iteration; `'final'` displays just the final output; `'notify'` (default) displays output only if the function does not converge.
- `MaxFunEvals` – Maximum number of function evaluations allowed.
- `MaxIter` – Maximum number of iterations allowed.
- `TolFun` – Termination tolerance on the function value.
- `TolX` – Termination tolerance on *x*.

Function *fminsearch* uses algorithm based on the Nelder-Mead simplex direct search method. This is a method that does not use numerical or analytic gradients as in fminunc or fmincon (see below). If *n* is the length of variable, a simplex in *n*-dimensional space is characterized by the *n*+1 distinct vectors that are its vertices. In two-dimensional, a simplex is a triangle; in three-dimensional, it is a pyramid. At each step of the search, a new point in or near the current simplex is generated. The function value at the new point is compared with the function's values at the vertices of the simplex and, usually, one of the vertices is replaced by the new point, giving a new simplex. This step is repeated until the diameter of the simplex is less than the specified tolerance. When the solving problem is highly discontinuous, fminsearch may be more robust than fminunc.

Function *fminunc* is generally referred to as unconstrained non-linear optimization of multivariable function. We used it in form

```
options = optimset('fminunc');
options.TolFun=1e-15;
options.TolX=1e-15;
options.MaxFunEvals=1200;
options.GradObj='on';
[min,fval,exitflg,output,grad,hessian]=fminun
c(@objective_f,input,options);
```

A variable `option` represents set of initial parameters of this function as above. Many parameters are same as parameters of the function fminsearch. We used special parameter GradObj sets `'on'`

- `GradObj` – gradient for the objective function. User in objective function defines it. The gradient must be provided to use the large-scale method. We used it as an optional parameter for the medium-scale method.

Function *fminunc* uses algorithm based on the BFGS (Broyden, Fletcher, Goldfarb, Shanno) Quasi-Newton method with a mixed quadratic and cubic line search procedure (in case of medium-scale optimization). This method uses the BFGS formula for updating the approximation of the Hessian matrix. The DFP (Davidon, Fletcher, Powell) formula, which approximates the inverse Hessian matrix, is selected by setting the `HessUpdate` parameter to `'dfp'` (and the `LargeScale` parameter to `'off'`). In case of Large-Scale Optimization an algorithm subspace trust region method based on the interior-reflective Newton method is used. Each iteration involves the approximate solution of a large linear system using the method of preconditioned conjugate gradients (PCG).

When it is needed to eliminate improper values of variables (e.g. negative values of capacitance) we implement function *fmincon*. This function finds a minimum of a constrained non-linear multivariable function. We used it in form

```
mat_A=[-1,0,0;0,-1,0;0,0,-1];
vec_b=[0;0;0];

options = optimset('fmincon');
options.TolFun=1e-15;
options.TolX=1e-20;
options.TolCon=1e-15;
options.MaxFunEvals=800;
options.GradObj='on';
[min,fval,exitflag,output,lambda_v,grad_v,hes
sian_v]=fmincon(@criteria_f,
input,mat_A,vec_b,[],[],[],[],[],options);
```

Variable `mat_A` represents the matrix **A** of the coefficients of the linear inequality constraints and `vec_b` represents corresponding right side vector **b** (i.e. **Ax** <= **b**).

Function *fmincon* uses algorithm based on the Sequential Quadratic Programming (SQP) method (in case of medium-scale optimization). Quadratic Programming (QP) subproblem is solved at each iteration. An estimate of the Hessian of the Lagrangian is updated at each iteration using the BFGS formula (see fminunc above). A line search is performed using a merit function. The QP subproblem is solved using an active set strategy.

## 4. NUMERICAL PROBLEMS

### 4.1. Linear balanced load

For comparison of the results obtained in numerical calculation with analytical solution we use the following simple problem. Line voltages of balanced network are

$$\sqrt{3}U = 380\,\mathrm{V},\ \omega = 2\pi f = 100\pi .$$

The load is linear (i. e. it draws harmonic currents) and symmetrical. Drawn currents are expressed in equation (12), where

$$I_1 = I_2 = I_3 = I = 10\mathrm{A},$$
$$\varphi_1 = \varphi_2 = \varphi_3 = \varphi = 60^\circ .$$

Compensation is done using static condenser in wye connection. Currents $i_{12}, i_{23}, i_{31}$ acc. Eq. (4), where acc. Eq. (9)

$$I_{ij} = 3{,}8.10^4 \pi C_i \qquad (13)$$

Substituting for $i_{11}, i_{12}, i_{13}$ in eq. (1) and solving optimization task we get

$$C_1 = C_2 = C_3 = 4{,}188.10^{-5}\,\mathrm{F},$$
$$\text{for} \quad F_{\min} = 37{,}5. \qquad (14)$$

For judging the environment of the found minimum of functional *F* according eq. (1) there is shown in Fig. 2 function

$$F = F(C_1, C_2) \text{ při } C_3 = 4{,}188.10^{-5}\,\mathrm{F}$$

in 3D representation. Optimization was done using three above-mentioned methods and the same results were achieved with the difference that function fmincon and fminunc showed higher accuracy of the result, but only in higher orders, which does not have any practical meaning.

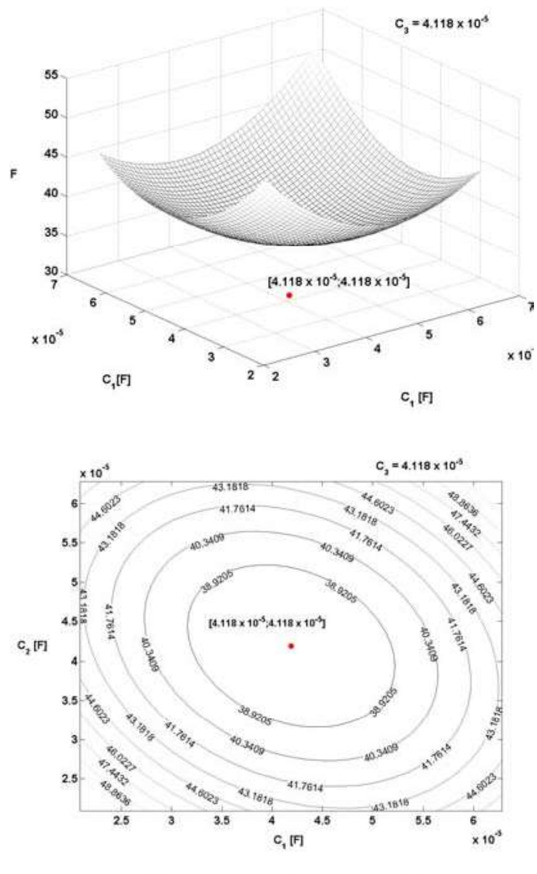


**Fig. 2** Objective function for problem 4.1

It is possible to solve this symmetrical problem also analytically, (see e.g. [2]). Obviously $C_1 = C_2 = C_3 = C$, where

$$C = \frac{I \sin\varphi}{3\omega U} = 4{,}188.10^{-5}\ \mathrm{F} \qquad (15)$$

### 4.2. Linear unbalanced load

Network is the same as in the previous example:

$\sqrt{3}U = 380\,\mathrm{V}$, $\omega = 100\pi$.

The load is linear, unbalanced; drawn currents are expressed by equations (12), where

$$I_1 = 10\,\mathrm{A}, \quad I_2 = 8\,\mathrm{A}, \quad I_3 = 12\,\mathrm{A}$$
$$\varphi_1 = 60^\circ, \quad \varphi_2 = 52^\circ, \quad \varphi_3 = 68^\circ \tag{16}$$

Compensation is done using two-poles $R_i\,L_i\,C_i$ $(i = 1,2,3)$ for the following values:

$$f_\mathrm{r} = 189\,\mathrm{Hz}, \; \omega_0 = 2\pi 189\,\mathrm{s}^{-1}, \; k = 0{,}1. \tag{17}$$

Through minimization of the functional (1) we obtain:

$C_1 = 2{,}826.10^{-5}$ F,

$C_2 = 4{,}015.10^{-5}$ F,

$C_3 = 4{,}846.10^{-5}$ F

According eq. (6) is

$L_1 = 0{,}1577$ H,

$L_2 = 0{,}1110$ H,

$L_3 = 0{,}0919$ H

And according eq. (7) is

$R_1 = 4{,}954\ \Omega$,

$R_2 = 3{,}486\ \Omega$,

$R_3 = 2{,}8889\ \Omega$

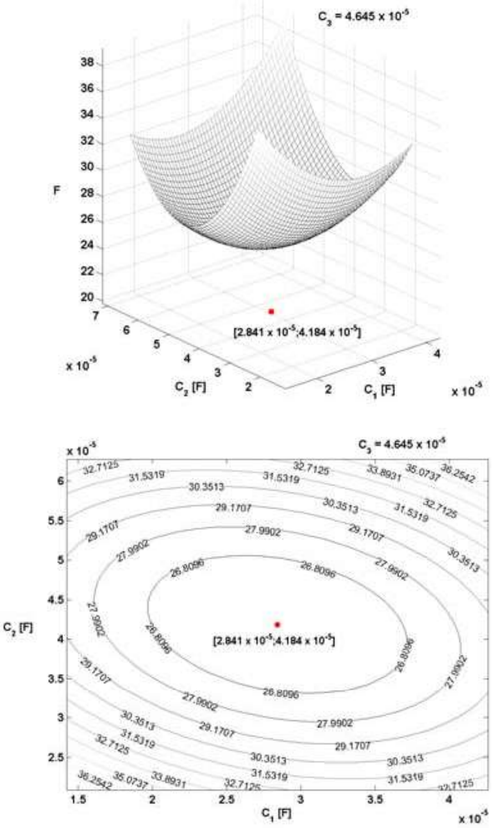


**Fig. 3** Objective function for problem 4.2

This case was solved again using all three methods. The best effect was achieved using function fminsearch. In case of function fminunc and fmincon the calculation got much longer and taking into consideration the character of the objective function course some numerical instabilities occurred as well. In Fig. 3 there is 3D representation of the objective function.

### 4.3. Nonlinear unbalanced load

The network is the same as in the previous problems. Instantaneous values of currents drawn by the load are (Fig. 4)

$$i_i = \begin{cases} 0 & \text{for } 0 < t < \alpha \\ I_i \sin(\omega t - \psi_i) & \text{for } \alpha < t < 2\pi,\ i = 1, 2, 3 \end{cases}$$

where

$$\psi_1 = -\varphi_1, \quad \psi_2 = -\varphi_2 - 2\pi/3, \quad \psi_3 = -\varphi_3 + 2\pi/3$$

Calculation is done for $\alpha = 45^\circ$ and for values $I_1$, $I_2$, $I_3$, $\varphi_1$, $\varphi_2$, $\varphi_3$ according eq. (16).

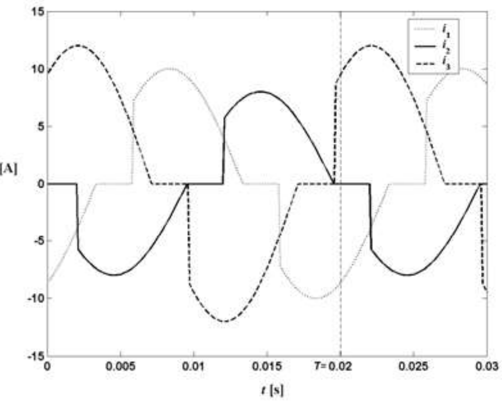


**Fig. 4** Time-dependency of the current of the load

Compensation is done using two-poles $R_i\,L_i\,C_i$ $(i = 1,2,3)$, for which eq. (17) is valid. Time dependency of function $(i_{11}^2 + i_{12}^2 + i_{13}^2)$ is in Fig. 5. Minimizing functional (1) we get

$C_1 = 2{,}841.10^{-5}$ F,

$C_2 = 4{,}184.10^{-5}$ F,

$C_3 = 4{,}645.10^{-5}$ F

$L_1 = 0{,}1568$ H,

$L_2 = 0{,}1065$ H,

$L_3 = 0{,}0959$ H

$R_1 = 4{,}926\ \Omega$,

$R_2 = 3{,}345\ \Omega$,

$R_3 = 3{,}013\ \Omega$

Application of the three above-mentioned methods had the same effect as the previous cases.

## 5. CONCLUSION

In this paper a method has been shown that enables to define the optimal values of parameters of compensation two-poles RLC, providing rigid supply mains. The proposed theory is valid for sinusoidal or nonsinusoidal, balanced or unbalanced three-phase power system. It can be easily extended to the power system with zero-sequence current, and/or voltages. Minimization of losses in line is not the only optimization problem solution suitable for practice. Objective function (1) can be formulated so that it is possible to design a filter for suppression of certain harmonic parts in time course of currents drawn from the network.

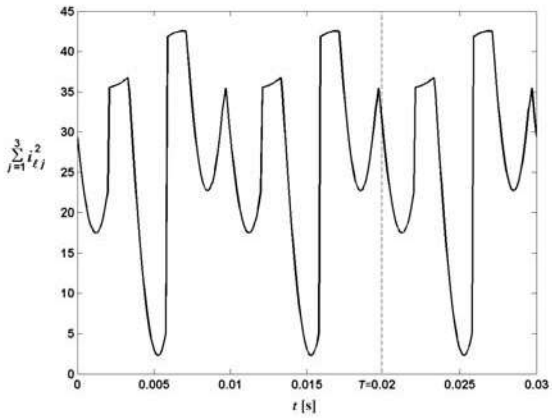


**Fig. 5** Time-dependency of instantaneous function $(i_{11}^2 + i_{12}^2 + i_{13}^2)$ from problem 4.3

## REFERENCES

[1] Mayer D., Páv M.: Contribution to the compensation of a three phase nonlinear power consumer. Elektrotechn. Obzor, 77 (1988), č. 2, pp. 75-78. (In Czech.)

[2] Mayer D.: Introduction to the electrical circuit theory. SNTL/ALFA, Praha 1981. (In Czech.)

[3] Mayer D.: Selected chapters from heavy current electrical engineering. Vydavatelství ZCU, Pilsen, 2002.

[4] Vasu E., Rao V. V. B., Sankaran P.: An optimization criterion for three phase reactive power compensation. Trans. IEEE, PAS-104 (1985), pp. 3216-3220.

[5] Optimization Toolbox User's Guide, The MathWorks, Inc., Natick, U.S.A., 2002.

## ACKNOWLEDGEMENT

Financial support of the Grant Agency of the Czech Republic (project No. 102/01/1401) is highly acknowledged.

## BIOGRAPHY

**Daniel Mayer**, Prof., Ing., DrSc. (1930). Received the Ing., PhD. and DSc degrees in electrical engineering from Technical University in Prague in 1952, 1958 and 1979, respectively. In 1959 Associate Professor at the University of West Bohemia in Pilsen, in 1968 Full Professor of the Theory of Electrical Engineering. Many years the head of the Dpt. of Theory of Electrical Engineering. Research interests: circuit theory, electromagnetic field theory, electrical machines and apparatus, history of electrical engineering. He published 6 books, more than 200 scientific papers and 10 patents. He is a Fellow of the IEE, member ICS, ISTET and UICEE, member of editorial boards of several international journals and leader of many grant projects (mayer@kte.zcu.cz).

**Petr Kropík**, Ing. (1971). He graduated from the Faculty of Applied Sciences (University of West Bohemia Pilsen) in 1995 in branch computer science. He was employ in SPT Telecom a. s. (in branch of computer networks, client–server database applications) and after three years he is the Assistant at the Department of Theory of Electrical Engineering, Faculty of Electrical Engineering (University of West Bohemia, Pilsen). He teaches the some subjects from computer science. His professional interest: optimisation methods, UNIX/LINUX operating systems, SQL databases and computer networks. Since 2001 is in external student for PhD (pkropik@kte.zcu.cz).