THE METHOD OF DESIGNING SWITCHED RELUCTANCE MOTORS BASED ON NELDER-MEAD ALGORITHM

Dariusz Kapelski, Tele- and Radio Research Institute, Warsaw, POLAND

Abstract

The aim of the study was a selection of geometric dimensions of a switched reluctance motor’s (SRM) rotor. The developed SRM is designed for operating at cryogenic temperatures, i.e. temperature of liquefied gases, such as hydrogen, oxygen, nitrogen and liquefied natural gas (LNG). An application for such motors can become: equipment for storage and transport such kinds of substances, modern cooling systems for superconducting transducers and even a fuel pump in spaceships.

This paper presents the method of designing of magnetic circuit of the electric motor, which is based on gradientless Nelder-Mead (N-M) algorithm. In the design process freeware software Scilab 5.4.1 and Femm 4.2. were used. Developed method is an example of a very simple optimization method. The criterion for optimization is the selection of the dimensions of the SRM rotor, which will give the highest value of mean torque as a function of a rotation angle. The paper shows selection of limits for the calculation algorithm, the final stop condition, and the influence of starting value on the number of iteration. The author also suggested solution to one of the fundamental disadvantages of the N-M method, influence of local minima of the objective function on the stability of the algorithm. The advantage of the proposed algorithm is the possibility of a parallel computing for different starting points, in order to reduce the risk of stopping the algorithm at a local minimum.

1. Introduction

In recent years it can be seen a development of new areas of application of electric motors. Exploration of space, cooling systems for superconducting applications and supercomputers, advanced medical equipment and chemical industries are the areas of application of cryogenic drives. Requirements for such devices include the placement of an electric motor in a cryogenic environment where it is exposed to continuous contact with the temperature of liquid gases [1,2].

Soft magnetic composite (SMC) materials are more likely to be used in a manufacturing of electromagnetic transducers, especially in electric motors. These materials consist generally of ferromagnetic particles distributed in a matrix of binding agent. SMC materials have a lot of advantages. One of the most important is the possibility to reduce production costs and relatively small influence of temperature on the resistivity of the such materials [5]. The Final Elements Method (FEM) calculations requires knowledge of magnetic properties of the soft magnetic materials exposed to liquid nitrogen temperature. Figure 1 shows relative magnetic permeability curves of the composite AncorLam at room and liquid nitrogen temperature. These characteristics has been used in simulation calculations.

![Figure 1. Magnetic permeability curves of AncorLam composite [3]](image)

It is planned the magnetic circuit of the motor will be made of AncorLam material. Stator will be composed of several components and is shown in Figure 2, outer diameter of stator is 86 mm.

The project of a stator takes into account the requirements of powder metallurgy. Production of this stator is carried out without additional machining. However, the rotor designing requires optimization of geometrical dimensions which takes into account the dimensions of a stator.
2. Nelder-Mead optimization method

The Nelder–Mead optimization method (downhill simplex method) is a commonly used nonlinear optimization technique, which is a well-defined numerical method for problems for which derivatives may not be known. The Nelder–Mead technique was proposed by John Nelder & Roger Mead in 1965 and is a technique for minimizing an objective function in a many-dimensional space [4]. For determination the directions of search and points at which the calculated value of the function it is used an object called a simplex. N-M algorithm generates a new test position by extrapolating the behavior of the objective function measured at each test points arranged as a simplex. The algorithm then chooses to replace one of these test points with the new test point and so the technique progresses. The simplest step is to replace the worst point with a point reflected through the centroid of the remaining N points. If this point is better than the best current point, then we can try stretching exponentially out along this line. On the other hand, if this new point isn’t much better than the previous value, then we are stepping across a valley, so we shrink the simplex towards a better point [4]. At each iteration, the vertices \( \{ \mathbf{x}_j \}_{j=1}^{n+1} \) of the simplex are ordered according to the objective function values (1)

\[
 f(\mathbf{x}_1) \leq f(\mathbf{x}_2) \leq \ldots \leq f(\mathbf{x}_{n+1})
\]

The \( \mathbf{x}_1 \) specifies as the best vertex, and the \( \mathbf{x}_{n+1} \) as the worst vertex. The algorithm uses four possible operations: reflection, expansion, contraction, and reduction, each being associated with a scalar parameter: \( \alpha \) (reflection), \( \beta \) (expansion), \( \gamma \) (contraction), and \( \delta \) (shrink). The values of these parameters satisfy \( \alpha > 0, \beta > 1, 0 < \gamma < 1 \), and \( 0 < \delta < 1 \). In the standard implementation of the Nelder-Mead method the parameters are chosen to be \( \alpha = 1, \beta = 2, \gamma = 1/2, \delta = 1/2 \). Let \( \overline{\mathbf{x}} \) be the centroid (the center of gravity of all points except \( \mathbf{x}_{n+1} \)) of the n best vertices (2).

\[
 \overline{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i
\]

Then the algorithm realize next steps of operations.

**Reflection**, compute the reflection point \( \mathbf{x}_r \) from

\[
 \mathbf{x}_r = \overline{\mathbf{x}} + \alpha (\overline{\mathbf{x}} - \mathbf{x}_{n+1})
\]

Evaluate \( f(\mathbf{x}_r) \). If

\[
 f(\mathbf{x}_1) \leq f(\mathbf{x}_r) \leq f(\mathbf{x}_{n+1})
\]

then replace \( \mathbf{x}_{n+1} \) with \( \mathbf{x}_r \).

**Expansion**, if \( f(\mathbf{x}_r) < f(\mathbf{x}_1) \) then compute the expansion point \( \mathbf{x}_e \) from

\[
 \mathbf{x}_r = \overline{\mathbf{x}} + \beta (\mathbf{x}_r - \overline{\mathbf{x}})
\]

and evaluate \( f(\mathbf{x}_e) \) if

\[
 f(\mathbf{x}_e) \leq f(\mathbf{x}_r)
\]

replace \( \mathbf{x}_{n+1} \) with \( \mathbf{x}_e \); otherwise replace \( \mathbf{x}_{n+1} \) with \( \mathbf{x}_r \).

**Contraction**, if \( f(\mathbf{x}_r) \geq f(\mathbf{x}_{n+1}) \), compute the inside contraction point \( \mathbf{x}_c \) from

\[
 \mathbf{x}_c = \overline{\mathbf{x}} - \gamma (\mathbf{x}_{n+1} - \overline{\mathbf{x}})
\]

and evaluate \( f(\mathbf{x}_c) \), if

\[
 f(\mathbf{x}_c) \leq f(\mathbf{x}_{n+1})
\]

replace \( \mathbf{x}_{n+1} \) with \( \mathbf{x}_c \); otherwise, go to reduction operation.

**Reduction**, for all without the best point, replace the point with

\[
 \mathbf{x}_i = \mathbf{x}_i + \delta (\mathbf{x}_i - \mathbf{x}_1)
\]

for \( i \in \{2, \ldots, n+1\} \).

Figure 3. Main operators of Nelder–Mead algorithm for a bi-dimensional problem A-starting simplex; B- reflection; C- expansion; D- contraction; E- reduction
The Nelder-Mead method may fail to converge to a critical point of $f$. The initial simplex is important, indeed, a too small initial simplex can lead to a local search, consequently the N-M can get more easily stuck. So this simplex should depend on the nature of the problem.

SciLab has implemented components using the NM algorithm to search for the minimum of a function. The proposed algorithm used the basic component using a form of the N-M algorithm, called `neldermead()`. The neldermead component is built on top of the optimbase and optimsimplex components.

The standard program code for the search of minimum function $f$ is shown below (12)

```plaintext
nm = neldermead_new ();
nm = neldermead_configure(nm,"-numberofvariables",2);
```

```
(12)
```

```
mm = neldermead_configure(nm,"-function",f);
nm = neldermead_configure(nm,"-x0",x0);
nm = neldermead_configure(nm,"-method","box");
nm = neldermead_configure(nm,"-boundsmin",[..]);
nm = neldermead_configure(nm,"-boundsmax",[..]);
nm = neldermead_configure(nm,"-simplex0method","randbounds");
nm = neldermead_search(nm);
```

The `neldermead_new()` creates a new neldermead object. `Neldermead_configure` method is used to configure the parameters of the problem. The neldermead solver can optimize problems with bounds. To do this, we can use box’s method, which projects the simplex into the bounds during the optimization. In this case, the initial guess ($x0$) must be located within the bounds. The `neldermead_search` function performs the search for the minimum. The `neldermead_display` function is used to display the state of the optimization and the `neldermead_get` is used to retrieve the optimum parameters [6].

3 A method of designing Switched reluctance motors.

3.1 Calculation algorithm

The solution to the problem of the influence of local minima on the stability of the algorithm is to conduct parallel computing. However, a necessary condition is to select several different starting points. This way calculation, can be carried out in parallel on different computers. This is a major advantage of the proposed solution. Figure 4 shows a flowchart of the calculation.

After determining the variability range of variables starting points are selected. Any carried out parallel computation requires a different starting points. Calculations can be performed in parallel on one or more computers.

After the calculation it is necessary to compare the final results and find the best solution.
Points P in the figure 4 was calculated using trigonometric functions (13).

\[
\begin{align*}
P_1(R_1 \cos(\alpha); R_1 \sin(\alpha)) \\
P_2(R_2 \cdot \cos(\beta); R_2 \cdot \sin(\beta)) \\
P_3(R_1 \cdot \cos(\alpha + \gamma); R_1 \cdot \sin(\alpha + \gamma)) \\
P_4(R_2 \cdot \cos(\beta + \delta); R_2 \cdot \sin(\beta + \delta))
\end{align*}
\]

(13)

where:

\[
\begin{align*}
\alpha &= a \cos(b/R_1) \\
\beta &= a \cos(b/R_2) \\
\gamma &= \pi - \alpha \\
\delta &= \pi - \beta
\end{align*}
\]

(14)

To draw a rotor model it was used the following Lua script functions: \text{mi\_drawline(), mi\_drawarc(), mi\_selectsegment(), mi\_setsegmentprop(), mi\_seteditmode('group'), mi\_selectgroup(), mi\_copyrotate().} However, the material in the rotor assigned and defined by the function: \text{mi\_addmaterial(), mi\_addbhpoints(bhcurve), mi\_addblocklabel(), mi\_selectlabel(), mi\_setblockprop().}

3.3 FEM simulation

To calculate the value of the objective function, it is necessary to implement of the simulation tests. In the study, the influence of the rotor dimensions on reluctance motor torque was calculated. FEM simulations allowed the determination the torque characteristics as a function of the angle of rotation, for a single motor phase supplied. The torque was determined on the basis of steady-state weighted stress tensor. The calculation of the torque and rotor motion were carried by the following program code (15):

```lua
delta = 2;
for step=0:15 do
    mi_seteditmode("group");
    mi_selectgroup(1);
    if step=0 then ang=0;
    else ang=delta; end;
    mi_moverotate(0,0,ang);
    mi_analyze;
    mo_selectblock((Rw+R1)/2,0);
    mo_selectblock(0,0);
    Tqr=mo_blockintegral(22);
    STqr=STqr + Tqr;
end;
```

(15)

where: delta - the rotation angle of the rotor; Tqr - static torque, determined on the basis of the stress tensor.

A average value of torque (Tsr) was determined based on the sum of the partial STqr, after the loop (15). In the simulation were as follows parameters:

- Current phase 10 A,
- The number of turns 20,
- Motor axial length 1 mm,
- Shaft material: titanium,
- Rotor and stator material: AncorLam.

3.4 Limitations and criterion an end of calculation

A very important aspect of calculations is the selection of final stop condition and the range of variation of dimensions. SciLab software has an implemented function that carries out N-M optimization algorithm. This function searches a minimum of the objective function. The FEM simulation gives a average value of torque. Optimization is designed to obtain the highest possible average torque. Therefore, the objective function must be described as an the inverse of average torque values Tsr (16)

\[
\min f = \frac{1}{Tsr}
\]

(16)

As a final stop condition there was selected a minimum evolution of the average values of torque (17), and a minimum number of iterations equal 20

\[
Tsr_n - Tsr_{n-1} < 0.0001
\]

(17)

In order to reduce computation time and to ensure the stability of the algorithm, it was introduced limits for values of the variables. SciLab allows to specify the limits of variation of vectors. This requires an usage the following program code (18)

```lua
neldermead_configure(nm,"-boundsmin",[dmin Rmin]);
neldermead_configure(nm,"-boundsmax",[dmax Rmax]);
```

(18)

In the code of the program (18), minimum width of the tooth \text{dmin} was 8 mm; and the \text{dmax} was 15 mm.

In the case of the inner radius of the rotor \text{R1}, the additional procedure was used. This procedure was used for limiting the range of variation, and protection against the situation in which the radius is less than the radius of the shaft. The inner diameter \text{R1} is described by equation (19), wherein variable is \text{R}

\[
R1 = R + Rw + 5
\]

(19)

where: \text{Rw}- radius of the shaft.
4. Results of simulations

For checking the stability of the algorithm tests were performed. The width of a rotor tooth \( d \) was calculated at a constant inner radius of the rotor \( R_1 \) for several starting points. Number of iterations and the calculated width of the tooth are shown in Table 1.

Table 1. Influence of value of starting point on number of iterations for one variable \( (d) \)

<table>
<thead>
<tr>
<th>Starting value ((d, R_1))</th>
<th>Stop value ( d )</th>
<th>Number of iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>(8; 15)</td>
<td>9.96875</td>
<td>22</td>
</tr>
<tr>
<td>(10; 15)</td>
<td>9.96868</td>
<td>22</td>
</tr>
<tr>
<td>(12; 15)</td>
<td>9.96798</td>
<td>24</td>
</tr>
<tr>
<td>(15; 15)</td>
<td>9.96872</td>
<td>21</td>
</tr>
</tbody>
</table>

For the different starting points the calculation algorithm always ended with similar results.

It was also carried out the calculations for the variable width of the rotor tooth \( d \) and for variable inner diameter of rotor \( R_1 \).

Table 2. Influence of value of starting point on number of iterations for two variables \((d, R_1)\)

<table>
<thead>
<tr>
<th>Starting value ((d, R_1)) [mm]</th>
<th>Optimal value ((d, R_1)) [mm]</th>
<th>Average value of torque [Nm]</th>
<th>Number of iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>(9; 15)</td>
<td>9.881836 ( R_1 = 11.818848 )</td>
<td>0.041833</td>
<td>40</td>
</tr>
<tr>
<td>(8; 18)</td>
<td>9.828278 ( R_1 = 12.140121 )</td>
<td>0.041835</td>
<td>48</td>
</tr>
<tr>
<td>(12; 16)</td>
<td>9.825912 ( R_1 = 11.797874 )</td>
<td>0.041830</td>
<td>38</td>
</tr>
<tr>
<td>(15; 12.5)</td>
<td>9.841187 ( R_1 = 12.140121 )</td>
<td>0.041812</td>
<td>39</td>
</tr>
<tr>
<td>(15; 15.5)</td>
<td>9.878278 ( R_1 = 12.140121 )</td>
<td>0.041822</td>
<td>40</td>
</tr>
<tr>
<td>(10; 17.5)</td>
<td>9.958587 ( R_1 = 18.590594 )</td>
<td>0.000001</td>
<td>31</td>
</tr>
<tr>
<td>(15; 18)</td>
<td>9.863037 ( R_1 = 12.343018 )</td>
<td>0.04180</td>
<td>40</td>
</tr>
<tr>
<td>(11; 14)</td>
<td>9.825678 ( R_1 = 12.156797 )</td>
<td>0.041832</td>
<td>48</td>
</tr>
</tbody>
</table>

In Figure 6 shows the torque characteristics as a function of the angle of rotation of a rotor SRM, for a few selected steps in calculations.

Results in the tenth iteration are very close to the optimum solution. The average value of torque achieved in the tenth iteration is 0.041514 Nm.

Conclusion

It is possible to get compacts with a tolerance of dimensions \( \pm 0.1 \) mm by the use of soft magnetic composites. Results obtained by application of the N-M algorithm have a much higher resolution. This allows for a substantial reduction of computation time by reducing the resolution of variables, or select another final stop conditions. Calculation time for each of a starting point was approximately 5 hours.

The results of calculations required averaging, due to the technological possibilities of powder metallurgy. The averaged results are shown in Table 3 (bold lines). The results allowed to design of a switched reluctance motor to drive a small pump of liquid nitrogen. Basic motor parameters are presented in table 3.

Table 3. The desired parameters of a prototype SRM

<table>
<thead>
<tr>
<th>Number of Phase</th>
<th>( \Phi_n )</th>
<th>Speed</th>
<th>Operating frequency of current</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>36 mm</td>
<td>18000 rpm</td>
<td>600 Hz</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stator outer diameter</th>
<th>( \Phi_n )</th>
<th>36 mm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stator inner diameter</td>
<td>( \Phi_n )</td>
<td>40.5 mm</td>
</tr>
<tr>
<td>Stator material</td>
<td>Ancholam</td>
<td></td>
</tr>
<tr>
<td>Rotor outer diameter</td>
<td>( \Phi_n )</td>
<td>40 mm</td>
</tr>
<tr>
<td>Rotor tooth width</td>
<td>( d )</td>
<td>9.9 mm</td>
</tr>
<tr>
<td>Rotor inner radius</td>
<td>( R_1 )</td>
<td>12.1 mm</td>
</tr>
<tr>
<td>Rotor material</td>
<td>Ancholam</td>
<td></td>
</tr>
<tr>
<td>Shaft diameter</td>
<td>( \Phi_n )</td>
<td>10 mm</td>
</tr>
<tr>
<td>Shaft material</td>
<td>Titanium</td>
<td></td>
</tr>
</tbody>
</table>

The operating parameters (shaft diameter and rotation speed) were selected so that the prototype motor could operate as a drive for liquid nitrogen pumping system [2]. Obtained geometric dimensions...
allowed to carry out FEM simulations of optimized structures. Figure 7 shows the distribution of magnetic flux density and magnetic flux in the magnetic circuit of SRM.

Figure 8 shows the characteristics of the static torque generated by one of the motor phases, with currents of 10; 20; 30 A.

A designing of motors for use in liquid gases requires a solution of many technical problems. Operating parameters of a presented SRM will be measured in liquid gas. The results of this work will be used to provide guidance for the design and selection of materials for cryogenic drives.

**Figure 7. Distribution of magnetic flux density.**

**Figure 8. Torque vs. angle characteristic for different value of current in one phase.**

**Bibliography**


**Author:**

M.Sc. Eng. Dariusz Kapelski
Tele- and Radio Research Institute
ul. Ratuszowa 11
03-450 Warszawa, Polska
tel. (+48 22)619 22 41 wew. 265
dariusz.kapelski@itr.org.pl

The author also plan to develop the algorithm. Further development works of an algorithm will lead to creation of a stable computing system that allows the calculation of geometrical dimensions of the SRM rotor motors with a complex shape. The number of variables in future algorithm will be significantly increased. The aim of work is to obtain an algorithm allowing for designing of the rotor tooth profile and achieve a given characteristic of static torque as a function of the angle of rotation.