

# A Comprehensive Approach to Convolutional Neural Networks Based Condition Monitoring of Permanent Magnet Synchronous Motor Drives

Dario Pasqualotto<sup>1\*</sup>, Mauro Zigliotto<sup>1</sup>

<sup>1</sup> Department of Management and Engineering, University of Padova, Stradella S. Nicola 3, Vicenza, Italy

\* E-mail: dario.pasqualotto@phd.unipd.it

## Abstract:

The increasing complexity of modern industrial systems calls for automatic and innovative predictive maintenance techniques. As suggested by the Industry 4.0 process, this demand translates in the need of *more-intelligent* drives. The present work is devoted to the use of a special kind of neural networks to interpret the data from motor currents for diagnostic purposes. The early detection of possible faults in the electrical motor allows programmed maintenance and reduces the risk of unplanned shutdowns. The innovation is in the overall approach to the neural network training, which does not call anymore for a large set of faulty motors. In this work we propose to generate a large training dataset by using a combination of tuned motor models and some data augmentation techniques. The result is a comprehensive and effective motor condition monitoring algorithm, whose hearth is a convolutionary neural network trained by a safe and cheap simulation-based dataset. The details of the design are fully reported in the paper. The method has been implemented in the laboratory and fully tested on both healthy and faulty permanent magnet synchronous motors. The generality of the proposed method also paves the way for the detection of other failures and to the application to different electrical motors.

## 1 Introduction

The present paradigm of on-site ac drives commissioning done by humans is more and more demanding and time-consuming. It also represents an impediment to the most advanced control strategies, that imply the availability of control experts for the tuning and maintenance. Beside that, the interconnection and cost of the installations leaves less room for malfunctions and performance degradation. Actually, there is a growing need to detect and identify any incoming failure as soon as possible [1] in order to improve the drive reliability. The research is increasingly focusing on techniques and methodologies able to give an augmented autonomy to modern industrial drives.

In the Mechatronics field, permanent magnet synchronous motors (PMSMs) are widely used in different applications, such as electrical vehicles, wind energy, home, and industrial appliances. Among the reasons, there are the simple and compact structure, easy manufacturing, and high power density, more precise control compared with other electrical motors, and high power factor over constant torque region [2]. In this context, the aim of this paper is to explore a new alternative for condition monitoring of PMSMs. The technique belongs to the knowledge-based systems class and makes use of convolutional neural networks (CNNs), a cutting-edge tool in artificial intelligence [3].

Several technical papers have dealt with the fault detection issue in electric motors and they have been collected and summarised in different reviews on condition monitoring for induction motors [4–6] or for PMSMs [7]. According to most of them, failure conditions on pmsm can be classified into three categories which are electrical, mechanical or magnetic faults. The electrical faults are related to the health conditions of the stator winding and they can mainly involve interturn shorts, open circuit faults or grounding [7–9]. Instead, mechanical faults are principally related to the rotor. They can be roughly classified in eccentricities (static, dynamic, mixed) and ball-bearing defects (to the inner race, outer race or to the

rotating balls) [10, 11]. Finally, magnetic faults concern the demagnetization of permanent magnets, which is the partial loss of the residual flux density [12]. As reported in [8], the causes of one or more of the above-mentioned failures can be classified in thermal, electrical, mechanical or environmental pollution.

Rather intuitively, each fault can be assessed more simply if a characteristic belonging to its own class is observed. For example, a ball bearing (mechanical) defect can be early detected by an accelerometer that senses the mechanical vibrations induced by the defect itself [13]. Nevertheless, this way leads to the use of many additional sensors that in turn pose again a problem of reliability.

Faiz et. al. in [2] proposed a revision and comparison of several current-based fault detection indexes for the demagnetization fault with some extensions also to eccentricities. The motor inductance monitoring may represent a valid alternative because it enables the separation between demagnetization and eccentricity faults. Nevertheless, it requires a current injection which interferes with the normal functioning in industrial applications.

A more effective approach derives by considering that all failures somehow affect the airgap flow distribution and consequently the magnetic path of the motor. In principle, any abnormal operating condition can be detected through the analysis of motor currents. The choice of the motor current signature analysis (MCSA) has great potentiality since it does not require any additional hardware with respect to a standard industrial drive, maintaining cost-effectiveness and reliability. The nowadays great question is to develop a procedure for the automatic recognition of the failure.

The solution of the condition monitoring problem is composed by three steps, namely the fault detection, isolation and identification. In other words, understand that a fault has occurred, detecting where and finally comprehend how much severe is the fault degree [1]. The complexity of the problem and the many contributing factors make the solution not trivial.

Given the recent growing trends of Internet of Things (IoT) and Industry 4.0 and essentially thanks to the modern availability and possibility to manage a huge amount of data and information the

knowledge-based or data-driven techniques are acquiring an increasing importance [3]. The basic idea is to develop an expert classifier, based on MCSA theory and trained on a big volume of historical data, that automatically detects an incoming failure. Artificial neural networks (ANNs) are suitable tools for this kind of problem as Nandi et al. report in [4]. They are gaining popularity because of their physical model-free solutions.

Among the first applications is the ANN technique applied by Chow et al. in [14] to detect incoming failures on ball bearings of an electric motor. In that study, the vibrations, measured through specific accelerometers, were used as index to understand the bearing health level and the classification was performed by an artificial intelligence-based (AI) algorithm. The study was soon followed by many others, always using the ANN as enabling technology for the condition monitoring problem [15].

Nevertheless, a recent research [16] has shown how a CNN performs better than the traditional multi-perceptron ANN, thanks to some inherent features that will be highlighted later on.

Promising results were reported in [17] and [18], in which a CNN was applied on induction motors in order to identify different types of fault. The main focus was on the transformation of a current sequence into a 2D image, in order to use a traditional 2D CNN as it is usually applied to image recognition. The former illustrates the preliminary findings using a specially designed CNN and a time/frequency-domain bearing vibration analysis, in order to detect ball bearing, rotor bar and winding insulation faults. The latter used an already LeNet-5 trained network, presenting a new signal-to-image conversion method aimed at eliminating the experts' experiences as much as possible.

One step beyond was performed by Ince in [19], demonstrating that 2D transformation is not necessary and a simple 1D convolution can be applied reaching a good level of accuracy in the dichotomic choice between healthy and faulty induction motors.

Finally, an application of a 1D CNN on PMSM monitoring was considered in [20], where all the main faults which can occur on a PMSM were successfully recognized.

The problem remains the CNN training, which requires a large batch of training data from the field. Without them, any technique is destined to remain almost a pure academic exercise.

This paper aims at proposing a solution to the generation of the training dataset which is effective and efficient from both economical and industrial points of view. At least two main points are addressed. The first is that in all the aforementioned cases the dataset generation was realized damaging a real motor for each fault and fault level. The set was then realized acquiring current records at different levels of speed and current. This methodology is very expensive, unpractical and scarcely affordable from an industrial point of view. A second aspect that requires attention is that so far a totally black box approach has been applied. Useful information on motor behaviour, features of the phase current patterns and network characteristics are often neglected. The lack is compensated by the CNN capabilities, at the cost of increasing enormously and uselessly the complexity of the network and the related training problems. In the electric drives world, this may be not a successful direction.

As said, the aim of the paper is to give an effective contribute to cover these shortcomings. In particular, thanks to modern software, it is now possible to realize a simulation-based generation of the training dataset which is here taken into consideration. This makes possible to create a huge amount of data in a very affordable way and to earn samples easily with different types and level of failures.

In this context, the paper also takes advantages of some improvements typical of the so-called data augmentation techniques, as detailed in Section 5.2. Actually, in CNN-based image processing, data augmentation is a mean to increase the available training dataset [21] or to improve the accuracy and robustness of the classifier [22]. Data augmentation is an hot and open issue in the industrial predictive maintenance field as the recent literature reports [23–25]. Its use is commented in Section 5.2 in this paper.

As another distinctive feature, the present paper includes the condition monitoring of fractional PMSM motors, that is, motors with non-integer number of slots per pole per phase. This is uncommon in the scientific literature, opposite to the use of fractional-slot motors

which is an increasing trend in the industrial world due to their specific advantages in both torque ripple minimisation and coils length [26]. Of course, the proposed approach is valid also for integer slot per pole per phase motors.

The paper also develops a useful "frequency normalization" technique in order to make current patterns less variant with the motor speed. Thanks to this, a very simple network structure is already capable to correctly classify the conditions of a PMSM into the following three categories: healthy, demagnetization fault and interturn fault. Detected faults have a broad degree of seriousness, yielding enhanced prediction capability.

The paper is organized as follows. Section 2 presents the models used for the generation of the training dataset. Each model is explained with particular care on the mathematical formulation and founding hypotheses. The expected current pattern during each fault condition is analysed at steady state, that is the condition in which the proposed condition monitoring operates. Section 3 reports the main properties and features of the CNN tool used for the condition monitoring process. The design steps of the network are also reported, for the sake of algorithm repeatability. Section 4 illustrates the design of experiments, along with the features of the selected CNN and the highlights on the training dataset. Experimental results are finally shown and commented in section 5, that is split in two parts.

The first one reports the validation of the PMSM drive simulation models, by comparing the output with the experimental ones. The models are then used to generate the phase current data patterns for the CNN training. The second part of the experiments shows the classification ability of the trained CNN, on real experimental patterns obtained from different PMSM motors, either healthy or with some degree of failure. Conclusive remarks and implementation hints are finally given.

## 2 Permanent Magnet Synchronous Motor models

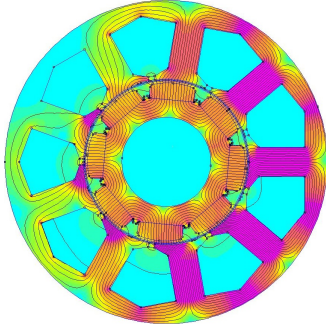
Each motor condition is characterized by a certain current signature. An accurate predictive algorithm is able to early detect a particular fault by recognising its relative signature. This concept may be easily extended to any motor fault. In particular, mechanical defects cause characteristic oscillations in the phase current [4]. By increasing the network complexity and setting new motor models for the generation of the training dataset, the signatures relative to any mechanical fault can be included.

The present work has limited the investigation to some of the possible faults, less prone to be detected by the commonly available techniques (i.e. accelerometers).

In order to generate the dataset of examples for the convolutional neural network training a proper model of each fault is necessary. Since not only the fundamental harmonic of current is of interest but also spurious ones, the model of the healthy motor should include proper finite element simulations [27]. The same concept is valid for demagnetized motor, while for interturn fault the model proposed in [28] is here adjusted to fit for the fractional-slot PMSM under test.

The choice of the reference frame for the motor models is worth a little deepening. The alternative is between a stationary frame, fixed to the stator, and a synchronous one, fixed to the rotor flux. As regards the healthy and demagnetised motors, the two frames are equivalent, since multiple harmonics of the fundamental have to be considered, so that both models become position-dependent. In case of different motor types, the choice of a synchronous reference frame may help in including magnetic nonlinearity. As regards the interturn fault, the choice of a stationary reference frame yields a constant inductance matrix with respect to the rotor position, opposite to the synchronous frame case. The analysis of different fault severity or locations also comes easily. Anyway, in case of different motors that suffer of saturation, a shift to a synchronous frame would be advantageous for a better matching with FEA methods.

The models selected for the present work are illustrated in detail below.



**Fig. 1:** Finite element analysis of the healthy PMSM of Table 1, when  $\theta_m = 0$  and  $i_q = 0$ .

### 2.1 Healthy motor

The PMSM electrical dynamic is described by the voltage balance equations in  $dq$  reference frame, synchronous to the rotor:

$$\mathbf{u}_{dq} = \mathbf{R}\mathbf{i}_{dq} + \frac{d\boldsymbol{\lambda}_{dq}}{dt} + \mathbf{J}\omega_{me}\boldsymbol{\lambda}_{dq} \quad \text{with} \quad \mathbf{J} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (1)$$

where  $\mathbf{u}_{dq} = [u_d, u_q]^T$ ,  $\mathbf{i}_{dq} = [i_d, i_q]^T$  and  $\boldsymbol{\lambda}_{dq} = [\lambda_d, \lambda_q]^T$  are the stator voltage, current and flux linkage vectors, respectively.  $\mathbf{R} = \text{diag}\{R, R\}$ , where  $R$  represents the phase resistance, while  $\omega_{me} = p\omega_m$  is the electromechanical speed, that is the pole pairs  $p$  times the mechanical speed  $\omega_m$ .

Due to the surface mounted permanent magnet structure, the effect of saturation in motor fluxes is almost negligible. Therefore a linear behaviour of the magnetic paths is assumed and it is described through the (constant) synchronous inductance matrix  $\mathbf{L} = \text{diag}\{L, L\}$ . Conversely, permanent magnet (PM) flux linkage  $\boldsymbol{\lambda}_{dq}^{mg}$  can be affected by local and partial demagnetisation, which creates flux spatial harmonics that finally modify the current harmonic content. A suitable model for the flux linkages spatial distribution can be the following:

$$\boldsymbol{\lambda}_{dq}(\theta_m) = \mathbf{L}\mathbf{i}_{dq} + \boldsymbol{\lambda}_{dq}^{mg}(\theta_m) \quad (2)$$

where  $\theta_m$  is the mechanical rotor position. It is worth noting that the space flux harmonics different from the fundamental one affect both axes.

The position-dependent distribution of (2) has been derived through a proper finite element analysis (FEA), at null stator currents. An example of the flux density across the PMSM under test (Table 1) at  $\theta_m = 0$  is reported in Fig 1.

Since PMs have the same permeability as the air, the FEA model was also used to derive the current-dependent flux linkage, and thus the inductance  $L$ , after removing the magnets from the drawing. Several simulations at different current levels have confirmed the linearity hypothesis with respect to the phase current, as stated above.

The PMSM model can be completed by considering the electromechanical torque:

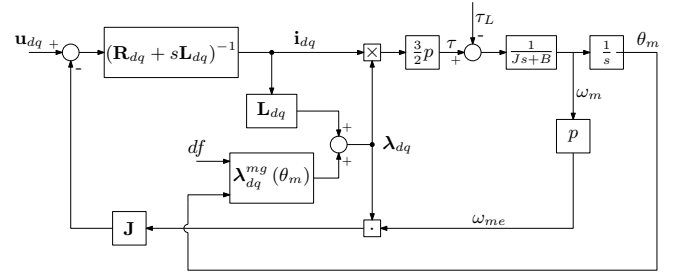
$$\tau = \frac{3}{2}p(\lambda_d i_q - \lambda_q i_d) \quad (3)$$

and by describing the mechanical dynamics by a first-order system

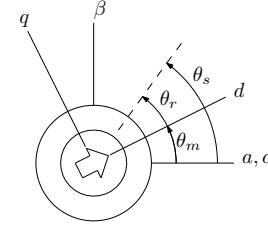
$$\tau = B\omega_m + J\frac{d\omega_m}{dt} + \tau_L \quad (4)$$

with  $J$  and  $B$  as rotor inertia and viscous friction, respectively, and  $\tau_L$  as load torque.

The complete model, obtained from equations (1)-(2) is shown in Fig 2. In the figure, the symbol  $\cdot$  represents the scalar product, while  $\times$  recalls the difference of the cross-products in (3). The PM flux



**Fig. 2:** Motor model for both healthy and demagnetised cases.



**Fig. 3:** Reference frames and angular displacements definition.

linkage model obtained by FEA incorporates the dependence on the rotor geometry and it is embedded in the look-up table (LUT) block  $\boldsymbol{\lambda}_{dq}^{mg}(\theta_m)$ . The same model will also be used for the motor under demagnetisation fault, as explained in the next section. Therefore, inside the block  $\boldsymbol{\lambda}_{dq}^{mg}(\theta_m)$  there are various LUTs, which differ in the demagnetisation factor  $df$ .

In the healthy case, the main component in both the flux linkage and the stator current is clearly the fundamental one. But in a real motor, also the 5-th and 7-th flux linkage harmonics have not a negligible effect on the currents [29]. Hence the need to consider those contributions in the current patterns used for the training.

The simulation model of Fig 2 was used to run the ac drive in several different working points. For each of them, the  $q$  current was stored for future use in the CNN training.

### 2.2 Motor with demagnetisation fault

Overtemperature and exposition to intense external fields can induce either a partial or global irreversible decrease of magnetisation in the PMSM rotor magnets [30]. The macroscopic effect is different in the two cases.

In case of homogeneous demagnetisation, all rotor poles are equally affected, so that the airgap flux density distribution maintains a very similar profile as in the healthy motor. In particular, the electrical behaviour of the motor is still represented by a three-phase balanced system and there is not the evidence of new harmonics eligible as indexes for the fault recognition. In that case the torque per ampere ratio decreases, and a comparison with the brand-new product may help in the fault detection.

More challenging is the case of partial demagnetisation, in which the flux density distribution is no more periodic in the electromechanical angle, that is, it depends on the considered pole pair. The distribution along the airgap is a function of the angular displacement, which can be expressed with respect to either the stationary ( $\theta_s$ ) or the synchronous ( $\theta_r$ ) reference frame, as depicted with the dashed line in Fig 3.

The Fourier series expansion of the flux density along the airgap  $B_g$  when the motor is partially demagnetized is the following:

$$B_g(\theta_r) = \sum_{n=1}^{\infty} \sum_{k=0}^{\infty} B_{g_{n,k}} \cos[(np \pm k)\theta_r] \quad (5)$$

where  $p$  is the number of polar pairs and  $n$  represents the index of harmonics characterizing the healthy behaviour, while the index  $k$  indicates the demagnetisation-related harmonics.

Actually, the flux density is not measured in a standard drive, but its influence on the stator currents through the back electromotive forces (b-EMF) is the key for the early detection of the fault.

Still in a general approach, the flux linkage of phase  $a$  is obtained by integrating the flux density (5) with respect to  $\theta_s$  obtaining (the details are in appendix 8):

$$\lambda_a^{mg}(\theta_s) = \sum_{n=1}^{\infty} \sum_{k=0}^{\infty} \lambda_{n,k}^{dm} \cos[(np \pm k)\theta_m] \quad (6)$$

Nevertheless, during the integration process that from the flux density takes to the flux linkages several harmonics may disappear. In principle, and in particular considering the case of fractional motors, it may frequently happen that the number of motor pole pairs  $p$  is different from the number of windings pole pairs  $p_w$ . The choices about these two values lead to different flux linkage distributions. Following the passages to get a general expression of the PMSM flux linkage, reported in appendix 8, it can be inferred that for a specific motor design not all the harmonics  $(n, k)$  are visible.

The present paper is relative to the fractional-slot PMSM whose data are reported in Table 1.

An accurate FEA and the mathematical analysis of appendix 8, applied to the considered motor, allows to identify the harmonics present in the stator flux linkage. Among them, the main ones are the fundamental derived from (6) by posing  $n = 1, k = 0$  and the fault-related harmonic, given by  $n = 1, k = 1$ . The amplitude of the main fault-related harmonic, which from now on is indicated by  $\lambda_a^{dm}$ , is eligible as significant index of the PM health. Due to the winding distribution of the motor with winding pole pairs  $p_w = 1$ ,  $\lambda_b^{dm}$  and  $\lambda_c^{dm}$  have the same distribution but shifted respect to  $\theta_m$  by  $2\pi/3$  and  $4\pi/3$  respectively.

$$\lambda_a^{dm}(\theta_m) = \lambda_{1,1}^{dm} \cos[(p+1)\theta_m] \quad (7)$$

Remembering that  $p = 4$ , one can easily argue that the three-phase system represented by the previous equation leads to a reverse phasor sequence rotating at  $-5\theta_m$ . With respect to a  $dq$  synchronous reference frame, the sequence rotates at  $-5\theta_m - \theta_{me} = -9\theta_m$ . As a result, under demagnetisation fault and at steady-state conditions a  $i_q$  current harmonic at  $9\omega_m$  is expected.

$$i_q^{dm}(t) = i_{1,1}^{dm} \cos(9\omega_m t) \quad (8)$$

In case of demagnetisation fault, the model is still described by the equation (1) and Fig 2. Time by time, the model was updated for different partial demagnetisation levels, by changing the  $\lambda_{dq}^{mg}(\theta_m)$  LUTs through customised FEA sessions. The modelling of the mechanical part remains unchanged. As in the healthy case, the generation of the CNN training datasets was obtained by changing the motor working points, storing at the same time the  $i_q$  current patterns.

### 2.3 Motor with interturn fault

Due to its peculiar mathematical description, the most effective dynamic state-space model of the motor with interturn fault is obtained in the  $\alpha\beta$  stationary reference frame. In principle, the present work adopts the modelling technique described by Vasoghi *et al.* in [28]. That method was suited for permanent magnet motors with an integer number of slots per pole and per phase. Here, the method is re-elaborated and extended also to fractional-slot PMSM motors.

The interturn fault is highlighted in Fig 4. Each phase is composed by  $N$  coils. For example here it is considered a loss of insulation in the first coil of phase  $b$ . The severity of the damage can be graduated by changing the value of the resistance  $r_f$ . The fault divides the coil into two parts  $1_f$  and  $1_h$ , characterised by the ratio between the faulty turns and total coil turns  $\mu = N_f/N_s$ .

The damage causes the phase current to split into two paths. The new path ( $i_f$ , Fig 4) drains part of the flux-producing current, with a consequent flux linkage reduction.

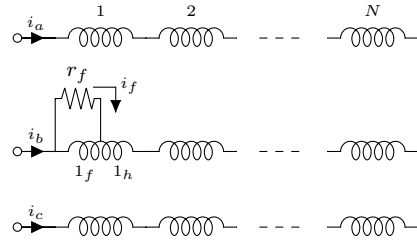


Fig. 4: Model of N-coil windings with an interturn fault on phase  $b$ .

Let  $R_{coil}$  be the total coil resistance and  $R_f = \mu R_{coil}$  and  $R_h = (1 - \mu) R_{coil}$  the resistances of the two portions  $1_f$  and  $1_h$ , with currents  $i_b - i_f$  and  $i_b$  flowing through them, respectively.

Treating the current in each coil (including the two sub-coils generated by the fault) as an independent state variable it is possible to obtain a system of  $(3 \cdot N) + 1$  equations that describes the electrical behaviour of the circuit. Then, substituting each current with the correspondent  $i_a, i_b$ , or  $i_c$  current, adding the voltage Kirchoff law on the closed loop created by  $r_f$  and reshuffling the equations, it is possible to obtain a 4-dimensional voltage balance system:

$$\begin{bmatrix} u_a \\ u_b \\ u_c \\ 0 \end{bmatrix} = \mathbf{R}_{abcf} \begin{bmatrix} i_a \\ i_b \\ i_c \\ i_f \end{bmatrix} + \mathbf{L}_{abcf} \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \\ i_f \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \\ -e_f \end{bmatrix} \quad (9)$$

where

$$\mathbf{R}_{abcf} = \begin{bmatrix} R & 0 & 0 & 0 \\ 0 & R & 0 & -R_f \\ 0 & 0 & R & 0 \\ 0 & -R_f & 0 & R_f + r_f \end{bmatrix} \quad (10)$$

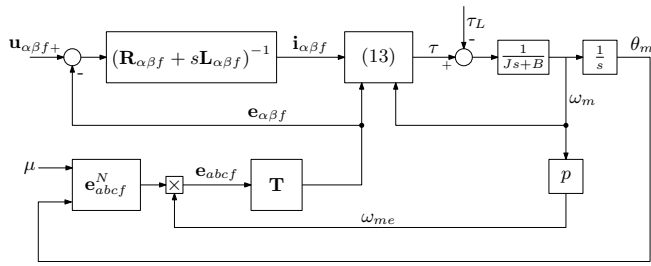
$$\mathbf{L}_{abcf} = \begin{bmatrix} L & M & M & M_{af} \\ M & L & M & M_{bf} \\ M & M & L & M_{cf} \\ M_{af} & M_{bf} & M_{cf} & L_f \end{bmatrix}$$

Symbols  $u_a, u_b, u_c$  represent the phase voltages while  $e_a, e_b, e_c$  are the b-EMFs. It is worth to note that only a portion  $e_f = \mu e_{coil}$  of the total b-EMF  $e_{coil}$  is induced in the faulty part  $1_f$  of the coil itself.

The knowledge of the inductance matrix  $\mathbf{L}_{abcf}$  comes from the availability of an accurate FEA analysis, which is an essential requirement of the proposed technique. Inductances  $L$  and  $M$  are the usual phase stator mutual and self-inductances. The fourth row (or column) contains the mutual and self-inductance between phases and the coil  $1_f$ .

While  $L$  and  $M$  are easily obtainable, the self- and mutual inductances related to the  $1_f$  coil need to include effect of  $\mu$ . In details,  $M_{af}$  is the mutual inductance between coil  $1_f$  and the entire winding  $a$ . Through a FEA it is possible to easily estimate the value of the mutual inductance  $M_{1a}$  between coil 1 of the faulty phase ( $b$  in this case) and phase  $a$ . This is performed imposing a constant current into the winding  $a$  and evaluating the flux linkage on the coil 1 of phase  $b$ . Then, knowing that the mutual inductance is proportional with the number of coil turns,  $M_{af}$  can be obtained scaling  $M_{1a}$  by the factor  $\mu$ . With the same approach the self-inductance  $L_f$  is equal to the self-inductance of coil 1 of phase  $b$  weighted by  $\mu^2$ .

Once the model in the natural  $abcf$  coordinates is achieved, the augmented space-vector model in the stationary reference frame  $\alpha\beta 0f$  is obtained by an extended version of Park transformation that



**Fig. 5:** Motor model for the interturn fault case.

makes use of the following transformation matrix:

$$\mathbf{T} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} & 0 \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & \sqrt{\frac{3}{2}} \end{bmatrix} \quad (11)$$

where the last element of the matrix is chosen in order to have:  $p_{abc f} = 3/2 \cdot p_{\alpha\beta 0 f}$ . That is the electrical power in  $abc f$  reference frame is  $3/2$  times the same power in  $\alpha\beta 0 f$ .

The homopolar component of the current  $i_0$  can be neglected, as it does not contribute to the torque production. The resulting three-dimensional model for the interturn fault is:

$$\mathbf{u}_{\alpha\beta f} = \mathbf{R}_{\alpha\beta f} \mathbf{i}_{\alpha\beta f} + \mathbf{L}_{\alpha\beta f} \frac{d\mathbf{i}_{\alpha\beta f}}{dt} + \mathbf{e}_{\alpha\beta f} \quad (12)$$

and it is shown in Fig 5. The effect of spatial harmonics is included in the model thanks to a LUT block ( $\mathbf{e}_{abc f}^N$ ) obtained experimentally. In particular, the three phase induced voltages are measured as function of the rotor position  $\theta_m$  and normalized with respect to  $\omega_m e$ . The fourth dimension of vector  $\mathbf{e}_{abc f}^N$  is estimated as a portion ( $\mu$ ) of the voltage induced in the faulty phase  $e_b$ .

The electromagnetic torque is estimated by an energy balance of the augmented motor model (12):

$$\tau = \frac{3 e_{\alpha} i_{\alpha} + e_{\beta} i_{\beta} - e_f i_f}{2 \omega_m} \quad (13)$$

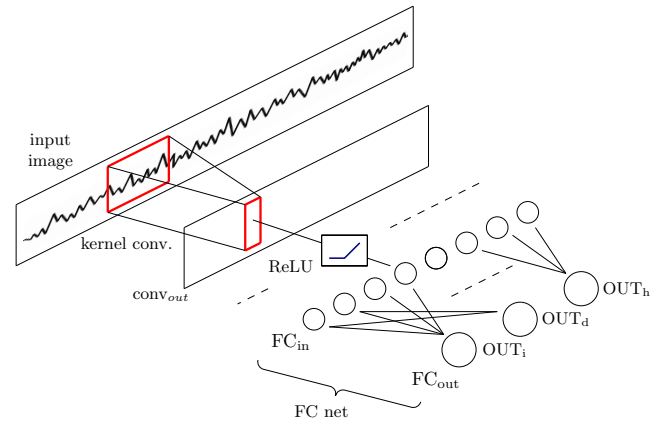
Finally, it is appropriate a consideration about the expected current patterns in case of interturn fault. In normal conditions the spectrum of the current vector is formed by the fundamental signal rotating at  $p\omega_m$  and its multiple harmonics. As usual, the phase of each harmonic is neglected since it has no significance in the proposed method.

$$i_a(p\omega_m) = \sum_{n=1}^{\infty} i_n \cos(np\omega_m t) \quad (14)$$

Multiples of the third harmonic are null due to the hypothesis of a three-phase balanced system. Clearly, when an interturn fault occurs the system balance is lost and the third harmonic and its multiples may appear in the current signal. This was also verified by simulation, from which it has been deduced that the major fault-related contribution is actually due to the third harmonic. It generates a positive phasor sequence in the three-phase system, which in turn generates an  $i_q$  current oscillation at  $2p\omega_m$  that can be profitably used as healthy index:

$$i_q^{it}(t) = i_1^{it} \cos(2p\omega_m t) \quad (15)$$

The  $i_q$  current was obtained under different interturn fault hypotheses (i.e. different values of  $\mu$ ) from the  $i_{\alpha\beta}$  current of the model represented in Fig 5, through the inverse of transformation (11). Then, the current patterns were stored for next use in the CNN training procedure.



**Fig. 6:** Example of a 1D convolutional neural network.

### 3 Convolutional Neural Networks

This section reports the fundamentals of convolutionary neural networks, whose properties and design issues will facilitate the understanding of the following paragraphs.

CNNs are a type of AI-based algorithm that are inspired by the behaviour of mammalian visual cortex [31, 32]. In particular, their principal similarity is in the procedure called feature extraction, which is the recognition of more or less complex characteristics that are present inside the image. This operation is performed by applying several filters, called kernels, through different layers in cascade. Such structure allows the recognition of characteristics that are as complex as the number of layers increases. This set of layers forms the convolutional stage of the CNN.

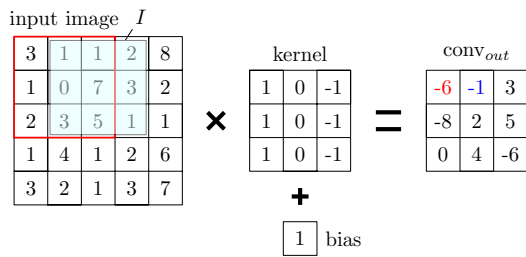
The next layer after the convolution is nonlinearity, that can be used to adjust or cut-off the generated output. For many years, sigmoid and tanh were the most popular non-linearity. More recently, the Rectified Linear Unit (ReLU) has been used more often, for its simpler definitions in both function and gradient [33]. The nonlinear layer helps in training and fitting nonlinear behaviours. Finally, a classical fully-connected (FC net) layer is used to classify the image among the possible output classes (Fig 6).

Generally, in image recognition a 2D discrete convolution is used. It is applied between the input image  $I$  and a kernel  $K$  characterized by a certain receptive field (i.e. height  $m$  and width  $n$ ) which is typically smaller than the dimensions of the image. The output is often called *feature map*. The operational definition of the convolution may vary, depending on whether or not it is of interest to maintain the commutative property [3]. In this work, the convolution  $S = K * I$  has been implemented as follows:

$$S(i, j) = \sum_m \sum_n I(i-1+m, j-1+n) K(m, n) + b \quad (16)$$

where it is intended that the matrix indices start from 1. The algorithm may also be defined as a *cross-correlation*, which is identical to a standard convolution, without flipping the kernel [3]. An example, reported in Fig 7, will help in understanding the procedure.

Firstly, an element-by-element (Hadamard) product between a sub-matrix of the input image and the kernel is performed. The portion of the input matrix that must be considered is identified by the indexes  $(i, j)$ . In Fig 7, the computation of  $S(1, 1)$  is obtained by the Hadamard product of a  $3 \times 3$  kernel matrix and sub-matrix of the same size located in the upper-left corner of  $I$ . The products (i.e. the elements of the Hadamard matrix) are summed together with a bias  $b$ , generating a value for  $S(1, 1)$  element. To compute  $S(1, 2)$ , one has to refer to the sub-matrix  $I$  highlighted in blue in Fig 7, and so forth.



**Fig. 7:** A convolution example. In case of 1D CNN, heights of input image, kernel and convolution output (conv<sub>out</sub>) are unitary.

### 3.1 CNN properties

Due to convolution, three are the main properties of a CNN that a normal fully connected neural network does not have: sparse interactions, parameter sharing and equivariance to translations. Sparse interactions and parameter sharing are properties which are reciprocally highly connected. Since the kernel dimension is generally smaller than the input image, the same weights (i.e. the elements of the kernel) are applied on different portion of the image. Therefore, the same parameters are shared among the neurons of the each layer, justifying the *sparse interaction* quality.

Opposite to traditional neural networks, that are fully connected, in CNNs the reduced kernel dimensions enable both a lower storage capability and a lighter computational load.

Finally, *equivariance* of a function means that if the input changes, the output changes too, accordingly. The property of equivalence to the translation of a CNN network means that if the input image is shifted in time, so will the output. Having in mind the pattern recognition problem, the equivariance is definitely useful, since the position of the fault within the input image is unknown.

### 3.2 CNN parameters

The design of a CNN is a very complex task, with a huge number of possibilities and parameters. This is the main concern when approaching the use of this powerful, but rather challenging analysis tool. In the following paragraphs some hints will be given in order to guide the designer in the choice, with specific reference to the condition monitoring problem.

The definition of a CNN requires many different parameters, which can be hyperparameters and learnable parameters.

The formers define the architecture of the network. Some examples are the number or type of layers that compose the CNN, the number of filters in each convolutional layer or the number of neurons within the filters.

The learnable parameters are the weights and biases of each neuron of the CNN. They are chosen after that the CNN structure is designed through the definition of all the hyperparameters.

The design of the CNN does not necessarily follow that order. In the present paper it has been found that a procedure that goes back and forth between the hyperparameters and the learnable ones may be the most suited way to come to satisfactory results. In the following, some details are given.

**3.2.1 Learnable parameters:** The training procedure defines the weights and biases inside the network. This problem is faced within a very powerful class of methods that are known as *supervised learning techniques*. A labelled training dataset is necessary, that is, a set of images to feed the network and that it tries to classify.

If the network performs its task properly, i.e. the CNN classifies the images as their are labelled, the weights are correctly tuned. If not, a method to update them is necessary.

A typical choice falls in the back-propagation error-based techniques and in particular in the simpler stochastic gradient descent (SGD) algorithm. The target of the SGD is the minimisation of a given *loss function*, which is evaluated at each iteration of the training dataset (*epoch*) by comparing the known (labelled) fault conditions with the ones predicted by the CNN.

After the loss is calculated, weights are updated by a small step in the direction of negative gradient of the loss itself. The dimension of the step is imposed by the *learning factor* coefficient, which governs the convergence speed. An upper limit to the leaning factor is given by the possible triggering of oscillations of the loss function around its minimum, without reaching it. A deeper analysis of SGD is done in [34] or still in [3].

It is of paramount importance that the training dataset covers the whole space of images that the network is asked to classify. When the CNN is trained on a very small subset, it will perform well only locally, since it has not a good extrapolation ability.

**3.2.2 Hyperparameters:** The focus moves now on the design of the CNN structure, i.e. the definition of the hyperparameters. A typical AI-related problem is the *overfitting* issue.

Overfitting is a situation such that the neural network is so closely fitted to the training set that it finds difficult to generalize the problem and make correct predictions for new data. This is directly related to the neural network model capacity, which defines the complexity of the tasks that a neural network is able to solve. The more well-structured the CNN, the larger its model capacity. On the other hand, if the CNN has a too big model capacity, it may tends to overfit training data and will be unable to elaborate new data.

The opposite condition, the *underfitting*, is also a problem. In that case, the model capacity is insufficient and the CNN does not track the real trend of training data. It tends to interpolate the experimental data by a too simple model and some crucial information are lost.

In the present work, it has been found that an iterative procedure can help in defining all the hyperparameters of the CNN. The training procedure is started with a CNN of minimum model capacity. The training set obtained by the drive models is splitted into two datasets (*cross-validation* method). The first is used for the training by the SGD algorithm. After each training epoch the SDG evaluates the loss function to decide whether to reiterate the training by starting a new epoch or to stop it. The end of the training can happen because the loss function has either reached a predetermined minimum value, or it has not improved during the last epochs.

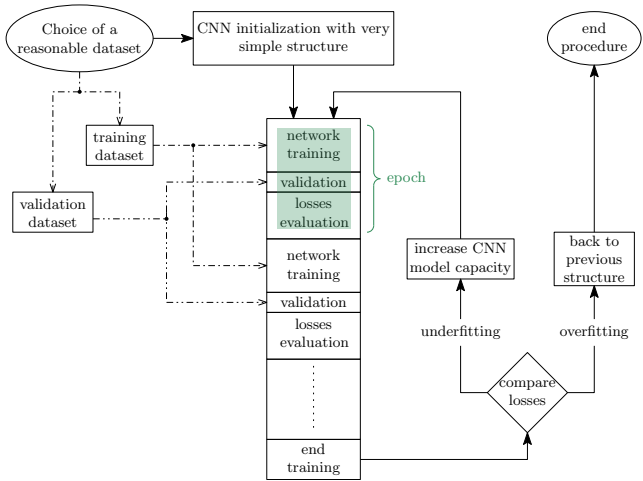
The second dataset, called *validation set* is used to check the CNN during the training outlined above, at regular intervals. The loss function is evaluated but (opposite to the training procedure) the check is not intended to modify the weights of the CNN. It just interrupts the training flow, verifying the ability of the network to recognise a dataset (i.e. a motor fault) that it has never seen before. At the end, the time evolution of the loss function of both the training and validation during the SGD training yields three possible situations:

- *overfitting*: the training accuracy is satisfactory, but the validation test on new images still returns a poor result. The CNN has to be simplified, for example reducing the number of layers.
- *underfitting*: both the training and validation still exhibit low accuracy. The CNN model capacity is too small and it shall be increased by either adding layers, increasing kernel sizes or making other decisions that lead to a more complex structure. Iteratively, the architecture of the CNN will become more and more structured till it edges an overfitting flaw.
- *well-tuned*: the predetermined overall accuracy (for example, 97% of positive matches) is reached and the validation test on new images returns a good result as well. The CNN is sized appropriately and the weights are well tuned. The training procedure has been successful.

The procedure outlined above was used to design the CNN that underlies the present condition monitoring technique. Even not revolutionary, still it represents an operative and effective way for the implementation. The various steps are summarised in Fig 8, while the CNN specific features are reported in section 4.2.

## 4 Design of Experiment

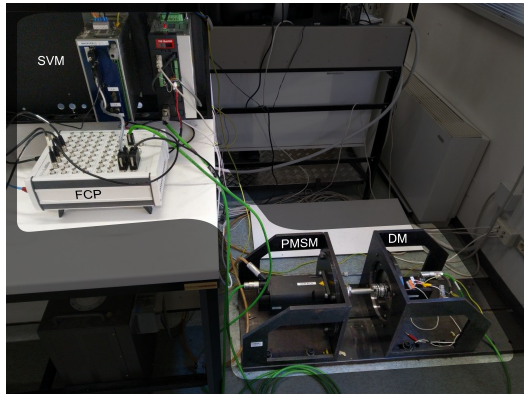
This section analyses the design and execution of experiments for the proposed condition monitoring of a fractional-slot permanent magnet synchronous motor. It is subdivided into a part related to the



**Fig. 8:** Proposed iterative design of CNN for condition monitoring.

**Table 1** Healthy PMSM nameplate data

Stator pole pairs ( $p_w$ )	1
Rotor pole pairs ( $p$ )	4
Nominal current ( $I_N$ )	3.8 A
Nominal speed ( $\Omega_{m,N}$ )	576 rad/s (5500 rpm)
Nominal torque ( $T_N$ )	2.6 N m
Stator resistance ( $R$ )	1.6 $\Omega$
Phase inductance ( $L$ )	9.15 mH
DC link voltage	540 V
PWM switching frequency	10 kHz



**Fig. 9:** Experimental laboratory setup.

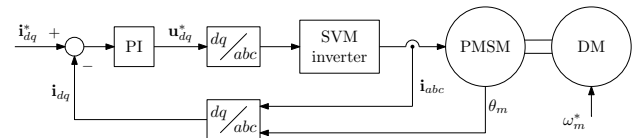
validation of the models necessary for the generation of the artificial training dataset and a part relative to the CNN design.

#### 4.1 PMSM drive models validation

One of the main innovation proposed in this paper is the combined use of motor models and CNN to get a comprehensive approach to condition monitoring. The motor models need to be validated before being used as generators of artificial dataset for the CNN training. The availability of a virtual unlimited dataset greatly enhance the industrial feasibility of the method, with respect to those based on the collection a great number of real faulty motors.

The validation of the models presented in the previous sections was achieved by comparing their behaviours with a real PMSM. It was a 1.5 kW PMSM whose main features are reported in Table 1. In order to validate the models under different loading conditions, the PMSM was stiffly coupled and dragged at constant speed  $\omega_m^*$  by a dragging motor (DM). The whole test bench is shown in Fig 9.

Actually, two prototypes of the same motor were manufactured with customised features. The first one was a healthy motor with



**Fig. 10:** Experimental current-controlled PMSM drive.

a modified winding. According to the modelling of section 2.3, a contact has been extracted from an intermediate point of the phase  $b$ . In this way, it was possible to insert an external variable resistance to simulate interturn faults of different severity. The absence of the resistance was the healthy motor, of course.

The second motor was used for the validation of the condition monitoring in case of demagnetisation. The rotor was modified removing the permanent magnets of a pole pair and replacing them with demagnetised elements, namely with the 20% of nominal remanence.

The experimental setup was completed by two space voltage modulated (SVM) inverters connected to the motors and controlled by a fast control prototype system (FCP, Fig 9), implementing the conventional field oriented control Fig 10.

The phase motor currents are of fundamental importance for the condition monitoring proposed in this paper. They were directly measured through the MicroLabBox ADCs, with a resolution of 16 bit for a full scale current of  $\pm 14$ A. The inverter switching frequency and the ADC sampling frequency were both of 10 kHz. The drives were operated at different speed and load conditions and the real currents were compared with those obtained by the Simulink models, for validation purposes.

The validation has involved both healthy and faulty motors and the related models. The first point to resolve was whether a frequency or time comparison was more appropriate. The former is a well-defined method, even if it involves the burden of a transformation in the frequency domain. It can be the best candidate in case you want to automate the validation procedure. The second enables a qualitative analysis of the trend of the signals over time. It is convenient because it provides at a glance information on the quality of the model. Although the validation was performed on different working conditions, only one per type of fault is here reported, for illustrative purposes.

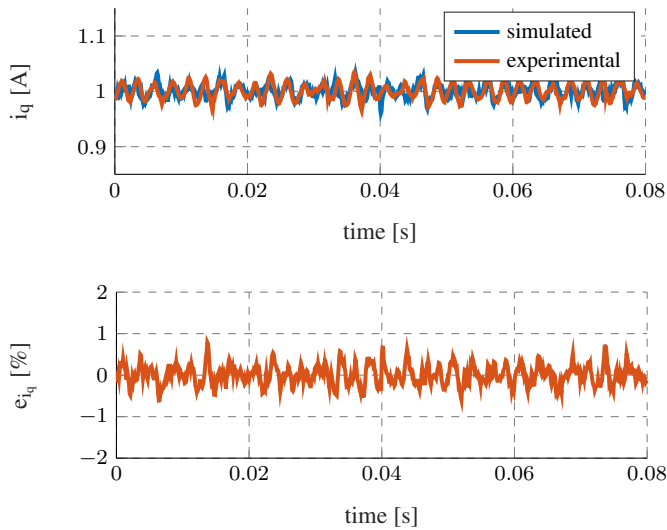
As described in section 2, an incoming fault is characterised by the emergence of new (fault) harmonics in the current spectrum. For what concerns the qualitative validation in the time domain, Fig 11 shows the  $i_q$  current in healthy conditions, at a speed of 104.7 rad/s and a load torque of about 0.5 Nm, corresponding to a current  $i_q = 1$  A.

The same working point was selected for the demagnetised PMSM and the interturn fault. The results of the comparisons are analysed in Fig 12 and Fig 13, respectively. It is worth to note how the demagnetisation of a single magnetic pole pair (out of 4) causes a marked oscillation at the frequency  $9\omega_m/2\pi = 150$  Hz, as required by (8).

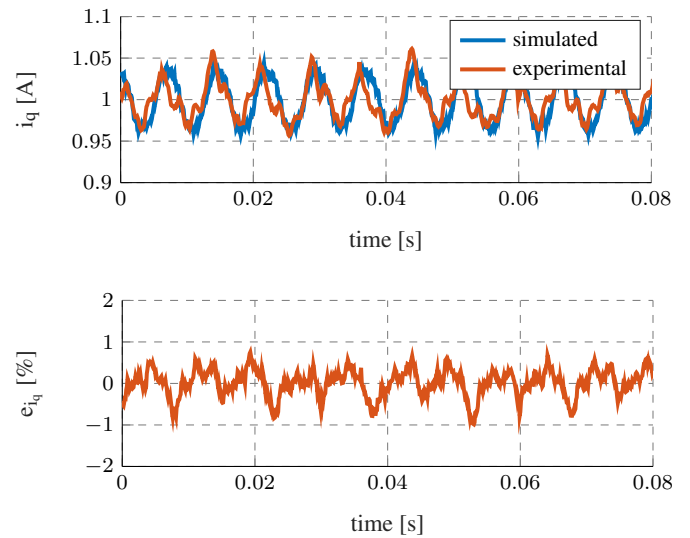
It is also evident that the interturn fault gives rise to an oscillation, due to the phase unbalance. Its period is of about 7.5 ms, as expected from (15) that indicates a frequency of  $2p\omega_m/2\pi = 133.3$  Hz.

All figures also track the percentage error in the  $i_q$  current between model and experiments, referred to the nominal motor current ( $3.8 A_{eff}$ , Table 1). In all cases the errors lie within the satisfactory range of  $\pm 1\%$ .

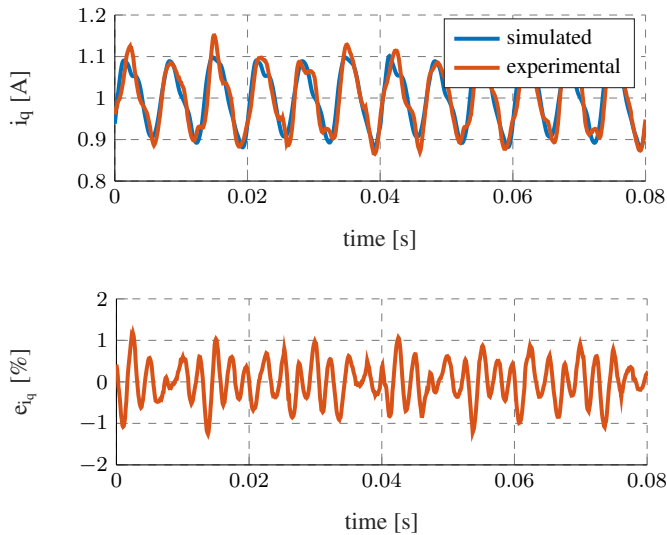
For what concerns the validation in frequency domain, Table 2 shows the comparison between the amplitudes of both the fundamental current components and fault harmonics, in case of simulation models (sim) and of experimental prototypes (exp). The percentage errors (err) are referred to the PMSM nominal current (Table 1). All data refer to the same working conditions described in Fig 11-Fig 13. It is easy to note the impressive similarity of the results, which confirms the accurate FEA of the motor. The strong correspondence between models and real prototypes will yield a trustworthy generation of artificial patterns for the CNN training, which is one of the key-features of the present work.



**Fig. 11:** Model and experimental  $i_q$  current time patterns, healthy PMSM. The percentage error is referred to the nominal motor current.



**Fig. 13:** Model and experimental  $i_q$  current time patterns, interturn fault. The percentage error is referred to the nominal motor current.



**Fig. 12:** Model and experimental  $i_q$  current time patterns, partially demagnetized PMSM. The percentage error is referred to the nominal motor current.

**Table 2** Model validation through FFT analysis of the  $i_q$  current

		sim [A]	exp [A]	err [%]
healthy	fundamental	1	0.999	0.018
	fault harmonic (absent)	-	-	-
demagnetized	fundamental	0.9997	1	-0.0055
	fault harmonic (@ 150 Hz)	0.108	0.102	0.111
interturn	fundamental	0.964	0.972	-0.141
	fault harmonic (@ 33.3 Hz)	0.039	0.038	0.025

#### 4.2 CNN design hints

The CNN was designed by adopting the approach explained in section 3. To make it easier to reproduce the conditioning monitoring system, some design hints are reported below.

Since the system is operating on current patterns, a 1D neural network was realized. The only one dimension is the  $q$  current record length that will be indicated with letter  $M$ .

At steady state, a healthy PMSM should exhibit a constant  $q$  current. As highlighted during the model validation stage, either a partial demagnetisation or an interturn fault cause oscillations to arise, because of harmonics at frequency different from the fundamental one.

It is expected that more severe is the fault level, the greater the amplitude of oscillations, while its angular frequency should identify the type of fault. The angular frequency is strictly dependent on the motor speed and this eases the evaluation of the right fault-related harmonics. In principle, a proper threshold on the oscillation amplitude could then implement a good condition monitoring. This apparently simple method has some implementation flaws, as the need of establishing the threshold and facing possible slight frequency shifts of the selected harmonic.

The use of a well-tuned convolutional neural network is highly preferable, as it takes the index in the recognition process not only a specific harmonic but the current time behaviour as a whole. The classification of the motor conditions is more robust and successful.

**4.2.1 Choice of kernel and current record sizes:** The condition monitoring is based on the analysis of  $i_q$  current time strings, each of them composed by  $M$  points, usually sampled at the beginning of every inverter switching period  $T_c = 100 \mu\text{s}$ . The mean value associated to the operating point of the motor is out of interest, and it is removed. After data post-processing, a healthy motor should return current patterns just noisy around zero. A first important choice is about the kernel size (or length) (Fig 7), that is the time period by which the  $i_q$  current record is scanned by convolution. The following considerations provide some hints in the trade-off.

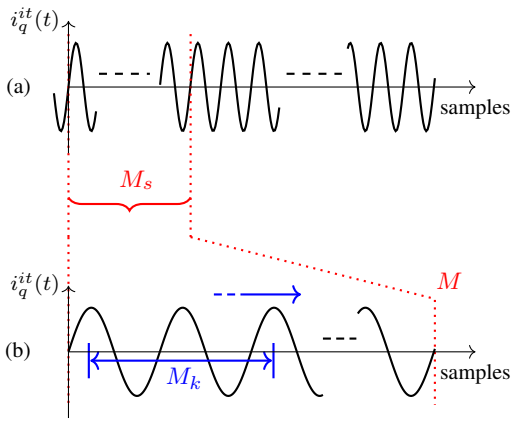
The oscillations due to the demagnetisation and interturn faults have a time period established by equations (8) and (15) and it is a function of the motor speed.

The kernel elements are the CNN weights that after the training will approximatively have a sinusoidal shape, at the fault-related harmonic frequency. Therefore, the minimum kernel length  $M_{min}$  that may guarantee the detection needs to include at least one period of oscillation in the worst conditions, i.e. at the minimum motor speed  $\omega_{min}$  and in presence of an interturn fault:

$$M_{min} \geq \frac{\pi}{p\omega_{min}T_c} \quad (17)$$

A longer kernel would contain more than one period of oscillation, with a detection capability preserved and even improved, at the cost of a longer processing time. A rough, but effective, rule of thumb is to set  $M_k \approx 2M_{min}$ , while the length of each  $i_q$  current record was set to  $M \approx 25M_{min}$ .





**Fig. 14:** Frequency normalization for kernel matching. The graphs are two  $i_q$  current records, (a) @  $\omega_m > \omega_{min}$ , (b) @  $\omega_{min}$ . After normalisation,  $M_s$  samples contain the same number of oscillations of  $M$  samples @  $\omega_{min}$ .

The technique is based on the effects of the faults into the air-gap flux density, which in turn affects the bemb and finally the current patterns. The effect is less evident at low speed and therefore below a certain minimum speed the algorithm cannot work. It has been found that a proper choice for the PMSM prototypes used during the experiments is to trigger the operation of the condition monitoring algorithm above the minimum operating speed  $\omega_{min} = 42 \text{ rad/s} = 400 \text{ rpm}$ , less than  $0.1 \cdot \Omega_{m,N}$ . In turn, that choice leads to a minimum kernel length  $M_{min} = 187$ ,  $M_k = 300$  and  $M = 5000$ , according to (17) and the related discussion.

**4.2.2 Frequency normalization:** For the sake of an easier implementation in the real drive, the  $i_q$  current time records always contain the same number  $M$  of samples. As the speed increases, more and more oscillation periods are present, as depicted in Fig 14, upper track. This is a big issue, since the kernel size and weights cannot be changed in real time.

When the PMSM runs at the minimum speed  $\omega_{min}$ , by construction  $M_{min}$  samples correspond to a whole period of oscillation and each  $i_q$  current record, composed by  $M$  elements contains  $M/M_{min} = 25$  whole oscillation periods. This is a design feature that must be preserved in any working condition. As the motor accelerate to a speed  $\omega_m$ , the  $i_q$  record more than 25 periods, so it must be scaled down by taking only the first  $M_s$  samples, calculated proportionally as

$$M_s = M \frac{\omega_{min}}{\omega_m} \quad (18)$$

so that the number of oscillations is still 25 at any speed and the effectiveness of the convolution is maintained.

## 5 Experimental Results

As it is evident from the previous sections, the road to a complete algorithm is a complex interweaving of interdisciplinary skills, ranging from the physics of electric motors to the design of appropriate neural networks.

Each phase involved theoretical studies, simulations and experimental sessions. Therefore, this section is entitled Experimental Results in the sense that it will show CNN's actual classification capabilities on PMSM prototypes that in no way took part to its training. First, some further implementation details are given below.

### 5.1 CNN implementation details

The convolutional neural network was implemented in Matlab through the Neural Network Toolbox. The training requires the design of an artificial dataset, composed by  $i_q$  current records

(images) obtained from the models validated above. Training samples among the three classes (healthy, demagnetized and motor with interturn fault) have to span the whole motor operating region in terms of both current and speed. Records are randomly mixed to compose epochs, as described in section 3.2.1. Once the kernel length  $M_k$  is fixed, it can be applied along each current pattern, for all the patterns of the epoch.

The convolutional network structure depicted in Fig 6 ends with a classification among healthy and two faulty conditions. In details, the CNN returns a real number for each neuron of the FC output layer. Namely,  $OUT_h$  for the healthy motor class,  $OUT_d$  and  $OUT_i$  for the demagnetised and interturn fault classes, respectively. Them alone are insufficient for a proper final decision on the PMSM state.

Another layer, called *softmax*, translates the neurons output in a percentage. Then, the label  $C$  of the class which has the higher probability to include the given  $i_q$  time record is computed by the simple expression:

$$C = \max_x \left[ \frac{OUT_x}{\sum_x OUT_x} \right]; \quad x = \{h, d, i\}. \quad (19)$$

By comparing  $C$  returned by (19) with the known state of the PMSM that has produced the current record, the training algorithm iteratively tunes the elements of the kernel as well as the weights of the FC net, as already described in Fig 8.

In the training dataset generation process, speed was imposed at different levels starting from  $\omega_{min} = 42 \text{ rad/s}$  at steps of  $20 \text{ rad/s}$  up to  $\Omega_{m,N} = 576 \text{ rad/s}$ , while the  $i_q$  current was controlled by steps of  $0.5 \text{ A}$  up to  $I_N = 3.8 \text{ A}_{eff}$ .

Several demagnetization levels were included in the training set, generating samples with PMs at 20%, 40% and 60% of nominal remanence. Interturn fault samples were achieved at the following different levels of the fault resistance  $r_f$ :  $0.01 \Omega$ ,  $0.1 \Omega$ ,  $1 \Omega$ .

The artificial training dataset (14700 current vectors, of  $M$  samples each) was divided randomly into two subsets: the first, composed by the 70% of the original dataset, was used for the training, while the remaining samples were used for the validation of the neural network (Fig 8).

### 5.2 Data augmentation techniques

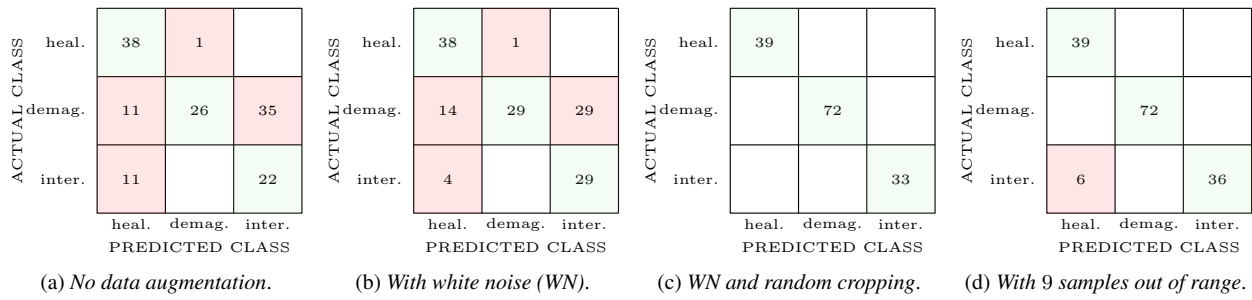
A recent review [35] has collected the main time series data augmentation methods, framing them in an accurate taxonomy. All methods have the goal of increasing the number of training samples.

The most challenging ones are the decomposition, the learning methods and finally the model-based methods. The formers decompose the initial training dataset in a series of statistical informations as trend, seasonality and residuals. Then they apply some transformations to these variables creating the augmented training dataset. While interesting, they are probably too complex for using in the present work. Learning-based methods assume that simple transformations applied to encoded inputs rather than to the raw inputs would produce effective data augmentation [35]. They represent a viable improvement and are worth a future deepening.

Last, model-based time series augmentation approaches typically involve modelling the dynamics of the time series with statistical models. The present implementation borrows some data augmentation techniques (white noise addition and random cropping), always with an eye to maintain the necessary overall simplicity. They have been applied to both simulation and experimental current patterns, as described below.

Firstly, a white noise was added to the  $i_q$  patterns obtained by the models, obtaining new patterns and therefore enlarging the artificial dataset.

Secondly, special attention was paid to the phase of the artificial faulty patterns. Since they are generated by a model, it is easy to make the mistake of polarising them with unrealistic conditions. This is particularly true for the initial conditions of the simulation that, if unchanged, may lead to faulty current patterns with always the same initial phase of the harmonics. It has been found that a good solution is to generate quite long current patterns and then obtain several artificial sequences by randomly cropping  $M$ -samples portions.



**Fig. 15:** Experimental confusion matrices. Classification results are listed by gradually including different data augmentation techniques to the training dataset. The rightmost matrix includes 9 experimental patterns out of the CNN training range.

The same cropping technique was used to triple the experimental dataset used to validate the classification capabilities of the CNN in Section 5.3.

### 5.3 CNN classification capabilities

The classification capabilities of the condition monitoring system were finally validated through an experimental set of samples collected on the custom motors available in the laboratory. None of the sets was used in the training of the CNN.

A set of 144 current vectors was obtained in different working conditions. In details, the experiment grid was obtained by regularly spacing the speed in the interval  $[1/10 \dots 1 \Omega_{m,N}]$  and the  $i_q$  current within  $[0.2 \dots 1 I_N]$ . As regards the interturn fault, fault resistances were selected in the range  $[1 \dots 15\Omega]$ . They were finally presented to the proposed condition monitoring system for classification in order to test its performances.

To evaluate the impact of the data augmentation techniques to the network classification accuracy, they were applied one after the other for the generation of the training dataset. After each improvement, the CNN was asked to classify the same experimental dataset.

As common practice in supervised learning algorithms, the results are reported in a so-called *confusion matrix*, a specific table layout (Fig 15) that allows the evaluation of performances at a glance. Each column of the matrix represents the instances in a predicted class while each row represents the instances in an actual class.

In the first case (Fig 15 (a)) neither noise addition (except quantization noise) nor random cropping were applied for the generation of each sample of the training dataset. Each sample pattern started from the same initial condition. In this case the classification results are very poor and the network is able to classify correctly only the 59.7% of the experimental motor conditions.

Some improvements were achieved by adding a white noise (WN) to the current patterns used to train the CNN (Fig 15 (b)), increasing the net accuracy up to 66.67%. The performance is still non-optimal because in this way the same initial phase is always imposed on the harmonics (Section 5.2), a condition that is obviously not respected by the experimental data.

The classifier accuracy was further increased by rand M-samples cropping within each simulation sample. This improvement in the training dataset produced the classification reported in Fig 15. In this case the recognition is perfect and all cases are correctly classified (Fig 15 (c)), confirming the validity of both the method and the proposed data augmentation hints.

Last, the experimental batch was enlarged by adding 9 further measurements at very low speed (26.2 rad/s), below the speed of 42 rad/s fixed as lower limit during the system design (section 4.2.1). Such new measurements were accomplished in the interturn fault case with  $r_f$  varying between  $[0.025 \Omega \dots 0.1 \Omega]$  with  $i_q = 1$  A.

This new test was intended to validate the recognition problems at low speeds of the motor. Even increasing the severity of the fault by strongly reducing the fault resistance  $r_f$ , the reduced rotor speed leads to some false positive predictions. In particular, the CNN correctly classifies only 3 current patterns over 9 (Fig 15 (d)).

## 6 Conclusive Remarks

An effective method for condition monitoring in the electric drives field has been presented in this paper. It falls into the category of knowledge-based systems, matching the *more-intelligent drives* paradigm.

The proposed condition monitoring technique is composed of two parts. The first one, performed offline, consists in the model setup and tuning, the training dataset generation and the CNN training. The second one, performed online, is relative to the experimental current pattern acquisition and their processing by the trained CNN, for a continuous monitoring of the health conditions of the motor. Therefore, the algorithm well fits for the Industry 4.0 paradigm and can be easily adapted to an IoT environment. The integration of the PMSM drive in an articulate and complex system eases the exchange of data between the components of the system itself. With such a structure it is possible to designate each task to the appropriate element. So the operations which need determinism and real-time behaviour, as for the motor control, are implemented in the slave inverter, while the condition monitoring of the motor can be performed by a master server which can be devoted also to other, more complex, but not deterministic tasks. The interconnection guarantees an online master-slave communication. The slave can acquire in real-time the motor current patterns sending them to the master, that performs the condition monitoring and takes the proper actions if it detects a potentially dangerous situation.

The use of 1-D convolutional neural networks fits for the simplicity that is required by the industrial field, while still providing the benefits of versatility and robustness typical of artificial intelligence.

Since the beginning, the approach was intended to be comprehensive, in the sense of addressing every theoretical and implementation aspect. An objective impediment to the straightforward transposition of the techniques used for image recognition to the field of electric motors is the lack of an adequate number of test cases..

To tackle the problem and as a distinctive feature of this work, we proposed the generation of artificial training patterns, obtained on motor models validated on custom prototypes. The effort is certainly less than either finding a vast archive of faulty current signatures, or reproducing the faults by damaging several motors.

The models were merged in the complete ac drive simulation, to get realistic current patterns. Actually, the second important feature of this work is that it uses only existing current sensors, present in any ac motor drive.

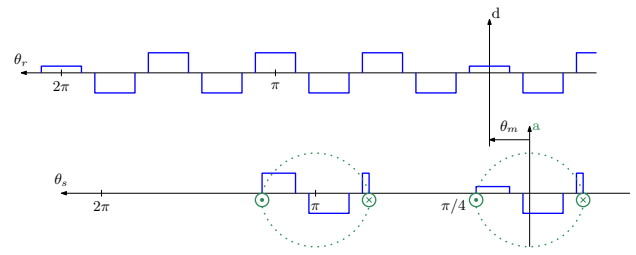
The datasets for both training and validation were enlarged by using simple data augmentation techniques. Some more advanced and challenging methods are worth deepening in a next research steps.

The paper includes all the design hints for reproducing the condition monitoring system. Experimental results relative to each stage have been included, and final verification of the classification ability, carried out on real prototypes, has confirmed the effectiveness and the practical feasibility.

The procedure is general and relevant for all the types of fault that have a measurable effect on phase currents. The approach is then easily extendible to other faults, as for example the ball bearing-related ones and to other types of motor as well.

## 7 References

- [1] Gao, Z., Cecati, C., Ding, S.X.: 'A survey of fault diagnosis and fault-tolerant techniques—part i: Fault diagnosis with model-based and signal-based approaches', *IEEE Transactions on Industrial Electronics*, 2015, **62**, (6), pp. 3757–3767
- [2] Faiz, J., Nejadi-Koti, H.: 'Demagnetization fault indexes in permanent magnet synchronous motors—an overview', *IEEE Transactions on Magnetics*, 2016, **52**, (4), pp. 1–11
- [3] Goodfellow, I., Bengio, Y., Courville, A.: 'Deep Learning'. (The MIT Press, 2016)
- [4] Nandi, S., Toliyat, H.A., Li, X.: 'Condition monitoring and fault diagnosis of electrical motors—a review', *IEEE Transactions on Energy Conversion*, 2005, **20**, (4), pp. 719–729
- [5] Zhang, P., Du, Y., Habetler, T.G., Lu, B.: 'A survey of condition monitoring and protection methods for medium-voltage induction motors', *IEEE Transactions on Industry Applications*, 2011, **47**, (1), pp. 34–46
- [6] Kumar, S., Mukherjee, D., Guchhait, P.K., Banerjee, R., Srivastava, A.K., Vishwakarma, D.N., et. al.: 'A comprehensive review of condition based prognostic maintenance (cbpm) for induction motor', *IEEE Access*, 2019, **7**, pp. 90690–90704
- [7] Wang, Z., Yang, J., Ye, H., et. al.: 'A review of permanent magnet synchronous motor fault diagnosis'. In: 2014 IEEE Conference and Expo Transportation Electrification Asia-Pacific (ITEC Asia-Pacific), (IEEE, 2014), pp. 1–5
- [8] Siddique, A., Yadava, G.S., Singh, B.: 'A review of stator fault monitoring techniques of induction motors', *IEEE Transactions on Energy Conversion*, 2005, **20**, (1), pp. 106–114
- [9] Bellini, A., Filippetti, F., Tassoni, C., Capolino, G.: 'Advances in diagnostic techniques for induction machines', *IEEE Transactions on Industrial Electronics*, 2008, **55**, (12), pp. 4109–4126
- [10] Rosero, J.A., Cusido, J., Garcia, A., Ortega, J.A., Romeral, L.: 'Broken bearings and eccentricity fault detection for a permanent magnet synchronous motor'. In: IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics, (IEEE, 2006), pp. 964–969
- [11] Jigyasu, R., Sharma, A., Mathew, L., Chatterji, S.: 'A review of condition monitoring and fault diagnosis methods for induction motor'. In: 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), (IEEE, 2018), pp. 1713–1721
- [12] Faiz, J., Mazaheri-Tehrani, E.: 'Demagnetization modeling and fault diagnosing techniques in permanent magnet machines under stationary and nonstationary conditions: An overview', *IEEE Transactions on Industry Applications*, 2017, **53**, (3), pp. 2772–2785
- [13] Patel, R.K., Agrawal, S., Joshi, N.C.: 'Induction motor bearing fault identification using vibration measurement'. In: 2012 Students Conference on Engineering and Systems, (IEEE, 2012), pp. 1–5
- [14] Li, B., Chow, M., Tipsuwan, Y., et. al.: 'Neural-network-based motor rolling bearing fault diagnosis', *IEEE Transactions on Industrial Electronics*, 2000, **47**, (5), pp. 1060–1069
- [15] Chow, M.Y., Sharpe, R.N., Hung, J.C.: 'On the application and design of artificial neural networks for motor fault detection - part i', *IEEE Transactions on Industrial Electronics*, 1993, **40**, (2), pp. 181–188
- [16] Gogul, I., Kumar, V.S.: 'Flower species recognition system using convolution neural networks and transfer learning'. In: 2017 4th International Conference on Signal Processing, Communications and Networking (ICSCN-2017), (IEEE, 2017), pp. 1–6
- [17] Mukhopadhyay, R., Panigrahy, P.S., Misra, G., et. al.: 'Quasi 1d cnn-based fault diagnosis of induction motor drives'. In: 2018 5th International Conference on Electric Power and Energy Conversion Systems (EPECS), (IEEE, 2018), pp. 1–5
- [18] Wen, L., Li, X., Gao, L., et. al.: 'A new convolutional neural network-based data-driven fault diagnosis method', *IEEE Transactions on Industrial Electronics*, 2018, **65**, (7), pp. 5990–5998
- [19] Ince, T., Kiranyaz, S., Eren, L., et. al.: 'Real-time motor fault detection by 1-d convolutional neural networks', *IEEE Transactions on Industrial Electronics*, 2016, **63**, (11), pp. 7067–7075
- [20] Kao, I., Wang, W., Lai, Y., et. al.: 'Analysis of permanent magnet synchronous motor fault diagnosis based on learning', *IEEE Transactions on Instrumentation and Measurement*, 2019, **68**, (2), pp. 310–324
- [21] Mikołajczyk, A., Grochowski, M.: 'Data augmentation for improving deep learning in image classification problem'. In: 2018 International Interdisciplinary PhD Workshop (IIPhDW), (IEEE, 2018), pp. 117–122
- [22] Fawzi, A., Samulowitz, H., Turaga, D., Frossard, P.: 'Adaptive data augmentation for image classification'. In: 2016 IEEE International Conference on Image Processing (ICIP), (IEEE, 2016), pp. 3688–3692
- [23] Meng, Z., Guo, X., Pan, Z., Sun, D., Liu, S.: 'Data segmentation and augmentation methods based on raw data using deep neural networks approach for rotating machinery fault diagnosis', *IEEE Access*, 2019, **7**, pp. 79510–79522
- [24] Jiang, X., Ge, Z.: 'Data augmentation classifier for imbalanced fault classification', *IEEE Transactions on Automation Science and Engineering*, 2020, pp. 1–12
- [25] Hu, T., Tang, T., Chen, M.: 'Data simulation by resampling—a practical data augmentation algorithm for periodical signal analysis-based fault diagnosis', *IEEE Access*, 2019, **7**, pp. 125133–125145
- [26] Carraro, E., Bianchi, N., Zhang, S., et. al.: 'Design and performance comparison of fractional slot concentrated winding spoke type synchronous motors with different slot-pole combinations', *IEEE Transactions on Industry Applications*, 2018, **54**, (3), pp. 2276–2284
- [27] Pando-Acedo, J., Rasselökin, A., Lehikoinen, A., et. al.: 'Hybrid fea-simulink modelling of permanent magnet assisted synchronous reluctance motor with unbalanced magnet flux'. In: 2019 IEEE 12th International Symposium on



**Fig. 16:** An example of the airgap flux density distribution in a PMSM.

- Diagnosics for Electrical Machines, Power Electronics and Drives (SDEMPED), (IEEE, 2019), pp. 174–180
- [28] Vaseghi, B., Nahid-Mobarakeh, B., Takorabet, N., et. al.: 'Modeling of non-salient pm synchronous machines under stator winding inter-turn fault condition: Dynamic model - fem model'. In: 2007 IEEE Vehicle Power and Propulsion Conference, (IEEE, 2007), pp. 635–640
- [29] Simon-Sempere, V., Burgos-Payan, M., Cerquides-Bueno, J.: 'Spatial filtering: A tool for selective harmonics elimination in the design of permanent-magnet synchronous motors', *IEEE Transactions on Magnetics*, 2012, **48**, (6), pp. 2056–2067
- [30] Ki-Chan, K., Seung-Bin, L., Dae-Hyun, K., et. al.: 'The shape design of permanent magnet for permanent magnet synchronous motor considering partial demagnetization', *IEEE Transactions on Magnetics*, 2006, **42**, (10), pp. 3485–3487
- [31] Hubel, D.H., Wiesel, T.N.: 'Receptive fields and functional architecture of monkey striate cortex', *The Journal of Physiology*, 1968, **195**, (1), pp. 215–243
- [32] Yamins, D.L.K., DiCarlo, J.J.: 'Using goal-driven deep learning models to understand sensory cortex', *Nature Neuroscience*, 2016, **19**, (3), pp. 356–365
- [33] Albawi, S., Mohammed, T.A., Al-Zawi, S.: 'Understanding of a convolutional neural network'. In: 2017 International Conference on Engineering and Technology (ICET), (IEEE, 2017), pp. 1–6
- [34] Thompson, C.M., Shure, L.: 'Image Processing Toolbox: For Use with MATLAB; [user's Guide]'. (MathWorks, 1995)
- [35] Wen, Q., Sun, L., Song, X., Gao, J., Wang, X., Xu, H.: 'Time series data augmentation for deep learning: A survey', *ArXiv*, 2020, **abs/2002.12478**

## 8 Appendix - Demagnetised Flux Linkage

In all generality, the number  $p$  of pole pairs of the permanent magnets in the rotor can be different from the number  $p_w$  of pole pairs in the windings.

An example with  $p = 4$  and  $p_w = 2$  is represented in Fig 16. The upper graph reports the airgap flux density distribution due to PMs referred to  $\theta_r$  (Fig 3). This reference frame with the whole waveform is shifted of  $\theta_m$  with respect to the stationary reference frame fixed to the axis of the winding  $a$ .

The lower graph shows the flux linkage of winding  $a$  due to the PM. In place of the distributed winding, its equivalent concentrated one is displayed. According to the electric machines theory, the magnetic behaviour is preserved by changing the number of turns according to a winding distribution factor  $k_w$  that depends on the constructive features of the winding.

The aim of the next passages is to qualitatively obtain the flux linkage harmonics of the phase  $a$ , in the specific case of partial demagnetisation of the rotor.

In a healthy motor, the airgap flux density is space-periodical with period  $2\pi$  in the angular coordinate  $p\theta_m$ . By using the Fourier series expansion, it can be expressed as follows:

$$B_g(\theta_r) = \sum_{n=1}^{\infty} \hat{B}_{g_n} \cos(np\theta_r) \quad (20)$$

Since only the frequency distribution of the harmonic content is of interest, the phase lag of each component is assumed to be zero, for simplicity. When a partial demagnetization occurs, the distribution (20) is modulated by a signal which is space-periodical in the

coordinate  $\theta_r$ , becoming:

$$B_g(\theta_r) = \left[ \sum_{n=1}^{\infty} \hat{B}_{g_n} \cos(np\theta_r) \right] \left[ \sum_{k=0}^{\infty} \alpha_k \cos(k\theta_r) \right] \\ = \sum_{n=1}^{\infty} \sum_{k=0}^{\infty} \hat{B}_{g_n} \alpha_k \frac{1}{2} \left[ \cos((np-k)\theta_r) - \cos((np+k)\theta_r) \right] \quad (21)$$

where  $\alpha_k$  is the coefficient that, for each harmonic  $k$ , represents the impact of demagnetization on the healthy waveform. The total flux linked by winding  $a$  can be expressed as:

$$\lambda_a^{mg} = k_w \sum_{i=0}^{p_w-1} \int_{-\frac{\pi}{2p_w} + \frac{2\pi i}{p_w}}^{\frac{\pi}{2p_w} + \frac{2\pi i}{p_w}} B_g(\theta_s) \frac{DL}{2} d\theta_s \quad (22)$$

with  $D$  and  $L$  as diameter at the airgap and stack length of the motor, respectively. It is worth noting that the sum with index  $i$  is general and makes equation (22) applicable for any value that  $p_w$  can assume.

Replacing (21) and  $\theta_r = \theta_s - \theta_m$  into (22) gives the following:

$$\lambda_a^{mg}(\theta_m) = \sum_{i=0}^{p_w-1} \sum_{n=1}^{\infty} \sum_{k=0}^{\infty} k_w \hat{B}_{g_n} \alpha_k \frac{DL}{4} \cdot \\ \int_{-\frac{\pi}{2p_w} + \frac{2\pi i}{p_w}}^{\frac{\pi}{2p_w} + \frac{2\pi i}{p_w}} \cos[(np-k)(\theta_s - \theta_m)] - \\ \cos[(np+k)(\theta_s - \theta_m)] d\theta_s \quad (23)$$

Solving the integrals in (23) in the integration variable  $\theta_s$  and repeatedly applying the sum ad difference identities gives:

$$\lambda_a^{mg}(\theta_m) = \sum_{i=0}^{p_w-1} \sum_{n=1}^{\infty} \sum_{k=0}^{\infty} k_w \hat{B}_{g_n} \alpha_k \frac{DL}{2} \left\{ \right. \\ \beta_{-k} \left[ \cos\left(\frac{2\pi i}{p_w}(np-k)\right) \cos((np-k)\theta_m) + \right. \\ \left. \sin\left(\frac{2\pi i}{p_w}(np-k)\right) \sin((np-k)\theta_m) \right] - \\ \beta_{+k} \left[ \cos\left(\frac{2\pi i}{p_w}(np+k)\right) \cos((np+k)\theta_m) + \right. \\ \left. \sin\left(\frac{2\pi i}{p_w}(np+k)\right) \sin((np+k)\theta_m) \right] \left. \right\} \quad (24)$$

where symbols  $\beta_{\pm k}$  are defined as

$$\beta_{\pm k} = \frac{1}{np \pm k} \sin\left(\frac{(np \pm k)\pi}{2}\right). \quad (25)$$

First of all, it can be observed that the following term is always equal to zero for each design choice:

$$\sum_{i=0}^{p_w-1} \sin\left(\frac{2\pi i}{p_w}(np \pm k)\right) = 0 \quad (26)$$

Then, (24) can be simplified as

$$\lambda_a^{mg}(\theta_m) = \sum_{i=0}^{p_w-1} \sum_{n=1}^{\infty} \sum_{k=0}^{\infty} k_w \hat{B}_{g_n} \alpha_k \frac{DL}{2} \left\{ \right. \\ \beta_{-k} \left[ \cos\left(\frac{2\pi i}{p_w}(np-k)\right) \cos((np-k)\theta_m) \right] - \\ \beta_{+k} \left[ \cos\left(\frac{2\pi i}{p_w}(np+k)\right) \cos((np+k)\theta_m) \right] \left. \right\} \quad (27)$$

Expression (27) is involved, but its particular form makes it possible to infer that for a given couple  $p, p_w$  (that is, for each specific motor geometry) only certain harmonics are present in the flux linkage. Among them, the lowest order one can be profitably used for fault detection, as in (7).