

# TOOLS CAPABLE OF PERFORMING OPTIMIZATION FOR ELECTRICAL MACHINES

A.B. Asaf Ali

## 1 INTRODUCTION

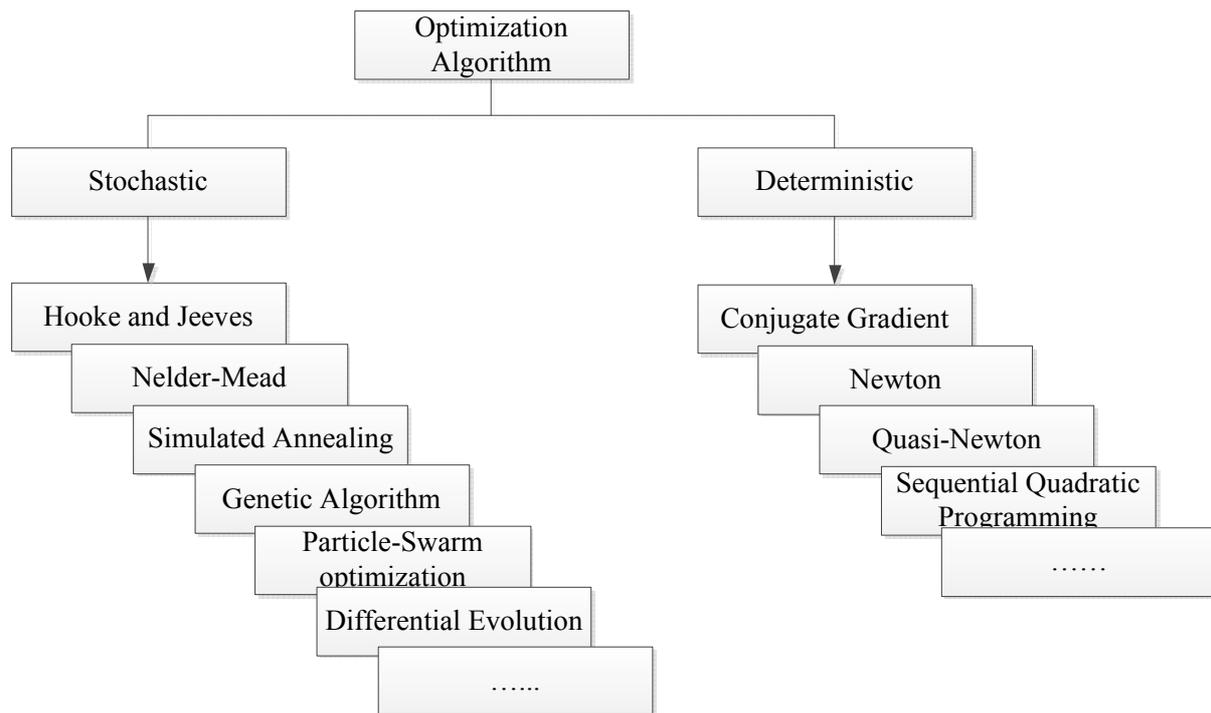
Although optimization techniques are widely implemented in various fields, it is also being used by the electrical machine design engineers to get insight about the magnetic circuit. The magnetic circuit parameters are very sensitive and it influences the output performance such as torque ripple, losses, efficiency and costs of the electrical machine. Thus optimization methods find a solution in the design criterion by fulfilling the given objective function either to minimize or to maximize under constrained or unconstrained geometrical parameters. For example the objective function can be minimizing torque pulsations or maximizing the efficiency, etc. In this report server exchange among Finite Element Software (FEM) from FLUX, CEDRAT [1] and the optimization tool FGot from G2ELAB [2] are discussed. A simple example with some geometry parameters by reducing the cogging torque of permanent magnet synchronous machine with tooth coil winding arrangements will be exemplified.

## 2 OPTIMIZATION METHODS

Basically the optimization methods are classified into deterministic and stochastic methods. The optimization methods algorithm can be understood from the tree shown in **Figure 2.1**. The mathematical explanation on minimizing the objective function with number of variable vectors  $\bar{X}$  can be written as

$$\begin{aligned} &\text{minimize function } f(\bar{X}) \\ &g_i(\bar{X}) \leq 0, 1 \leq i \leq m \\ &h_i(\bar{X}) = 0, 1 \leq i \leq k \end{aligned} \tag{2.1}$$

The objective function can be a single or a multi-objective function.  $g_i$  and  $h_i$  are equality and inequality constraints. The deterministic methods are local optimum methods that use gradient methods to find the objective function whereas the stochastic methods find the global optimum based on evolution methods. There are also algorithms combining both methods [3].



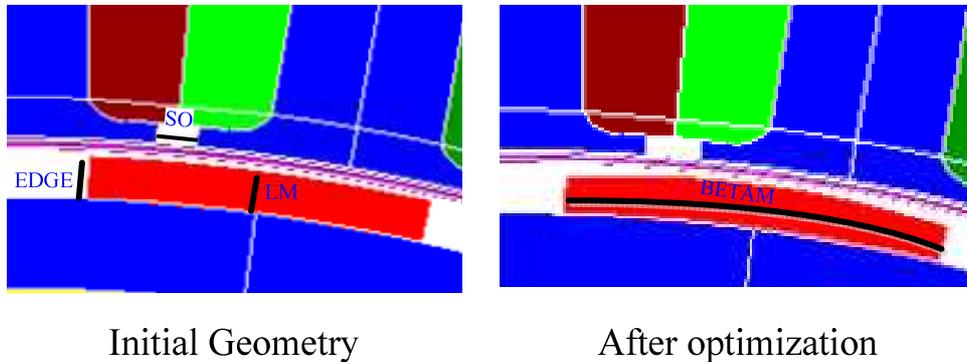
**Figure 2.1** Optimization Algorithm Methods

The theory behind the optimization techniques can be found in the literature [4]. All the methods should be carefully interpreted. Sometimes the convergence is poor in local optimum; it can be improved with the help of good initial values. Sometimes other methods based on evolutionary strategy are able to find global optimum. In this report a Genetic Algorithm based on evolution strategy with mono objective function is illustrated. Optimization can be done in system level and also in component level (Electrical Machines). When performing an optimization for electrical machines, its parameters can be directly optimized with saturation dependent numerical calculations provided possibility to combine FEM and optimization tools. When performing an optimization for electrical machines the following steps are necessary:

1. Selecting the variables or geometry dimensional parameters to be optimized. These can be poles, height and width of slot, outer radius or inner radius, flux density in teeth, etc.
2. Based on physical limits, setting up constrained or unconstrained limits for the parameters.
3. Providing good initial values for the parameters based on experience.
4. Preparing the objective function for which the variables should be optimized.
5. Selecting the optimization algorithm to perform optimization

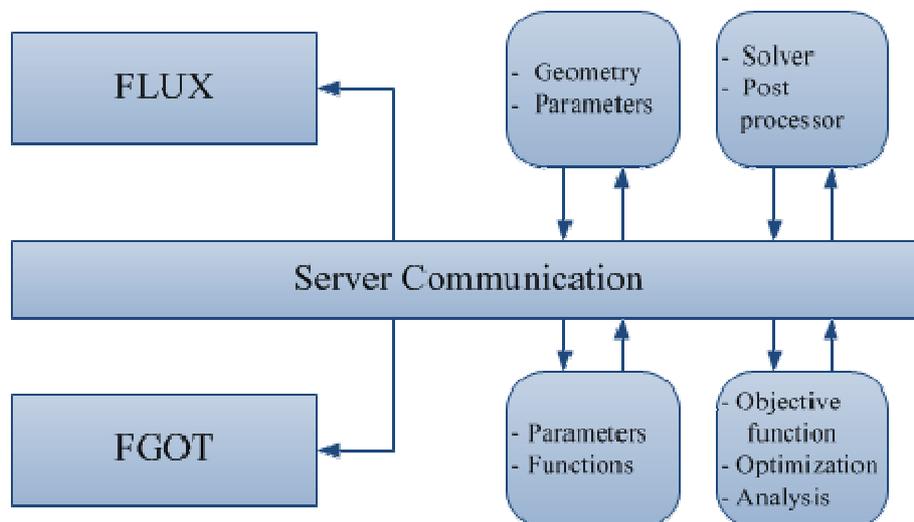
### 3 EXAMPLE - OPTIMIZATION FOR A PM-MACHINE

In this section a problem with cogging torque minimization is shown as an example. To illustrate this example a machine with high torque and low speed is considered. The machine geometry for both before and after optimization is shown in **Figure 3.1**.



**Figure 3.1** Machine Geometry

The machine stator winding is of tooth coil winding type with 2/3 arrangement. The rotor structure is surface mounted permanent magnet type. The machine parameters such as slot opening (SO), magnet width (BETA), height of the magnet (LM) and edge of the magnet (EDGE) are varied during the optimization process. This geometry is created in FLUX environment. This model should be prepared to make the optimization in FGot software. **Figure 3.2** shows the exchange of information between the two software programs.



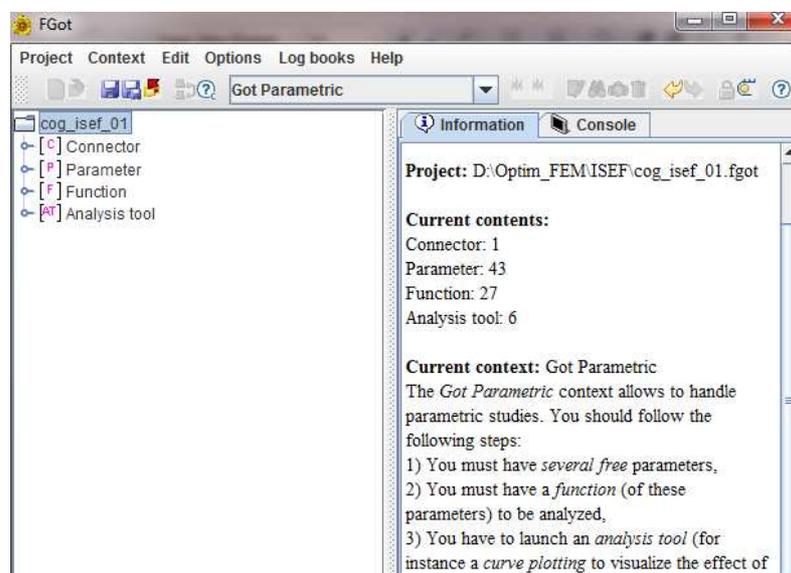
**Figure 3.2** Software Communication

After launching FLUX, the project file is opened in the FLUX supervisor and is ready to exchange information. Later FGot is launched and the communication is established between both the software. The connection set-up is shown in tutorials in the FGot help. The name Parameters, Functions, Optimization and Analysis are sub menus in FGot. The parameters that

have to be optimized are created in FGot. The upper limits and lower limits of the parameters should be given. Functions are provided to create constraints and objective functions. Sensors and Input/Output parameters in FLUX are outputs as a function in FGot. A macro in the form of “pyflux” file is created to get the results from the post-processor in FLUX. For every parameter variation this file will update the Sensors and Input/Output Parameters. This will be read as an output function in FGot. Thus FGot will make optimization by varying its input parameters and verifying its output functions until convergence of the objective function is reached.

### 3.1 General Menu

This section concentrates more on the FGot software and gives an overview how FGot and FLUX are connected. **Figure 3.3** below shows the FGot dialog menu. More details about the software are given in the tutorials [2].

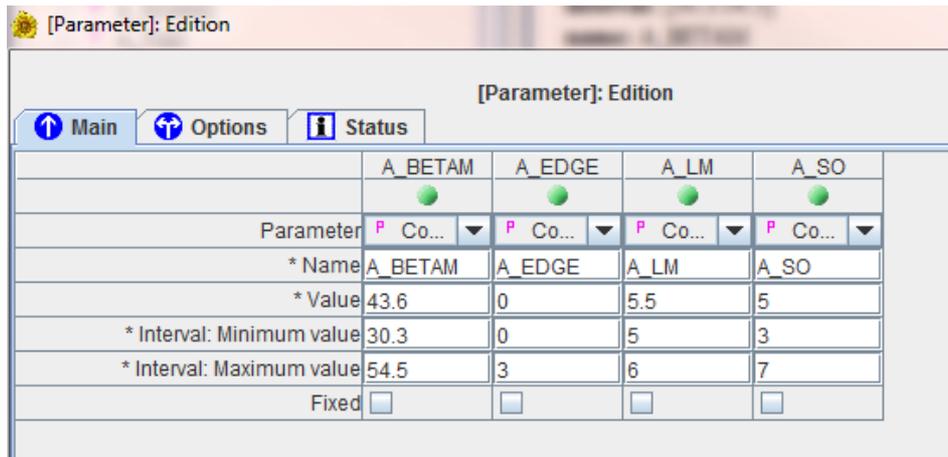


**Figure 3.3** FGot menu

The menu tabs give information about Log books, Help, etc. The right side dialogue contains information about the project and the console to view the messages. The left tree contains parameters, functions, optimization algorithm and analysis tool.

#### 3.1.1 Parameters in FGot

Once FLUX is connected to FGot all parameters in FLUX are available. **Figure 3.4** shows the parameters menu where the parameter values necessary to be optimized are entered. In this menu the range of parameter variation can be entered. This range should be selected based on physical insight of the geometry limitations. If the parameters are fixed then the initial value for the optimization is the default value entered. Rounding of the parameters, continuous or discrete parameters can be selected here. Though the winding number of turns is discrete the current version of the software is not able to consider the number of turns in discrete form.



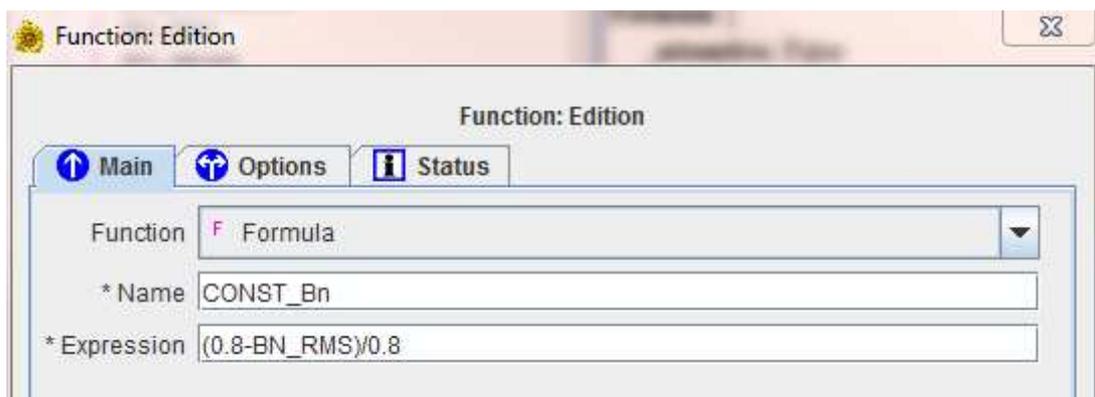
The screenshot shows a software window titled "[Parameter]: Edition" with three tabs: "Main", "Options", and "Status". The "Main" tab is active. Below the tabs is a table with four columns representing parameters: A\_BETAM, A\_EDGE, A\_LM, and A\_SO. Each column has a green status indicator at the top. The table contains the following data:

	A_BETAM	A_EDGE	A_LM	A_SO
Parameter	P Co...	P Co...	P Co...	P Co...
* Name	A_BETAM	A_EDGE	A_LM	A_SO
* Value	43.6	0	5.5	5
* Interval: Minimum value	30.3	0	5	3
* Interval: Maximum value	54.5	3	6	7
Fixed	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Figure 3.4** Parameters for optimization

### 3.1.2 Functions in FGot

In the function menu it is possible to enter the constraints and objective functions of the problem. The constraint air-gap flux density used in this example is shown in **Figure 3.5**.



The screenshot shows a software window titled "Function: Edition" with three tabs: "Main", "Options", and "Status". The "Main" tab is active. Below the tabs are input fields for defining a function:

- Function: F Formula
- \* Name: CONST\_Bn
- \* Expression:  $(0.8 - BN\_RMS) / 0.8$

**Figure 3.5** Function block

The constraints and objective functions are entered as normalized values. It is better for the optimization program to avoid error during optimization. Once the “pyflux” file is executed in FLUX the output parameters are transferred as functions. For example, BN\_RMS is the flux density in the air-gap obtained in FEM transferred to FGot as output.

### 3.1.3 Optimization in FGot

The next procedure is to select the optimization algorithm and to prepare the optimization problem for the selected algorithm. As explained earlier there are many algorithms and it depends on the user to select. For example, best solutions can be displayed as Pareto Frontier in the multi objective optimization. The number of evolutions will be more and it takes longer time to solve.

In this example a genetic algorithm for the global optimization is selected with a mono-objective function. The objective function is penalized. The optimization is a standard constrained problem with inequality constraints. The following specifications are summarized for the example problem:

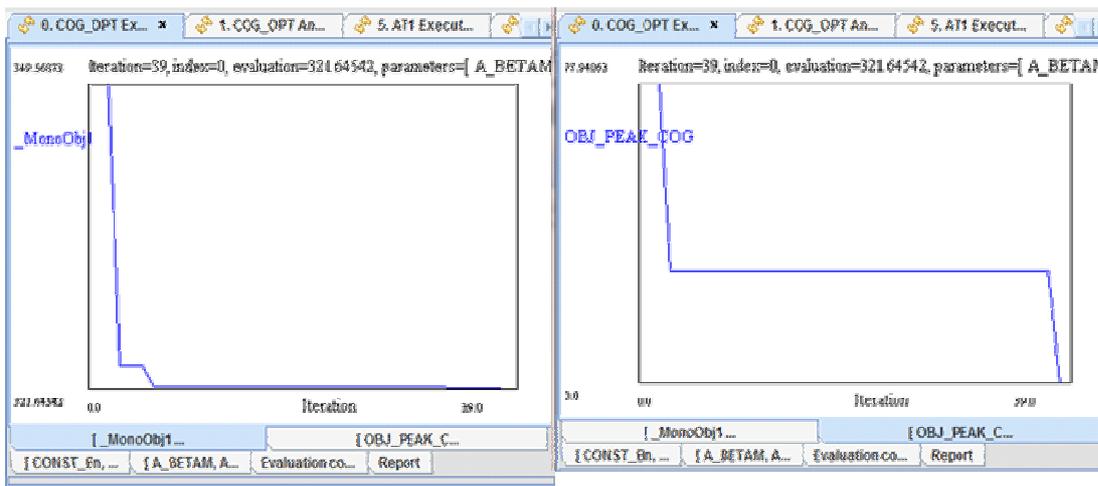
The objective of the design is,

- Minimizing the cogging torque

The constraints in the example are,

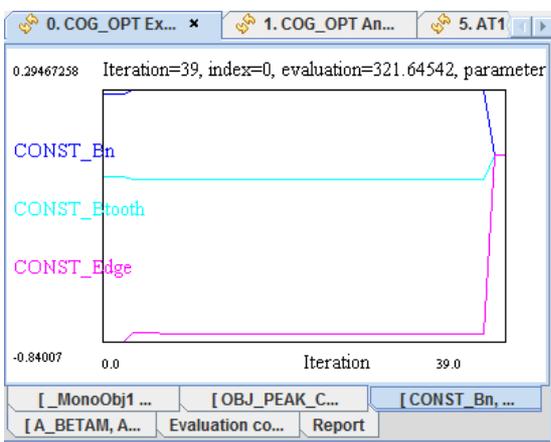
- Flux density in the air-gap:  $B \leq 0.8 \text{ T}$
- Flux density in the teeth:  $B \leq 1.5 \text{ T}$

The problem is solved and the final results are shown in the **Figures 3.6 – 3.8**. **Figure 3.6** shows the mono-objective function for different iterations.

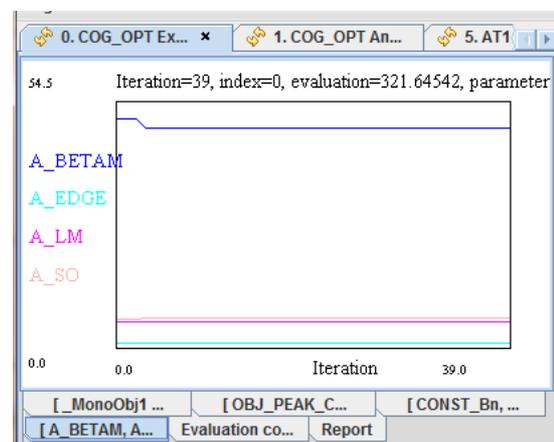


**Figure 3.6** Mono-Objective function (left) and Objective function (right)

For the selected iterations there were approximately 321 evaluations for 40 iterations. The evaluation time can be saved with the help of screening methods to make the computational time faster. **Figure 3.6** right shows the objective function for the number of iterations. **Figure 3.7** shows the constraints for different iterations and **Figure 3.8** the parameters for different iterations.



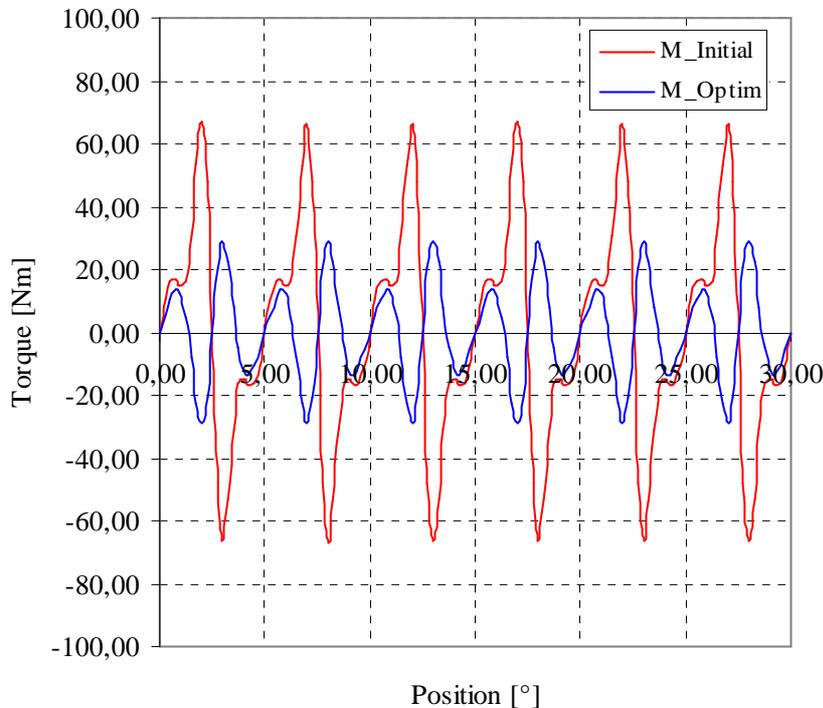
**Figure 3.7** Constraints vs Iterations



**Figure 3.8** Parameters vs Iterations

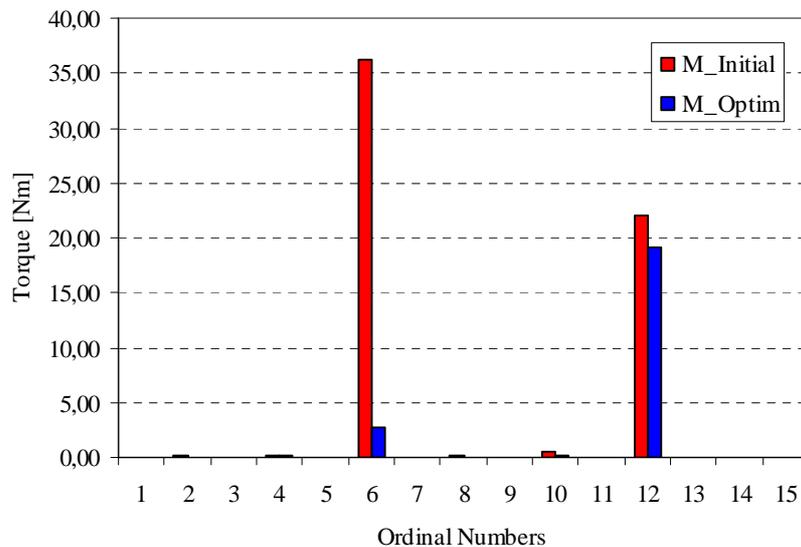
## 4 COGGING TORQUE COMPARISON

There are two kinds of torque pulsations in PPSM: Cogging torque exists under no-load conditions and a torque ripple exists under load conditions. The cogging torque of the initial geometry and the optimized geometry are shown in **Figure 4.1**.



**Figure 4.1** Cogging Torque comparison

The red curve in **Figure 4.1** is for the initial geometry, the blue curve is for the optimized geometry. For the given objective function and the constraints the peak to peak torque is minimized. In this example the peak torque in the optimized model is reduced to 44% when compared with the initial model. **Figure 4.2** shows the Fourier analysis of the curve.



**Figure 4.2** Cogging Torque Harmonics

The spectrum shows that in the optimized design the 6<sup>th</sup> harmonics is considerably reduced but the 12<sup>th</sup> harmonics is not reduced when compared to the initial design.

The final parameters obtained from the optimization program are tabulated below (**Table 1**).

**Table 1** Final Optimized Parameters

Variables	Characteristics	Values [mm]
SO	Slot opening	6,53
LM	Magnet height	5,72
EDGE	Magnet edge	1,12
BETAM	Magnet width	48,12

## 5 SUMMARY

Cogging torque minimization methods by directly coupling the FEM with an optimization program is shown in this report. The design optimization parameters are slot opening, magnet height, magnet edge and magnet width. Constraints for the analysis have been made to parameters like flux density in the teeth and in the air-gap. There are various optimization algorithms in the Fgot software. The evolutionary strategy is time consuming but this method is also capable of providing a good solution for the electrical machine design. The genetic algorithm is selected to make quick results with less number of generations.

## 6 REFERENCE

- [1] FLUX, CEDRAT, Version 10.4
- [2] Fgot, Prof. Jean-Louis Coulomb, Grenoble Electrical Engineering Laboratory, G2ELAB, Grenoble-INP, Grenoble, France
- [3] Krotsch J, Piepenbreier B: *Hybrid Algorithm for Multi-objective Optimization of PMSM using massively distributed Finite Element Analysis*, 12<sup>th</sup> International Conference on Optimization of Electrical and Electronic Equipment, OPTIM 2010
- [4] Boldea I., Tutelea L: *Electrical Machines Steady State, Transients and Design with MATLAB*, CRC press, 2010, ISBN 978-1-4200-5572-6