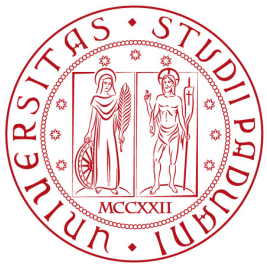


Università degli Studi di Padova

DIPARTIMENTO DI INGEGNERIA INDUSTRIALE
Corso di Laurea Magistrale in Ingegneria dell'Energia Elettrica

The Harmonic Balance method for the solution of nonlinear circuit and field problems with harmonic sources and its integration with a commercial simulation software

Relatore:
Prof. Piergiorgio Alotto

Laureando:
Alessandro Giust

Anno Accademico 2015-2016

Sommario

Il metodo del Bilanciamento Armonico è una tecnica matematica per risolvere equazioni differenziali non lineari con soluzioni periodiche.

In questo lavoro viene applicato il metodo del Bilanciamento Armonico (HB) per studiare problemi sia circuitali sia campistici non lineari.

La prima parte dell'elaborato è dedicata a fornire una formulazione matematica del metodo, dopodichè questo verrà utilizzato per studiare circuiti elettrici non lineari. Verranno presentate le varie tecniche e sarà implementato e testato il relativo codice su un esempio specifico.

La parte pricipale di questa tesi consiste nell'applicazione del metodo del Bilanciamento Armonico allo studio di campi elettromagnetici non lineari, sarà quindi trattato nel dettaglio il metodo degli Elementi Finiti con Bilanciamento Armonico (HBFEM). Verrà perciò derivato il metodo HBFEM per capire come strutturare un codice che implementi tale tecnica.

L'originalità di questo lavoro è legata alla scelta di integrare il metodo del Bilanciamento Armonico in un solutore commerciale agli elementi finiti, al contrario di quanto già avvenuto con vari codici di ricerca basati sul metodo FEM. Verrà illustrato come sia possibile implementare nel software commerciale COMSOL il metodo del Bilanciamento Armonico tramite MATLAB in modo semplice sfruttando al massimo le routine standard del predetto solutore FEM. Sarà quindi testato il metodo su alcuni esempi significativi in modo da dimostrare l'accuratezza e le performance del codice comparandole con le classiche simulazioni time-dependent.

Abstract

The Harmonic Balance (HB) method is a mathematical technique to solve nonlinear differential equations with periodic solutions.

In this work we apply the HB method both to nonlinear electrical circuit and field problems.

After a first more theoretical part dedicated to the mathematical formulation of the method, we will apply the HB analysis for the study of nonlinear circuits. We will present the various techniques and we will implement and test the relative code in a specific example.

The main part of this thesis consists in the study of the HB method applied to nonlinear electromagnetic fields, therefore we will treat in detail the so called Harmonic Balance Finite Element Method (HBFEM). We will derive the HBFEM in order to understand how it is possible to set up a code which implements this technique.

The originality of this work is related to the choice of integrating the HB analysis in an existing FEM commercial solver in respect of what has already been done with research codes based on FEM. We will show how it is possible to implement in the commercial software COMSOL the HB method via MATLAB in a simple way using as much as possible the standard routines of the chosen FEM solver. We will test our method on some significant examples in order to show the accuracy and the performance of the code compared with the standard time-dependent calculations.

Contents

Chapter 1

The Harmonic Balance Method

The Harmonic Balance (HB) method is a technique to determine the steady-state solution of nonlinear problems.

It was firstly introduced to study nonlinear differential equation, in this sense the works for example of Cesari [1] and Urabe [2] show the mathematical derivation of the method focusing on its convergency properties and presenting it as a Galerkin's approximation tecnique using Fourier basis and Fourier test functions.

Moreover the Harmonic Balance method gained an important role in various fields of engineering. A relevant application is the simulation RF and microwave circuits which exhibit strong nonlinear characteristics. In the same way the HB can be used to study power electronics devices such as rectifiers and switching converters, moreover it has also gained an important role in the analysis of large scale systems such as HVDC and renewable energy installations.

The Harmonic Balance was then extended and combined with Finite Element analysis to study nonlinear electromagnetic fields problems, giving rise to the so called Harmonic Balance Finite Element method (HBFEM). The articles of Yamada and Bessho [3] are the first that present this technique.

Speaking generally, the HB method is a (hybrid) frequency-domain method which combines time and frequency representations in such a way to achieve an advantage in comparison with the classical time-consuming transient analysis. Infact if only the steady-state solution is of interest, the HB method can determine it directly while a transient solver has to explicitly solve several time-stepping computations that could become really expensive in particular with large time costants.

During the years several slightly different formulations of the HB method have been developed ([4]-[5]-[6]), mostly to reduce the main drawback of such a method. The main problem of the HB is the size and the complexity of the system of equation that arises from its application. This fact is crucial for the HBFEM and a careful implementation has to be carried out to improve the simulation speed.

In this sense Bìrò and Preis [7] proposed a novel approach wich consists in solving the system which derives from the application of the HBFEM using a fixed point iteration technique to decouple each harmonic, hence each harmonic can be solved in parallel.

The key concept of the HB is to find the solution of the problem using a frequency representation obtained with a truncated Fourier series expansion or other algorithms like the Fast Fourier Transform. The number of harmonics that has to be taken into account is not always easy to determine, more harmonics are considered and more accurate will be the solution but at the same time the computational cost will increase.

The HB method has been applied to several different problems. As mentioned before it is widely used to analyse nonlinear circuits, the HBFEM instead has been used to solve eddy current problems and to analyse electrical machines such as transformers, electrical motors and generators and other electromagnetic devices.

1.1 Mathematical Framework

In order to introduce the HB method let's consider the first order nonlinear differential equation:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \dot{x} = f(x(t)) \tag{1.1}$$

where x and $f(x,t)$ are vectors of the same dimension and $f(x(t))$ is periodic in t of period 2π. Assuming that the solution $x(t)$ admits a Fourier series representation:

$$x(t) = \alpha_0 + \sum_{n=1}^{\infty} \alpha_n \sin nt + \beta_n \cos nt \tag{1.2}$$

the HB provides a strategy to find an approximate solution $\tilde{x}(t)$ considering a limited number of harmonic. The solution can then be expressed as a trigonometric polynomial of order N:

$$\tilde{x}(t) = a_0 + \sum_{n=1}^{N} a_n \sin nt + b_n \cos nt \tag{1.3}$$

Let's now introduce the definition of residual:

$$r(x(t)) = \dot{x} - f(x(t)) \tag{1.4}$$

Clearly if $\varepsilon(x(t)) = 0$, $\dot{x}$ is the exact solution of the problem. Then assuming a periodic solution $x(t)$ to exists (Eq.1.2), a resonable way of obtaining a good approximation is to assume a correct functional form for $\tilde{x}(t)$ (Eq.1.3):

$$\tilde{x} = \tilde{x}(t, a_0, a_1, b_1, ..., a_N, b_N) \tag{1.5}$$

and then choose the unknown coefficients in Eq.1.5 to minimize the residual. This procedure is a Galerkin's approximation technique using trigonometric polynomials for the test and trial functions. The class of Galerkin's methods will be analysed in detail in the next chapters.

The coefficients can be found imposing the ortogonalization of the residuals, so that defining the scalar product for two generic functions as:

$$\langle u, v \rangle = \frac{1}{T} \int_0^T u(t)v(t)\, dt \tag{1.6}$$

the resultant system that has to be solved becomes:

$$\begin{cases} \langle r(\tilde{x}), 1 \rangle = 0 \\ \langle r(\tilde{x}), \cos(t) \rangle = 0 \\ \langle r(\tilde{x}), \sin(t) \rangle = 0 \\ \qquad \vdots \\ \langle r(\tilde{x}), \cos(Nt) \rangle = 0 \\ \langle r(\tilde{x}), \sin(Nt) \rangle = 0 \end{cases} \tag{1.7}$$

The functions $\{1, \cos(t), \sin(t), \ldots, \cos(Nt), \sin(Nt)\}$ are linearly indepent and form a basis for the test space.

Let's now recast the residual in such a way to write explicitely system 1.7. From Eq.1.3 we have:

$$\dot{\tilde{x}}(t) = \sum_{n=1}^{N} (n a_n \cos(nt) - n b_n \sin(nt)) \tag{1.8}$$

For the sake of conciseness the following notation for the Fourier coefficients of $f(\tilde{x}, t)$ will be used:

$$\{f(\tilde{x}, t)\}_{nc} = \frac{1}{\pi} \int_0^{2\pi} f(\tilde{x}, s) \cos(ns)\, ds \tag{1.9}$$

$$\{f(\tilde{x},t)\}_{ns} = \frac{1}{\pi}\int_0^{2\pi} f(\tilde{x},s)\sin(ns)\,ds \tag{1.10}$$

Note that although $\tilde{x}$ has no harmonics of order greater than N, the function $f(\tilde{x},t)$ in general possesses harmonics of all order. Therefore $f(\tilde{x},t)$ is written as:

$$f(\tilde{x},t) = \sum_{n=0}^{\infty}\left(\{f(\tilde{x},t)\}_{nc}\cos(nt)\,dt + \{f(\tilde{x},t)\}_{ns}\sin(nt)\,dt\right) \tag{1.11}$$

Finally the residual can be rewritten:

$$\begin{aligned} r &= \dot{\tilde{x}} - f(\tilde{x}) \\ &= \sum_{n=1}^{N}(na_n\cos(nt) - nb_n\sin(nt)) - \sum_{n=0}^{\infty}\left(\{f(\tilde{x},t)\}_{nc}\cos(nt) + \{f(\tilde{x},t)\}_{ns}\sin(nt)\right) \\ &= \sum_{n=0}^{N}[(na_n - \{f(\tilde{x},t)\}_{nc})\cos(nt) + (-nb_n - \{f(\tilde{x},t)\}_{ns})\sin(nt)] \\ &+ \sum_{n=N+1}^{\infty}\left(\{f(\tilde{x},t)\}_{nc}\cos(nt) + \{f(\tilde{x},t)\}_{ns}\sin(nt)\right) \end{aligned} \tag{1.12}$$

Then to minimize the residual it's necessary to determine the unknowns $\{a_0, a_1, b_1, \ldots, a_N, b_N\}$ such that:

$$\begin{cases} \{f(\tilde{x},t)\}_0 = 0 \\ na_n - \{f(\tilde{x},t)\}_{nc} = 0 \\ -nb_n - \{f(\tilde{x},t)\}_{ns} = 0 \end{cases} \tag{1.13}$$

System 1.13, which is equal to 1.7, is a nonlinear system of $(2N+1)$ equations in $(2N+1)$ unknowns thas has to be solve numerically, its roots are the estimate of the exact Fourier coefficient of the solution. It's clear that the computational effort increases rapidly with increasing the number of harmonics, at the same time to achieve an accurate solution N must be sufficiently high. Moreover the term $\sum_{n=N+1}^{\infty}(\{f(\tilde{x},t)\}_{nc}\cos(nt)+\{f(\tilde{x},t)\}_{ns}\sin(nt))$ in Eq.1.12 vanishes in the theoretical case of considering an infinite number of harmonics for the approximate solution.

In order to clarify the previous mathematical derivation and make some other considerations we analyze the following example.

Example 1. We want to solve the equation:

$$m\ddot{y} + c\dot{y} + k_1 y + k_3 y^3 = A\cos(\omega t) \tag{1.14}$$

using the HB method.

This nonlinear differential equation is known as Duffing equation and it is used to model certain oscillators for example electrical circuits or mechanical pendulums. The coefficients m, c, k_1 and k_3 are the parameters of the mass, damping and stiffness of the system respectively, A and ω are the external excitation force amplitude and frequency of the oscillator.

In accordance with the HB method, the solution of a nonlinear system is assumed to be of the form of a truncated Fourier series:

$$y(t) = a_0 + \sum_{n=1}^{N} a_n\cos n\omega t + b_n\sin n\omega t \tag{1.15}$$

The Fourier expansions of the first and second derivatives of the solution are:

$$\dot{y}(t) = \sum_{n=1}^{N} n\omega(a_n\cos n\omega t + b_n\sin n\omega t) \tag{1.16}$$

$$\ddot{y}(t) = \sum_{n=1}^{N} -n^2\omega^2(a_n \cos n\omega t + b_n \sin n\omega t) \tag{1.17}$$

In this particular example even the Fourier expansion of a cubic term is needed, we will denote it as:

$$y^3(t) = \hat{a}_0 + \sum_{n=1}^{N} \hat{a}_n \cos n\omega t + \hat{b}_n \sin n\omega t \tag{1.18}$$

where:

$$\hat{a}_0 = \frac{1}{T}\int_0^T \left(a_0 + \sum_{n=1}^{N} a_n \cos n\omega t + b_n \sin n\omega t\right)^3 dt \tag{1.19}$$

$$\hat{a}_n = \frac{1}{T}\int_0^T \left(a_0 + \sum_{n=1}^{N} a_n \cos n\omega t + b_n \sin n\omega t\right)^3 \cos(n\omega t)\, dt \tag{1.20}$$

$$\hat{b}_n = \frac{1}{T}\int_0^T \left(a_0 + \sum_{n=1}^{N} a_n \cos n\omega t + b_n \sin n\omega t\right)^3 \sin(n\omega t)\, dt \tag{1.21}$$

Substituting Eqs.1.15-1.21 into Eq.1.14 and equating coefficients associated with each harmonic yields:

$$\begin{cases} k_1 a_0 + k_3 \hat{a}_0 = 0 \\ -m\omega^2 a_1 + c\omega b_1 + k_1 a_1 + k_3 \hat{a}_1 = A \\ -m\omega^2 b_1 - c\omega a_1 + k_1 b_1 + k_3 \hat{b}_1 = 0 \\ \qquad \vdots \\ -mN^2\omega^2 a_N + cN\omega b_N + k_1 a_N + k_3 \hat{a}_N = 0 \\ -mN^2\omega^2 b_N - cN\omega a_N + k_1 b_N + k_3 \hat{b}_N = 0 \end{cases} \tag{1.22}$$

which is a nonlinear system of $(2N+1)$ equations in $(2N+1)$ unknowns.

Solving system 1.22 requires the analytical expressions for the nonlinear functions $\hat{a}_0$, $\hat{a}_n$, $\hat{b}_n$ in terms of a_0, a_N and b_N. When using only the fundamental harmonic component, system 1.22 is simplified and leads to:

$$\begin{cases} -m\omega^2 a_1 + c\omega b_1 + k_1 a_1 + k_3(\frac{3}{4}a_1^3 + \frac{3}{4}a_1 b_1^2) = A \\ -m\omega^2 b_1 - c\omega a_1 + k_1 b_1 + k_3(\frac{3}{4}b_1 a_1^2 + \frac{3}{4}b_1^3) = 0 \end{cases} \tag{1.23}$$

which is a nonlinear system of two equations in two unknowns that can be easilly solved numerically. However, if the nonlinearity of the equation is strong, the high-order harmonic components may have a considerable effect and can contribute significantly to the whole solution and consequently, a truncated Fourier series expansion may make the solution less accurate. On the other hand, if more harmonic components are considered for the analysis, then system 1.22 can become really complex to solve even with numerical techniques.

At the moment we have applied the HB method without considering the property of the nonlinear differential equations, now we want to give some conditions in order to garantee the existance of the periodic solution of a differential equation. Moreover we can make some considerations regarding the existance of the Galerkin approximation of the solution and give an estimation of the error we make applying the HB method considering different numbers of harmonic.

In this sense, let's consider a given real periodic system of differential equations:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = X(x,t) \tag{1.24}$$

where x and $X(x,t)$ are vectors of the same dimension, and $X(x,t)$ is periodic in t of period 2π. It can be proved the following theorem:

Theorem 1. *If there is an isolated periodic solution $x = x(t)$ of Eq.1.24 lying inside a closed bounded region of the x-space D, then there exist Galerkin approximations $x = \bar{x}(t)$ to any order $m \geq m_0$, lying in D provided m_0 is sufficiently large. Such Galerkin approximations $x = \bar{x}_m, (t)$ converges uniformly as $m \to \infty$ to the exact solution $x = \hat{x}(t)$ together with their first order derivatives.*

The above theorem asserts that, if there is an isolated periodic solution lying inside D, then there always exist Galerkin approximations of all orders sufficiently high with errors as small as we desire. Now let us suppose we have obtained numerically a Galerkin approximation $\bar{x}_m(t)$. We want to verify the existance of an exact periodic solution related to the Galerkin procedure and estimate the error of the approximated solution. For this purpose we state the following theorem:

Theorem 2. *Assume that Eq.1.24 has a periodic approximate solution $x = \bar{x}(t)$ lying inside D and consider a continuous periodic matrix $A(t)$ such that the multipliers of the linear homogeneous system:*

$$\frac{\mathrm{d}y}{\mathrm{d}t} = A(t)y \tag{1.25}$$

are all different from one. Let $\Phi(t)$ be the fundamental matrix of Eq.1.25 such that $\Phi(0) = I$ (I is the unit matrix) and $H(t,s) = H_{kl}(t,s)$ be a piecewise continuous matrix such that:

$$H(t,s) = \begin{cases} \Phi(t)[I - \Phi(2\pi)]^{-1}\Phi^{-1}(s), & 0 \leqslant s \leqslant t \leqslant 2\pi \\ \Phi(t)[I - \Phi(2\pi)]^{-1}\Phi(2\pi)\Phi^{-1}(s), & 0 \leqslant s \leqslant t \leqslant 2\pi \end{cases} \tag{1.26}$$

Let M be a positive constant such that:

$$\left(2\pi \max_{0\leq t\leq 2\pi} \int_0^{2\pi} \sum_{k,l} H_{kl}^2(t,s)\, ds\right)^{\frac{1}{2}} \leq M \tag{1.27}$$

and r be a nonnegative constant such that:

$$\left\| \frac{\mathrm{d}\bar{x}(t)}{\mathrm{d}t} - X[\bar{x}(t), t] \right\| \leq r \tag{1.28}$$

where the symbol $||\cdot||$ denotes the Euclidean norm. Then, if there are a positive constant δ and a non-negative constant $k < 1$ such that:

(i) $D_\delta = \{x : ||x - \bar{x}|| \leqslant \delta, t \in L\} \subset D$

(ii) $||J(x,t) - A(t)|| \leq \frac{k}{M}, \forall x : ||x - \bar{x}(t)|| \leqslant \delta, t \in L$

(iii) $\frac{Mr}{1-k} \leqslant \delta$

where $J(x,t)$ is the Jacobian matrix of $X(x,t)$ with respect to x, then the given system 1.24 has one and only one exact solution $x = \hat{x}(t)$ in D_δ and this is an isolated periodic solution. Further, for $x = \hat{x}(t)$, we have:

$$||\bar{x}(t) - \hat{x}(t)|| \leq \frac{Mr}{1-k} \tag{1.29}$$

In Theorem 2 let us suppose $\bar{x}(t)$ is a certain computed Galerkin approximation of order m, we denote it as $\bar{x}_m(t)$. Let us take $J[\bar{x}(t), t]$ as $A(t)$. Then by numerical integration of Eq.1.25, we can find the value of M and by numerical computation of the Fourier coefficients of $\frac{\mathrm{d}\bar{x}(t)}{\mathrm{d}t} - X[\bar{x}(t), t]$ we can find the value of r. Then we can easily check the existence of the constants δ and k satisfying conditions (i)-(ii)-(iii). If there are such constants, then, by Theorem 2, we can assert the existence of a periodic exact solution, and by Eq.1.29 we have an error bound of the computed Galerkin approximation. If the constants δ and k satisfying (i)-(ii)-(iii) do not exist, then the order of the Galerkin approximation has to be raised. Such a procedure always ends at a certain finite order in case the given system has

an isolated periodic solution lying inside D. In other words, in such a case there always exists a Galerkin approximation for which the conditions of Theorem 2 are fulfilled, and therefore, the existence of an exact isolated periodic solution can be asserted. The formal demonstration of the previous theorems can be found in [2].

Now we want to apply the obtained results in a numerical example.

Example 2. We want to solve the equation:

$$\ddot{x} + x^3 = \sin(t) \tag{1.30}$$

In order to obtain an accurate result we adopt an high order approximation considering 15 harmonics. Numerical calculation leads to:

$$\begin{aligned} \bar{x}(t) = {} & 1.431189037\sin(t) - 0.126915530\sin(3t) \\ & + 0.009754734\sin(5t) - 0.000763601\sin(7t) \\ & + 0.000059845\sin(9t) - 0.000004691\sin(11t) \\ & + 0.000000368\sin(13t) - 0.000000029\sin(15t) \end{aligned} \tag{1.31}$$

We note the rapid convergence of the Fourier series. The solution obtained using the HB method is compared with the one computed using the MATLAB function `ode45` which implements an explicit Runge-Kutta formula, the steady state solutions are depicted in Fig. 1.1.

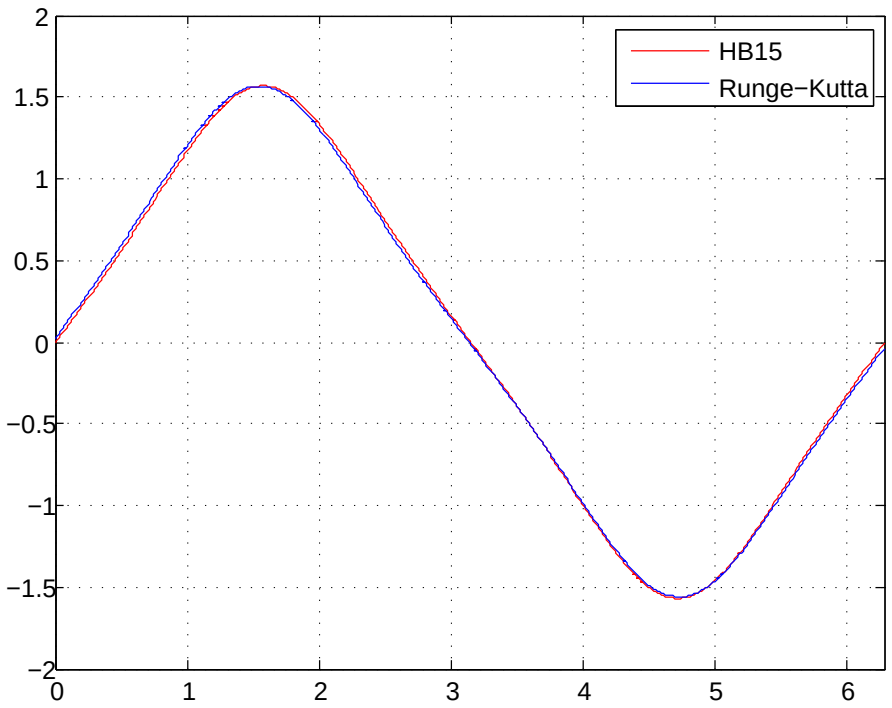


Figure 1.1: Steady state solution computed using Runge-Kutta and HB

Then we can compute r and M, they respectively result equal to:

$$r = 713\text{e}{-9} \tag{1.32}$$

$$M = 11.4107 \tag{1.33}$$

Now, by means of Theorem 2, let us check the existence of an exact isolated periodic solution $x = \hat{x}(t)$ and seek the error bound of the Galerkin approximation $x = \bar{x}(t)$ of Eq.1.31. Eq.1.30 is rewritten in the form of a first order system as follows:

$$\begin{cases} \dot{x} = y \\ \dot{y} = -x^3 + \sin(t) \end{cases} \tag{1.34}$$

The Jacobian matrix of the right hand side is:

$$J(x) = \begin{bmatrix} 0 & 1 \\ -3x^2 & 0 \end{bmatrix} \tag{1.35}$$

Let us consider the region:

$$D_\delta = \{x : |x - \bar{x}(t)| \leq \delta\} \tag{1.36}$$

Then from Eq.1.35, for any $x \in D_\delta$:

$$||J[x] - J[\bar{x}(t)]|| \leq 3|x^2 - \bar{x}^2(t)| \leq 3\delta(\delta + 2|\bar{x}(t)|) \tag{1.37}$$

and consequently, from Eq.1.31, $|x - \bar{x}(t)| \leq \delta$ implies:

$$||J[x] - J[\bar{x}(t)]|| \leq 3\delta(\delta + 3.137375670) \tag{1.38}$$

Then we have to seek δ and $k < 1$ such that:

$$3\delta(\delta + 3.137375670) \leq \frac{k}{11.42} \tag{1.39}$$

$$\frac{11.42 \cdot 72\text{e}{-8}}{1-k} = \frac{822.24\text{e}{-8}}{1-k} \leq \delta \tag{1.40}$$

Hence, let us suppose:

$$\delta \leq 1\text{e}{-5} \tag{1.41}$$

Eq.1.40 can be replaced by the stronger inequality:

$$\delta \leq \frac{k}{3 \cdot 3.1374 \cdot 11.42} = \frac{k}{107.487324} \tag{1.42}$$

Then from Eq.1.40 and Eq.1.42, we have:

$$\frac{822.24\text{e}{-8}}{1-k} \leq \delta \leq \frac{k}{107.487324} \tag{1.43}$$

Now let us consider the inequality:

$$822.24\text{e}{-8} \cdot 107.487324 = 8.8380377\text{e}{-4} \leq k(1-k) \tag{1.44}$$

This results satisfied for $k = 1\text{e}{-3}$. Anyway for this value of k we have:

$$\frac{822.24\text{e}{-8}}{1-k} = 823.06306\text{e}{-8} \tag{1.45}$$

$$\frac{k}{107.487324} = 9.303\text{e}{-6} \tag{1.46}$$

Consequently, taking Eq.1.41 into consideration, we see that we have only to choose δ so that:

$$8.231\text{e}{-6} \leq \delta \leq 9.303\text{e}{-6} \tag{1.47}$$

This shows that there are indeed positive constants δ and $k < 1$ satisfying Eq.1.39-1.40.

From this result, by Theorem 2, we see that the given equation 1.30 which is equivalent to Eq.1.34 has one and only one exact isolated periodic solution $x = \hat{x}(t)$ in the region D_δ. Furthermore we see that:

$$|\bar{x}(t) - \hat{x}(t)| \leq 823.06306\text{e}{-8} \tag{1.48}$$

This gives the error bound of the Galerkin approximation $x = \bar{x}(t)$ of Eq.1.31.

Chapter 2

Harmonic Balance Analysis of Nonlinear Circuits

This chapter deals with the analysis of nonlinear circuit using the Harmonic Balance method. Harmonic balance analysis is applicable to a wide variety of microwave and RF problems which exhibit strong nonlinear properties to obtain the steady state solution of the circuit. Moreover Transient analysis was formulated before harmonic balance methods. Thus, the existence of harmonic-balance analysis implies that transient methods are not adequate for many kinds of circuits. In fact, the methods are often complementary: harmonic balance works well where transient analysis does not, and transient analysis usually outperforms harmonic balance in the kinds of problems where it is applicable. Clearly nonlinear circuits which exhibit slow dynamic and therefore long transient regimes suite well for the application of the Harmonic Balance method.

There are two main formulation of the same method namely the Nodal Harmonic Balance method and the Piecewise Harmonic Balance method. The first one considers an entire circuit applying the Kirchhoff's laws in every node giving rise to a large and in general full nonlinear system. In this case each variable is represented by a Fourier series expansion whose coefficients are the unknowns of the problem. It's a relatively easy approach but due to the high number of variables is not more widely use.

The Piecewise Harmonic Balance method instead has emerged as the more convenient approach and will be analysed in detail. Nowadays, due to its diffusion, the Piecewise Harmonic Balance method is simply identified as Harmonic Balance and so will be done in this work.

2.1 Formulation of the Method

To introduce the Harmonic Balance method let's consider only single-tone circuits, ones having periodic excitations at a single fundamental frequency. This includes sinusoidal excitations and even periodic nonsinusoidal excitations as long as they can be expressed by a Fourier series. In later sections, we will show how Harmonic Balance analysis can be applied to circuits having more complex excitations.

The Piecewise Harmonic Balance method, or simply Harmonic Balance method, consists in dividing the circuit that has to be analysed in a linear and in a nonlinear subsystem connected by M ports (Fig. 2.1). The criterium for the subdivision of the circuit is based on two main concepts. First of all assuming the minimum number of nonlinear ports and therefore of unknown reduces the numerical complexity of the problem, on the contrary augmenting the number of them greatly simplifies the description of the nonlinear subsystem.

The idea of harmonic balance is to find a set of port voltage that gives the same currents in both the linear-network equations and the nonlinear-network equations, these currents has to satisfy Kirchoff's current law at each port. When that set is found, it must be the solution of the circuit.

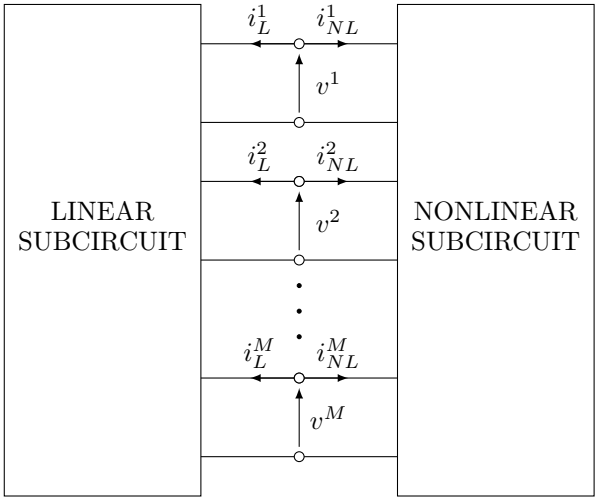


Figure 2.1: General nonlinear circuit partitioned into linear and nonlinear subcircuit

The voltage of the generic port m can be expressed with the Fourier series expansion:

$$v^m = \mathrm{Re}\left(\sum_{n=0}^{N_H} V_n^m \cdot e^{jn\omega_0 t}\right) \tag{2.1}$$

where the phasors V_n^m are the unknowns of the problem. The unknows can be riarranged in the vector:

$$\mathbf{V} = \begin{bmatrix} V_0^1 & V_1^1 & \dots & V_{N_H}^1 & V_0^2 & \dots & V_{N_H}^2 & V_0^M & \dots & V_{N_H}^M \end{bmatrix}^T \tag{2.2}$$

Note that $\mathbf{V}$ is a vector of $M(N_H+1)$ componenents. Applying the Kirchhoff's current law at each port it must be verified:

$$i_L^m(t) + i_{NL}^m(t) = 0 \qquad m = 1, ..., M \tag{2.3}$$

where $i_L^m(t)$ indicates the generic linear current i at port m while $i_{NL}^m(t)$ represents the nonlinear current. The same equation has to be true for each harmonic, so that passing in the frequency domain and then defining the vectors $\mathbf{I}_L$ and $\mathbf{I}_{NL}$ similarly to what has been done with $\mathbf{V}$ it's possible to write:

$$\mathbf{I}_L + \mathbf{I}_{NL} = \mathbf{0} \tag{2.4}$$

Since the current and voltage signals are real all the vectors previously defined in frequency domain present hermitian symmetry, thus the negative frequency components are the complex conjugate of the positive ones. This observation permits to take into account only positive frequencies plus the DC term, thus the complexity of the problem is greatly reduced.

In order to find the solution, the two subcircuits have to be analysed separetely. In particular the linear subsystem is analysed in the frequency domain using the superposition principle while the nonlinear one is studied in the time domain. Then the two analysed must be combined in order to minimize the error due to the balancing of each harmonic.

The linear circuit can be replaced by its Norton equivalent at each port (Fig. 2.2), leading to the relation:

$$\mathbf{I}_L = \mathbf{Y} \cdot \mathbf{V} + \mathbf{I}_{eq} \tag{2.5}$$

where $\mathbf{I}_{eq}$ is the vector of the equivalent currents and $\mathbf{Y}$ is the admittance matrix. The equivalent current vector similarly to what has been done for the voltages is expressed as:

$$\mathbf{I}_{eq} = \begin{bmatrix} I_{eq,0}^1 & I_{eq,1}^1 & \dots & I_{eq,N_H}^1 & I_{eq,0}^2 & \dots & I_{eq,N_H}^2 & I_{eq,0}^M & \dots & I_{eq,N_H}^M \end{bmatrix}^T \tag{2.6}$$

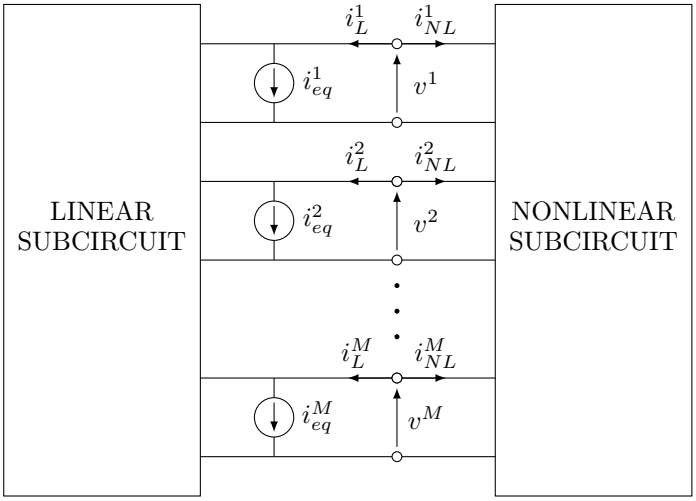


Figure 2.2: Norton equivalent of the linear subcircuit

The admittance matrix is a sparse and block matrix, each submatrix is a diagonal whose elements are the value $Y^{i,j}$ between port i and port j at each harmonic. Therefore defining $\mathbf{Y}$ as:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}^{1,1} & \mathbf{Y}^{1,2} & \dots & \mathbf{Y}^{1,M} \\ \mathbf{Y}^{2,1} & \mathbf{Y}^{2,2} & \dots & \mathbf{Y}^{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{Y}^{M,1} & \mathbf{Y}^{M,2} & \dots & \mathbf{Y}^{M,M} \end{bmatrix} \tag{2.7}$$

each submatrix can be expressed:

$$\mathbf{Y}^{i,j} = \begin{bmatrix} Y_0^{i,j}(0) & 0 & \dots & 0 \\ 0 & Y_1^{i,j}(\omega_0) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Y_{N_H}^{i,j}(N_H \cdot \omega_0) \end{bmatrix} \tag{2.8}$$

Even the vector $\mathbf{I}_{eq}$ will be sparse with non zero terms only for the fundamental frequency. The elements of the admittance matrix $\mathbf{Y}$ can be evaluated using the definition of self and mutual admittance:

$$Y_{i,i} = \left.\frac{i_i}{v_i}\right|_{v_n=0,\forall n\neq i} \tag{2.9}$$

$$Y_{i,j} = \left.\frac{i_j}{v_i}\right|_{v_n=0,\forall n\neq i} \tag{2.10}$$

The self and mutual admittances at each port have to be calculated for each frequency solving different circuits, for this purpose methods like Nodal Analysis or Tableau Analysis can be used.

The calculation of the non linear current follows a different approach. First of all, the vector of the time-domain voltages at all ports is computed from the voltage phasors by means of an inverse Fourier transform:

$$\mathbf{v}(t) = \mathcal{F}^{-1}(\mathbf{V}) \tag{2.11}$$

Then, knowing the component equations $\mathbf{i}_{NL}(t) = f(\mathbf{v}(t))$ at each port, the phasors of the currents flowing into the nonlinear subcircuit are computed using the Fourier transform:

$$\mathbf{I}_{NL} = \mathcal{F}(\mathbf{i}_{NL}(t)) \tag{2.12}$$

Therefore the procedure is:

$$\mathbf{V} \xrightarrow{\mathcal{F}^{-1}} \mathbf{v}(t) \xrightarrow{f(v(t))} \mathbf{i}_{NL}(t) \xrightarrow{\mathcal{F}} \mathbf{I}_{NL} \tag{2.13}$$

We underline that the previous operation has to be done using the Discrete Fourier Transform or the FFT algorithm, therefore the number of samples must satisfy the hypothesis of the Shannon's sampling theorem in order to avoid aliasing or any other numerical error. In particular for the FFT the number of time samples must be twice the number of frequency components. Moreover when strong nonlinearities are involved oversampling is advisable.

Finally the problem consists in solving the nonlinear system:

$$\mathbf{F}(\mathbf{V}) = \mathbf{I}_L(\mathbf{V}) + \mathbf{I}_{NL}(\mathbf{V}) = \mathbf{0} \tag{2.14}$$

The unknowns of the system are the voltages, or more exactly the phasors of the truncated Fourier series expansions of the voltages at the ports connecting the linear and the nonlinear subcircuits. Anyway, once the number of harmonics has been chosen, the number of unknowns is fixed and doesn't depend on the number of nonlinear components effectively present in the circuit.

Eq. 2.14 represents a test to determine whether a trial set of port voltage components is the correct one; that is, if $\mathbf{F}(\mathbf{V}) = \mathbf{0}$, then $\mathbf{V}$ is a valid solution.

It also represents an equation that can be solved to obtain the portvoltage vector, $\mathbf{V}$. The values of the phasors are found by an iterative numerical algorithm, given the nonlinearity of the equations. $\mathbf{F}(\mathbf{V})$, called the current-error vector, represents the difference between the current calculated from the linear and nonlinear subcircuits, at each port and at each harmonic, for a trial-solution vector $\mathbf{V}$.

A generic circuit with M ports and N_H+1 harmonics (N_H armonics+DC term) gives rise to a nonlinear system with $M(N_H + 1)$ variables; considering the real and imaginary parts the total number of unknowns grows to $2M(N_H + 1)$. A number of algorithms have been proposed for solving this problem but the most common choice is to use Newthon-Raphson's method, which is described by the following recurrence relation:

$$\mathbf{V}^{(k+1)} = \mathbf{V}^{(k)} - (\mathbf{J}^{(k)})^{-1} \cdot \mathbf{F}(\mathbf{V}^{(k)}) \tag{2.15}$$

where $\mathbf{J}$ is the Jacobian matrix computed as follows:

$$\mathbf{J} = \left.\frac{\mathrm{d}\mathbf{F}(\mathbf{V})}{\mathrm{d}\mathbf{V}}\right|_{\mathbf{V}=\mathbf{V}^{(k)}} \tag{2.16}$$

The structure of the Jacobian is:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial F_1^0}{\partial V_1^0} & \frac{\partial F_1^0}{\partial V_1^1} & \cdots & \frac{\partial F_1^0}{\partial V_N^{NH}} \\ \frac{\partial F_1^1}{\partial V_1^0} & \frac{\partial F_1^1}{\partial V_1^1} & \cdots & \frac{\partial F_1^1}{\partial V_N^{NH}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_N^{NH}}{\partial V_1^0} & \frac{\partial F_N^{NH}}{\partial V_1^1} & \cdots & \frac{\partial F_N^{NH}}{\partial V_N^{NH}} \end{bmatrix} \tag{2.17}$$

where the subscripts denote the ports while superscipts the harmonic order. Moreover it can also be seen that the Jacobian matrix can be calculated as:

$$\mathbf{J} = \mathbf{Y} + \frac{\mathrm{d}\mathbf{I}_{NL}}{\mathrm{d}\mathbf{V}} \tag{2.18}$$

The Jacobian contains the derivatives of all the components of the error vector with respect to the components of $\mathbf{V}$. Thus, it contains information on the sensitivity of changes in every component of $\mathbf{F}$ resulting from changes in any component of $\mathbf{V}$. The Jacobian matrix can be computed analytically, if the nonlinear function is known in analytical form, or numerically

by incremental ratio. The analytical derivation, however, has better numerical properties, and it is advisable when available.

The algorithm will hopefully converge towards the correct solution, and will be stopped when the error decreases below a limit value. The error is actually the vector of the error currents, real and imaginary parts, at each node and for each harmonic frequency; convergence will be assumed to be reached when its norm will be lower than a desired accuracy level:

$$\|\mathbf{F}^{(k)}\| < \varepsilon \tag{2.19}$$

Alternatively, the algorithm is stopped when the solution does not vary any more:

$$\|\mathbf{V}^{(k+1)} - \mathbf{V}^{(k)}\| < \delta \tag{2.20}$$

The values of ε and δ rispectively depend on the current and voltage levels in the circuit.

A critical point in the algorithm is the choice of the first guess. A well chosen first guess will considerably ease the convergence of the algorithm to the correct solution. If the circuit is mildly nonlinear, the linear solution, obtained for a low-level input, will probably be a good starting point and it is usally used as a first guess. If the circuit presents strong nonlinearities, a continuation method will probably be the best choice. With this approach, the input level is increased stepwise, using the result of the previous step as a first guess. This method is often called Source Stepping method. In this case the Newton-Raphson iteration becomes:

$$\mathbf{V}^{(k+1)} = \mathbf{V}^{(k)} - \mathbf{h}(\mathbf{V}^{(k)})(\mathbf{J}^{(k)})^{-1} \cdot \mathbf{F}(\mathbf{V}^{(k)}) \tag{2.21}$$

Another issue is related to the number of harmonics (N_H) that has to be taken into account. Selecting N_H too small results in poor accuracy, and often poor convergence; conversely, selecting N_H larger increases the accuracy but slows the solution process and increases the use of computer memory. Therefore the choice of N_H is not straightforward and must be evaluated on case by case basis.

The final algorithm is summarized in the flowchart of Fig. 2.3.

The main computational effort is the construction of the Jacobian and the solution of the related linear system in each Newton-Raphson iteration. In this sense Quasi-Newton methods can be used. With these approaches the Jacobian $\mathbf{J}_k$ is approximated using the matrices $\mathbf{J}_1, \ldots, \mathbf{J}_{k-1}$, in such a way to make the linearized system easy to be solved. Another approach consists in calculating the Jacobian only after a specific number of iterations, if the error does not decrease or decreases to slowly the exact Jacobian is updated.

The described formulation is based on Kirchhoff's voltage law where the unknown is the voltage and the circuit elements are described as admittances. Moreover it can happen in some cases that certain elements do not have an admittance representation and cannot be used directly to form the admittance matrix. One common solution is to approximate the unrealizable element by realizable ones, the way to circumvent the problem is explained in [8].

Alternatively, Kirchhoff's current law can be used to formulate the method, with the current being the unknown, and the circuit elements described as impedances. While no problem usually arises for the linear elements, it can be seen that the nonlinear elements are usually voltage-controlled nonlinear conductances or capacitances. However, any alternative form of Kirchhoff's equations is allowed as a basis for the harmonic balance algorithm in the cases in which the nonlinear elements have a different representation.

Finally to conclude, we underline that there are several other methods used for the analysis of nonlinear circuits which are not discussed in this work, for example the ones based on the Volterra series expansion. An overview of these other techniques can be found in [8].

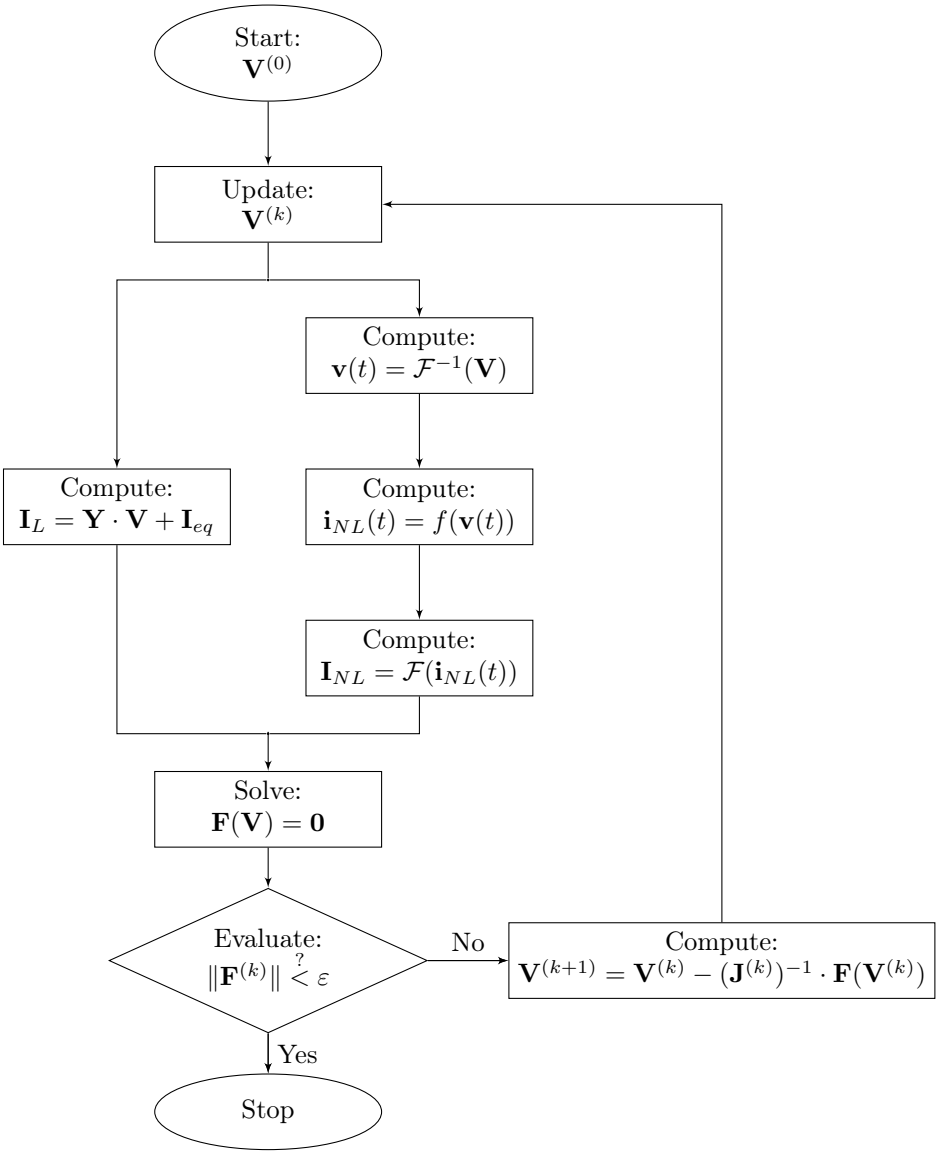


Figure 2.3: Flowchart of the algorithm

2.2 Multi-tone Harmonic Balance Analysis

At the moment we have only considered single-tone circuits, ones having periodic excitations at a single fundamental frequency. Now we want to extend the HB analysis to nonlinear circuits excited by several noncommensurate sources. The concept of harmonic-balance analysis is illustrated by Fig.2.1, which shows a nonlinear circuit partitioned into linear and nonlinear subcircuits. In the case of single-tone excitation, the voltages and currents are periodic, and thus have a fundamental-frequency component and a number of harmonics. In the case that the frequency components are not harmonically related the circuit partitioning remains equally valid but requires a slightly different formulation to take into account noncommensurate frequencies.

We now consider the case where the excitation may have two or more noncommensurate frequencies, and the frequency components of the currents and voltages are no longer harmonically related. We restrict our discussion to the two-tone case, the extension to greater numbers of excitations is straightforward, however the computational burden increases quickly, usually limiting the effective analysis capabilities to no more than three tones.

First of all a more complex Fourier series for the signals is needed, composed of two tones:

$$v(t) = \sum_{n_1=-\infty}^{+\infty} \sum_{n_2=-\infty}^{+\infty} V_{n_1,n_2} e^{(jn_1\omega_1 + jn_2\omega_2)t} \tag{2.22}$$

If the two basic frequencies ω_1 and ω_2 are noncommensurate, the signal is said to be quasi-periodic. The unknowns of the problem are still the phasors of the voltage, but now they are not relative to the harmonics of a periodic signal, they rather represent a complex spectrum. The series in Eq.2.22 must be truncated so that only important terms are retained, a proper choice increases the accuracy of the analysis while limiting the numerical effort. The expansion of a single-tone signal is truncated so that the neglected harmonics have negligible amplitude. The same principle holds for a multi-tone analysis. The frequency spectrum includes all the frequencies that are a linear combinations of the two basic frequencies:

$$\omega_{n_1,n_2} = n_1\omega_1 + n_2\omega_2 \tag{2.23}$$

where ω_1 and ω_2 are the frequencies of the two excitations. The sum of the absolute values of the two indices $n = |n_1| + |n_2|$ is the order of the harmonic component. The frequencies that will be considered in the analysis is made up of the harmonics and of the intermodulation products of the ecxitations. The choice of the maximum number of harmonics that has to be considered for each frequency and of the maximum order of intermodulation is a very complicated subject and is determined considering the nonlinearities of the circuit analyzed ([9]).

The general structure of the harmonic balance algorithm, as described before, still holds, the main modification is related to the Fourier transform which becomes a multi-dimensional Fourier transform.

The multi-dimensional Fourier transform for a function $\bar{v}(t_1, t_2)$ of two variables, each with its own periodicity, is defined as:

$$\bar{v}(t_1, t_2) = \sum_{n_1=-\infty}^{+\infty} \sum_{n_2=-\infty}^{+\infty} \bar{V}_{n_1,n_2} e^{(jn_1\omega_1 t_1 + jn_2\omega_2 t_2)} \tag{2.24}$$

Each variable is sampled over its own periodicity, in this way, a two dimensional grid of samples is obtained. If the signal has a limited frequency spectrum, and if the number of samples satisfies Shannon's sampling theorem, we can compute the two-dimensional grid of coefficients in the two-dimensional Fourier series expansion of Eq.2.22, the complete calculation can be found in [9].

The samples are taken at the sampling time instants:

$$t_{k_1} = \frac{T_1}{2N_1 + 1} k_1, \qquad k_1 = -N_1, \ldots, N_1 \tag{2.25}$$

$$t_{k_2} = \frac{T_2}{2N_2+1}k_2, \qquad k_2 = -N_2, \ldots, N_2 \tag{2.26}$$

summing up to a number of samples:

$$N_{tot} = (2N_1+1)(2N_2+1) \tag{2.27}$$

Once the phasors are computed, the original two-tone voltage is readily obtained as:

$$v(t) = \bar{v}(t,t) \tag{2.28}$$

as can be seen from Eq.2.24.

The multi-tone analysis has some important shortcomings. The large number of samples requires a comparably large and often prohibitive amount of computation time. The large amount of computation time is not the only problem that this method introduces. The large number of arithmetic operations necessary to form the Fourier transform reduces numerical precision, causing the result to be inaccurate. This is especially troublesome when the analysis includes both large and small frequency components, the usual situation in multitone analysis.

2.3 A Practical Example

The algorithm presented in the previous section has been implemented and tested in a simple nonlinear circuit, namely the half-wave rectifier of Fig. 2.4. The main script and functions implemented are described in Appendix A.

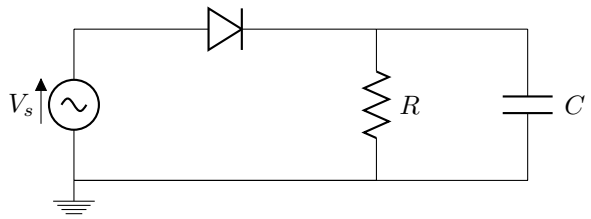


Figure 2.4: Half-wave rectifier

This circuit presents one linear port, the RC load, one nonlinear port, the diode, and a single sinusoidal excitation. Even if the structure of the circuit is quite simple, the strong nonlinearity introduced by the diode must be treated carefully in order to ensure the convergence of the algorithm. In this sense a continuation method like the Source Stepping method turned out to be efficient.

Preliminarly it is necessary to partition the circuit in a linear and nonlinear subcircuits. In this case the subdivision is straightforward. The necessary partition of the circuit is represented in Fig.2.5.

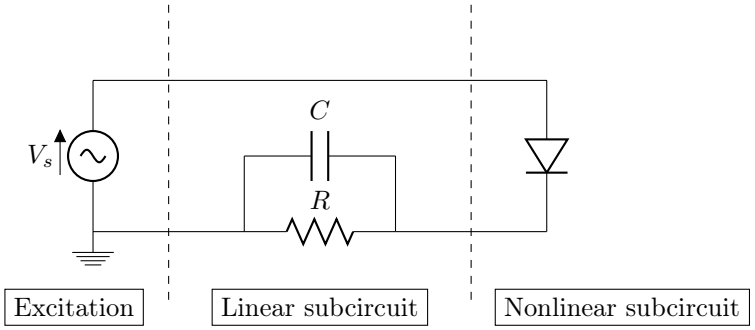


Figure 2.5: Circuit partition

Several tests have been carried out, taking into account different numbers of harmonics to check the accuracy of the algorithm, and considering different values for the RC load, then the simulations have been validated using the commercial circuit simulator PSpice.

The voltage source is a sinusoidal signal:

$$v_s = V_{max} \sin(\omega t) \tag{2.29}$$

with $V_{max} = 1$ V and $\omega = 20$ kHz, the period is denoted with T and it results equal to 5×10^{-5} s.

The diode is described by:

$$i_d = I_d(e^{(\frac{v}{V_t})} - 1) \tag{2.30}$$

with reverse current $I_d = 0.1$ nA and threshold voltage $V_t = 0.025$ V. The value of the resistor is initially set to $R = 100\ \Omega$ while the capacitance is $C = 0.1$ mF.

First of all the circuit has been simulated considering different numbers of harmonics (N_H). The Fourier Transform and the Inverse Fourier Transform have been implemented using the Fast Fourier Transform (FFT) algorithm. This implies that the number of harmonics that has to be taken into account should be choosen as 2^n with $n \in \mathbb{N}$ to greatly increase the efficiency of the FFT. Otherwise computing the Discrete Fourier Transform in matrix notation is equally valid but leads to a less efficient code. The steady state solutions in one period are represented in Fig. 2.6 - 2.7.

The same implementation is valid even changing the parameters of the load or the frequency and amplitude of the excitation.

Augmenting the number of harmonics the solution computed using the HB tends to be more accurate. Moreover the solution can be computed in the frequency domain to check and compare the accuracy of each harmonic computed with the HB and with the circuit simulator. In this sense the figures 2.8 - 2.9 represent up to 16-th harmonic the spectrum of the load voltage and current.

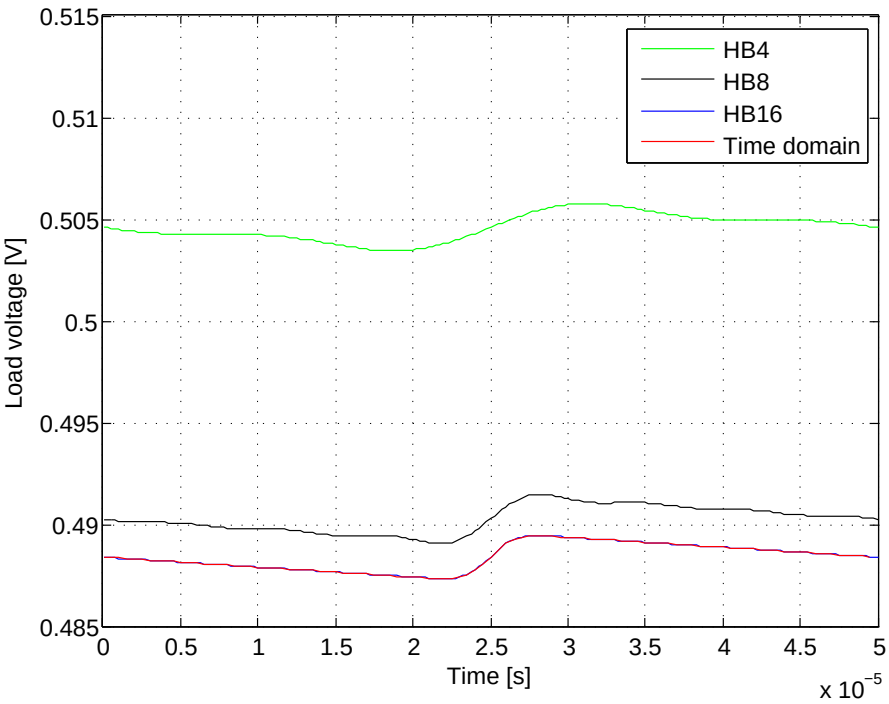


Figure 2.6: Load voltage

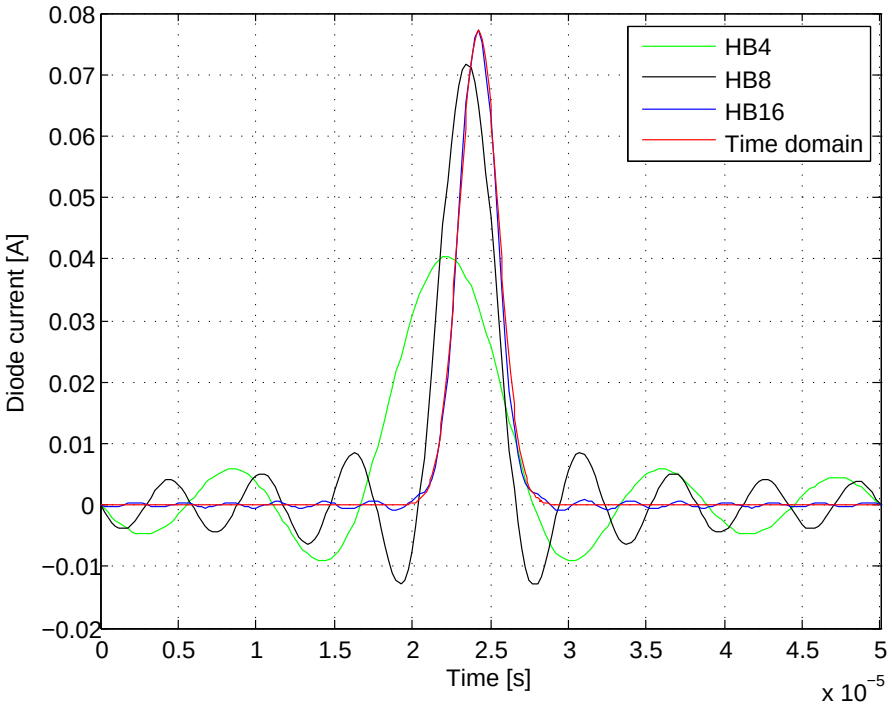


Figure 2.7: Diode current

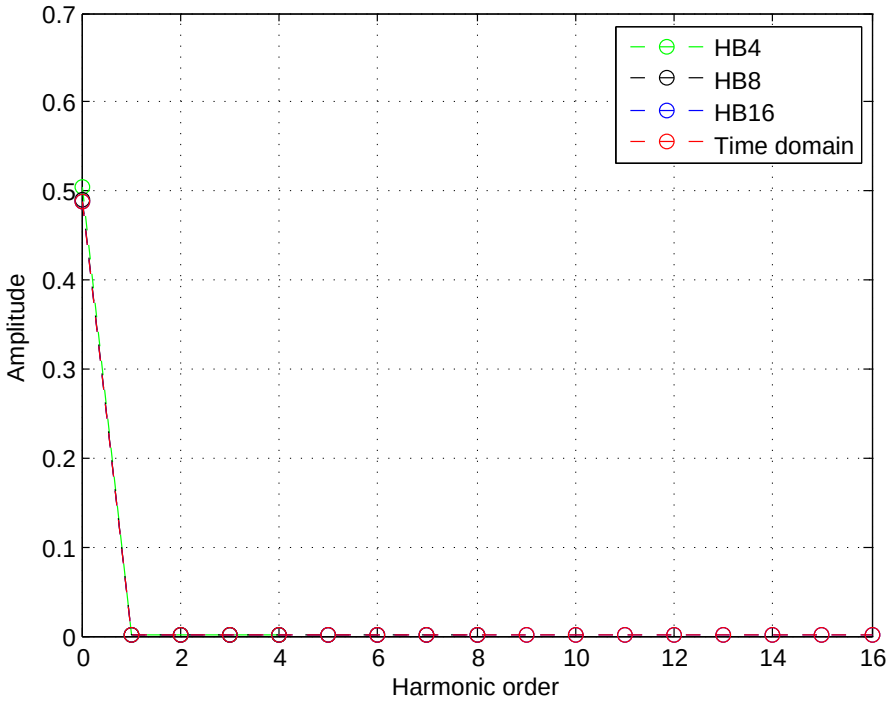


Figure 2.8: Harmonic content of the load voltage

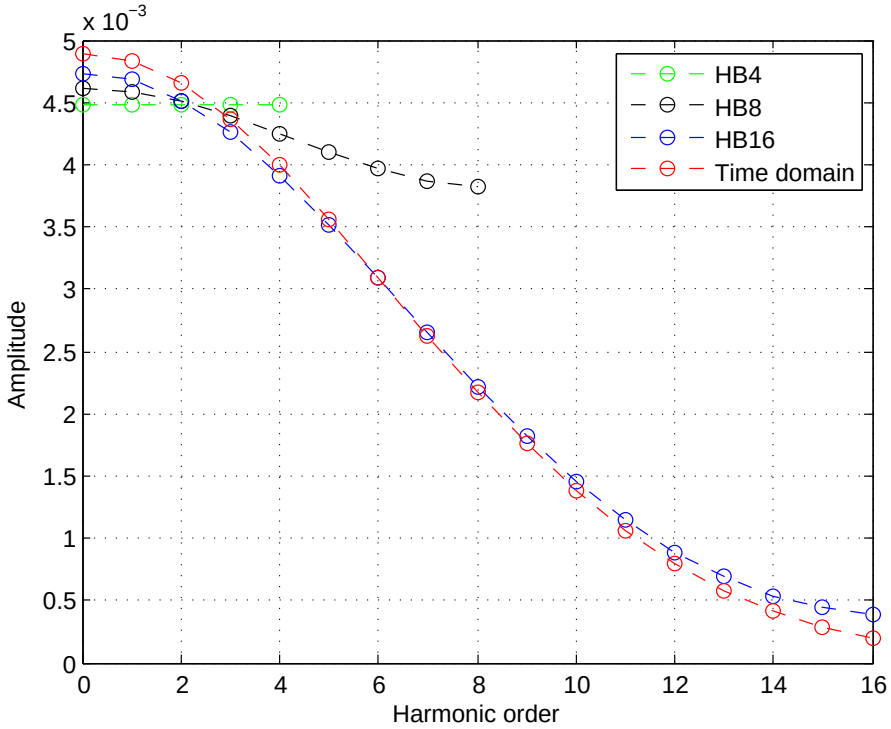


Figure 2.9: Harmonic content of the diode current

Chapter 3

The Harmonic Balance Finite Element Method

The analysis of electromagnetic problems is typically performed using numerical techniques such as Finite Element Method. The presence of non-linearity also causes the impossibility to use time-harmonic solvers, in these cases it is therefore necessary to carry out time-dependent simulations that can take long calculation times. If only the steady-state solution is of interest is possible to use the HBFM that allows to solve the problem without considering the transient regime.

The aim of this chapter is to introduce the HBFEM method considering both a strong-coupled method and an approach with decoupling of harmonics. The HBFEM will be derived considering nonlinear eddy currents problems.

3.1 Maxwell's Equations

In order to introduce eddy currents problem we have to recall briefly the Maxwell's equation. In differential form they are expressed as follows:

- Faraday's law
$$\nabla \times \boldsymbol{E} = -\frac{\partial \boldsymbol{B}}{\partial t} \tag{3.1}$$
- Ampere-Maxwell's law
$$\nabla \times \boldsymbol{H} = \boldsymbol{J} + \frac{\partial \boldsymbol{D}}{\partial t} \tag{3.2}$$
- Gauss' law
$$\nabla \cdot \boldsymbol{D} = \rho \tag{3.3}$$
- Gauss' law for magnetism
$$\nabla \cdot \boldsymbol{B} = 0 \tag{3.4}$$

We also have to include the charge conservation law:

$$\nabla \cdot \boldsymbol{J} = -\frac{\partial \rho}{\partial t} \tag{3.5}$$

which derives from the previous set of equations. Moreover we can add the constitutive relations:

$$\boldsymbol{B} = \mu \boldsymbol{H} \tag{3.6}$$

$$\boldsymbol{D} = \varepsilon \boldsymbol{E} \tag{3.7}$$

$$\boldsymbol{J} = \sigma \boldsymbol{E} \tag{3.8}$$

In the quasi-static limit Eq.3.2 and Eq.3.5 get simpler neglecting the terms related to the time derivative, therefore they can respectively be rewritten:

$$\nabla \times \boldsymbol{H} = \boldsymbol{J} \tag{3.9}$$

$$\nabla \cdot \boldsymbol{J} = 0 \tag{3.10}$$

The previous set of equations is the starting point for the next results.

3.2 Eddy Current Problems

We want to derive now the eddy current PDE starting from the previous set of equations. Introducing the vector potential $\boldsymbol{A}$ defined as:

$$\boldsymbol{B} = \nabla \times \boldsymbol{A} \tag{3.11}$$

Eq.3.1 can be rewritten as:

$$\nabla \times \boldsymbol{E} = -\frac{\partial}{\partial t}\left(\nabla \times \boldsymbol{A}\right) \tag{3.12}$$

and therefore:

$$\nabla \times \left(\boldsymbol{E} + \frac{\partial \boldsymbol{A}}{\partial t}\right) = 0 \tag{3.13}$$

From the previous equation, with the appropriate topological hypothesis, is possible to introduce the electric scalar potential V, and recast Eq.3.13 in such a way to express $\boldsymbol{E}$ as:

$$\boldsymbol{E} = -\frac{\partial \boldsymbol{A}}{\partial t} - \nabla V \tag{3.14}$$

so that the current density becomes:

$$\boldsymbol{J} = \sigma\left(-\frac{\partial \boldsymbol{A}}{\partial t} - \nabla V\right) \tag{3.15}$$

Therefore substituting the current density in Eq.3.9 and using Eq.3.11 we arrive at the final equation that describes eddy current problems:

$$\nabla \times (\nu \nabla \times \boldsymbol{A}) + \sigma \frac{\partial \boldsymbol{A}}{\partial t} = -\sigma \nabla V \tag{3.16}$$

with:

$$-\nabla \cdot \left(\sigma \nabla V + \sigma \frac{\partial \boldsymbol{A}}{\partial t}\right) = 0 \tag{3.17}$$

The previous derivation is known as $\boldsymbol{A}, V - \boldsymbol{A}$ formulation. A similar calculation can be done in term of current vector potential $\boldsymbol{T}$ and magnetic scalar potential Φ, in this case we talk about $\boldsymbol{T}, \Phi - \Phi$ formulation. In 2D problems we have:

$$\boldsymbol{A} = A_z(x, y) \tag{3.18}$$

therefore Eq.5.14 can be simplified to a scalar equation:

$$-\nabla \cdot \nu \nabla A_z + \sigma \frac{\partial A_z}{\partial t} = J_{sz} \tag{3.19}$$

3.3 The Finite Element Method

In order to introduce the HBFEM we have to recall the basis of the Finite Element Method. We underline that explain formally the entire theory behind the Finite Element method is not the object of this work, we want only to derive the main aspects which will be used in the next sections.

Let's consider the generic domain Ω and the differential problem:

$$\begin{aligned} Au(\boldsymbol{x}) &= f(\boldsymbol{x}), \quad \boldsymbol{x} \in \Omega \\ Gu(\boldsymbol{x}) &= 0, \quad \boldsymbol{x} \in \partial\Omega \end{aligned} \tag{3.20}$$

where A and G are generic linear differential operator, for example A is the Laplacian considering the Laplace or the Poisson equation. We denote by $\mathcal{S}$ the functional space which contains all the continuous functions defined in Ω.

The idea is to find an approximated solution $\hat{u}_n$ of the exact solution $\bar{u}$ in a finite dimension subspace $\mathcal{S}_n \subset \mathcal{S}$ as a linear combination of one of his basis:

$$\hat{u}_n = \sum_{j=1}^{n} \alpha_j \varphi_j(\boldsymbol{x}) \tag{3.21}$$

therefore we have reduced the problem to the determination of the unknown coefficients α_j, we have passed from a continuous problem to a discrete one. Note that this kind of approximation garantees the mean convergence of the approximated solution to the exact one and not necessarily the uniform one. The peculiarity of this approach, with respect to other method for example the Finite Difference method which discretize the differential operator, is that the Finite Element method seek directly for an approximated solution of the problem. To determine these coefficients we make use of the Galerkin Weighted Residual approach.

There are also other variational methods to evaluate these coefficients for example the Ritz method, on the other hand the Galerkin approach is more general and can solve any differential equation while the Ritz method requires the minimization of a functional. Anyhow it can also be demonstrated that if A is self adjoint and positive defined the equations obtained applied Galerkin or Ritz coincide.

3.3.1 Galerkin Weighted Residual Approach

To explain the Galerkin approach we consider an approximate solution as expressed in Eq.3.21 and we define the residual:

$$r_n = A(\hat{u}_n) - f \tag{3.22}$$

The idea is to impose proper conditions in order to determine the coefficients $\{\alpha_j\}_{j=1}^n$ which minimize the Eq.3.22 in $\mathcal{S}_n$. The residual will be null only if $\mathcal{S}_n$ contains the exact solution $\bar{u}$ of the problem.

Then we define the linearly independent test functions $\{w_i\}_{i=1}^n$ and we impose the orthogonality conditions:

$$\langle r_n, w_i \rangle = 0, \quad i = 1, \ldots, n \tag{3.23}$$

therefore we have:

$$\int_\Omega (A(\hat{u}_n) - f) w_i \, d\Omega = 0, \quad i = 1, \ldots, n \tag{3.24}$$

we have obtained a system with unknowns $\{\alpha_i\}_{i=1}^n$.

The integrals of Eq.3.24 describe the weak form of Eq.3.20. Explicitely we have:

$$\int_\Omega A(\hat{u}_n) w_i \, d\Omega = \int_\Omega f w_i \, d\Omega, \quad i = 1, \ldots, n \tag{3.25}$$

In conclusion we have obtained the problem of finding $\hat{u}_n \in \mathcal{S}_n$ so that:

$$a(\hat{u}_n, w) = (f, w), \quad \forall w \in \mathcal{L}_n \tag{3.26}$$

$a(\cdot,\cdot)$ and $(\cdot,\cdot)$ are bilinear forms while $\mathcal{L}_n$ is the functional space with $\{w_i\}_{i=1}^n$ as a basis.

The integrals of Eq.3.24 lead to different method depending on the choice of the test functions. We adopt Galerkin's choice and therefore we have:

$$w_i = \varphi_i, \quad i = 1, \ldots, n \tag{3.27}$$

and therefore Eq.3.24 becomes:

$$\int_\Omega (A(\hat{u}_n) - f)\varphi_i \, d\Omega = 0, \quad i = 1, \ldots, n \tag{3.28}$$

which impose that the residual must be orthogonal to the n basis functions.

3.3.2 Choice of the Trial and Test Functions

The FEM is a Galerkin technique for whom we have $\{\varphi_i\}_{i=1}^n = \{w_i\}_{i=1}^n$ and therefore $\mathcal{S}_n \equiv \mathcal{L}_n$. The Finite Element method assumes the functions $\{\varphi_i\}_{i=1}^n$ equal to piecewise continuous polynomials with local supports. The definition of the local support of each function is determined partitionig the domain Ω in polygonal elements, called finite elements, Ω_i so that:

$$\Omega = \bigcup_{i=1}^{m} \Omega_i \tag{3.29}$$

The interpolating polynomials can be determined easily using the Lagrange interpolating theory, in this work we mostly deal with 2-dimensional problems therefore we recall briefly the theory related to this particular case.

Let's then suppose to have a 2D domain and the related finite element mesh of triangular linear elements.

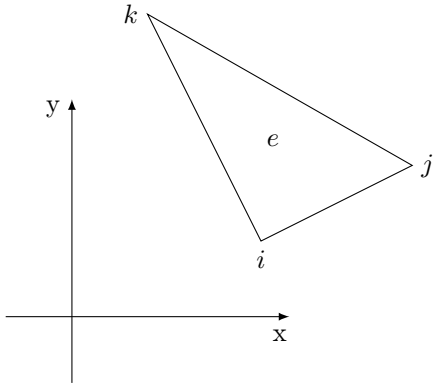


Figure 3.1: Generic triangular element

In this case the shape functions are:

$$\begin{aligned}
\varphi_i^e(x,y) &= \frac{a_i + b_i x + c_i y}{2\Delta} \\
\varphi_j^e(x,y) &= \frac{a_j + b_j x + c_j y}{2\Delta} \\
\varphi_k^e(x,y) &= \frac{a_k + b_k x + c_k y}{2\Delta}
\end{aligned} \tag{3.30}$$

with:

$$\begin{aligned} a_i(x,y) &= x_j y_k - x_k y_j \\ b_i(x,y) &= y_j - y_m \\ c_i(x,y) &= x_k - x_j \end{aligned} \tag{3.31}$$

the other coefficients are obtained immediately with a counterclockwise permutation of the indeces, Δ is the cross section of the generic element e. Regarding for example the shape function φ_i^e, it is equal to one on node i and is null in j and k.

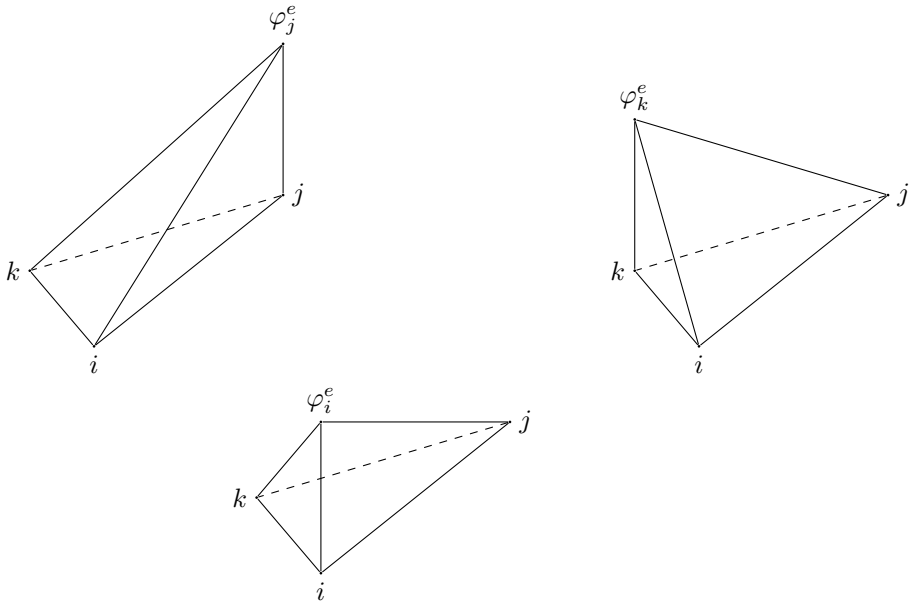


Figure 3.2: Local basis functions for a linear triangular element

Then the function φ_i will be the union of the local basis functions $\varphi_i^e(x,y)$ defined in all the triangular elements which present the node i as a vertex, the same concept is valid for $\varphi_j^e(x,y)$ and $\varphi_k^e(x,y)$.

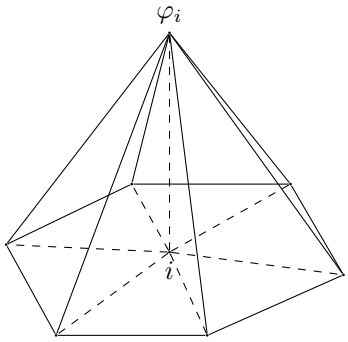


Figure 3.3: Basis function of the generic node i

We recall we have considered only triangular linear elements, if it is necessary to increase the order of the approximation more nodes must be added, anyhow the analysis can be easily extended to higher order elements or to one or three dimensional problems.

In conclusion we note that the approximated solution is a $\mathcal{C}^0$ function even if higher order interpolations are used.

3.4 HBFEM: Strong Coupled Approach

The aim of this section is to give a presentation of the HBFEM considering a strong coupled approach which means that all the harmonics are coupled together giving rise to a single large nonlinear system that has to be solved to find the final solution, to explain the method we have to make some preliminary consideration. First of all we are dealing with nonlinear eddy current problems, in this case the nonlinearity is given by the dependence of the magnetic permeability on the magnetic field, the constitutive relation expressed by Eq.3.6 has to be rewritten as:

$$\boldsymbol{B} = \mu(|\boldsymbol{H}|)\boldsymbol{H} \tag{3.32}$$

$$\boldsymbol{H} = \nu(|\boldsymbol{B}|)\boldsymbol{B} \tag{3.33}$$

if we are considering respectively the magnetic permeability or the magnetic reluctivity.

Moreover, in order to apply the HB method, we assume a periodical excitation source. However, due to the nonlinearity, the result will in general not present the same harmonic content of the excitation, and will be approximated by a Fourier series expansion.

We recall that we are engaged in the solution of the problem:

$$\nabla \times (\nu \nabla \times \boldsymbol{A}) + \sigma \frac{\partial \boldsymbol{A}}{\partial t} = \boldsymbol{J} \tag{3.34}$$

with the appropriate boundary and initial conditions. In this section we derive a real equivalent formulation while a slightly different version of the method in the complex domain will be presented later.

We present now the standard application of the HB method to the partial differential equation 3.34 discretized using FEM. First of all, we have to apply Galerkin to this equation making use of a triangulation of linear finite elements but at the moment, we make some preliminary hypothesis. We consider:

- the field is 2-dimensional
- the problem is quasi-stationary
- the saturated core is isotropic.
- the hysteresis is not considered

Therefore, we have already seen the vector potential $\boldsymbol{A} = (0, 0, A_z(x, y))$ satisfies, in the region of interest surrounded with some boundary conditions, Eq.3.19. Now we can apply the Galerkin procedure and the weighting functions are the same as the shape functions $\varphi_i(x, y)$. We obtain:

$$-\int_\Omega \nabla\varphi_i \cdot \nabla A_z \, d\Omega + \int_\Omega \varphi_i \sigma \frac{\partial A_z}{\partial t} \, d\Omega = \int_\Omega \varphi_i J \, d\Omega \tag{3.35}$$

We are seeking for the steady-state solution, therefore all variables are approximated as a Fourier series expansion:

$$A_z^i(t) = \sum_{n=1}^{\infty} A_{ns}^i \sin(n\omega t) + A_{nc}^i \cos(n\omega t) \tag{3.36}$$

$$J^i(t) = \sum_{n=1}^{\infty} J_{ns}^i \sin(n\omega t) + J_{nc}^i \cos(n\omega t) \tag{3.37}$$

$$B_x^e(t) = \sum_{n=1}^{\infty} B_{xns}^e \sin(n\omega t) + B_{xnc}^e \cos(n\omega t) \tag{3.38}$$

$$B_y^e(t) = \sum_{n=1}^{\infty} B_{yns}^e \sin(n\omega t) + B_{ync}^e \cos(n\omega t) \tag{3.39}$$

Similarly, we can express the reluctivity as:

$$\nu(t) = \sum_{n=0}^{\infty} \nu_{ns}^{i} \sin(n\omega t) + \nu_{nc}^{i} \cos(n\omega t) \tag{3.40}$$

The superscript i and e denotes the generic node and element respectively. Now we substitute the previous expansions in Eq.3.35 and we apply the Harmonic Balance equating the coefficients on both side. This calculation leads to the following matrix equation for a generic element:

$$\begin{aligned} &\frac{1}{4\Delta} \begin{bmatrix} (b_1b_1 + c_1c_1)\boldsymbol{R} & (b_1b_2 + c_1c_2)\boldsymbol{R} & (b_1b_3 + c_1c_3)\boldsymbol{R} \\ (b_2b_1 + c_2c_1)\boldsymbol{R} & (b_2b_2 + c_2c_2)\boldsymbol{R} & (b_2b_3 + c_2c_3)\boldsymbol{R} \\ (b_3b_1 + c_3c_1)\boldsymbol{R} & (b_3b_2 + c_3c_2)\boldsymbol{R} & (b_3b_3 + c_3c_3)\boldsymbol{R} \end{bmatrix} \boldsymbol{A} \\ &+ \frac{\sigma\omega\Delta}{12} \begin{bmatrix} 2\boldsymbol{D} & \boldsymbol{D} & \boldsymbol{D} \\ \boldsymbol{D} & 2\boldsymbol{D} & \boldsymbol{D} \\ \boldsymbol{D} & \boldsymbol{D} & 2\boldsymbol{D} \end{bmatrix} \boldsymbol{A} = \frac{\sigma\Delta}{3}\boldsymbol{J} \end{aligned} \tag{3.41}$$

where:

$$\boldsymbol{A} = [A_{1s}^1, A_{1c}^1, A_{2s}^1, A_{2c}^1, \ldots, A_{1s}^{N_n}, A_{1c}^{N_n}, A_{2s}^{N_n}, A_{2c}^{N_n}, \ldots]^T \tag{3.42}$$

$$\boldsymbol{J} = [J_{1s}^1, J_{1c}^1, J_{2s}^1, J_{2c}^1, \ldots, J_{1s}^{N_n}, J_{1c}^{N_n}, J_{2s}^{N_n}, J_{2c}^{N_n}, \ldots]^T \tag{3.43}$$

with N_n we denote the total number of nodes and Δ is the cross section of the generic element e. The value of a_i, b_i and c_i is given by Eq.3.31.

The matrix $\boldsymbol{R}$ is symmetric and presents the following structure:

$$\boldsymbol{R} = \frac{1}{2} \begin{bmatrix} 2\nu_0 - \nu_{2c} & \nu_{2s} & \nu_{2c} - \nu_{4c} & -\nu_{2s} + \nu_{4s} & \ldots \\ & 2\nu_0 + \nu_{2c} & \nu_{2s} + \nu_{4s} & \nu_{2c} + \nu_{4cs} & \ldots \\ & & 2\nu_0 - \nu_{6c} & \nu_{6s} & \ldots \\ & & & 2\nu_0 + \nu_{6c} & \ldots \\ & & & & \ddots \end{bmatrix} \tag{3.44}$$

$\boldsymbol{D}$ is a diagonal block matrix, it results:

$$\boldsymbol{D} = \begin{bmatrix} 0 & 1 & & & \\ -1 & 0 & & & \\ & & 0 & 2 & \\ & & -2 & 0 & \\ & & & & \ddots \end{bmatrix} \tag{3.45}$$

The system equation for the entire region is obtained by the same procedure as the conventional FEM and is solved by the iteration procedure for a nonlinear static field. The main feature is that the calculation concerned with time is not included and the procedure of the calculation is the same as the nonlinear static FEM. The procedure we have just described is the classical way to apply the Harmonic Balance method to a FEM discretization. The main disadvantage of this method is the complicated assembly and resolution of a large and dense system of nonlinear equations, this is mainly due to the fact we have considered explicitely the Fourier expansion of the reluctivity.

Now we want to derive a different approach which partially cures this drawback. In order to solve Eq.3.34 numerically, we need a discretization in time and space. In this case, instead of discretizing the equation in space using FEM and then solving the resulting ODE, we take advantage of the periodicity of the current source applying the Harmonic Balance first. For the sake of clarity we assume a pure cosinusoidal excitation in the form:

$$\boldsymbol{J}(\boldsymbol{x}, t) = \boldsymbol{J}(\boldsymbol{x}) \cos(\omega t) \tag{3.46}$$

note that the method is valid even if the current source presents a multi harmonic content. Then for the solution we are seeking we make the ansatz:

$$\boldsymbol{A}(\boldsymbol{x},t) \simeq \sum_{k=0}^{N}[\boldsymbol{A}_k^c(\boldsymbol{x})\cos k\omega t + \boldsymbol{A}_k^s(\boldsymbol{x})\sin k\omega t] \tag{3.47}$$

hence, the magnetic field $\boldsymbol{H} = \boldsymbol{H}(\boldsymbol{B}) = \boldsymbol{H}(\nabla\times\boldsymbol{A})$ is periodic as well and can be rewritten in the form:

$$\boldsymbol{H}(\nabla\times\boldsymbol{A}) = \boldsymbol{H}_0(\nabla\times\boldsymbol{A}) + \sum_{k=1}^{N}[\boldsymbol{H}_k^c(\nabla\times\boldsymbol{A})\cos k\omega t + \boldsymbol{H}_k^s(\nabla\times\boldsymbol{A})\sin k\omega t] \tag{3.48}$$

3.4.1 Reduction of the Harmonics

In order to semplify the derivation of the method it's possible to make some preliminary consideration on the harmonic content of the solution. In order to keep notation simple, we use $\boldsymbol{A}$ to denote the sequence of Fourier coefficients and $\boldsymbol{A}(t)$ to signify the periodic function that is determined by these coefficients according to Eq.3.47, and similarly for $\boldsymbol{J}$.

Since odd modes $\cos((2k+1)\omega t)$, $\sin((2k+1)\omega t)$ present half-wave symmetry, the condition:

$$v\left(t+\frac{\pi}{\omega}\right) = -v(t) \tag{3.49}$$

for the generic function $v = v_0 + \sum_{k=1}^{N} v_k^c(\cos k\omega t) + v_k^s(\sin k\omega t)$ is an equivalent characterization of the property $v_{2k}^c = v_{2k}^s = 0$, the excitation $\boldsymbol{J}$ presents this characteristic. We want to show that if the previous hypothesis is verified for the excitation, even the solution has to present only odd harmonics.

For a given right hand side $\boldsymbol{J}(\cdot)$, we get the unique solution $\boldsymbol{A}$. Shifting the right hand side to $\tilde{\boldsymbol{J}} = \boldsymbol{J}(\cdot + \frac{\pi}{\omega})$ we obtain the solution $\tilde{\boldsymbol{A}}$, but with $\bar{\boldsymbol{J}} = -\boldsymbol{J}$ we obtain the result $\bar{\boldsymbol{A}} = \tilde{\boldsymbol{A}}$ because $\boldsymbol{J}(\cdot + \frac{\pi}{\omega}) = -\boldsymbol{J}(\cdot)$. Moreover $\boldsymbol{A}(\cdot + \frac{\pi}{\omega})$ is a periodic solution of Eq.3.34 with right hand side $\tilde{\boldsymbol{J}}$, then we have $\tilde{\boldsymbol{A}}(\cdot) = \boldsymbol{A}(\cdot + \frac{\pi}{\omega})$. On the other hand $-\boldsymbol{A}$ solves Eq.3.34 with right hand side $\bar{\boldsymbol{J}}$ and so $-\boldsymbol{A} = \bar{\boldsymbol{A}}$. To summarize:

$$\boldsymbol{A}\left(\cdot + \frac{\pi}{\omega}\right) = \tilde{\boldsymbol{A}}(\cdot) = \bar{\boldsymbol{A}}(\cdot) = -\boldsymbol{A}(\cdot) \tag{3.50}$$

wich means that the solution we are looking for satisfies Eq.3.49 so it's proved that $\boldsymbol{A}$ can be expressed as a sum of odd harmonics.

We can also state a similar property for the magnetic field, infact we have:

$$\begin{aligned}\boldsymbol{H}\big(t+\frac{\pi}{\omega}\big) &= \nu\Big(\big|\nabla\times\boldsymbol{A}\big(t+\frac{\pi}{\omega}\big)\big|\Big)\nabla\times\boldsymbol{A}\big(t+\frac{\pi}{\omega}\big)\\ &= \nu\big(\big|-\nabla\times\boldsymbol{A}(t)\big|\big)\big(-\nabla\times\boldsymbol{A}(t)\big)\\ &= -\boldsymbol{H}(t)\end{aligned} \tag{3.51}$$

3.4.2 Time Discretization using the Harmonic Balance Method

We have already assumed that the solution can be expressed as a finite sum of harmonics as stated in Eq.3.47, now we have to apply the HB method to find out the nonlinear system that has to be solve to determine the unknown coefficients.

Substituting Eq.3.47 and Eq.3.48 in Eq.3.34 we obtain:

$$\begin{aligned}&\nabla\times\sum_{k=0}^{N}[\boldsymbol{H}_k^c(\nabla\times\boldsymbol{A})\cos k\omega t + \boldsymbol{H}_k^s(\nabla\times\boldsymbol{A})\sin k\omega t]\\ &\quad+\omega\sigma\sum_{k=0}^{N}k[\boldsymbol{A}_k^s(\nabla\times\boldsymbol{A})\cos k\omega t - \boldsymbol{A}_k^c(\nabla\times\boldsymbol{A})\sin k\omega t]\\ &\quad=\sum_{k=0}^{N}[\boldsymbol{J}_k^c\cos k\omega t + \boldsymbol{J}_k^s\sin k\omega t]\end{aligned} \tag{3.52}$$

We test this equation with $\cos(m\omega t)$ and $\sin(m\omega t)$ and integrate by t, taking advantage of the orthogonality:

$$\begin{aligned}
&\frac{\omega}{\pi}\int_0^{\frac{2\pi}{\omega}} \cos(k\omega t)\cos(m\omega t)\,dt = \delta_{km} \\
&\frac{\omega}{\pi}\int_0^{\frac{2\pi}{\omega}} \sin(k\omega t)\cos(m\omega t)\,dt = 0 \\
&\frac{\omega}{\pi}\int_0^{\frac{2\pi}{\omega}} \sin(k\omega t)\sin(m\omega t)\,dt = \delta_{km}
\end{aligned} \tag{3.53}$$

where δ_{km} is the Kronecker delta. Together with the fact that all even harmonics are zero and neglecting the zero order harmonic, this calculations lead to the following nonlinear system of equations:

$$\nabla \times \begin{bmatrix} \boldsymbol{H}_1^c(\nabla\times\boldsymbol{A}) \\ \boldsymbol{H}_1^s(\nabla\times\boldsymbol{A}) \\ \vdots \\ \boldsymbol{H}_N^c(\nabla\times\boldsymbol{A}) \\ \boldsymbol{H}_N^s(\nabla\times\boldsymbol{A}) \end{bmatrix} + \omega\sigma \begin{bmatrix} 0 & 1 & & & \\ -1 & 0 & & & \\ & & \ddots & & \\ & & & 0 & N \\ & & & -N & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_1^c \\ \boldsymbol{A}_1^s \\ \vdots \\ \boldsymbol{A}_N^c \\ \boldsymbol{A}_N^s \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_1^c \\ \boldsymbol{J}_1^s \\ \vdots \\ \boldsymbol{J}_N^c \\ \boldsymbol{J}_N^s \end{bmatrix} \tag{3.54}$$

In a more compact way:

$$\nabla \times \boldsymbol{H}(\nabla\times\boldsymbol{A}) + \omega\sigma\boldsymbol{D}\boldsymbol{A} = \boldsymbol{J} \tag{3.55}$$

where $\boldsymbol{D}$ is the same matrix introduced in Eq.3.45. Note that the time dependance of the problem has been eliminated and we have obtained a nonlinear system which only presents space dependents terms. Moreover $\boldsymbol{A}$ is the vector of all the Fourier coefficients, which means that every harmonic is coupled with the others.

Considering a 2D problem the structure of system 3.54 can be simplified, infact in this case the eddy current problem is described as:

$$-\nabla\cdot\nu(|\nabla A|)\nabla A + \sigma\frac{\partial A}{\partial t} = J \tag{3.56}$$

Then the approximate solution turns out to be:

$$A(\boldsymbol{x},t) \simeq \sum_{k=0}^{N}[A_k^c(\boldsymbol{x})\cos k\omega t + A_k^s(\boldsymbol{x})\sin k\omega t] \tag{3.57}$$

In this case the coefficients $A_k^c(\boldsymbol{x}) = A_k^c(x,y)$ and $A_k^s(\boldsymbol{x}) = A_k^s(x,y)$ become scalar functions. Then we momentarily introduce the potential:

$$\boldsymbol{\Psi}[A](\boldsymbol{x},t) = \nu(|\nabla A|)\nabla A(\boldsymbol{x},t) \tag{3.58}$$

that can also be approximated as:

$$\boldsymbol{\Psi}[A](\boldsymbol{x},t) \simeq \sum_{k=0}^{N}[\boldsymbol{\Psi}_k^c[A](\boldsymbol{x})\cos k\omega t + \boldsymbol{\Psi}_k^s[A](\boldsymbol{x})\sin k\omega t] \tag{3.59}$$

The system 3.54 can be rewritten:

$$-\nabla\cdot\begin{bmatrix} \boldsymbol{\Psi}_1^c[A] \\ \boldsymbol{\Psi}_1^s[A] \\ \vdots \\ \boldsymbol{\Psi}_N^c[A] \\ \boldsymbol{\Psi}_N^s[A] \end{bmatrix} + \omega\sigma \begin{bmatrix} 0 & 1 & & & \\ -1 & 0 & & & \\ & & \ddots & & \\ & & & 0 & N \\ & & & -N & 0 \end{bmatrix} \begin{bmatrix} A_1^c \\ A_1^s \\ \vdots \\ A_N^c \\ A_N^s \end{bmatrix} = \begin{bmatrix} J_1^c \\ J_1^s \\ \vdots \\ J_N^c \\ J_N^s \end{bmatrix} \tag{3.60}$$

In compact notation:

$$-\nabla\cdot\boldsymbol{\Psi}[\boldsymbol{A}] + \omega\sigma\boldsymbol{D}\boldsymbol{A} = \boldsymbol{J} \tag{3.61}$$

3.4.3 Space Discretization using the Finite Element Method

We have derived a nonlinear system of equation with space dependent coefficients, to solve this system we use the Finite Element Method. For the sake of semplicity we will analyse the 2D problem, therefore we have to solve Eq.3.61. We suppose to discretize the domain Ω with a triangulation of linear finite elements, next we want to build up the matrices and the resulting system of equations, we proceed analysing separately the terms of Eq.3.61.

The variational formulation associated to Eq.3.61, written in weak form, leads to:

$$\int_\Omega \boldsymbol{\Psi}[\boldsymbol{A}] \cdot \nabla \boldsymbol{v} + \omega\sigma \boldsymbol{D}\boldsymbol{A} \cdot \boldsymbol{v}\, d\Omega = \int_\Omega \boldsymbol{J} \cdot \boldsymbol{v}\, d\Omega \tag{3.62}$$

where $\boldsymbol{v}$ is a vector valued test function and the gradient is considered applied entry-wise. Using the standard nodal basis function, it is easy to treat the term $\omega\sigma\boldsymbol{D}$. It gives rise to the Mass matrix $\boldsymbol{M}$ defined as:

$$\boldsymbol{M} = \begin{bmatrix} 0 & D_1 & & & & & & \\ -D_1 & 0 & & & & & & \\ & & 0 & D_2 & & & & \\ & & -D_2 & 0 & & & & \\ & & & & \ddots & & & \\ & & & & & & 0 & D_N \\ & & & & & & -D_N & 0 \end{bmatrix} \tag{3.63}$$

with:

$$D_{k_{i,j}} = k\omega\sigma \int_\Omega \varphi_i \cdot \varphi_j\, d\Omega \tag{3.64}$$

where φ_i and φ_j are the usual linear basis function. Note that considering N harmonics and a mesh with N_n nodes, the solution is a vector of $2N_nN$ elements:

$$\begin{aligned} \boldsymbol{A} &= [\boldsymbol{A}^c_1, \boldsymbol{A}^s_1, \dots, \boldsymbol{A}^c_N, \boldsymbol{A}^s_N]^T \\ &= [A^c_{1,1}, \dots, A^c_{1,N_n}, A^s_{1,1}, \dots, A^s_{1,N_n}, \dots, A^c_{N,1}, \dots, A^c_{N,N_n}, A^s_{N,1}, \dots, A^s_{N,N_n}]^T \end{aligned} \tag{3.65}$$

The term $-\nabla\cdot\boldsymbol{\Psi}[\boldsymbol{A}]$ is not as easy to treat as the previous one, this is due to the presence of the nonlinear reluctivity $\nu(|\nabla A|)$ which couples all the unknowns. For the sake of clarity we will derive the analysis considering an approximation made by only the first harmonic, then the results obtained can be generalized to a generic number of harmonics. We have to calculate the integral:

$$\int_\Omega \boldsymbol{\Psi}[\boldsymbol{A}] \cdot \nabla \boldsymbol{v}\, d\Omega = \int_\Omega \nu(|\nabla A|)\nabla \boldsymbol{A} \cdot \nabla \boldsymbol{v}\, d\Omega \tag{3.66}$$

Considering only the first harmonic we can approximate:

$$\boldsymbol{\Psi}[\boldsymbol{A}] \simeq \boldsymbol{\Psi}^c_1 \cos(\omega t) + \boldsymbol{\Psi}^s_1 \sin(\omega t) \tag{3.67}$$

with:

$$\begin{aligned} \boldsymbol{\Psi}^c_1 &= \frac{2}{T}\int_T \boldsymbol{\Psi}[\boldsymbol{A}] \cos(\omega t)\, dt \\ &= \frac{2}{T}\int_T \nu(|\nabla A|)\nabla A \cos(\omega t)\, dt \\ &= \frac{2}{T}\int_T \nu(|\nabla A|)\nabla (A^c_1 \cos(\omega t) + A^s_1 \sin(\omega t)) \cos(\omega t)\, dt \\ &= \frac{2}{T}\int_T \nu(|\nabla A|)(\nabla A^c_1 \cos(\omega t) + \nabla A^s_1 \sin(\omega t)) \cos(\omega t)\, dt \end{aligned} \tag{3.68}$$

and similarly:

$$\boldsymbol{\Psi}^s_1 = \frac{2}{T}\int_T \nu(|\nabla A|)(\nabla A^c_1 \cos(\omega t) + \nabla A^s_1 \sin(\omega t)) \sin(\omega t)\, dt \tag{3.69}$$

We define the vectors:

$$\boldsymbol{\Psi} = [\boldsymbol{\Psi}_1^c, \boldsymbol{\Psi}_1^s]^T \tag{3.70}$$

$$\boldsymbol{v} = [v_1^c, v_1^s]^T \tag{3.71}$$

then we can calculate the integral:

$$\int_\Omega \boldsymbol{\Psi}[\boldsymbol{A}] \cdot \nabla \boldsymbol{v}\, d\Omega = \int_\Omega \boldsymbol{\Psi}_1^c \cdot \nabla v_1^c + \boldsymbol{\Psi}_1^s \cdot \nabla v_1^s\, d\Omega \tag{3.72}$$

defining:

$$\begin{aligned}
\alpha &= \frac{2}{T}\int_T \nu(|\nabla A|)\cos(\omega t)\cos(\omega t)\, dt \\
\beta &= \frac{2}{T}\int_T \nu(|\nabla A|)\cos(\omega t)\sin(\omega t)\, dt \\
\gamma &= \frac{2}{T}\int_T \nu(|\nabla A|)\sin(\omega t)\sin(\omega t)\, dt
\end{aligned} \tag{3.73}$$

If we substitute Eq.3.68 and Eq.3.69 in Eq.3.72, using the definitions 3.73, we obtain:

$$\begin{aligned}
\int_\Omega \boldsymbol{\Psi}_1^c \cdot \nabla v_1^c + \boldsymbol{\Psi}_1^s \cdot \nabla v_1^s\, d\Omega &= \int_\Omega (\alpha \nabla A_1^c + \beta \nabla A_1^s) \cdot \nabla v_1^c + (\beta \nabla A_1^c + \gamma \nabla A_1^s) \cdot \nabla v_1^s\, d\Omega \\
&= \int_\Omega \alpha \nabla A_1^c \cdot \nabla v_1^c\, d\Omega + \int_\Omega \beta \nabla A_1^s \cdot \nabla v_1^c\, d\Omega \\
&+ \int_\Omega \beta \nabla A_1^c \cdot \nabla v_1^s\, d\Omega + \int_\Omega \gamma \nabla A_1^s \cdot \nabla v_1^s\, d\Omega
\end{aligned} \tag{3.74}$$

Now we can discretize Eq.3.74 using a set of linear basis functions, we denote it as:

$$\{\varphi_i\}_{i=1}^{N_n} = \{\varphi_1, \varphi_2, \ldots, \varphi_{N_n}\} \tag{3.75}$$

then we have:

$$A_1^c = \sum_{j=1}^{N_n} \eta_j \varphi_j \tag{3.76}$$

$$A_1^s = \sum_{j=1}^{N_n} \xi_j \varphi_j \tag{3.77}$$

In order to clarify the structure of the Stiffness matrix we proceed to calculate explicitely the local Stiffness matrix of a generic element e with vertex i, j and k analysing each term of the sum in Eq.3.74 separetely. In the case we are treating the local Stiffness matrix is composed of four submatrix itself:

$$\boldsymbol{K}^{(e)} = \begin{bmatrix} \boldsymbol{K}_{(cc)}^{(e)} & \boldsymbol{K}_{(sc)}^{(e)} \\ \boldsymbol{K}_{(cs)}^{(e)} & \boldsymbol{K}_{(ss)}^{(e)} \end{bmatrix} \tag{3.78}$$

Considering for example the term $\boldsymbol{K}_{(cc)}^{(e)}$, using Galerkin's approach for the choice of the test functions we have:

$$\boldsymbol{K}_{(cc)}^{(e)} = \begin{bmatrix} \int_\Omega \alpha \nabla\varphi_i \cdot \nabla\varphi_i\, d\Omega & \int_\Omega \alpha \nabla\varphi_i \cdot \nabla\varphi_j\, d\Omega & \int_\Omega \alpha \nabla\varphi_i \cdot \nabla\varphi_k\, d\Omega \\ \int_\Omega \alpha \nabla\varphi_j \cdot \nabla\varphi_i\, d\Omega & \int_\Omega \alpha \nabla\varphi_j \cdot \nabla\varphi_j\, d\Omega & \int_\Omega \alpha \nabla\varphi_j \cdot \nabla\varphi_k\, d\Omega \\ \int_\Omega \alpha \nabla\varphi_k \cdot \nabla\varphi_i\, d\Omega & \int_\Omega \alpha \nabla\varphi_k \cdot \nabla\varphi_j\, d\Omega & \int_\Omega \alpha \nabla\varphi_k \cdot \nabla\varphi_k\, d\Omega \end{bmatrix} \tag{3.79}$$

The generic term is therefore:

$$K_{(cc)i,j}^{(e)} = \int_\Omega \alpha \nabla\varphi_i \cdot \nabla\varphi_j\, d\Omega \tag{3.80}$$

and similarly for the other matrices:

$$K^{(e)}_{(sc)i,j} = \int_\Omega \beta \nabla \varphi_i \cdot \nabla \varphi_j \, d\Omega \tag{3.81}$$

$$K^{(e)}_{(cs)i,j} = \int_\Omega \beta \nabla \varphi_i \cdot \nabla \varphi_j \, d\Omega \tag{3.82}$$

$$K^{(e)}_{(ss)i,j} = \int_\Omega \gamma \nabla \varphi_i \cdot \nabla \varphi_j \, d\Omega \tag{3.83}$$

The local Stiffness matrix is therefore a 6×6 matrix. The global Stiffness matrix is assembled starting from the local contributions.

Finally the construction of the right hand side consists in the calculation:

$$\int_\Omega \boldsymbol{J} \cdot \boldsymbol{v} \, d\Omega = \int_\Omega J^c_1 v^c_1 + J^s_1 v^s_1 \, d\Omega = \int_\Omega J^c_1 v^c_1 \, d\Omega + \int_\Omega J^s_1 v^s_1 \, d\Omega \tag{3.84}$$

To sum up the final system presents a Stiffness matrix $\boldsymbol{K}$ of $2N_nN$ equations in $2N_nN$ unknowns. In order to assemble efficiency the Stiffness matrix is necessary to proceed as in the stadard FEM implementation, therefore it is essential to build up the local Stiffness matrices element by element and then assemble the global matrix $\boldsymbol{K}$. Note that the final system does not present time dependent term, infact even the coefficients α, β and γ have only space dependency. The evaluation of the integrals in Eq.3.73 has to be performed numerically, tipically using Gaussian integration. We recall that the previous derivation has been developed considering only a first harmonic approximation, with some calculation the analysis can be extended to a generic number of harmonics. Moreover the final system we have derive still present the nonlinearity due to the terms containing the reluctivity. This problem is typically solved using Newton-Raphson, in this case the derivation of the algorithm is not straightforward. Infact to apply Newton-Raphson is necessary to build up the Jacobian matrix, which presents a complicated structure. Moreover the linearized system obtained is nonsymmetric and is tipically solved using GMRES with ILUT preconditioning. Anyway the problem has been treated by different authors, the formal derivation of the Newton-Raphson method applied to the HBFEM can be found in [4]-[10].

3.4.4 Complex Equivalent Formulation

In the previous section a real equivalent strong coupled approach has been presented, another way to solve the problem is to derive a complex equivalent formulation. This can be done similarly to the real equivalent case using a complex Fourier approximation for the solution:

$$\boldsymbol{A}(\boldsymbol{x}, t) = \mathrm{Re}\left(\sum_{k=0}^{N} \boldsymbol{A}_k e^{jk\omega t} \right) \tag{3.85}$$

then the derivation is similar the one presented in the previous section.

Another way to solve the problem in the complex domain is to recast Eq.5.14 directly in the frequency domain to eliminate the time dependancy. It can be shown that Eq.5.14 rewritten in the frequency domain leads to:

$$\nabla \times (\underline{\nu} \star \nabla \times \underline{\boldsymbol{A}}) + \sigma \star \mathcal{T}(\underline{\boldsymbol{A}}) = -\sigma \star \nabla \underline{V} \tag{3.86}$$

where $\star$ denotes convolution, $\underline{(\cdot)}$ is a frequency domain spectrum and $\mathcal{T}$ is the operator representing the Fourier transform of the time derivative:

$$\mathcal{T} : g(\omega) \rightarrow j\omega g(\omega) \tag{3.87}$$

In the previous section it has been shown that the magnetic field and the magnetic flux density presents only even harmonic, similarly it can be demonstrate that the material property as the permeability present only odd harmonics. Therefore in this case we have

to consider explicitely the truncated spectrum of the reluctivity, similarly to what has been done in the first real formulation we have previously presented. Note that in practice only few components are considered, both for the permeability and for the magnetic field and the magnetic flux density, therefore the operator $\star$ in Eq.3.86 as to be seen as a truncated convolution. This operation can be represented as the multiplication of a matrix with a vector. Considering the generic field quantities $\underline{a}$ and $\underline{b}$ with odd harmonic pattern and a material function $\underline{\tau}$ with even pattern, we can introduce the Toeplitz notation and therefore we obtain the following equivalent representation for the truncated convolution:

$$\begin{bmatrix} \tau_0 + \tau_2 Cg(\cdot) & \bar{\tau_2} \\ \tau_2 & \tau_0 \end{bmatrix} \begin{bmatrix} \underline{a}_1 \\ \underline{a}_3 \end{bmatrix} = \begin{bmatrix} \underline{b}_1 \\ \underline{b}_3 \end{bmatrix} \tag{3.88}$$

$$\begin{bmatrix} \tau_0 & \tau_2 & & \\ \bar{\tau_2} & \tau_0 & \tau_2 & \\ & \bar{\tau_2} & \tau_0 & \tau_2 \\ & & \bar{\tau_2} & \tau_0 \end{bmatrix} \begin{bmatrix} \underline{a}_3 \\ \underline{a}_1 \\ \underline{a}_{-1} \\ \underline{a}_{-3} \end{bmatrix} = \begin{bmatrix} \underline{b}_3 \\ \underline{b}_1 \\ \underline{b}_{-1} \\ \underline{b}_{-3} \end{bmatrix} \tag{3.89}$$

$$\begin{bmatrix} \tau_0 & \operatorname{Re}\tau_2 & -\operatorname{Im}\tau_2 & \\ \operatorname{Re}\tau_2 & \tau_0 + \operatorname{Re}\tau_2 & \operatorname{Im}\tau_2 & \operatorname{Im}\tau_2 \\ -\operatorname{Im}\tau_2 & \operatorname{Im}\tau_2 & \tau_0 - \operatorname{Re}\tau_2 & \operatorname{Re}\tau_2 \\ & \operatorname{Im}\tau_2 & \operatorname{Re}\tau_2 & \tau_0 \end{bmatrix} \begin{bmatrix} \operatorname{Re}\underline{a}_3 \\ \operatorname{Re}\underline{a}_1 \\ \operatorname{Im}\underline{a}_1 \\ \operatorname{Im}\underline{a}_3 \end{bmatrix} = \begin{bmatrix} \operatorname{Re}\underline{b}_3 \\ \operatorname{Re}\underline{b}_1 \\ \operatorname{Im}\underline{b}_1 \\ \operatorname{Im}\underline{b}_3 \end{bmatrix} \tag{3.90}$$

where Cg denotes the complex conjugate and the subscripts denote the harmonic index. The previous notation is valid considering two harmonics, the same concept can be also extended to a generic pattern. If we consider a 2D problem and a triangulation of linear finite element we can again express $\underline{A}$ as:

$$\underline{A} = \sum_{j=1}^{N_n} \underline{A}_j \varphi_j \tag{3.91}$$

applying Galerkin, the resulting system of equations is:

$$(\boldsymbol{K} \star + \boldsymbol{M} \star \mathcal{T})\underline{x} = \underline{f} \tag{3.92}$$

with:

$$\underline{k}_{ij} = \int_\Omega \underline{\nu} \nabla\varphi_i \cdot \nabla\varphi_j \, d\Omega \tag{3.93}$$

$$m_{ij} = \int_\Omega \sigma \varphi_i \varphi_j \, d\Omega \tag{3.94}$$

$$\underline{x}_j = \underline{A}_j \tag{3.95}$$

$$\underline{f}_i = \int_\Omega \sigma \star \nabla \underline{V} \varphi_i \, d\Omega \tag{3.96}$$

We underline that σ and m_{ij} are scalars while $\underline{\nu}$ and $\underline{k}_{ij}$ are vectors, explicitely regarding the term related to the Stiffness matrix we have:

$$\boldsymbol{K} \star \underline{x} = \sum_{j=0}^{N_n} \underline{k}_{ij} \star \underline{x}_j, \qquad i = 1, \ldots, N_n \tag{3.97}$$

which means that each convolution has to be calculated as a matrix-vector multiplication using Toeplitz notation as stated in Eq.3.88-3.89-3.90. The Mass matrix and the excitation can be expressed similarly to what has been done for the Stiffness matrix.

In this case the nonlinear system 3.92 is tipically solved using Picard iteratione ([5]-[11]), inbetween two successive steps k, adaptive underrelaxation is indispensable to ensure global convergence. The Newton-Raphson method is not used due to the problems that arise in the construction of the Jacobian. Applying Picard iteration, if $\tilde{\underline{x}}$ is the solution of the linearized system:

$$(\boldsymbol{K}^{(k)} \star + \boldsymbol{M} \star \mathcal{T})\tilde{\underline{x}} = \underline{f}^{(k)} \tag{3.98}$$

the new solution:

$$\underline{\tilde{x}}^{(k+1)} = \gamma\underline{\tilde{x}} + (1-\gamma)\underline{\tilde{x}}^{(k)} \tag{3.99}$$

is determined in terms of the relaxation factor γ. The relaxation factor is selected out of the set $1, 0.625, 0.5, 0.25, 0.125$ in order to minimize the nonlinear residual $||\underline{f}^{(k+1)} - (\boldsymbol{K}^{(k+1)} \star + \boldsymbol{M} \star \mathcal{T})\underline{x}^{(k+1)}||$. The linear system that arises in each iteration is symmetric and positive defined thus is tipically solved using Quasi-Minimal Residual (QMR), or Conjugate Gradient (CG) with Symmetric Successive Overrelaxation (SSOR) preconditioner.

3.5 HBFEM: Decoupling of the Harmonics

In the previous section we have analysed the HBFEM considering a strong-coupled approach both in real and complex domain. Now we want to formulate the same method considering each harmonic decoupled from the other in such a way to study each harmonic independently. It has been widely shown that the nonlinear term containing the permeability or the reluctivity couple all the unknowns, so that the decoupling is trivial in the theoretical linear case while for nonlinear problems special techniques are needed. If a decoupling is performed, we obtain a number of systems of equations that equals the number of harmonics we are considering, each system with as many unknowns as there are degrees of freedom in the FEM approximation.

We recall we are engaged with the solution of the equation:

$$\nabla \times (\nu \nabla \times \boldsymbol{A}) + \sigma \frac{\partial \boldsymbol{A}}{\partial t} = \boldsymbol{J} \tag{3.100}$$

applying Galerkin we obtain:

$$\boldsymbol{K}[\nu(\boldsymbol{x}(t))]\boldsymbol{x}(t) + \boldsymbol{M}(\sigma)\frac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}t} = \boldsymbol{f}(t) \tag{3.101}$$

where the dependance of the Stiffness matrix $\boldsymbol{K}$ on ν and of the Mass matrix $\boldsymbol{M}$ on σ is explicitely shown. Since the reluctivity depends on the field, ν depends on $\boldsymbol{x}$ and hence on t as indicated.

In this case in order to apply the HB method, we use a complex Fourier series for the solution considering N harmonics:

$$\boldsymbol{x}(t) \simeq \boldsymbol{x}_N(t) = \mathrm{Re}\left(\sum_{k=1}^{N} \boldsymbol{X}_k e^{jk\omega t}\right) \tag{3.102}$$

where $\boldsymbol{X}_k$ is the complex Fourier coefficient of the k-th harmonic at the angular frequency $k\omega$. It can be computed as:

$$\boldsymbol{X}_k = \mathscr{F}_k(\boldsymbol{x}) = \frac{1}{T}\int_T \boldsymbol{x}(t)e^{-jk\omega t}\,dt \tag{3.103}$$

If we substitute Eq.3.102 in Eq.3.101 and computing N Fourier coefficients of both sides, a system of equations with N times as many unknowns is obtained as there are degree of freedom in $\boldsymbol{x}(t)$:

$$\mathscr{F}_m\{\boldsymbol{K}[\nu(\boldsymbol{x}_N)]\boldsymbol{x}_N\} + jm\omega\boldsymbol{M}(\sigma)\boldsymbol{X}_m = \mathscr{F}_m(\boldsymbol{f}), \qquad m = 1, \ldots, N \tag{3.104}$$

In the linear term in Eq.3.104, the Fourier coefficients of the m-th harmonic appear only. The time derivative in Eq.3.101 corresponds to a multiplication by $jm\omega$ in the frequency domain. The right hand side can be computed directly using Eq.3.103. On the other hand, the nonlinear term containing the permeability or the reluctivity depending on the unknown solution couple all Fourier coefficients to each other. Therefore, due to the nonlinearity, we cannot solve each harmonic independently.

3.5.1 Linear Problems

In the linear case the permeability and the reluctivity does not depend on the solutions, in this case the decoupling is trivial. Infact, the system of ordinary differential equation 3.101 becomes linear since $\boldsymbol{K}$ does not depend on $\boldsymbol{x}(t)$. Then the Fourier coefficients related to the Stiffness matrix simply become:

$$\mathscr{F}_m\{\boldsymbol{K}(\nu)\boldsymbol{x}_N\} = \boldsymbol{K}(\nu)\boldsymbol{X}_m \tag{3.105}$$

Consequently each harmonic of Eq.3.104 becomes decoupled and can be determined independently:

$$[\boldsymbol{K}(\nu) + jm\omega\boldsymbol{M}(\sigma)]\boldsymbol{X}_m = \mathscr{F}_m(\boldsymbol{f}), \qquad m = 1, \dots, N \tag{3.106}$$

Therefore we have obtained N independent linear systems.

3.5.2 Fixed-Point Method for Nonlinear Problems

In nonlinear problems each harmonic is coupled with harmonics due to the presence of the permeability, to overcome this problem a fixed-point iteration technique is used.

The fixed-point iteration method for the solution of nonlinear equations reduces the problem to finding the fixed point of a nonlinear function. The fixed point $\boldsymbol{x}_{FP}$ of the function $\boldsymbol{G}(\boldsymbol{x})$ is defined as:

$$\boldsymbol{x}_{FP} = \boldsymbol{G}(\boldsymbol{x}_{FP}) \tag{3.107}$$

Then to proceed in the derivation we state the Banach Fixed Point Theorem:

Theorem 3 (Banach)**.** *Let $D \subset \mathbb{R}^n$ closed, and a contraction $\boldsymbol{G} : D \to \mathbb{R}^n$. Then there exists a unique $\boldsymbol{x}_{FP} \in D$ with $\boldsymbol{G}(\boldsymbol{x}_{FP}) = \boldsymbol{x}_{FP}$ and for any $\boldsymbol{x}^{(0)} \in D$ the fixed point iterates given by $\boldsymbol{x}^{(s+1)} = \boldsymbol{G}(\boldsymbol{x}^{(s)})$ converges to $(\boldsymbol{x}_{FP})$ as $s \to \infty$.*

We recall that a contraction is a function for which exists a contraction number $-1 < q < 1$ so that for any $\boldsymbol{x}$ and $\boldsymbol{y}$:

$$||\boldsymbol{G}(\boldsymbol{x}) - \boldsymbol{G}(\boldsymbol{y})|| < q||\boldsymbol{x} - \boldsymbol{y}|| \tag{3.108}$$

where $||\cdot||$ is a suitable norm.

A general nonlinear equation $\boldsymbol{F}(\boldsymbol{x}) = 0$ can be transformed in a fixed point problem by selecting a suitable linear operator $\boldsymbol{L}$ and defining $\boldsymbol{G}$ as:

$$\boldsymbol{G}(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{L}^{-1}\boldsymbol{F}(\boldsymbol{x}) \tag{3.109}$$

the fixed point iteration then becomes:

$$\boldsymbol{L}^{(s)}\boldsymbol{x}^{(s+1)} = \boldsymbol{L}^{(s)}\boldsymbol{x}^{(s)} + \boldsymbol{F}(\boldsymbol{x}^{(s)}), \qquad s = 0, 1, \dots \tag{3.110}$$

where the superscript s of $\boldsymbol{L}^{(s)}$ indicates that linear operator $\boldsymbol{L}$ can be changed at each iteration step to accelerate convergence.

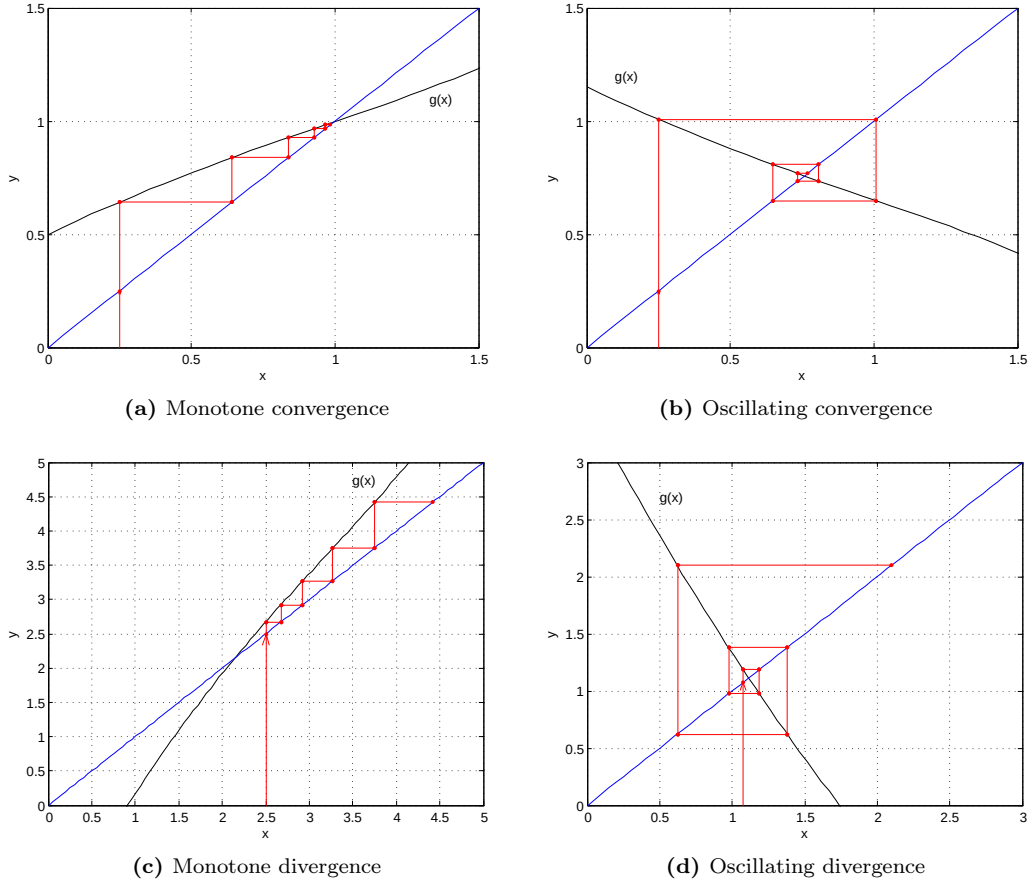
In this case the choise of the linear operator is straightforward, the permeability or reluctivity has to be set to a value independent of the magnetic field. This value, μ_{FP} or ν_{FP}, is not necessarily independent of the space coordinates, hence we can have $\mu_{FP} = \mu_{FP}(\boldsymbol{r})$ or $\nu_{FP} = \nu_{FP}(\boldsymbol{r})$, therefore the fixed point permeability or reluctivity can vary in the domain but has to be independent of the field and hence on time. In the same way as the general linear operator $\boldsymbol{L}$, μ_{FP} or ν_{FP} can also change at each iteration step.

Then we define the fixed point reluctivity as:

$$\nu = \nu_{FP} + (\nu - \nu_{FP}) \tag{3.111}$$

so that we substitute Eq.3.111 in Eq.3.100 and we obtain:

$$\nabla \times (\nu_{FP}\nabla \times \boldsymbol{A}) + \sigma\frac{\partial \boldsymbol{A}}{\partial t} = \boldsymbol{J} + \nabla \times ((\nu_{FP} - \nu)\nabla \times \boldsymbol{A}) \tag{3.112}$$



(a) Monotone convergence

(b) Oscillating convergence

(c) Monotone divergence

(d) Oscillating divergence

Figure 3.4: Converge of the FP method for $x = g(x)$

Galerkin's method applied to Eq.3.112 leads to the ordinary differential equation system:

$$\boldsymbol{K}(\nu_{FP})\boldsymbol{x}(t) + \boldsymbol{M}(\sigma)\frac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}t} = \boldsymbol{f}(t) + \boldsymbol{K}(\nu_{FP} - \nu)\boldsymbol{x}(t) \tag{3.113}$$

Eq.3.113 implies an iterative method, namely:

$$\boldsymbol{K}(\nu_{FP}^{(s)})\boldsymbol{x}^{(s+1)} + \boldsymbol{M}(\sigma)\frac{\mathrm{d}\boldsymbol{x}^{(s+1)}}{\mathrm{d}t} = \boldsymbol{f} + \boldsymbol{K}(\nu_{FP}^{(s)} - \nu^{(s)})\boldsymbol{x}^{(s)} \tag{3.114}$$

The permeability or reluctivity distributions $\mu^{(s)}$ or $\nu^{(s)}$ are determined from the solution $\boldsymbol{x}^{(s)}(t)$ which means that, in contrast to $\mu_{FP}^{(s)}$ or $\nu_{FP}^{(s)}$, they are time-dependent.

In each iteration step a linear ordinary differential equation system has to be solved, there are two approach to solve this system.

The first one consists in a time discretization using a backward Euler scheme to obtain:

$$\boldsymbol{K}[\nu(\boldsymbol{x}_m)]\boldsymbol{x}_m + \boldsymbol{M}(\sigma)\frac{\boldsymbol{x}_m - \boldsymbol{x}_{m-1}}{\Delta t} = \boldsymbol{f}_m \tag{3.115}$$

in this case m denotes time instants. Then one applies the HB method to Eq.3.115 giving rise to a block-structured system that has to be solve to determine the solution. The formal derivation of the method can be found in [7].

We focus our attention to a second method which use the HB method directly in the frequency domain. The basic idea of the harmonic balance method is that if the trigonometric Fourier coefficients of two functions are equal, then, under appropriate conditions, these

two functions are equal. Then if we apply the HB to Eq.3.114 equating the coefficients we obtain:

$$[\boldsymbol{K}(\nu_{FP}^{(s)}) + jm\omega\boldsymbol{M}(\sigma)]\boldsymbol{X}_m^{(s+1)} = \mathscr{F}_m[\boldsymbol{K}(\nu_{FP}^{(s)} - \nu^{(s)})\boldsymbol{x}^{(s)}(t) + \boldsymbol{f}(t)], \qquad m = 1, \ldots, N \tag{3.116}$$

The nonlinear iterations for solving Eq.3.116 are terminated when the maximum and the mean change of $\mu^{(s)}$ or $\nu^{(s)}$ beetween two iteration become less than a suitable threshold.

Eq.3.116 consists in N decoupled linear system, since these are independent from each other, they can be solved in parallel with each core responsible for the solution of one harmonic $\boldsymbol{X}_m^{(s+1)}$. Once these parallel computations are ready, the right hand side for the next iteration can be determined by first computing the solution in the time domain using Eq.3.102 and then carrying out the Fourier decomposition as in Eq.3.116. In this part of the process the parallelization is not possible, anyway the computational effort necessary for it is negligible in comparison to the solution of the large linear algebraic systems. The final algorithm is summarized in the flowchart of Fig.3.5.

Now we want to carry out an analysis of the convergence of the Fixed-Point HBFEM. Let's consider a nonlinear magnetic domain, it can be demonstrated that a value for the fixed-point permeability $\mu_{FP} < 2\mu_{min}$ ensures convergence in the static case. Here μ_{min} is the minimum slope of the BH-curve, which is equal to the permeability of vacuum. However, numerical investigations ([12]-[13]) show that higher values for μ_{FP} accelerate convergence, but if μ_{FP} exceeds a certain value, the fixed-point method diverges. The aim is to choose a fixed-point permeability higher than μ_0 which leaves the method convergent.

A first approach consists in the calculation of the maximum value of the magnetic flux density within a period. Therefore considering a generic point P we can calculate:

$$\boldsymbol{B}_{max} = \max\{\boldsymbol{B}(P,t)\}, \qquad 0 \leq t < T \tag{3.117}$$

The corresponding permeability for the maximum of the magnetic flux density at this point is given by:

$$\mu(P) = \frac{|\boldsymbol{B}_{max}|}{|\boldsymbol{H}(\boldsymbol{B}_{max}|)} \tag{3.118}$$

Therefore this value is choosen to be the fixed point permeability at the point P. For the iterative algorithm, the fixed point permeability in each integration point is replaced by Eq.3.118 at each iteration step. It is assumed that this choice will improve the convergence rate, since the value determined by Eq.3.118 is, in general, higher than μ_0.

Another approach relies in the determination of the optimal value for the fixed point permeability which garantees the acceleration of the convergence of the algorithm. The analysis of the convergence is carried out using the Banach Fixed Point theorem, in this case we have the function $\boldsymbol{G}(\boldsymbol{x})$ defined as:

$$\boldsymbol{G}(\boldsymbol{x}) = \mathscr{F}^{-1}[\boldsymbol{K}(\nu_{FP}) + jm\omega\boldsymbol{M}]^{-1}\mathscr{F}_m[\boldsymbol{K}(\nu_{FP} - \nu)\boldsymbol{x} + \boldsymbol{f}] \tag{3.119}$$

Since $\boldsymbol{x}$ depends on time and space we have to use two L_2 norms:

$$||\boldsymbol{x}||_{L_2(\Omega)} = \sqrt{\int_\Omega \boldsymbol{x}^2 \, d\Omega} \tag{3.120}$$

$$||\boldsymbol{x}||_{L_2(0,T)} = \sqrt{\int_T \boldsymbol{x}^2 \, dt} \tag{3.121}$$

In order to get the contraction number it is necessary to compute:

$$\boldsymbol{G}(\boldsymbol{x}) - \boldsymbol{G}(\boldsymbol{y}) = \mathscr{F}^{-1}[\boldsymbol{K}(\nu_{FP}) + jm\omega\boldsymbol{M}]^{-1}\mathscr{F}_m[\boldsymbol{K}(\nu_{FP} - \nu)(\boldsymbol{x} - \boldsymbol{y}) + \boldsymbol{f}] \tag{3.122}$$

This calculation ([12]) leads to an estimation for the contraction number:

$$q \leq C\left\| \frac{\nu_{FP} - \nu}{\nu_{FP}} \right\|_{L_2(0,T)} \tag{3.123}$$

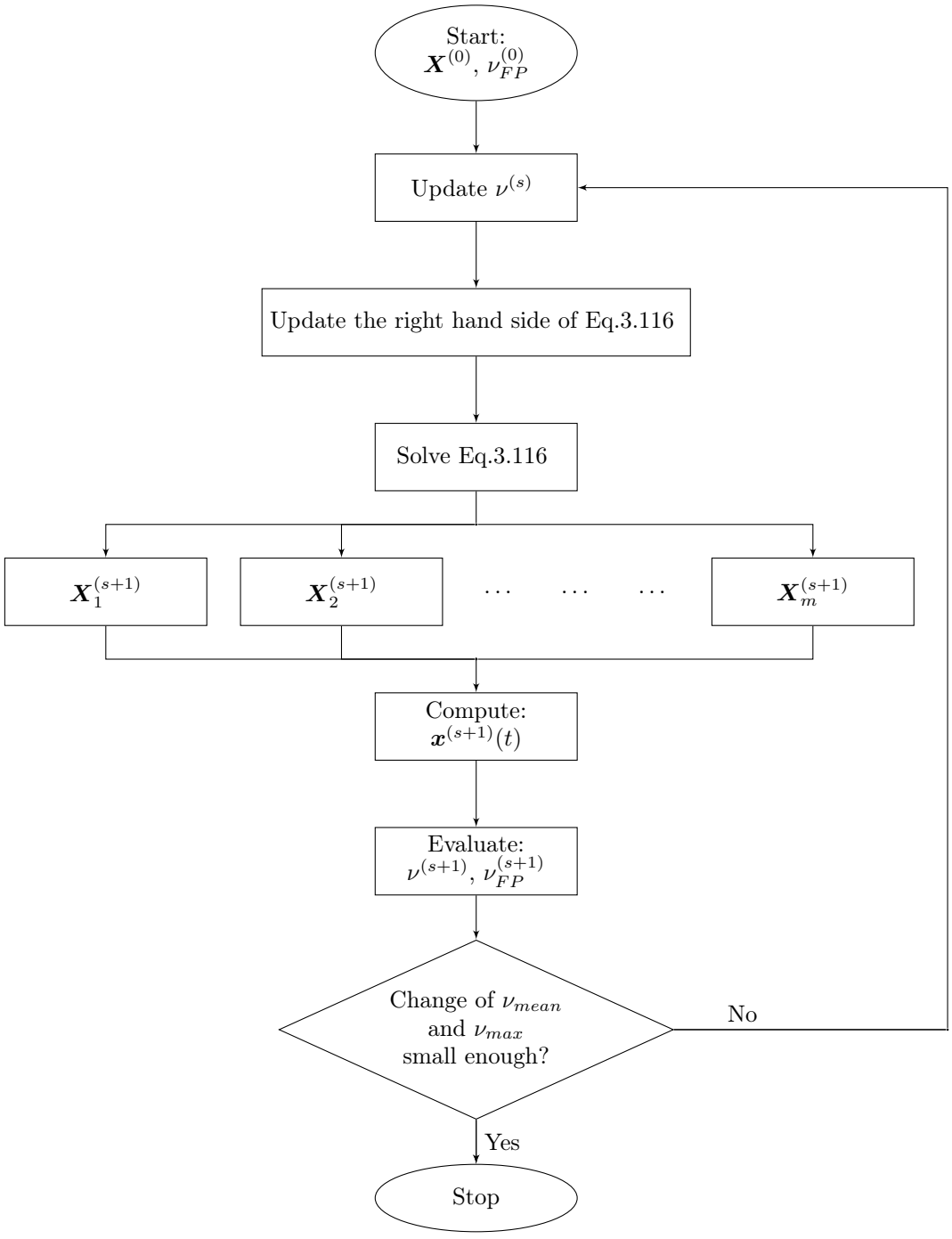


Figure 3.5: Flowchart of the Fixed Point HBFEM

$$q \leq C \left\| \frac{\nu_{FP} - \nu}{\nu_{FP}} \right\|_{L_\infty(0,T)} \tag{3.124}$$

where C is a constant near to one and independent from ν_{FP}. This estimation indicates that the contraction number strongly depends on the fixed-point reluctivity. The optimal value can be determined minimizing the right hand side of Eq.3.123-3.124. This minimization leads to the following values for ν_{FP}:

$$\nu_{FP} = \frac{\int_T \nu^2(t)\, dt}{\int_T \nu(t)\, dt} \tag{3.125}$$

$$\nu_{FP} = \frac{\min_{t\in[0,T]}(\nu) + \max_{t\in[0,T]}(\nu)}{2} \tag{3.126}$$

Once we have estimated the contraction number we can analyse the convergece of the method taking into account that the change of the reluctivity between two iteration can be expressed as:

$$\nu^{(s+1)} - \nu^{(s)} = (q-1)(1 - B'(H)\nu^{(s)})(\nu^{(s)} - \nu^{(s-1)}) \tag{3.127}$$

where $B'(H)$ is the derivative of the BH-curve. Eq.3.127 yields a condition for the convergence of the algorithm, the fixed-point reluctivity has to verify the equation:

$$||\nu(1 - B'(H)\nu)||_{L_\infty(0,T)} < \nu_{FP} \tag{3.128}$$

Then we can conclude that the best choice for the fixed-point reluctivity is:

$$\nu_{FP}^{opt} = \max \left\{ \frac{\int_T \nu^2(t)\, dt}{\int_T \nu(t)\, dt}, \frac{\min_{t\in[0,T]}(\nu) + \max_{t\in[0,T]}(\nu)}{2} \right\} \tag{3.129}$$

however if this value does not satisfy Eq.3.128, the optimal fixed-point reluctivity must be changed to:

$$\nu_{FP}^{opt} = ||\nu(1 - B'(H)\nu)||_{L_\infty(0,T)} \tag{3.130}$$

Selecting larger values for the optimal fixed-point reluctivity ensure convergence but lead to a higher number of iteration, on the contrary smaller value for the fixed-point parameter lead to divergence. The analysis has been carried out considering the fixed-point reluctivity, the optimal fixed-point permeability can be obtained in a similar way. Therefore to accelerate the convergence the fixed-point reluctivity can be assigned in each Gaussian integration point in accord with Eq.3.129-3.130, moreover considering linear elements, the fixed-point reluctivity is constant in each element of the mesh. In conclusion it can be seen that the condition given in Eq.3.130 depends on the BH curve, in particular for strongly saturating curves it can lead to values similar to the ones obtained using the first approach presented before.

Chapter 4

Implementation of the Harmonic Balance FEM

In this chapter we want to show how it is possible to implement the HBFEM in a commercial FEM solver. This type of implementation has already been done in various research codes, now in this work we want to extend this concept to a commercial software. We chose to implement the HB method in the commercial FEM software COMSOL Multiphysics using the MATLAB language. We made this choice due to the possibility to interface MATLAB and COMSOL easily using the LiveLink *for* MATLAB functionality of COMSOL, in particular we have used the AC\DC module to treat nonlinear electromagnetic problems. The aim is to demonstrate the feasibility of this approach and therefore give an alternative to the standard time-dependent simulations, moreover we want to use as much as possible the potentialities of COMSOL and it will be shown that embedding the HB method in such a software is relatively easy.

4.1 Choice of the Method

We have already shown that the HBFEM can be derived using different formulations, in particular we have analysed a strong coupled method, both in real and complex domain, and an approach with decoupling of the harmonics. We want to verify which approach leads to a feasible implementation in COMSOL taking into account the limitations due to the fact we want to use a commercial FEM solver.

We start our analysis considering the strong coupled approach. In the previous chapter we have shown that the most difficult part of the method consists in building up the Stiffness matrix. Infact due to the presence of the nonlinear reluctivity each harmonic is coupled with the others and a special calculation has to be performed, this means that the Stiffness matrix that arises from the application of the HBFEM can not be constructed using the standard functionality of COMSOL. On the contrary the term related to the Mass matrix can be determined considering each harmonic independently and therefore is easy to compute. Moreover the size of the Stiffness matrix of the HBFEM is $N_n(2N+1)$ where N_n is the number of degree of freedom of the model and N is the number of harmonics we are dealing with, again we note that to build up the Stiffness matrix it should be necessary to write a complete new code, in the same way to solve the nonlinear system using Newton-Raphson requires the implementation of a specific algorithm. This could be done if it is necessary to write a specfic code that implement the HBFEM starting from scratch but this is not the object of this work. The previous problems turned out to be difficult to be overcome and therefore we decided to focus our attention on the decoupled method.

Considering the approach with decoupling of harmonics it is immediate to note that the problems previously cited are overcome in a simple and natural way. In this case infact both the Stiffness and Mass matrix can be constructed considering each harmonic indipendently and therefore standard routines can be used. Moreover it is not necessary to use a method

such as Newton-Raphson due to the fact that the linearization is carried out using the fixed point method. It only remains to understand how to use the fixed point reluctivity ν_{FP}, this will be clarified in the next sections. For this reasons we decided to adopt this second approach and implement the fixed point HBFEM.

4.2 Implementation of the Fixed Point HBFEM

Once that the choice of the method has been performed, we want to show how to implement the fixed point HBFEM in COMSOL. In the previous chapter a complete analytical derivation has been presented, now to understand how to implement the algorithm we focus our attention on the iterative scheme obtained by the application of the fixed point and HB method:

$$[\boldsymbol{K}(\nu_{FP}^{(s)}) + jm\omega\boldsymbol{M}(\sigma)]\boldsymbol{X}_m^{(s+1)} = \mathscr{F}_m[\boldsymbol{K}(\nu_{FP}^{(s)} - \nu^{(s)})\boldsymbol{x}^{(s)}(t) + \boldsymbol{f}(t)], \qquad m = 1, \dots, N \quad (4.1)$$

Eq.4.1 tells us that in order to find the solution we have to solve N decoupled linear system for each iteration of the algorithm, the crucial part to implement the method is the handle of the fixed point parameter and consequently the construction of the matrices $\boldsymbol{K}(\nu_{FP})$ and $\boldsymbol{K}(\nu_{FP} - \nu)$. Now we analyse each passage to solve a complete model. The procedure we follow is represented in Fig.4.1.

4.2.1 Construction of the Model

This is a preliminary phase, to solve a model it is necessary to build up the geometry, set the boundary conditions, impose the excitations and construct the mesh. This is tipically done in any FEM software, in this case this process can be done using the GUI of COMSOL or directly via MATLAB. Anyway the final result has to be a MATLAB script with the instructions to construct the model.

4.2.2 Calculation of the Stiffness and Mass Matrix

This is the crucial part of the algorithm and special procedures has to be developed. Once we have constructed the model we can develop the working process to solve the our problem using the fixed point HBFEM. We recall the algorithm is described by Eq.4.1, hence is necessary to determine $\boldsymbol{K}(\nu_{FP})$, $\boldsymbol{M}(\sigma)$ and $\boldsymbol{K}(\nu_{FP} - \nu)$. Note that the Mass matrix has to be calculated only in the first iteration given the fact that it does not depend on the reluctivity.

We start our analysis from the calculation of $\boldsymbol{M}(\sigma)$. This term is easy to be determined, it can be calculated using the built-in function `mphmatrix`, the known excitation $\boldsymbol{f}$ can be calculated similarly.

The Stiffness matrix $\boldsymbol{K}(\nu_{FP})$ has to be calculated imposing the value of ν_{FP} in the model, we used the fixed point permeability μ_{FP} to ease the implementation of the algorithm. If we assume a constant value for the fixed point permeability, it means that $\boldsymbol{K}(\nu_{FP})$ has to be calculated only in the first iteration. Therefore once we have set in the model the choosen value of the fixed point permeability it is sufficient to calculate the Stiffness matrix using again the function `mphmatrix`. On the contrary adopting the adaptive method a different implementation is required. In this second case $\boldsymbol{K}(\nu_{FP})$ has to be calculated on each iteration, moreover μ_{FP} is not constant in the whole domain and assumes different values in each Gaussian integration points. In our implementation we use linear elements, in this case μ_{FP} is constant in each element of the mesh, so that it is necessary to create a MATLAB function that sets the optimal value of the permeability in each element of the mesh, in particular the COMSOL variable `meshelement` turned out to be useful. Moreover is higly recommended to vectorize the function avoiding `for` loops over the elements to speed up the code.

We also have to determine the matrix $\boldsymbol{K}(\nu_{FP} - \nu^{(s)}(t))$. In an iteration this calculation has to be performed for each time instant in wich we sample $\mu^{(s)}(t)$. The value of $\mu^{(s)}(t_i)$

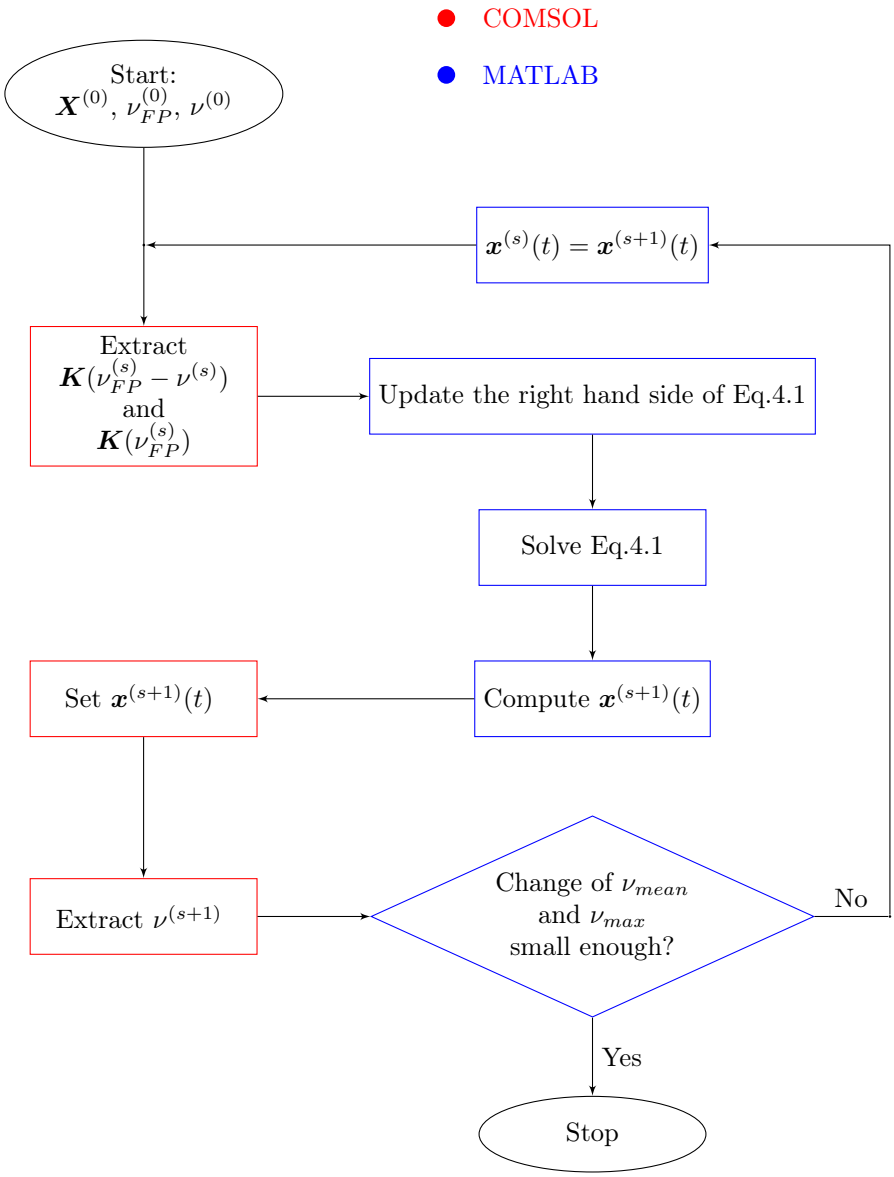


Figure 4.1: Flowchart of the algorithm

is calculated from the solution $\boldsymbol{x}^{(s)}(t_i)$, to do that we set the time domain solution in the model using the function `setU` and then we extract the permeability, which is constant in each element, using for example `mphinterp`. This process has to be done for each time instant in which we have previously calculated the solution. We resume the previous procedure in the following pseudocode:

Algorithm 1

```
N_ts=Number of time instants
for i ← 1, N_ts do
    Set x(t_i)
    Extract μ(t_i)
    Set (μ_FP − μ(t_i))
    Compute K(ν_FP − ν(t_i))
end for
```

Once we have done this process we can update the right hand side as:

$$\mathscr{F}_m[\boldsymbol{K}(\nu_{FP}^{(s)} - \nu^{(s)})\boldsymbol{x}^{(s)}(t) + \boldsymbol{f}(t)] = \mathscr{F}_m[\boldsymbol{K}(\nu_{FP}^{(s)} - \nu^{(s)})\boldsymbol{x}^{(s)}(t)] + \mathscr{F}_m[\boldsymbol{f}(t)] \tag{4.2}$$

In order to perform this calculation we used the FFT algorithm of MATLAB.

4.2.3 Solution of the Linear Systems

At this point it only remains to solve the linear systems that arise from the previous calculation. We can distinguish two cases.

The first one is the eddy current free problem, in this case Eq.4.1 simpifies to:

$$\boldsymbol{K}(\nu_{FP}^{(s)})\boldsymbol{X}_m^{(s+1)} = \mathscr{F}_m[\boldsymbol{K}(\nu_{FP}^{(s)} - \nu^{(s)})\boldsymbol{x}^{(s)}(t) + \boldsymbol{f}(t)], \qquad m = 1, \dots, N \tag{4.3}$$

Therefore it appears that $\boldsymbol{K}(\nu_{FP}^{(s)})$ does not change considering different harmonics, it means that we have obtained a linear system with multiple right hand side. In this case we solve Eq.4.3 adopting a factorizarion of the Stiffness matrix. Using for example an $\boldsymbol{LU}$ factorization we obtain the well known form:

$$\begin{cases} \boldsymbol{L}\boldsymbol{y}_m = \boldsymbol{b}_m \\ \boldsymbol{U}\boldsymbol{x}_m = \boldsymbol{y}_m \end{cases} \qquad m = 1, \dots, N \tag{4.4}$$

The first system is solved using forward substitutions while the second one is solved by backward substitutions. The main advantage of this approach relies in the fact that the factorization of the matrix has to be done only once. Infact the same triangular factor $\boldsymbol{L}$ and $\boldsymbol{U}$ are used to solve the m linear systems. We underline that this procedure is internally carried out using the MATLAB function `mldivide`, which automatically adopts the most efficient algorithm for the factorization of the given matrix.

Considering eddy current the previous considerations are not valid, in this case we have to solve Eq.4.1. This is the part that should be solved in parallel in order to increase the efficiency of the code. However, in this work we have limited our analysis to 2D problems using linear finite elements. In this case the matrices that arise from a FEM discretization are relatively small, therefore good speed is achieved even without parallelization. It is clear that to tackle 3D problem or use higher order elements the complexity dramatically grows and parallelization is required. Moreover we note that the adoption of direct method as the LU factorization is tipically a good choice for relatively small matrices as the one we treat in this work. When the dimension of the linear system increase, for example in 3D problems, is usually required to use iterative method in order to reduce the computational time and effort.

4.2.4 Termination Criterion

The iterations has to be performed until a suitable criterion is satisfied.

A first approach is to stop the algorithm once the variation of magnetic vector potential in each node of the finite element between two consecutive iterations is below of an imposed threshold.

The criterion we use in this work is the one proposed in [7], it consists in imposing a condition on the permeability. In particular when the maximum and the mean relative variation of the permeability between two consecutive iterarion is below of an imposed threshold we stop the iterative process. This means that the solution is not varying anymore and we have reached convergence. For most of the problems a good choice is to impose a threshold equal to 0.1% for the mean relative variation and 1% for the maximum relative variation of the permeability in the integration points. These conditions are easily checked in MATLAB during each iteration of the algorithm.

4.2.5 Post-processing

An important detail is the post-processing phase and consequently the visualization of the results.

Due to the fact we are working with MATLAB we have the possibility to visualize and elaborate the obtained results in the MATLAB Command Window or in COMSOL. To facilitate the post-processing of the results the code devoleped automatically sets the results in COMSOL, in this way we can work on the solved model directly through the GUI of this software without the necessity to develope other code. We can also run several simulations using different number of harmonics and load easily the computed solutions in COMSOL (Fig.4.2).

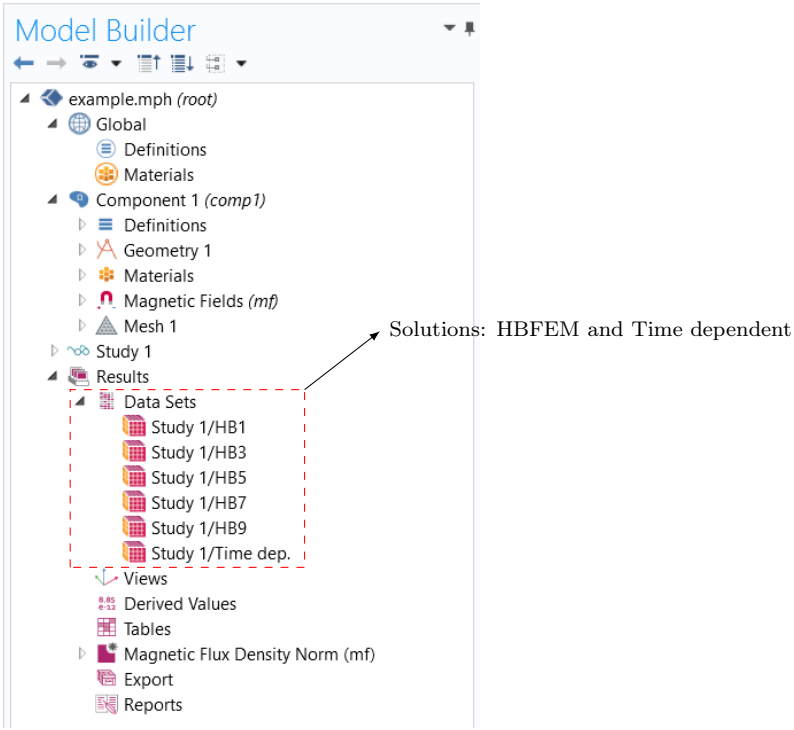


Figure 4.2: Screenshot of the Model Builder with the solutions of the HBFEM

Setting the solutions through MATLAB is easily performed using the function `setU`, we underline that the determined values has to be imposed for each time instant in which we have computed the solutions. Once the solution is set is sufficient to create the solution and load the corresponding model.

We note that the application of the HB requires the calculation of the solution both in time domain and in frequency domain, therefore it is possible to see the computed magnetic vector potential harmonic by harmonic. Even in this case is easy to set the solution in each node of the mesh and load the corresponding model in COMSOL similarly to what can be done with the steady state solution in time domain.

4.3 Working Process

To conclude this chapter we resume the working process we have developed in the previous section, for this purpose we refer to Fig.4.3.

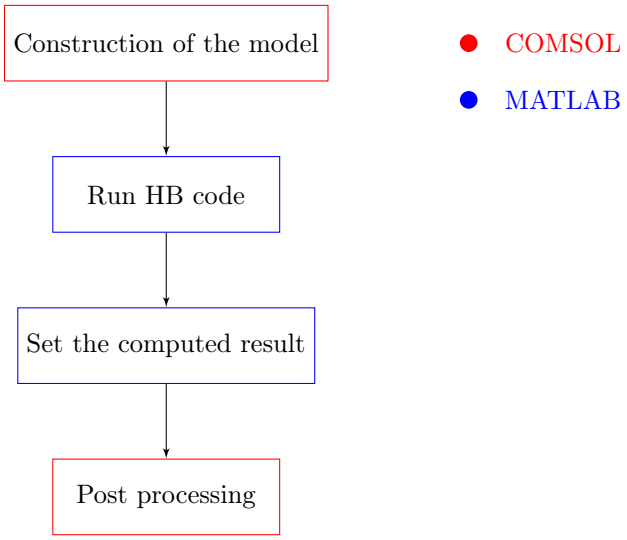


Figure 4.3: Working process to solve a model

As can be seen from the previous figure the working process is linear and quite simple and does not require complicated manipulations. The code which implement the HB method has to be slightly changed and adapted according to the specific model we have to analyse, for example if we consider a problem with eddy currents or not. The main script and functions implemented are described in Appendix B. The application of the method developed is the objective of the next chapter.

After the model has been created in COMSOL, the obtained script is processed via MATLAB applying the code developed which implements the HB method. Once the solution has been calculated, the obtained values of the magnetic vector potential are set in the corresponding model. At this point the model can be loaded and the post processing phase can be done directly through the GUI of COMSOL.

The principles we have followed to propose this process are mainly the simplicity of the working flow and its applicability to a wide range of different models. In the next chapter we will follow the procedure developed and we will test the code proposed in some practical examples.

We note that a lot of parts of the procedure are tipically done to solve any kind of model, for example the creation of the geometry and the set up of the boundary conditions, infact the core of the working flow is the code which implements the HB method and it is also the process which takes the most of the computational time. Moreover the time required to solve the corresponding model depends on the model itself. The computational time is crucial for a feasible and wide utilisation of our code. At the moment we have seen through numerical investigations that the time required to reach the steady state solution using a time dependent simulations is less than the one required to our code. The code developed can be further ammeliorated but there are some serious limitations due to the fact we are dealing with a commercial solver. Infact the use of some built-in COMSOL functions requires

a relative long amount of time, for example the repeated use of the function `mphmatrix` used to construct the Stiffness matrix in each time sample turned out to be particularly inefficient. It should be essential to understand if these problems can be overcome in order to make more attractive the implementation we have proposed.

Chapter 5

HBFEM: Application of the Method

In this chapter we want to apply the method proposed studying some significative 2D models. In order to validate the code we confront the solution obtained using the HBFEM with one steady state solution obtained using a standard time-dependent simulation, the results must present similar behaviour. We will analyse three different models that permit to make some useful considerations.

5.1 Ferromagnetic Yoke

The first example to test the code consists in a nonlinear ferromagnetic core and a current driven coil surrounded by air. The geometry of the model is represented in Fig.5.1.

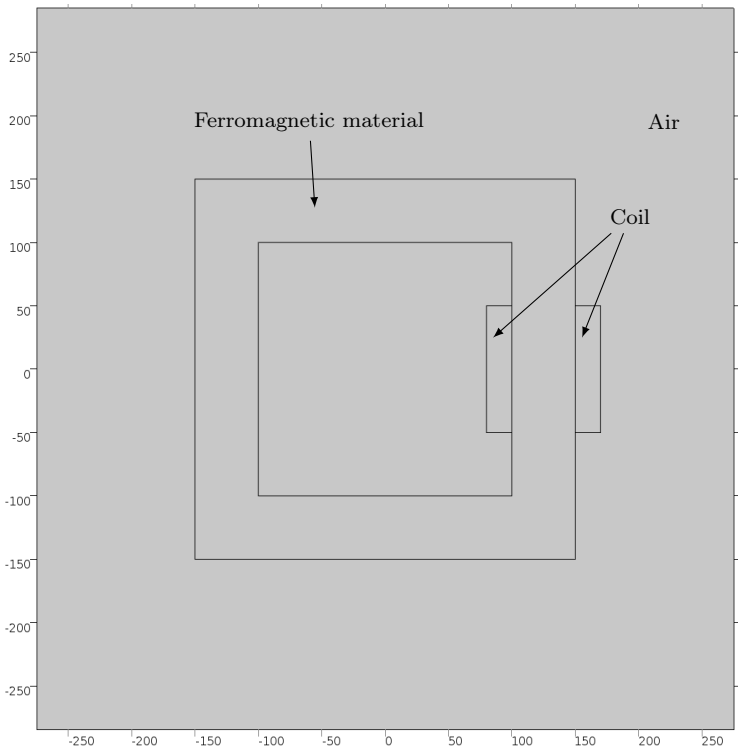


Figure 5.1: Nonlinear ferromagnetic yoke

In this first example we consider an eddy current free problem, infact the conductivity

of the magnetic material is set to zero. Moreover we assume to have a uniform distribution of the current density in the coil, to have this it is sufficient to impose a null conductivity and impose the desidered current density excitation. In order to achieve a strong saturation condition we impose a sinusoidal current density with peak value equal to $1 \times 10^6\,\mathrm{A/m^2}$, the frequency is set to 50 Hz. For the magnetic yoke we used a COMSOL built-in silicon steel whose BH curve is depicted in Fig.5.2.

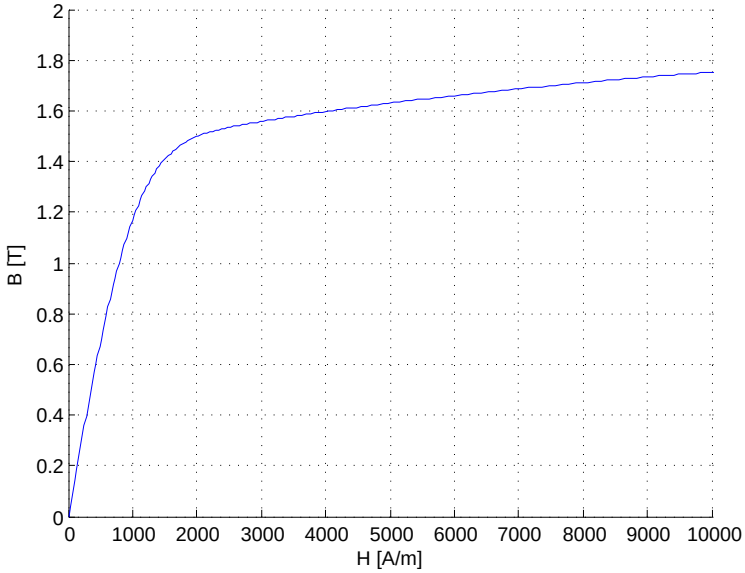


Figure 5.2: BH curve of the magnetic material

We recall we use a linear finite element discretization. Then we have run different simulations using the HBFEM considering different number of harmonics and we have compared the obtained results with a standard time dependent simulation. The solution is obtained in term of magnetic vector potential A_z. In particular we can evaluate the solution in the nodes represented in Fig.5.3.

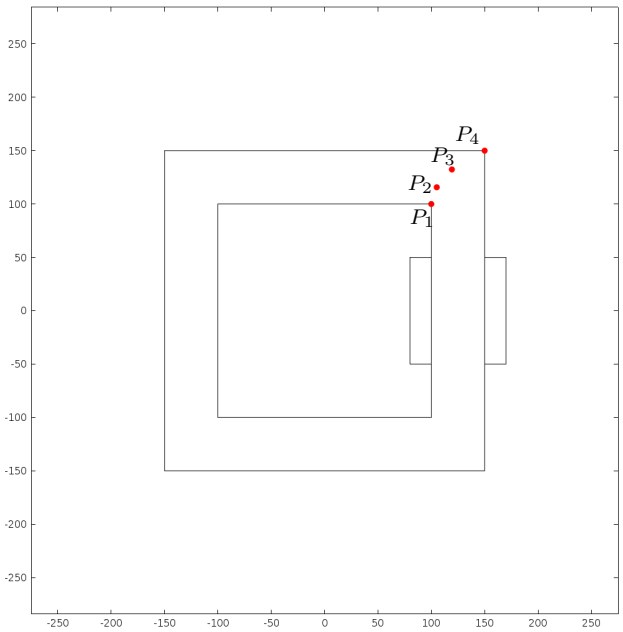
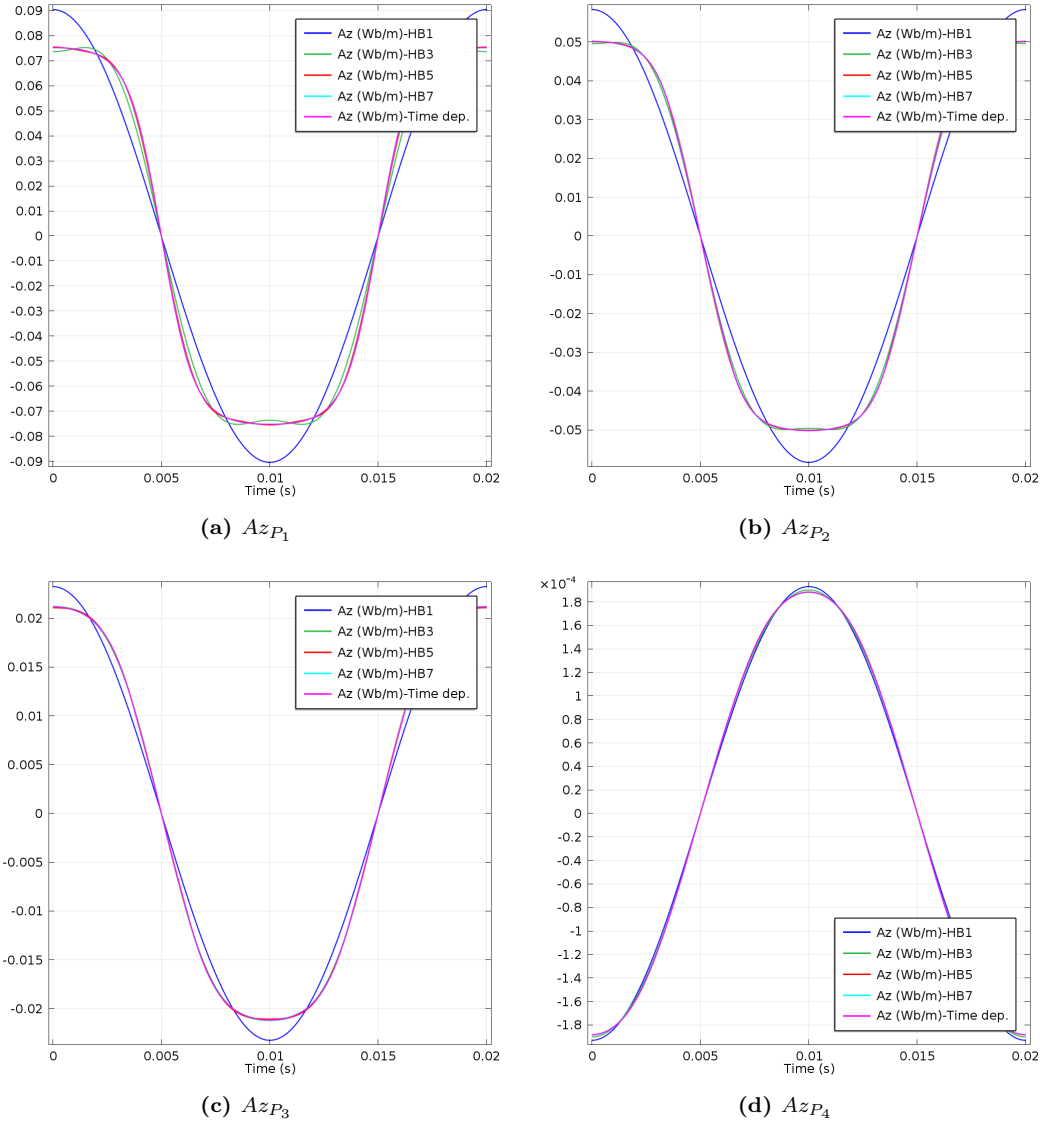


Figure 5.3: Nodes P_1, P_2, P_3 and P_4



(a) Az_{P_1} **(b)** Az_{P_2}

(c) Az_{P_3} **(d)** Az_{P_4}

Figure 5.4: Magnetic vector potential in points P_1, P_2, P_3 and P_4

The obtained solutions are depicted in Fig.5.4, moreover having imposed a pure sinusoidal excitation the solution presents only even harmonics. The results calculated using the HBFEM are closed to the ones obtained via time dependent simulation, increasing the number of harmonics the solution tends to be more accurate. In particulare in point P_4, where the saturation is very low, the solution is well approximated even using a small number of harmonic components. Moreover the fundamental quantity is the magnetic flux density, in this sense we can proceed similarly to what has been done for the magnetic vector potential. Using linear finite elements we ensure that the magnetic flux density is constant within each element. Therefore we can compare the magnetic flux density in some specific elements obtained using the HBFEM and the time dependent simulation. We choose three elements with different levels of saturation, we identify them by the coordinate of their barycenter (Fig.5.5). Even in this case the magnetic flux density tends to the time dependent solution increasing the number of harmonics, in particulare for point E_1 where the saturation is strong a relatively high number of components is needed in order to achieve an accurate result (Fig.5.6).

We have already discussed about the importance of a proper choice for the fixed point permeability, now we want to use this example to clarify some aspects. A correct choice of

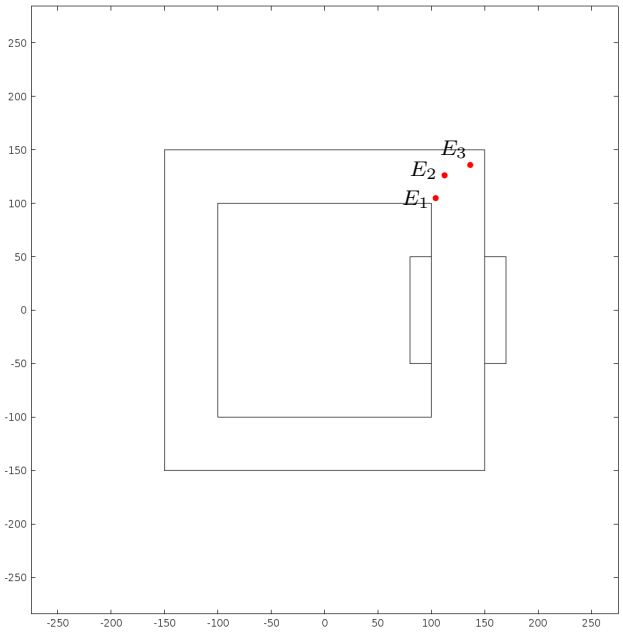
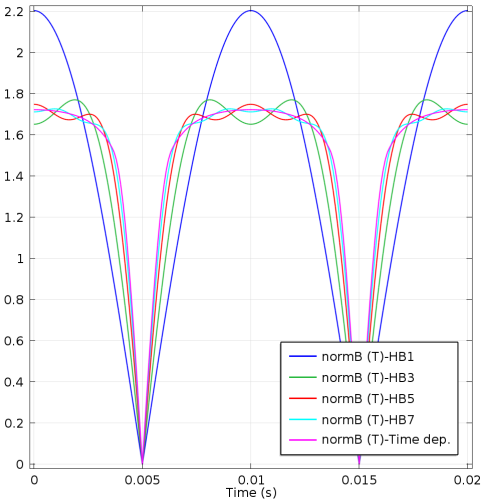


Figure 5.5: Elements E_1, E_2, E_3

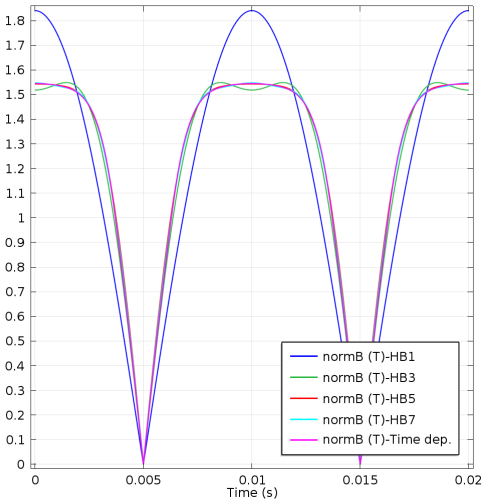
the fixed point permeability is essential to guarantee the convergence of the algorithm and the speed of convergence. The higher the value of μ_{FP} is, the faster the convergence. On the other hand if the value of the fixed point permeability is too high the algorithm does not converge. Moreover it's not possible to know a priori which is the maximum admissible value for μ_{FP}. In this sense an adaptive approach is useful. Using an adptive method infact the algorithm converges whatever it is the initial value of the fixed point permeability and the number of iterations to reach convergence is greatly reduced. There are several approaches to solve this problem, we have choosen the first one presented in Chapter 3 and formally derived in [13]. It's easy to implement and leads to a strong reduction of the iteration needed to reach convergence. In this example, due to the strong nonlinearity intentionally introduced, we have calculated the value of μ_{FP} using the adaptive method and then we have slightly decreased its values in order to ensure the convergence of the algorithm.

In Fig.5.7-5.8 it is shown the rate of convergence of the HBFEM considering different value for the fixed point permeability.

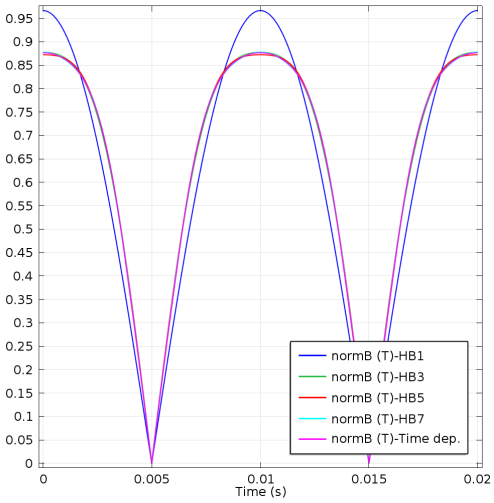
The stopping criterion is choosen to be on the maximum and on the mean variation variation of the permeability between two consecutive iterations. If the value of the fixed point permeability is taken higher than $50\,\mathrm{H/m}$ the algorithm does not converge, on the contrary using the adaptive method the initial value of μ_{FP} can be choosen arbitrarily. Moreover the speed of convergence of the adaptive approach is in any case higher.



(a) $||B||$ in E_1



(b) $||B||$ in E_2



(c) $||B||$ in E_3

Figure 5.6: Norm of the magnetic flux density in points E_1, E_2, E_3

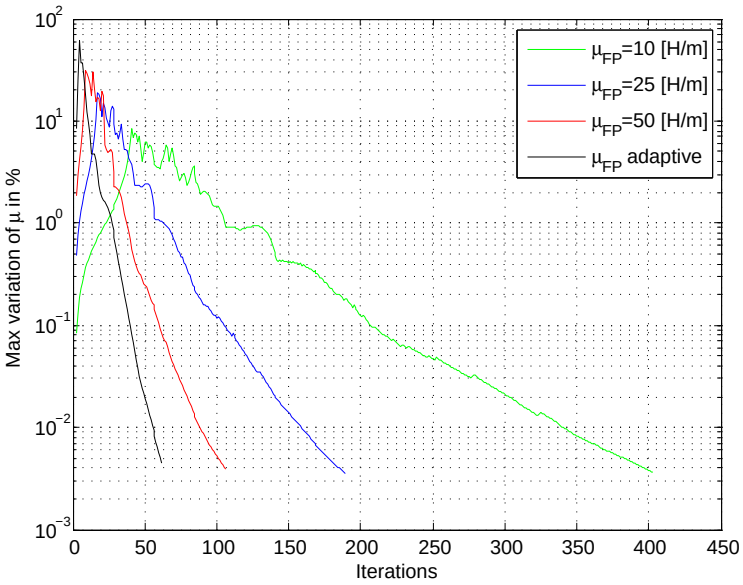


Figure 5.7: Maximum variation of the permeability

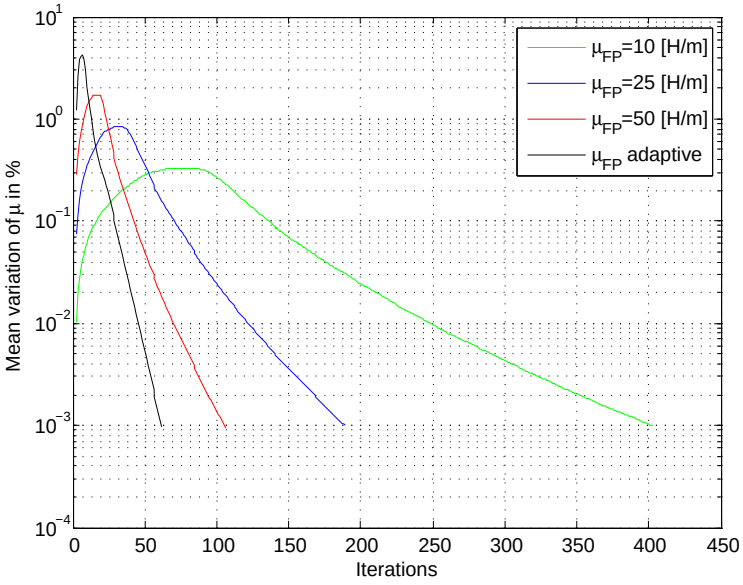


Figure 5.8: Mean variation of the permeability

5.2 Single-phase Transformer

The HBFEM is widely adopted in the analysis of electrical machine which tipically present relatively slow dynamic, we refer both to rotating and static machines. In this second example we apply our code to a realistic problems namely the single-phase transformer represented in Fig.5.9, the extension to three-phase transformers or reactances is effortless.

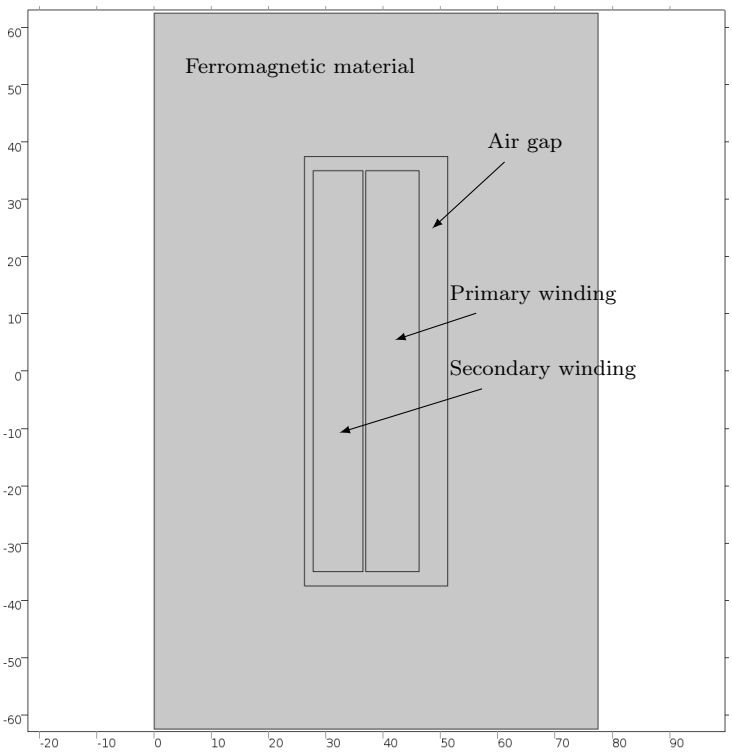


Figure 5.9: Single-phase trasformer

Due to the simmetry, we have analysed only one half of the trasformer. In this case we have used a different nonlinear ferromagnetic material, its BH curve is depicted in Fig.5.10. Even in this example the conductivity of the ferromagnetic material is set to zero.

Regarding the excitation we used a voltage driven multi-turn coil. The extension to this kind of source does not imply any significant changes in the code developed. It's only necessary to remember that in this case the total current of the coil has to be considered as an unknown of the problem, anyway the structure of the code remains the same infact the Stiffness matrix and the known excitation vector are coherently built up in COMSOL. The imposed voltage is sinusoidal with a peak amplitude equal to 125 V and frequency set to 50 Hz.

As we have done in the previous example we check the accuracy of the solution obtained using our code compared to the one obtained via time dependent simulation. Therefore we identify some elements with different level of saturation (Fig.5.11) and we calculate the norm of the magnetic flux density in a period (Fig.5.12).

Even in this example the magnetic flux density obtained using the HB method tends to be closer to the steady state solution of the time dependent calculation if a reasonable number of harmonics is taken into account. The same considerations hold for the magnetic vector potential in each node of the mesh.

Moreover with our code we can easily extend the analysis to our important concepts. Applying the HBFEM infact we deal both with the time dependent solution and with the frequency domain solution considering each harmonic decoupled, this means we can derive important results harmonic by harmonic. In particular we have the real distribution of each harmonic of the magnetic flux density in the nonlinear ferromagnetic core (Fig.5.13),

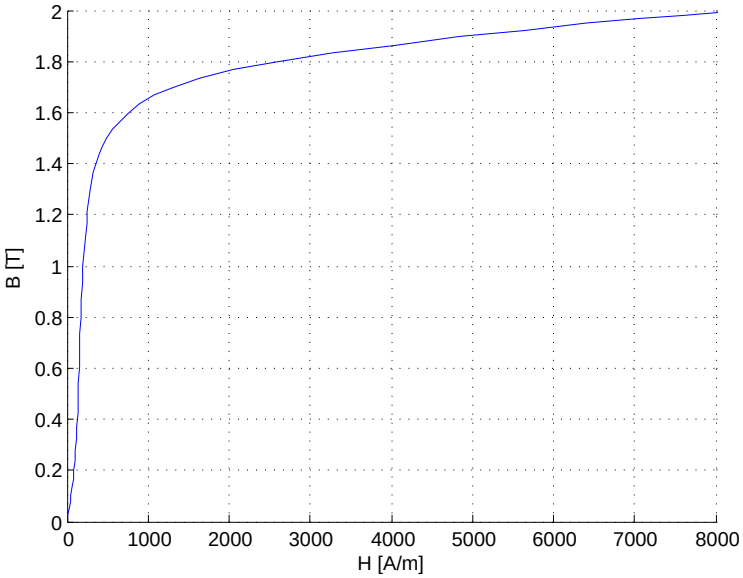


Figure 5.10: BH curve of the magnetic material

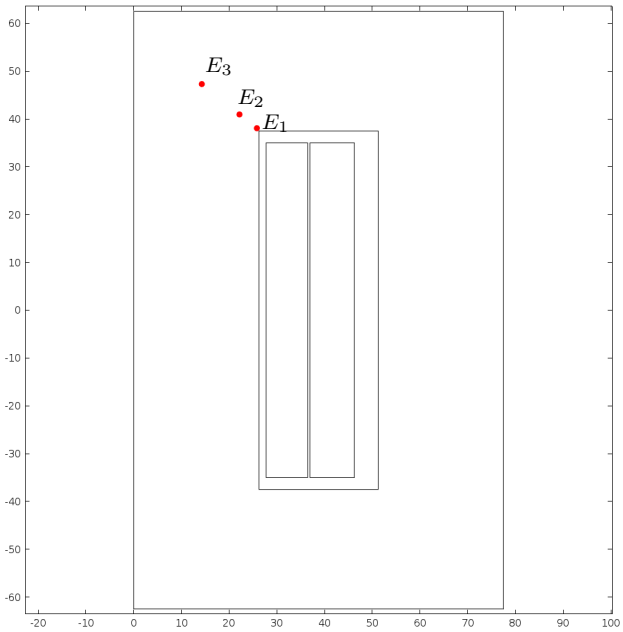
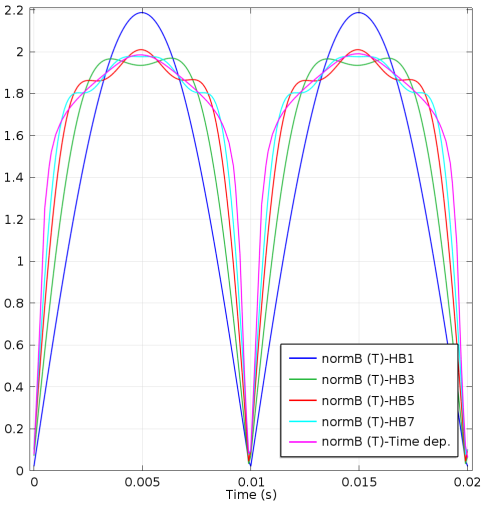
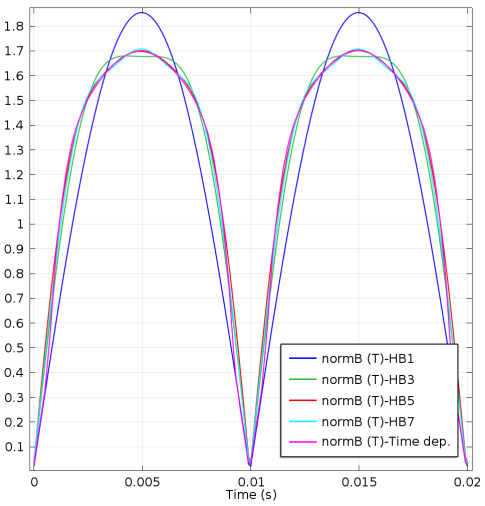


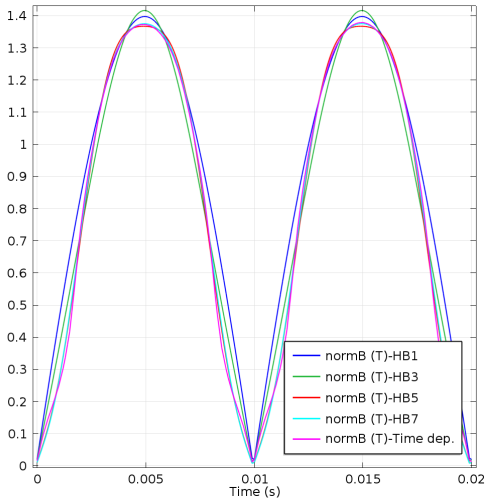
Figure 5.11: Elements E_1, E_2, E_3



(a) $||B||$ in E_1



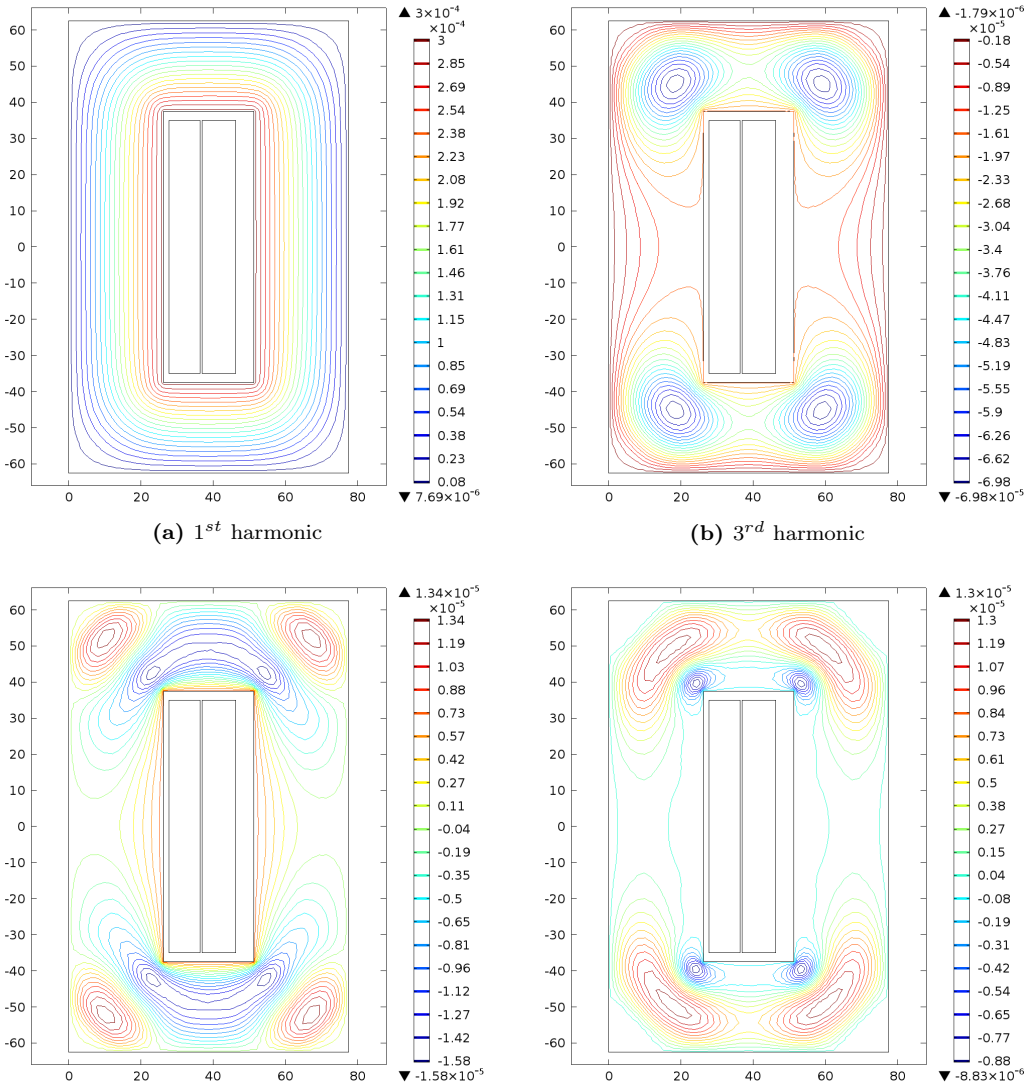
(b) $||B||$ in E_2



(c) $||B||$ in E_3

Figure 5.12: Norm of the magnetic flux density in points E_1, E_2, E_3

therefore it's possible to check harmonic by harmonic how the transformer is magnetically loaded. The amplitudes and the phases are directly calculated usign the HBFEM.



(a) 1^{st} harmonic

(b) 3^{rd} harmonic

(c) 5^{th} harmonic

(d) 7^{th} harmonic

Figure 5.13: Flux lines of the harmonics of B

We underline this is the correct frequency domain solution for each harmonic and it's totally different from carrying out the corresponding time-harmonic simulations which formally does not make sense for the nonlinear problems we are treating.

5.3 Coil Over a Nonlinear Ferromagnetic Plate

The third example we analyse consists in an eddy current problem, the geometry of the model is depicted in Fig.5.14-5.15.

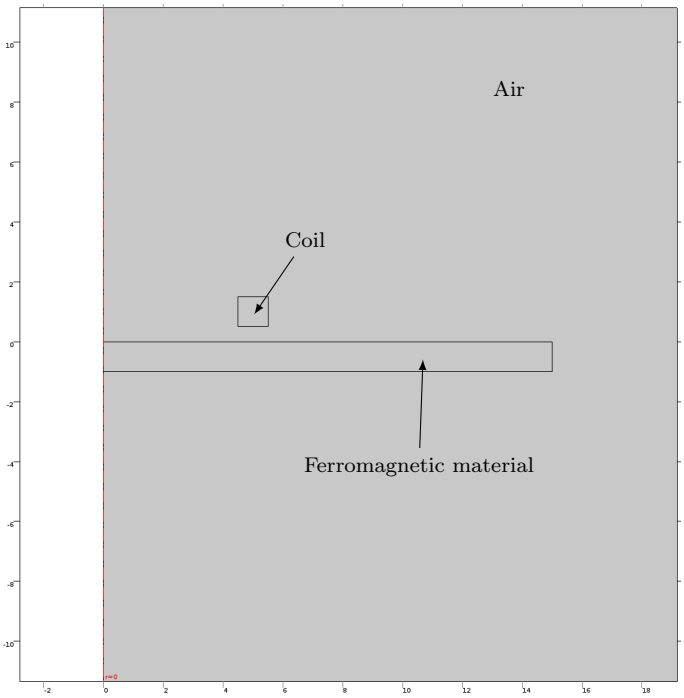


Figure 5.14: Geometry of the model

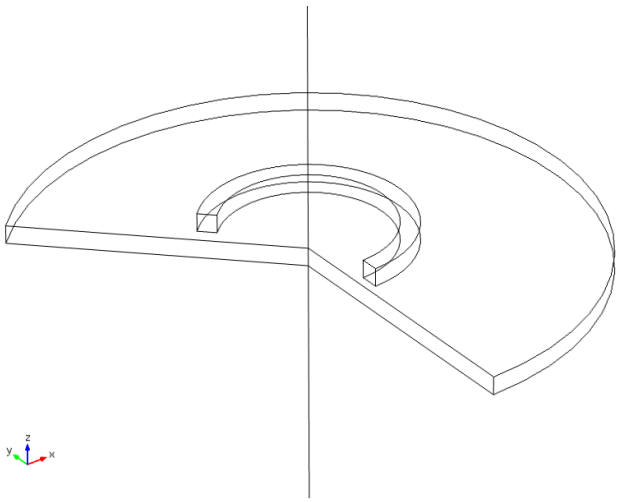


Figure 5.15: Slice 3D view of the model

We use the same steel of the second example whose BH curve is represente din Fig.5.10, in this case the conductivity is equal to 8.41 MS/m. The current driven coil is fed by a sinusoidal current density with peak value equal to $2.5 \times 10^7\,\mathrm{A/m^2}$, the frequency is set to 50 Hz.

We remark that example concerns an eddy current problem, the code developed is equally valid for this kind of study. It is sufficient to consider also the imaginary part of the linear systems that arises from the application of our code. Moreover the model is studied using a 2D axisymmetric domain. This means the spatial variable (x, y) must be changed to (r, z)

in order to avoid error when using the HBFEM. No other modifications are needed. Even for this example the steady state solution of the time dependent simulation are compaed with the one obtained using the HBFEM.

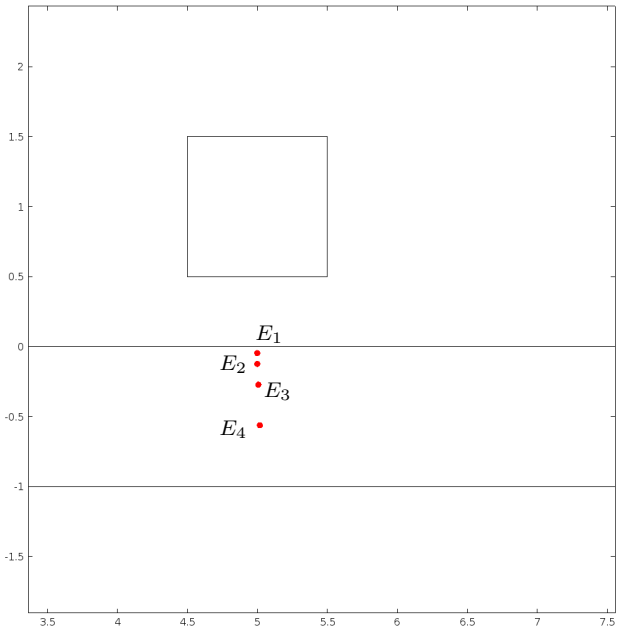
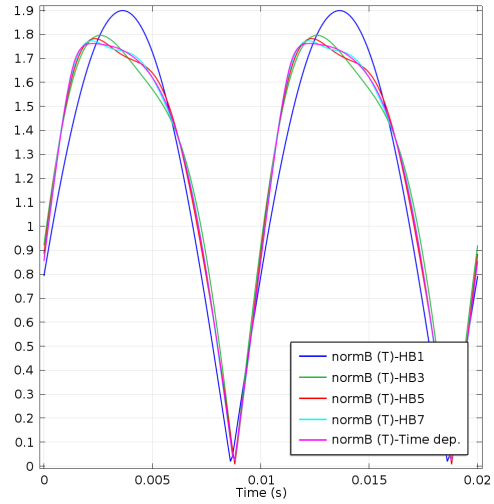


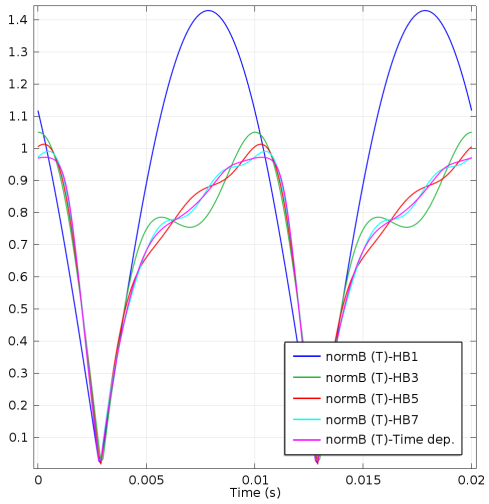
Figure 5.16: Elements E_1, E_2, E_3 and E_4

Again we identify some element (Fig.5.16) and we compare the norm of the magnetic flux density. The obtained results are depicted in Fig.5.17. As in the previous examples the solution computed using the HBFEM tends to the one calculated via time dependent simulation when the number of harmonics taken into account increases.

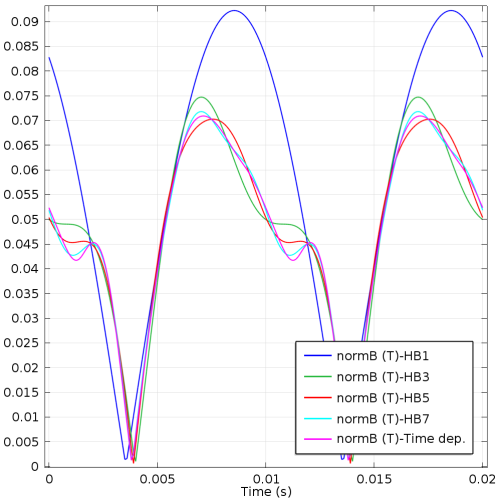
For this model we can check other quantities related to the eddy current induced in the ferromagnetic plate. For example we can compute the current density in the barycenter of the elements E_1, E_2, E_3 and E_4, to achieve accurate results we have considered an approximation made by nine harmonics (Fig.5.18). Another important quantity is the one related to the Joule losses (Fig.5.19). Even in this case there is a good agreement in the computed solution.



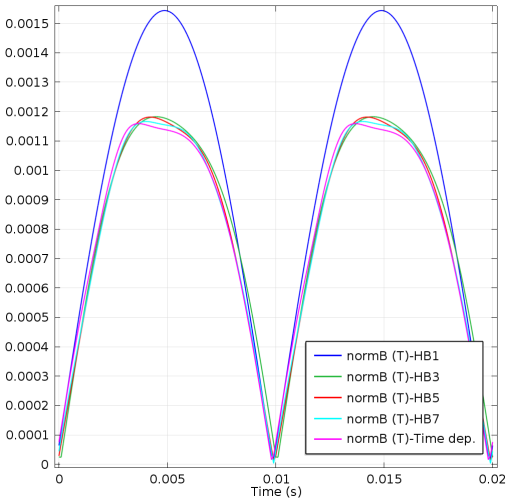
(a) $||B||$ in E_1



(b) $||B||$ in E_2

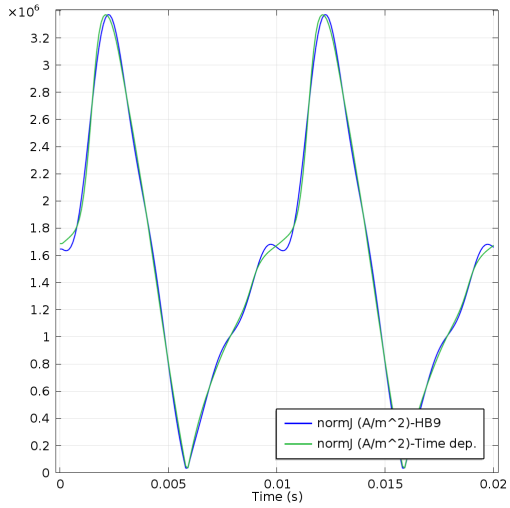


(c) $||B||$ in E_3

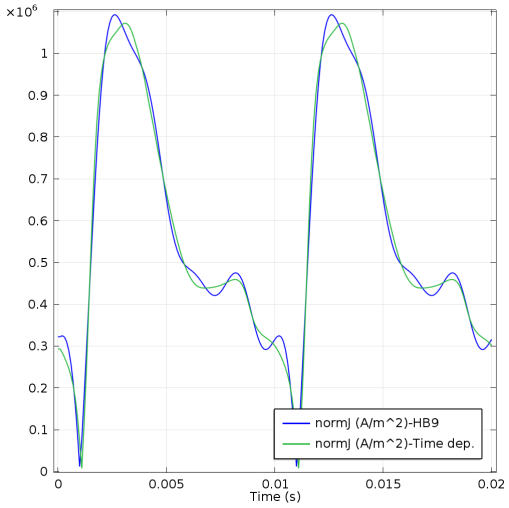


(d) $||B||$ in E_4

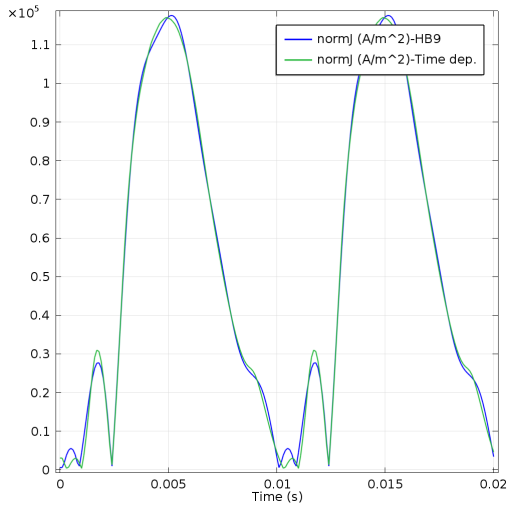
Figure 5.17: Norm of the magnetic flux density in elements E_1, E_2, E_3 and E_4



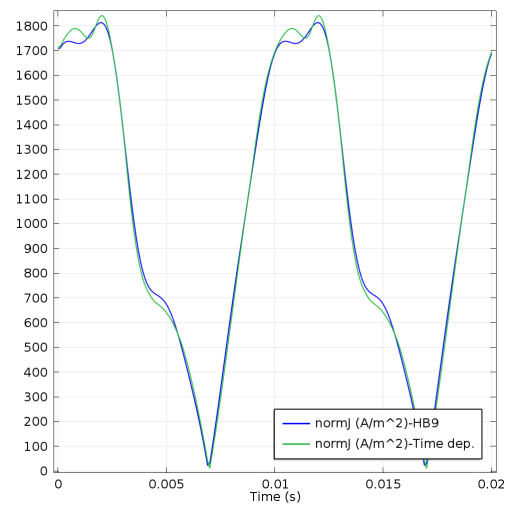
(a) $||J||$ in E_1



(b) $||J||$ in E_2



(c) $||J||$ in E_3



(d) $||J||$ in E_4

Figure 5.18: Norm of the current density in points E_1, E_2, E_3 and E_4

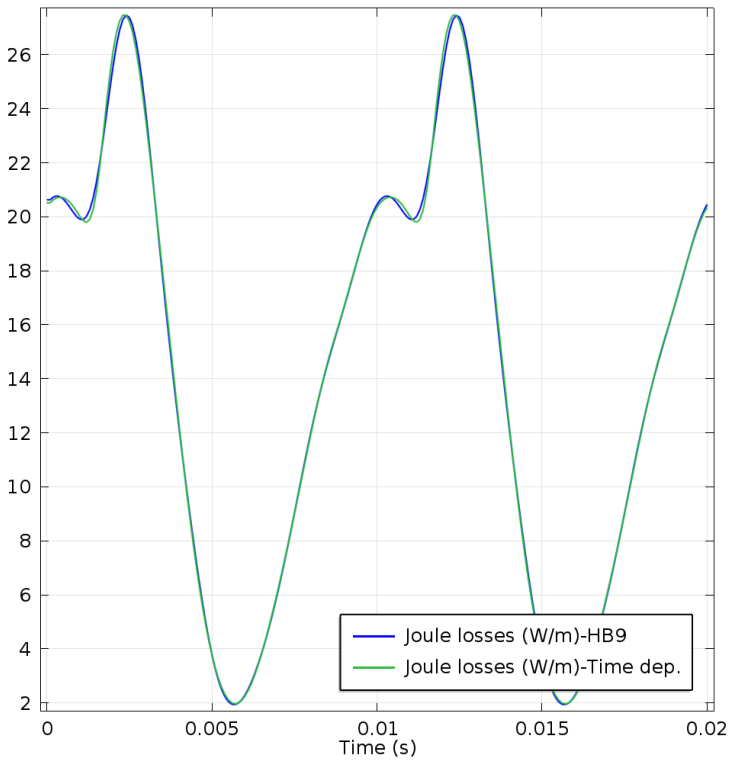


Figure 5.19: Joule losses

Chapter 6

Conclusions

In this work we have treated the Harmonic Balance analysis applied both to electrical circuit and field problems.

The first step consisted in giving a mathematical formulation of the method, presenting its properties and peculiarities, in order to have a theoretical base to develop our work.

Then we have applied the method analysing nonlinear electrical circuits. We have shown how the HB analysis applied to nonlinear circuits works and how to implement a code to solve this kind of problems. Then the algorithm developed has been tested in an example and observations about the solutions obtained have been presented.

The main part of this thesis consisted in the study of the HB method applied to field problems, therefore we have treated in detail the so called Harmonic Balance Finite Element Method. We have derived this method presenting various formulations and we have presented its pros and cons in the determination of the steady-state solution of time-dependent nonlinear problems with respect to standard time-stepping calculations. Then we have shown the working process to develop a code based on the HBFEM. The originality of this work is related to the choice of integrating the HB analysis in an existing FEM commercial software in respect of what has already been done with research codes based on FEM. We have implemented our code and we have tested it in some significative problems comparing the solutions obtained with the ones calculate using standard time-dependent simulations in order to show the accuracy of our method. In this way we have demonstrated the correctness of the method showing how the determined solutions tends to be really accurate when a suitable number of harmonics is taken into account for the approximation.

During this thesis some problems have emerged, moreover due to the complexity and vastness of the subject there are several possible future developments of this work.

For example we have limited our method considering only 2D problems and discretizations made by linear finite elements. It could be interesting to extend the analysis considering higher order elements and to investigate the possibility to treat 3D problems which are the real candidates for whom HBFEM becomes really attractive.

Further analysis could include the application of the HB method to coupled nonlinear circuit and field equations and the study of devices with moving parts, for example rotating electrical machines.

Nevertheless, the main development has to be the improvement of the code in order to make it more efficient. Infact at the moment, the code implemented is still not competive with the standard time-dependent simulations. The calculation times can be reduced but there are some serious limitations due to the fact we are dealing with a commercial solver. It should be essential to understand if these problems can be overcome in order to make more attractive the implementation we have proposed.

Appendix A

HB Code for Nonlinear Circuits

In this appendix is presented the MATLAB code which implements the HB analysis applied to the nonlinear circuit represented in Chapter 2. The main script and the principal functions are presented below.

A.1 Main Script

```
%HB analysis half-wave rectifier
clear all
close all
clc

%Read the netlist
fid=fopen('RC1.txt');
F=textscan(fid,'%f %f %s %f %s');
fclose(fid);

%Input
%Number of harmonics
h=16;
%Omega
w_o=20*10^3*2*pi;
%Excitation
Vs=F {4}(1,1);
V_s=sparse(h+1,1);
V_s(2,1)=Vs;
%Diode
Id=100*10^-12;
vt=0.025;
gmin=10^-12;

%Admittance matrices
Ys=buildYs(F,w_o,h,Vs);
Y12=buildY12(F,w_o,h);

%Solve HB
v_t=ones(2*h,1);
v_f=1/h.*fft(v_t);
it_max=1000;
lambda=0.001;
d_lambda=lambda;
i_diodo=zeros(2*h,1);
H=zeros(2*h,2*h);
norma=1;
it=1;
while norma>10^-12 && it<it_max

    %Time domain
    v_f=v_f(1:h+1,1);
    i_diode=Id.*(exp(v_t./vt)-1)+v_t.*gmin;
    di_diode=Id/vt.*(exp(v_t/vt))+gmin;
```

```
    i_t=i_diode;

    %Frequency domain
    i_f=1/h.*fft(i_t);
    i_f=i_f(1:h+1,1);
    f=lambda.*Ys*V_s+Y12*v_f+i_f ;
    norma=norm(f);

    %Jacobian matrix
    H=h.*ifft(1/h.*fft(diag(di_diode)).').';
    H=H(1:h+1,1:h+1);
    J=Ys+H;

    %Update v_f and v_t
    v_f=v_f-(J)\f;
    v_fc=flipud(conj(v_f(2:h,1)));
    v_f=[v_f ;v_fc];
    v_t=h.*ifft(v_f);
    V(:,it)=v_t;

    lambda=lambda+d_lambda;
    it=it+1;
end
```

A.2 Admittance Matrix

```
function [Y12] = buildY12( F,w,h )
%Builds up the admittance matrix between nonlinear ports performing Tableau
%analysis

%Incidence matrix
[x,y]=size(F);
N=[F{1,1},F{1,2}];
[l,p]=size(N);
n=max(max(N));
A_c=zeros(n,l);
cols=[1:l]';
Ap=sparse(N(:,1),cols,ones(l,1),n,l) ;
An=sparse(N(:,2),cols,-ones(l,1),n,l);
A_c=Ap+An;
A=A_c(1:n-1,1:l);
s=zeros(l,1);

%Shortcircuit excitation
Cr1=char(F{y-2});
for i=1:l
    if  Cr1(i,1)=='V'
        F{y-1}(i,1)=0;
        F{y-2}(i,1)={'R'};
    end
end

%Shortcircuit nonlinearities
Cr=char(F{y});
 k=1;
 for i=1:l
    if  Cr(i,1)=='N'
        F{y-1}(i,1)=0;
        F{y-2}(i,1)={'V'};
        num_NL(k,1)=i;
        k=1+k;
        NL=k-1;
    end
 end

%Solve for each harmonic
Cr1=char(F{y-2});
R=sparse(l,l);
```

```
G=sparse(l,l);
C=sparse(l,l);
L=sparse(l,l);

for r=1:NL

    F{y-1}(num_NL(r,1),1)=1;

        for k=0:h;
                    for i=1:l
                        if Cr1(i,1)=='R';
                            R(i,i)=(F{y-1}(i,1));
                            G(i,i)=-1;
                        elseif Cr1(i,1)=='V'
                            G(i,i)=1;
                            s(i,1)=(F{y-1}(i,1));
                        elseif Cr1(i,1)=='L'
                            G(i,i)=1;
                            L(i,i)=-(F{y-1}(i,1));
                        elseif Cr1(i,1)=='C'
                            R(i,i)=1;
                            C(i,i)=-(F{y-1}(i,1));
                        end
                    end

         M1=[A zeros(n-1,l) zeros(n-1,n-1);
             zeros(l,l) -eye(l) A';
             R G zeros(l,n-1)];
         M2=[zeros(n-1,l) zeros(n-1,l)  zeros(n-1,n-1);
             zeros(l,l)  zeros(l,l)  zeros(l,n-1);
             L C zeros(l,n-1)];

          tn=[zeros(n-1,1); zeros(l,1); s];
          M12=(M1+j*k*w*M2);
          x=M12\tn;
          I=x(1:l,1);
          V=x(l+1:l+l,1);

          T((k+1),:)=I(num_NL);

        end

 T1(:,r)=reshape(T,NL*(h+1),1);

 F{y-1}(num_NL(r,1),1)=0;

end

%Build the admittance matrix
 for g=1:NL;
   for m=1:NL
      for n=1:h+1
         Y12(n+(m-1)*(h+1),n+(g-1)*(h+1))=-T1(n+(g-1)*(h+1),m);
      end
   end
 end

end
```

Appendix B

HBFEM Code

We report now the HB code that has been implemented and integrated with COMSOL. The script are presented in the following sections for both eddy current free problems and eddy current problems.

B.1 Eddy Current Free Problem

```
clc
close all
clear all

mu_fp=10;
save('mu_fp','mu_fp')
[model1] = modello1;
model1.sol('sol1').runAll;
[U] = mphgetu(model1);
M1= mphmatrix(model1 ,'sol1','Out', {'K','L'});
[meshstats,meshdata] = mphmeshstats(model1);
info = mphxmeshinfo(model1,'soltag','sol1');
elem_idx = find(meshdata.elementity{1}==1 | meshdata.elementity{1}==2 | ...
    meshdata.elementity{1}==3 | meshdata.elementity{1}==22)';
bound = [];
for i = 1:length(elem_idx)
bound = [bound;info.elements.edg.dofs(:,elem_idx(i))];
end
bound=unique(bound)+1;
M1.K(bound,:)=0;
for rr=1:numel(bound);
     M1.K(bound(rr,1),bound(rr,1))=1;
end
zn_idx=2;
[coord,el]=coord_bar(meshdata,zn_idx);
Nh=5;
Nts=2*Nh+1;
sol_f=zeros(numel(U),Nh+1);
sol_f(:,2)=U/2;
sol_f=[sol_f,fliplr(conj(sol_f(:,1:2*Nh+1)))];
sol_t=Nts*ifft(sol_f,[],2);
[model2]=modello2;
perm_vect=ones(numel(meshdata.elementity{1,2}),1);
save('C:\Program Files\COMSOL\COMSOL50\Multiphysics\yoke\perm_vect','perm_vect');
[model3] = modello3;

it=2;
perm_matrix_old=zeros(numel(el),Nts);
perm_matrix=zeros(numel(el),Nts);
perm_tot=zeros(size(meshdata.elem{1,2},2),Nts);
rhs_t=zeros(numel(U),Nts);

it_max=1000;
t=[0:20e-3/Nts:20e-3-20e-3/Nts];
```

```
while (mean_var>0.001 || max_var>0.01) && (it<it_max)

   %Set solution, extract permeability
     for m=1:Nts
         model2.sol('sol1').setU(sol_t(:,m)) ;
         model2.sol('sol1').createSolution ;
         perm_matrix(:,m)=mphinterp(model2,'(mf.normB)/((mf.normH)*4*pi*10^-7)','coord',coord);
         for n=1:numel(el)
             perm_tot(el(n,1),m)=perm_matrix(n,m);
         end
     end

  %Max and mean percentage variation
   max_var=100*max(max(abs(perm_matrix-perm_matrix_old)))/max(max(perm_matrix_old));
   mean_var=100*((sum(sum(abs(perm_matrix-perm_matrix_old))))/(numel(perm_matrix)))/...
   (((sum(sum(abs(perm_matrix_old))))/(numel(perm_matrix))));

  %Adaptive mu_fp
   perm_matrix_old=perm_matrix;
   perm_min=min(perm_tot,[],2);
           for r=1:size(meshdata.elem{2},2)
              mu_fp(r,1)=perm_min(r,1);
           end
    mu_fp=mu_fp./5;
    save('C:\Program Files\COMSOL\COMSOL50\Multiphysics\yoke\mu_fp','mu_fp');

  %K(mu_fp)
   M1= mphmatrix(model1 ,'sol1','Out', {'K'});
   M1.K(bound,:)=0;
          for r=1:numel(bound);
              M1.K(bound(r,1),bound(r,1))=1;
          end

  %K(mu_fp-mu)
          for m=1:Nts
              perm_vect=perm_tot(:,m);
              save('C:\Program ...
                  Files\COMSOL\COMSOL50\Multiphysics\yoke\perm_vect','perm_vect');
              M3 =mphmatrix(model3 ,'sol1','Out', {'K'});
              matrix=real(M3.K);
              matrix(bound,:)=0;
               for r=1:numel(bound);
                  matrix(bound(r,1),bound(r,1))=1;
               end
              %Time domain rhs
              rhs_t(:,m)=matrix*sol_t(:,m);
          end

  %Frequency domain rhs
   rhs_f=1/Nts*fft(rhs_t,[],2);
   rhs_f(:,2)=(rhs_f(:,2)+M1.L/2);
   rhs_f(:,Nts)=(rhs_f(:,Nts)-M1.L/2);
   rhs_f=2*rhs_f;
   rhs_f=rhs_f(:,1:Nh+1);

  %Frequency domain solution
   sol_f=M1.K\rhs_f;
   sol_f=[sol_f,fliplr(conj(sol_f))];
   sol_f=sol_f(:,1:2*Nh+1);
   sol_f=sol_f/2;

  %Time domain solution
   sol_t=Nts*ifft(sol_f,[],2);

   it=it+1;

end
```

B.2 Eddy Current Problem

The code for eddy current problems is equal to the one presented in the previous section. The only difference consists in the solution of the linear systems, in this case even the imaginary part must be taken into account. The lines that has to be changed are presented below.

```
%Frequency domain solution
sol_f(:,1)=zeros(size(meshdata.vertex,2),1);
sol_f(:,2)=M1.K\rhs_f(:,2);
for r=4:2:(Nh+1);
    sol_f(:,r)=(real(M1.K)+1j*(r-1)*imag(M1.K))\(rhs_f(:,r));
    sol_f(:,r-1)=zeros(size(meshdata.vertex,2),1);
end
sol_f=[sol_f,fliplr(conj(sol_f))];
sol_f=sol_f(:,1:2*Nh+1);
sol_f=sol_f/2;
```

B.3 Set the Elementwise Permeability

In each iteration the permeability of each element must be updated. This is done with the function `set_permeability`.

```
function [permeability] = set_permeability(x,y,meshelement)

load('C:\Program Files\COMSOL\COMSOL50\Multiphysics\eddy current\perm_vect')
load('C:\Program Files\COMSOL\COMSOL50\Multiphysics\eddy current\mu_fp')

mu=(1./mu_fp-1./perm_vect).^-1;
permeability=mu(meshelement);

end
```

Bibliography

[1] L. Cesari, *Functional analysis and periodic solutions of nonlinear differential equations. Contributions to Differential Equations.* Michigan Math. Journ, 1963.

[2] M. Urabe, *Galerkin's Procedure for Nonlinear Periodic Systems.* Mathematics Research Center Madison, Wisconsin, U.S.A., 1965.

[3] S. Yamada and K. Bessho, "Harmonic field calculation by the combination of finite element analysis and harmonic balance method", *IEEE Transaction on Magnetics*, vol. 6, 1988.

[4] J. Gyselinck, P. Dular, C. Geuzaine, and W. Legros, "Harmonic-balance finite-element modeling of electromagnetic devices: a novel approach", *IEEE Transaction on Magnetics*, vol. 38, 2002.

[5] H. D. Gersem, H. V. Sande, and K. Hameyer, "Strong coupled multi-harmonic finite element simulation package", *COMPEL*, 2001.

[6] F. Bachinger, U. Langer, and J. Schoberl, "Numerical analysis of nonlinear multiharmonic eddy current problems", *SFB Numerical and Symbolic Scientific Computing*,

[7] O. Bíró, G. Koczka, and K. Preis, "Finite element solution of nonlinear eddy current problems with periodic excitation and its industrial applications", *ELSEVIER Applied Numerical Mathematics*, vol. 79, 2014.

[8] S. Maas, *Nonlinear Microwave and RF Circuits.* Artech House, 2003.

[9] F. Giannini and G. Leuzzi, *Nonlinear Microwave Circuit Design.* Wiley, 2004.

[10] F. Bachinger, U. Langer, and J. Schoberl, "Effcient solvers for nonlinear time-periodic eddy current problems", *comput. Vis. Sci.,9*, 2006.

[11] H. D. Gersem, S. Vandewalle, and K. Hameyer, "Krylov subspace methods for harmonic balanced finite element methods", 2001.

[12] G. Koczka, S. Außerhofer, O. Bíró, and K. Preis, "Optimal convergence of the fixed-point method for nonlinear eddy current problems", *IEEE Transaction on Magnetics*, vol. 45, 2009.

[13] S. Außerhofer, O. Bíró, and K. Preis, "A strategy to improve the convergence of the fixed-point method for nonlinear eddy current problems", *IEEE Transaction on Magnetics*, vol. 44, 2008.

[14] N. Bianchi, *Calcolo delle Macchine Elettriche col Metodo degli Elementi Finiti.* Cleup, 2001.

[15] S. Yamada, K. Bessho, and P. Biringer, "Calculation of nonlinear eddy-current problems by the harmonic balance finite element method", *IEEE Transaction on Magnetics*, vol. 5, 1991.

[16] A. Gasull, "A theoretical basis for the harmonic balance method", 2001.

[17] Z. K. Peng, Z. Q. Lang, S. Billings, and G. R. Tomlinson, "Comparisons between harmonic balance and nonlinear output frequency response function in nonlinear system analysis", *ELSEVIER Journal of sound and vibration*, 2007.

[18] G. Gambolati and M. Ferronato, *Lezioni di Metodi Numerici per l'Ingegneria.* Progetto, 2014.

[19] M. Kolmbauer, "The multiharmonic finite element and boundary element method for simulation and control of eddy current problems", PhD thesis, Linz, 2012.

[20] D. Copeland and U. Langer, "Domain decomposition solvers for nonlinear multiharmonic finite element equations", Tech. Rep., 2009.

[21] D. Masotti, "La tecnica del bilanciamento armonico come strumento per la simulazione e l'analisi di stabilità e di rumore di sistemi non lineari: dal circuito di potenza con induttori isteretici all'oscillatore a radiofrequenza", PhD thesis, Bologna, 1995.