C Programming Language Based Finite Element Analysis of Permanent Magnet Synchronous Motors

Gergely Kovács, Miklós Kuczmann
Széchenyi István University (Hungary)
kovacsg@maxwell.sze.hu, kuczmann@maxwell.sze.hu

Abstract— The paper presents the antecedents of my doctoral work, which were comparisons of two different design software tools in the case of a PMS motor. The paper also presents the future plans of my doctoral work, which is a finite element based analysis software for research of induction motor and PMS motors.

I. INTRODUCTION

The computer-aided design is one of the important part of the electric engine development. Electric engine development has been at the Széchenyi István University, as well and my part of this development is to design and optimization a brushless DC (direct current) motor family, which are will be applied with bicycles and smaller motors.

There are also two bigger projects where these motors will be applied. One of them is a development of hybrid E-VAN, which is an adapted Ford-truck. The second one is an electric car which is an individual development by the students and teachers of the university.

The main aspect of this PMS motor (Permanent Magnet Synchronous motor) [1] development is to reduce the weight and the size of the engine but the torque and losses of the motor not to change. There are more ways to design the PMS motor. The paper shows these possibilities furthermore a structure of a C programming language based development environment is introduced which will be used for electric motor development. First of all, two different PMS motor possibilities are shown, after that the structure of the C programming language based development environment is introduced.

II. DEVELOPMENT OF THE PMS MOTOR WITH INFOLYTICA MOTOR SOLVE

The Infolytica MotorSolve [2] is an electric motor design software for brushless DC motor. In this case the motor design is made by the help with different templates. By the help of the change of the sizes of the schemes can have been designed the electric motor furthermore the Fig. 1, shows some templates of the magnets of the rotor.

Fig. 1. Some template of the slots of the stator in Infolytica MotorSolve.

The Fig. 2, shows some templates of the slots of the stator.

Fig. 2. Some template of the slots of the stator in Infolytica MotorSolve.

The parameters of the electric motor are calculated by the help of an automated-FEA (Finite Element Analysis) solver, for example torque, losses, power, and the others. The easy applicability is the advantage of the program. The disadvantage of the program that the motor designing is possible by the help only with some defined templates, which means that there is no way to design a motor with optional geometry.

III. DEVELOPMENT OF THE PMS MOTOR WITH COMSOL MULTIPHYSICS

The COMSOL Multiphysics [3,4] is a finite element based software for the modeling and simulation of any physics-based system. In this case calculations on optional geometry have been able to make with the program; however the preprocessing is more difficult for instance to draw the model, or to set the boundary conditions. The Fig. 3, shows some possibilities of settings.

Fig. 3. The graphical user interface of COMSOL.

The simulated model has been modeled as a static magnetic field problem, where the following Maxwell’s equations can be used [5-11]

\[ \nabla \times \mathbf{H} = \mathbf{J}_e, \text{ in } \Omega_e \cup \Omega_m, \]  
\[ \nabla \cdot \mathbf{B} = 0, \text{ in } \Omega_e \cup \Omega_m. \]  

118
Here \( H \) is the magnetic field intensity, \( J_0 \) is the source current density, \( B \) is the magnetic flux density. The \( H \) magnetic field intensity can be expressed as \[5-10\]

\[
H = \begin{cases} 
\nu_0 B, & \text{in air}, \quad \Omega_a, \\
\nu_0 \nu, B, & \text{in magnetic material}, \quad \Omega_m.
\end{cases}
\] (3)

Here \( \nu_0 \) is the reluctivity of vacuum and \( \nu \) is the relative reluctivity. The air region is denoted by \( \Omega_a \) and the magnetically region is denoted by \( \Omega_m \). The magnetic flux density can be expressed as

\[
B = \nabla \times A,
\] (4)

where is the magnetic vector potential \([2, 8]\). This expression is satisfied (2), because of the identity \( \nabla \cdot \nabla \times \nu = 0 \) for any vector function \( \nu = \nu(\mathbf{r}) \). Substituting (4) to the (1) and (2) and using the (3) constitutive relations can be obtained by the following partial differential equations:

\[
\nabla \times (\nu_0 \nabla \times A) = J_0, \text{ in } \Omega_a,
\] (5)

and

\[
\nabla \times (\nu_0 \nu_\nu \nabla \times A) = J_0, \text{ in } \Omega_m.
\] (6)

The divergence of the magnetic vector potential can be selected according to Coulomb gauge,

\[
\nabla \cdot A = 0,
\] (7)

which is satisfied automatically in two dimensional problems \([2, 8]\). In two dimensional case the source current density has only \( z \) component, moreover the magnetic field intensity vector and the magnetic flux density vector have \( x \) and \( y \) components,

\[
J_0 = J_{0z}(x,y)e_z,
\] (8)

\[
H = H_x(x,y)e_x + H_y(x,y)e_y,
\] (9)

\[
B = B_x(x,y)e_x + B_y(x,y)e_y.
\] (10)

The magnetic vector potential has only \( z \) component

\[
A_z = A_z(x,y)e_z,
\] (11)

and the \( x \) and \( y \) components of the magnetic flux density can be described as

\[
B_x(x,y) = \frac{\partial A_z}{\partial y},
\] (12)

and

\[
B_y(x,y) = -\frac{\partial A_z}{\partial x}.
\] (13)

The boundary conditions of a two dimensional static magnetic field problem can be formulated as

\[
(v \nabla \times A) \times \mathbf{n} = 0, \text{ on } \Gamma_n.
\] (14)

The problem was calculated by the help with the (5) and (6) partial differential equations and the (13) and (14) boundary conditions.

The main advantage of the COMSOL, that solution of optional geometry is possible but the preprocessing and postprocessing period is more difficult and for example the solution the transient analysis to calculate the torque of the motor is very slow. The static magnetic field problem was calculated by the help with these partial differential equations and boundary conditions.

IV. COMPARING THE TWO DIFFERENT DEVELOPMENT SOFTWARE

The Fig. 4, shows the scheme of the developed permanent magnet synchronous motor which is designed by the help with Infolytica MotorSolve \([9]\).

The outer diameter of the motor is 205mm furthermore the inner diameter of the motor is 187mm. The rotor type is exterior and it has 28 Neodymium magnets. The stator has 36 slots with three phase double layers windings. The type of the rotor and the stator material is M19. The maximum power of the PMSM is 1200W, as well as the maximum rotational speed of the motor is 1000RPM. In this case the delivered torque is about -64Nm. When the rotational speed is about 100RPM then the delivered torque is 11.8Nm and the motor has 200W power.

The simulation results were compared focusing the magnetic potential, and the magnetic flux density of the PMSM motor in the case of 1000 RPM rotational speed. The same place were chosen for the calculation of the magnitude of the magnetic potential and the magnetic flux density.
The Fig. 5, shows the simulation result of the magnetic potential calculated by the Infolytica MotorSolve.

![Fig. 5. The simulation results of the magnetic potential with the help Infolytica MotorSolve.](image1)

The Fig. 6, shows the simulation result of the magnetic potential by the help of COMSOL Multiphysics.

![Fig. 6. The simulation results of the magnetic potential with the help COMSOL Multiphysics.](image2)

Comparing the simulation results which were calculated two different design software tools they are similar in the case of 1000RPM rotational speed.

The Fig. 7, shows the simulation result of the magnetic flux density calculated by the Infolytica MotorSolve.

![Fig. 7. The simulation results of the magnetic flux density with the help Infolytica MotorSolve.](image3)

The Fig. 8, shows the simulation result of the magnetic flux density by the help of COMSOL Multiphysics.

![Fig. 8. The simulation results of the magnetic flux density with the help COMSOL Multiphysics.](image4)

Comparing the simulation results which were calculated two different design software tools they are similar in the case of 1000RPM rotational speed.

The simulation results of the PMSM were compared with each other focusing the delivered torque in the case of 1000RPM rotational speed, as well. Calculating the delivered torque with Infolytica MotorSolve is -68,4 Nm and with COMSOL Multiphysics is -67,25 Nm in the case of maximal rotational speed.

Comparing the simulation results are similar to each other which means the two different design software tools are convenient to design PMS motors. The main advantage of the Infolytica MotorSolve is that the development of the motor is easier than with COMSOL Multiphysics. Disadvantage of the first program is that the motor design is possible by the help with only some predefined templates. The main advantage of the COMSOL Multiphysics is the possibility of designing the PMS motors with optional geometries; however the method of this development is more difficult with COMSOL Multiphysics.

A C programming language based development environment is decided to develop due to the disadvantages of the other two different development software tools.

V. C PROGRAMMING LANGUAGE BASED DEVELOPMENT ENVIRONMENT

The new development is actually a finite element based solver for calculate parameters of induction motors and PMS motors. The Fig. 9, shows the structure of the development environment.

![Fig. 8. The scheme of the development environment.](image5)

The structure of the environment consists three parts.
The first part is the preprocessing part, which consists of a sort of CAD software to draw the model geometry and the GMSH [12]. The GMSH is a three-dimensional finite element mesh generator.

The second part is the finite element based solver. The meshed geometry of the problem is calculated and solved by the help of the PETSc [13]. The PETSc is a C language based package on Linux, with partial differential equations and matrix operations can be used. The finite element based solver is written by the help with this package.

The last part is the postprocessing part. This part consists of the VTK [14], and the Qt [15]. The VTK is a C++ programming language based visualization toolkit on Linux environment. The GUI (Graphical User Interface) will be developed by the help with Qt environment, which is also a C++ programming language based software on Linux environment.

Static magnetic field problems are solved by the help with the C programming language based development structure at present.

VI. CONCLUSION AND FUTURE PLANS

Two different finite element based development environment were compared with each other, which are disadvantages on the case of induction and PMS motor development.

A C programming language based development environment is being improved due to the disadvantages.

In the future the finite element solver will be improve to calculate parameters of induction and PMS motors. The electric engines will be calculated by the help with this environment using parallel computations, as well. The development environment will be applied to my doctoral work, where I would like to investigate arrangements of the magnets and airgaps of the rotor of the BLDC motor to develop more energy efficient BLDC motors.

ACKNOWLEDGMENT

“TÁMOP-4.2.2.A-11/1/KONV-2012-0012: Basic research for the development of hybrid and electric vehicles - The Project is supported by the Hungarian Government and co-financed by the European Social Fund”

REFERENCES