



UNIVERSITA' DEGLI STUDI DI PADOVA

Dipartimento di Ingegneria Industriale

Corso di Laurea Magistrale in Ingegneria dell'Energia Elettrica

## Multi-Objective Surrogate Optimization of a Surface Permanent Magnet Synchronous Machine

Relatore: Prof. Nicola Bianchi

Supervisori: Prof. Anouar Belahcen, Aswin Balasubramanian

Matteo Schiavon

Anno Accademico 2019/2020

Copyright © 2020 Matteo Schiavon



---

**Author** Matteo Schiavon

---

**Title** Multi-Objective Surrogate Optimization of a Surface Permanent Magnet Synchronous Machine

---

**Degree programme** Electrical Energy Engineering

---

**Major**

**Code of major**

---

**Supervisor** Prof. Anouar Belahcen & Prof. Nicola Bianchi

---

**Advisor** Aswin Balasubramanian, M.Sc.

---

**Date** July 29, 2020

**Number of pages** 90+2

**Language** English

---

**Abstract**

Synchronous machines are nowadays widely used and, due to the increasing energy demand, the development of ever higher efficiency is essential for energy saving. Optimization is playing a crucial role to achieve this goal. Moreover, after the Pareto concept has been introduced, multi-objective optimization became more and more employed among real-world engineering problems. In the last decades, the necessity for minimizing costs and obtaining efficient products at the same time led to an evolution process in both design and optimization methods. Nowadays, analytical modelling is typically used only during the preliminary stage and the actual design is entrusted to a finite element analysis. However, the time-stepping method in FEA becomes computationally highly expensive for machine optimization, which often requires a large number computations. Thus, by applying a response surface methodology to the design space it is possible to obtain surrogate functions, which play the role of actual functions in the optimization process and allow to save lots of time. Despite this, when dealing with problems having several optimization variables and a wide domain, surrogates are very likely to be inaccurate. Based on this notions, the development of a fast and accurate optimization design method in case of synchronous machine induced a thought of employing a clustering algorithm. To wrap it all up, a clustering-based surrogate optimization design is applied to a 16 kW surface permanent magnet SM and the achievements are illustrated. A two-dimensional time-stepping finite element method, integrated in Aalto's FCSMEK software, was used to perform all stages of the design requiring a simulation.

---

**Keywords** Permanent magnet machines, Energy efficiency, Finite element analysis, Design optimization, Particle swarm optimization, Pareto optimization, Sampling methods, Clustering algorithms

---



## Preface

This master's thesis has been accomplished in the Department of Automation and Electrical Engineering in Aalto University, which hosted me as an Erasmus student between January 2020 and July 2020. I would like to express my gratitude to my Prof. Nicola Bianchi for giving me this excellent opportunity. I would also like to express my sincere appreciation to my supervisor Prof. Anouar Belahcen and to my first advisor M.Sc. Aswin Balasubramanian who welcomed me in the warmest way making my stay enjoyable and productive.

I would like to thank Osaruyi Osemwinyen, Md Masum Billah, Avaz Ruzibaev, Alireza Nemat Saberi, Floran Martin, Ismet Gurbuz and other colleagues at Electromechanics group for creating a friendly working environment. I am also very grateful to my flatmates Enrico, Zaradacht, Mohamed, Zakaria, Imad, Joel, Erick and Eloi who made me feel like in a big family. Thanks to all of my friends here in Finland for making my Erasmus an unbelievable experience.

Finally, my deepest love and gratitude to my parents, my brother and Elena for giving me the opportunity to come to Finland. Despite the pandemic, their infinite support guided my stay and encouraged my work.

Espoo, July 29, 2020

Matteo Schiavon



# Contents

<b>Abstract</b>	<b>3</b>
<b>Preface</b>	<b>5</b>
<b>Contents</b>	<b>7</b>
<b>Symbols and Abbreviations</b>	<b>9</b>
<b>1 Introduction</b>	<b>13</b>
1.1 Background . . . . .	13
1.2 Thesis Objectives . . . . .	13
1.3 Thesis Structure . . . . .	14
<b>2 Literature Review</b>	<b>15</b>
2.1 Introduction to Synchronous Machine . . . . .	15
2.1.1 History of the machine and Development of Permanent Magnets	15
2.1.2 Working Principles and Machine Structure . . . . .	16
2.1.3 Inductances . . . . .	19
2.1.4 Losses Estimation . . . . .	21
2.2 Optimization . . . . .	23
2.2.1 Particle Swarm Optimization . . . . .	24
2.2.2 Multi-Objective Optimization . . . . .	29
2.3 Design of Experiments . . . . .	33
2.3.1 Sampling . . . . .	34
2.3.2 Surrogate Model . . . . .	37
2.3.3 Clustering . . . . .	40
<b>3 Research Methodology</b>	<b>45</b>
3.1 Design Theory of Surface Permanent Magnet Synchronous Machine .	47
3.1.1 Initial Data and Choice of Materials . . . . .	47
3.1.2 Main Dimensions, Air Gap and Stator Windings . . . . .	49
3.1.3 Tooth Width and Stator Slot Dimensions . . . . .	52
3.1.4 Height of Yokes and Permanent Magnets . . . . .	53
3.1.5 Resistance and Inductances Calculation . . . . .	54
3.2 Clustering Process . . . . .	56
3.2.1 Latin Hypercube Sampling . . . . .	56
3.2.2 Candidates Selection . . . . .	57
3.2.3 Clustering . . . . .	57
3.3 Construction of Surrogates . . . . .	58
3.4 Optimization . . . . .	59
3.4.1 Multi-objective Optimization Routine . . . . .	60
3.5 Design Assessment . . . . .	62
3.6 Introduction to FCSMEK . . . . .	63
3.7 Analytical Calculations . . . . .	65

3.7.1	Routine for Analytical Calculations . . . . .	67
<b>4</b>	<b>Results and Discussion</b>	<b>71</b>
4.1	Computational Burden and Convergence . . . . .	71
4.2	Optimization by Box-Behnken Design in a SPMSM . . . . .	73
4.2.1	Method Without Clustering . . . . .	74
4.2.2	Clustering-Based Method . . . . .	76
4.2.3	Comparative Analysis of the Accuracy . . . . .	78
4.2.4	Achievements using the Clustering . . . . .	80
4.3	Further Observations . . . . .	83
<b>5</b>	<b>Conclusions</b>	<b>85</b>
	<b>References</b>	<b>87</b>
<b>A</b>	<b>Errors in Surrogates Functions</b>	<b>91</b>
A.1	Mean Absolute Error . . . . .	91
A.2	Mean Relative Error . . . . .	91
A.3	Standard deviation . . . . .	91
<b>B</b>	<b>Relative Hypervolume</b>	<b>92</b>



# Symbols and Abbreviations

## Symbols

$A$	linear current density [A/m]
$a$	number of parallel paths in winding
$B$	magnetic flux density [T]
$B_{1\text{peak}}$	peak value of the fundamental air gap flux density [T]
$B_{\text{yoke}}$	maximum flux density of the yokes [T]
$B_{\text{max}}$	maximum value of the rectangular air gap flux density [T]
$C_1$	individual cognition parameter
$C_2$	social learning parameter
$c$	number of candidates for the LHS
$D_r$	external rotor diameter (air gap diameter) [m]
$D_s$	internal stator diameter (air gap diameter) [m]
$F$	distribution function
$f$	supply frequency [Hz], fitness/objective function
$f_s$	surrogate function
<b><math>gbest</math></b>	global best position
$H_c$	coercivity [A/m]
$h_{\text{PM}}$	height of the permanent magnet [m]
$h_{\text{yr}}$	height of the rotor yoke [m]
$h_{\text{ys}}$	height of the stator yoke [m]
$k_{\text{Cus}}$	space factor for copper
$k_{\text{Fe}}$	space factor for iron
$k_{w1}$	winding factor of the fundamental
$k_\rho$	experimental factor for Schuisky equation [W s <sup>2</sup> /m <sup>4</sup> ]
$L_{\text{md}}$	direct-axis magnetizing inductance [H]
$L_{\text{mq}}$	quadrature-axis magnetizing inductance [H]
$l$	axial length of the machine [m]
$l'$	effective core length [m]
$l_{\text{av}}$	average length of a coil turn [m]
$l_{\text{rpm}}$	axial length of the permanent magnets [m]
$M$	number of clusters
$m$	number of phases
$N$	number of samples, number of turns in series in a phase winding
$n$	synchronous speed [1/s], number of optimization variables
$n_s$	synchronous speed [rpm]
$P$	shaft power [W], number of particles of the PSO population
$P_{\text{Cu}}$	stator resistive losses [W]
$P_{\text{ex}}$	additional losses [W]
$P_{\text{Fe}}$	iron losses [W]
$P_\phi$	mechanical losses [W]
$P_{\phi,f}$	friction losses [W]
$P_{\phi,w}$	windage and ventilator losses [W]

$p$	number of pole pairs
$\mathbf{pbest}$	personal best position
$Q$	total number of stator slots
$q$	number of slots per pole and phase
$R$	resistance [ $\Omega$ ]
$R_{DC}$	DC equivalent resistance [ $\Omega$ ]
$rand_1$	random value parameter for cognitive component
$rand_2$	random value parameter for social component
$s$	number of samples of the RSM
$U_{m \text{ tot}}$	peak value of the total magnetic voltage [A]
$U_{m \delta}$	magnetic voltage over the air gap [A]
$U_{m \text{ ts}}$	magnetic voltage over the stator tooth [A]
$U_{m \text{ PM}}$	magnetic voltage over the permanent magnet [A]
$U_{m \text{ ys}}$	magnetic voltage over the stator yoke [A]
$U_{m \text{ yr}}$	magnetic voltage over the rotor yoke [A]
$\mathbf{V}$	velocity vector
$V_r$	rotor volume [ $\text{m}^3$ ]
$v_R$	tangential velocity of the rotor surface [m/s]
$W$	winding pitch [m]
$w$	inertia weight, weighted-sum method values
$\mathbf{X}$	position vector
$X_d$	direct-axis reactance [H]
$X_q$	quadrature-axis reactance [H]
$x$	optimization variable
$z_Q$	number of conductors in a slot
$\alpha_{Cu}$	temperature coefficient of copper resistivity [ $\text{K}^{-1}$ ]
$\alpha_{PM}$	relative width of the permanent magnet
$\boldsymbol{\beta}$	surrogate coefficients vector
$\beta$	surrogate coefficient
$\delta$	air gap width [m]
$\Theta$	temperature rise in copper wires [ $^{\circ}\text{C}$ ]
$\sigma_{Cu}$	conductivity of copper [ $\text{S/m}$ ]
$\sigma_{Ftan}$	tangential tension [Pa]
$\tau_p$	pole pitch [m]
$\tau_u$	stator slot pitch [m]
$\Phi$	total flux [Wb]
$\Phi_m$	air gap (main) flux [Wb]
$\Phi_{\sigma}$	leakage flux [Wb]
$\omega$	electric angular velocity [rad/s]

## Operators

$\sum_{i=1}^n$	sum over index $i$ , with $i = 1 \dots n$
$\mathbf{A} \cdot \mathbf{B}$	dot product of vectors $\mathbf{A}$ and $\mathbf{B}$

## Abbreviations

2D	two-dimensional
3D	three-dimensional
AC	alternating current
BBD	Box-Behnken design
CCD	central composite design
DC	direct current
DOE	design of experiments
FEA	finite element analysis
FEM	finite element method
GA	genetic algorithms
LHS	latin hypercube sampling
MAE	mean absolute error
MOO	multi-objective optimization
MRE	mean relative error
PM	permanent magnet
PMSM	permanent magnet synchronous machine
PRS	polynomial response surface
PSO	particle swarm optimization
RSM	response surface methodology
SOO	single-objective optimization
SPMSM	surface permanent magnet synchronous machine
STD	standard deviation
SyRM	synchronous reluctance machine



# 1 Introduction

## 1.1 Background

The synchronous machine has long been the most important of the electromechanical power conversion devices, playing a key role both in the production of electricity and in certain special drive applications. The machine soon underwent rapid developments and permanent magnets immediately proved their potential, being to replace the stator wound poles of a conventional motor. Since the beginning, the evolution process of the electrical machine has been progressing with the aim of optimizing [1] [2]. As most of the engineering problems, also in the permanent magnet synchronous machines the main concerns have always been maximizing earns and minimizing losses [3]. Also optimization itself has had its evolutionary process; in fact, before the introduction of the Pareto concept it was not fully exploited.

In general, the concept of optimization can be applied to every type of engineering design, including simple analytical equations. Although the analytical modelling of any kind of electrical machine have been formalized by many authors in the last decades, such models are typically used only during the preliminary design stage because they are obviously based on hypothesis in order to simplify their modelling. Consequently, the design procedure of such machines requires a finite element analysis (FEA) is employed to evaluate the design aspects disregarded in the first stage. In recent works, automated design procedures assisted by optimization algorithms and coupled with FEA have proved to be very efficient. However, several design iterations can be needed and the computational time can be easily become unacceptable [4]. Several methodologies and techniques came up in order to decrease the computational burden of an optimization design assisted by FEA. The Box-Behnken design (BBD), which is a response surface method, has revealed to be a powerful tool for this purpose, proving to be at the same time quite accurate. It permits to express the experimental relation between design variables and design functions (objectives and constraints) through simple second-degree polynomial functions. The latter are then called surrogate functions and are the ones being evaluated in the optimization routine [5].

## 1.2 Thesis Objectives

This work performs a multi-objective optimization design of a 160 kW surface permanent magnet synchronous machine. The objectives of the study, as the previous section has anticipated, will be the efficiency and the cost of the motor. The method which has been developed was based on the need to identify an alternative to the classic surrogate BBD that, in our case, did not lead to valuable solutions. For this purpose, a surrogate design based on clustering is proposed. We will see in which way and the reason why the clustering can ensure reliability to the surrogate optimization design.

### **1.3 Thesis Structure**

The thesis is divided into five chapters. In Chapter 1 background and motivation of the thesis are explained. Chapter 2 presents the literature review of the theoretical background and the history of synchronous machines, with a focus on permanent magnet synchronous machine. Chapter 3 introduces the design theory and the methodology which has been developed. Chapter 4 contains the the results and findings of the developed method. Chapter 5 depicts the final achievements of the conducted research work and the indication of future research perspectives related to this thesis.

## 2 Literature Review

Designing a surface permanent magnet synchronous machine (SPMSM) demands theoretical knowledge in rotating electrical synchronous machines, thus academic aspects are treated in Section 2.1. To pursue a multi-objective optimization (MOO) design certain optimization principles are crucial; they are discussed in Section 2.2. Design of experiments (DOE), which is a systematic and rigorous approach that applies techniques to data collection, is discussed in Section 2.3.

### 2.1 Introduction to Synchronous Machine

Nowadays, synchronous machines are still used primarily for large generators of electrical power at hydro, nuclear, or thermal power stations; quite commonly they can have power ratings of several hundred MVA. Like most rotating machines, a synchronous machine can operate also as a motor both in large sizes, like pumps in generating stations, and in small sizes. Depending on the application they may be requested either constant speed or variable speed, for example in industrial drives [2].

Since the outset, the evolution process of the electrical machine has been evolving with the intention of optimizing. These machines have become virtually ideal converters of energy and nowadays they require to have more and more technically and economically standards. This was permitted by constant innovation and enhancements in manufacture, testing and operation conditions [1].

The introduction of permanent magnets (PM) was, in all likelihood, the most important advancement in the development of the synchronous machine. Other key aspects that have led to major progress were the consolidation of a robust theoretical knowledge and software-assisted design.

#### 2.1.1 History of the machine and Development of Permanent Magnets

The beginnings of the synchronous machine are found in the last decades of the 18th century. Once several prominent scientists devised fundamental concepts for multiphase alternating-current machines, in 1887 F. A. Haselwander built the first three-phase synchronous generator and tried it out in practice. After a few years, in 1891, another decisive landmark in the development of this machine took place when the first long-distance transmission of high voltage three-phase power from Lauffen to Frankfurt began. Consequently, many countries committed themselves to a common effort for the improvement of the synchronous machine. Not surprisingly, in the past, it has long been the most remarkable of the electromechanical power-conversion devices, playing the major role in the production of electricity [1] [6].

The introduction of permanent magnets derived came from the necessity to design high-efficient machines. The manufacture and experimentation of these began in the early twentieth century with the production of carbon, cobalt and wolfram steels. A remarkable improvement was due to the discovery of aluminium-nickel-cobalt (AlNiCo) materials, in the 1930s, which are still used nowadays at temperature of 300 °C and above. Ferrites were first introduced in the 1950s and, due to their low price,

they are still dominant on the market. Another considerable step was taken in the 1960s, when the compounds of rare-earth metals and cobalt were invented. After the discovery of these materials, the last important advance was the neodymium-iron-boron (NdFeB) permanent magnets in 1983, which still nowadays yield the highest energy product [7].

### 2.1.2 Working Principles and Machine Structure

The main feature of the synchronous machine is that, unlike asynchronous, the current linkage component created by the rotor operates independently of the stator, this might explain why they're often known to as doubly excited machines [7].

In steady state conditions, the rotor rotates at the same constant speed of the air gap stator field; it is called synchronous speed. As long as all constraints of any kind are respected, after a brief synchronization transient, the machine will run at the synchronous speed, proportional to the frequency and inversely proportional to the pole pairs:

$$n_s = \frac{60 \cdot f}{p} \quad (1)$$

where  $n_s$  stands for the synchronous speed of the machine,  $f$  represents the supply frequency of the stator winding and  $p$  the number of pole pairs.

Since they're double excited, the flux in the air gap is therefore the resultant of the fluxes due to both rotor and stator current. This implies that this machine can draw both lagging and leading reactive current from the alternating current (AC) supply system but, if the rotor field winding provides just the necessary excitation, the stator will draw no reactive current and the motor will operate at a power factor equal to one [2]. Still, since the air gap flux caused by the stator current is not compensated spontaneously, the current in the rotor field has to be adjusted when needed in order to work in the ideal working condition of unitary power factor [7]. If the rotor excitation current is decreased, lagging reactive current will be drawn from the AC source to aid magnetization by the rotor field current. On the other hand, if the rotor field current is increased, leading reactive current will be drawn from the AC source to oppose magnetization by the rotor field current. Both cases will result in a power factor other than one [2].

Over time, the stator structure of this type of machine has become practically consolidated, with distributed three phases AC windings which originates the rotating stator magnetic field. On the other hand, the rotor construction of a synchronous machine can be various and it leads to different types of machine depending on the excitation system, as depicted in Figure 1: separately excited synchronous machine, synchronous reluctance machine (SyRM) and permanent magnet synchronous machine (PMSM) [7]. .

In a separately excited synchronous machine the rotor is powered with a single-phase winding supplied with direct current (called field winding current), which can be conducted via slip rings to the rotor. In this way a quite fast-responding control for the excitation current is easily achievable. Besides this, it is well known that the main disadvantage of this machine is the requirement of constant maintenance



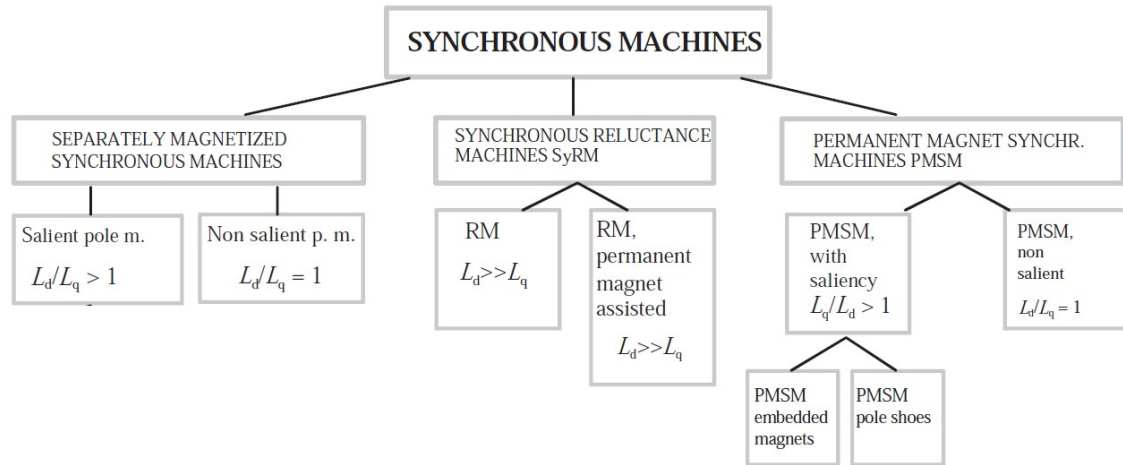


Figure 1: Synchronous machine family sorted according to the type of excitation system [8].

for the brushes and, for this reason, several brushless excitation systems have been developed [7].

Synchronous reluctance machine is the simplest of synchronous machines; the rotor consists of soft magnetic material, such as laminated silicon steel, which has multiple projections acting as salient magnetic poles through magnetic reluctance. The objective is to create a maximum difference between the direct-axis inductance, which is minimize, and the quadrature one, which is maximize. In order to improve some aspects of the machine, embedding magnets on the q-axis can help to reduce even further the q-axis reactance. In this case they are so named permanent-magnet-assisted synchronous reluctance machines [7].

In permanent magnet synchronous machines the current linkage component by

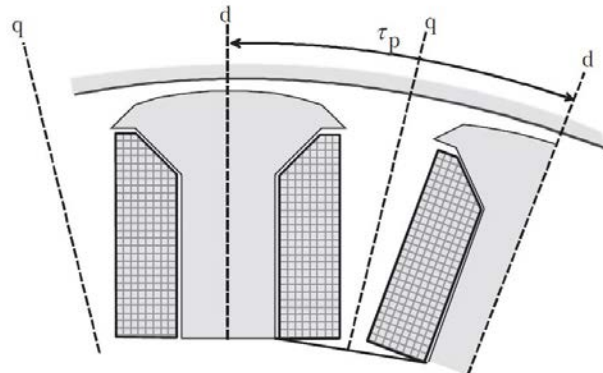


Figure 2: Salient poles with windings in a separately excited synchronous machine [8]. Direct-axis, d, and quadrature-axis, q, have a remarkable effect on salient-pole machines behaviour [8].

the rotor is generated directly by the permanent magnets[7].

According to their magnetic circuits, electrical machines can be further divided into two main categories, as depicted in Figure 1: salient-pole machines, in which the field windings are concentrated, and nonsalient-pole machines, where the magnetizing windings are spatially distributed [8]. This definition is clearly related to a separately excited synchronous machine, but the concept can be easily extended to any type of machine.

Another aspect that must be considered is the difference between d-axis and q-axis. In general, a direct-axis quantity is one whose magnetic effect is along the field pole axis, while a quadrature-axis quantity is one whose magnetic effect is along the perpendicular to field pole axis [9].

## Permanent Magnet Synchronous Machine

We have seen that permanent magnets are used on the rotor instead of a wound field. The advantages of the permanent magnet type are that no supply is needed for the rotor so the construction can be robust and reliable. Moreover, the steady state efficiency and power factor at full load are in most cases better than the equivalent induction motor [10].

The disadvantage is that the excitation is fixed, thereby controlling power factor via excitation is no longer possible. In addition, the designer must either choose the shape and disposition of the magnets to match the requirements of one specific load, or seek a general purpose compromise [10].

Early permanent magnets suffered from the tendency to be demagnetised by the high stator currents during starting, and from a restricted maximum allowable temperature. Much improved versions using high coercivity rare-earth magnets were developed during the 1960s to overcome these problems [10].

The features of a PMSM are determined from the rotor construction; Figure 3 illustrates different permanent magnet rotor constructions. If the magnets are set on the rotor surface, as illustrated in 3(a), the rotor is in principle nonsalient, since the relative permeability of NdFeB magnets is around one. Embedded magnets, like in 3(b), give rise to a machine in which the quadrature-axis synchronous inductance is higher than the direct-axis synchronous one. Also the pole shoe construction, 3(c) produces a similar inductance ratio.

It is well known that embedding the magnetic material completely inside the rotor structure wastes a significant proportion of the flux produced by the magnet in leakage components of the rotor. On the other hand, embedded magnets are both mechanically and magnetically protected.

Synchronous reluctance machine has torque density not as high as an equivalent PM machine while its power factor is rather low. PMs, either ferrite magnets or rare-earth magnets, are inserted into a SyRM's flux barriers to improve its performance, as shown in 3(d). Thus, it becomes a hybrid machine and its characteristics lie in between SyRM and PM machine with the benefit that torque and power factor can be significantly improved [11].

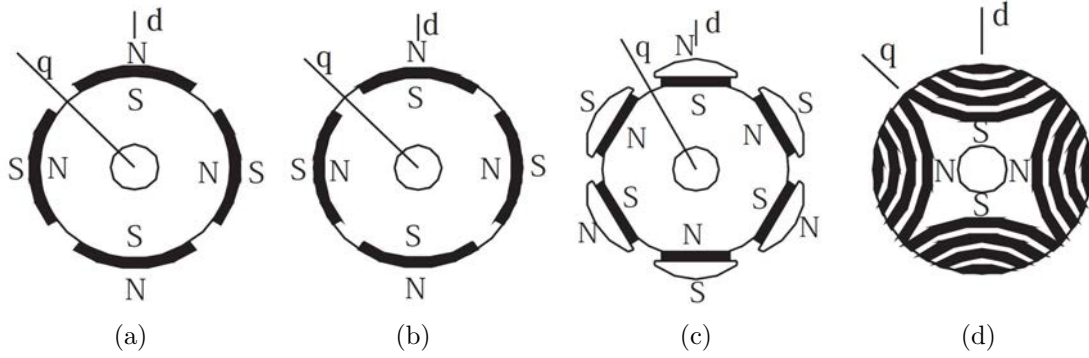


Figure 3: Different rotors a permanent magnet synchronous machine [8]: (a) surface permanent magnets; (b) embedded permanent magnets; (c) salient pole shoe; (d) reluctance rotor equipped with permanent magnets.

### 2.1.3 Inductances

In general rotating electrical machines theory, the total flux  $\Phi$  consists of the main flux (air gap flux)  $\Phi_m$  and the leakage flux  $\Phi_\sigma$ . The electromechanical energy conversion lies in the air gap flux portion, while the leakage part does not participate. The main flux has to cross the air gap of the machine, since it plays fundamental role of connecting electromagnetically both the stator and the rotor (that is why it is also named air gap linkage flux). On the other hand, leakage fluxes, which occurs both in rotor and stator, comprise all the flux components that do not cross the air gap and those components crossing the air gap that do not participate in the formation of the main flux linkage. The air gap flux corresponds to the magnetizing inductance  $L_m$ , whereas the leakage component corresponds to a leakage inductance  $L_{s\sigma}$  [7].

Along the rotor surface the air gap may vary in direct and quadrature directions, which leads to different inductances in the direct-axis and quadrature-axis, respectively  $L_d$  and  $L_q$ . In both directions, the inductances consist of the magnetizing inductance,  $L_{md}$  and  $L_{mq}$ , and the leakage inductance  $L_{s\sigma}$  [7]:

$$L_d = L_{md} + L_{s\sigma} \quad (2)$$

and

$$L_q = L_{mq} + L_{s\sigma}. \quad (3)$$

In the case of a machine operating at a constant frequency (constant speed), reactances can be employed:

$$X_{d/q} = 2\pi f L_{d/q}. \quad (4)$$

More precisely, direct-axis synchronous reactance is defined as the reactance offered to the stator flux when the peak of the rotating stator magnetic field is directed along the d-axis; this situation is represented in Figure 4(a). Under this condition, the air gap width is minimum hence the reluctance is also minimum and the stator flux is maximum [9].

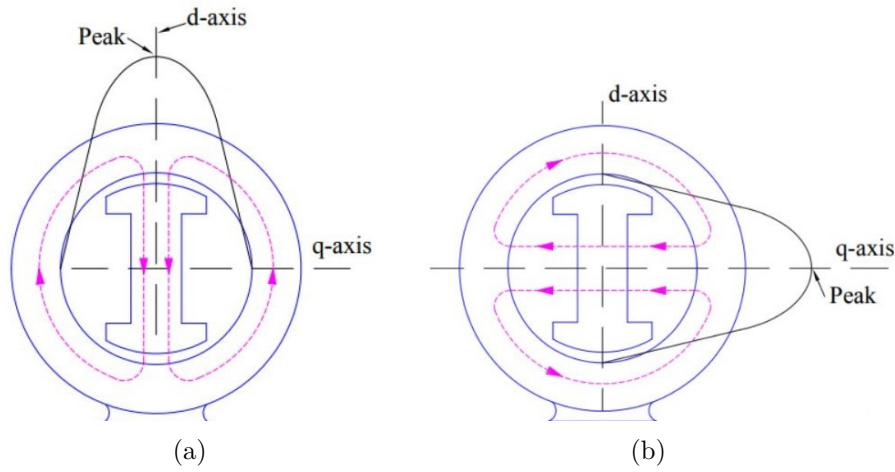


Figure 4: Salient pole synchronous machine [9]: (a) direct-axis synchronous reactance; (b) quadrature-axis synchronous reactance.

When the peak of stator magnetic field coincides with the quadrature-axis, the reactance offered to the stator flux is called quadrature-axis reactance. In Figure 4(b), the peak the stator rotating field is along the perpendicular to the d-axis. In this case, the air gap width is maximum thus the reluctance is maximum as well and the stator flux is minimum [9].

As depicted in Figure 4, in salient-pole synchronous machine the air gap is clearly non-uniform and, in particular,  $L_d$  is bigger than  $L_q$  as anticipated by Figure 1. But due to uniform air gap in nonsalient-pole synchronous machine, both  $L_d$  and  $L_q$  are equal; for instance, this happens in the case of a SPMSM, where the air gap remains constant and the machine is electrically symmetrical because of unitary relative permeability of NdFeB magnets.

### Leakage Flux Components

The calculation of leakage inductances from the structural dimensions of the machine is a rather demanding task. Thus, in this paragraph it is only given a brief and representative explanation.

The flux components that do not cross the air gap self-evidently belong to the leakage flux. Among these components, represented in Figure 5, there are:

- Pole leakage flux;
- Slot leakage flux;
- Tooth tip leakage inductance;
- End winding leakage flux.

To determine the leakage flux, leakage flux paths have to be investigated. Paths of pole leakage fluxes of a salient-pole machine are shown in Figure 5(a), while flux

paths of both pole and slot leakage of a nonsalient-pole machine are illustrated in Figure 5(b). The tooth tip leakage inductance is determined by the magnitude of leakage flux flowing in the air gap outside the slot opening. This flux leakage is illustrated in the upper part of Figure 5(c) and its current linkage in the slot causes a potential difference between the teeth on opposite sides of the slot opening. End winding leakage flux results from all the currents flowing in the end windings, as represented in Figure 5(d).

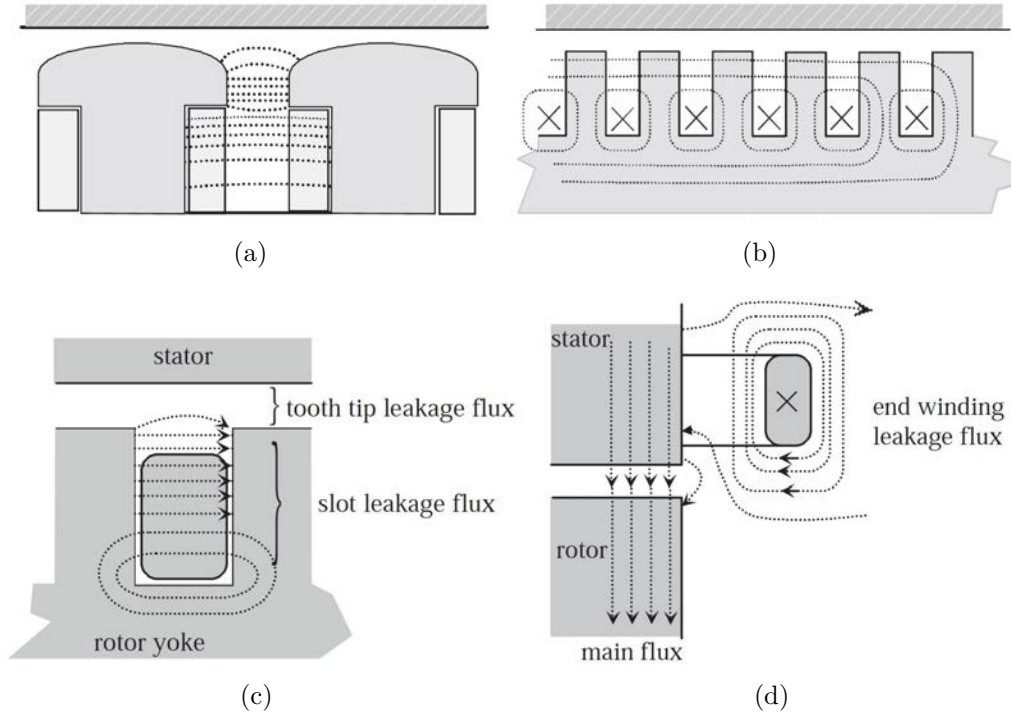


Figure 5: Leakage flux components not crossing the air gap [8]: (a) pole leakage fluxes of a salient-pole machine; (b) pole and slot leakage of a nonsalient-pole machine; (c) leakage flux components of a coil winding contained in slots of a nonsalient-pole machine; (d) end winding leakage flux.

Leakage fluxes that cross the air gap are included in the air gap flux. An air gap flux does not completely link the windings of the machine to each other. The reasons for incomplete linking are: short pitching, skewing and the spatial distribution of the windings. A weakening of the linking between the stator and rotor windings is not usually regarded as leakage, but it has to be taken into account. The spatial distribution of the windings causes an air gap (or harmonic) flux leakage. These leakage flux components crossing the air gap are included in the air gap flux [8].

#### 2.1.4 Losses Estimation

The machine efficiency can be determined by calculating or measuring the total losses. The most convenient way of defining the losses of an electrical machine is classifying

the loss components according to their origin or structure [6]. The losses in the synchronous machine, like other electrical machines, can be classified as following:

- The resistive losses are caused by the current flowing through stator and rotor windings and also rotor bars. The resistive losses occur in the stator winding due to the supply current, while in case of separately excited synchronous machine, the direct current (DC) is responsible for resistive losses in rotor winding. The resistive losses can be estimated based on the DC equivalent resistance  $R_{DC}$  value of the winding as:

$$P_{Cu} = m I_{rms}^2 R_{DC}, \quad (5)$$

where  $m$  is the number of phases and  $I_{rms}$  is the root mean square value of the current. A more accurate estimation of resistive losses can be achieved by means of the AC resistance of the phase winding instead of the DC one [6].

- The iron losses are usually given by manufacturers in a combined way, which means that even if hysteresis losses differ from eddy current losses, these two components are not usually given separately. According to standard calculations, the core losses are found by subdividing the magnetic circuit of the machine in  $n$  components, in which the flux density is approximately constant. Once the masses  $m_{Fe,n}$  of the different areas  $n$  have been computed have been calculated, the iron losses of the different parts of the machine can be approximated as follows:

$$P_{Fe,n} = k_{Fe,n} P_{15} \left( \frac{\widehat{B}_n}{1.5} \right)^2 m_{Fe,n}, \quad (6)$$

where  $\widehat{B}_n$  is the peak value of the flux density and  $P_{15}$  indicates the losses per mass unit at the flux density peak value of 1.5 T at 50 Hz. For manual calculations, the empirical correction coefficients  $k_{Fe,n}$  takes empirically into account various phenomena as: pure sinusoidal flux variation never occurs alone in any parts of the machine, field harmonics are present, stresses created in the punching of the sheet increase the loss index [8].

- The additional losses are caused by several different phenomena, some of them can be very difficult to calculate accurately. They include for example, the losses caused by the skin effect in conductors, which have not been taken into account in  $P_{Cu}$  by using the  $R_{DC}$ . Permeance harmonic losses in the rotor surface permanent magnets caused by the stator slot openings are also considered in the additional losses. However, due to the difficulty of the calculations and to the impossibility of having an accurate estimation, they can be assumed as:

$$P_{ex} = 0.005 \cdot P. \quad (7)$$

- The mechanical losses are a consequence of different mechanical issues, like bearing friction and windage. Bearing friction losses  $P_{\rho,f}$  depend on the shaft speed, bearing type, properties of the lubricant and the load on the bearing



and, according to SKF manufacturer [12], bearing friction losses in the normal operating conditions are:

$$P_{\rho,f} = 0.5\Omega\mu F D_{bearing}, \quad (8)$$

where  $\Omega$  is the angular frequency of the shaft supported by a bearing,  $\mu$  the friction coefficient,  $F$  the bearing load and  $D_{bearing}$  the inner diameter of the bearing [7]. Windage losses become more and more significant with increasing machine speed. These losses are a consequence of the friction between the rotating surfaces and the surrounding gas, usually air. An experimental equation that include both windage and ventilator losses  $P_{\rho,w}$  was given by Schuisky [13]:

$$P_{\rho,w} = k_{\rho} D_r (l + 0.6\tau_P) v_r^2, \quad (9)$$

where  $k_{\rho}$  is an experimental factor (Table 9.2 [8]) and  $v_r$  the surface speed of the rotor.

- The permeance harmonic losses  $P_{PMEC}$  are produced by the stator slot openings; however, the calculation is a rather laborious task [14].

## 2.2 Optimization

Maximizing earns and minimizing losses have always been two fundamental design concerns in engineering problems. Since, for most fields of knowledge, the complexity of optimization problems increases as technology and science advance, there's a growing need in develop methods and algorithms useful both for enhance and speeding up the design. Quite often, examples of engineering problems that require an optimization process in the planning stage are in energy conversion and distribution and in mechanical design [3].

The aim of any optimization study is to determine a vector of variables  $\mathbf{X} = [x_1, x_2, \dots, x_n]$  that minimizes, or maximizes, the function  $f(\mathbf{X})$ . This is called fitness function, or objective function, and it may assess how good or bad is the set of variables  $\mathbf{X}$ . On the other hand, the vector  $\mathbf{X}$  is known as position vector; it represents a variable model and it is a  $n$ -dimensional vector, where  $n$  represents the number of variables affecting the objective that may be determined in the problem [3].

In order to find the optimum, such investigation can be performed through several criteria and for this reason there are as many approaches that one could perform. In spite of a wide range of methods that could be used, there is not a main one that is considered to be the best for any kind of problems. One optimization approach that is appropriate for a problem might not be so for another one; it depends on several features (for example, whether the function is differentiable and its concavity). To solve a problem, one must understand different optimization methods and must be able select the algorithm that best fits on the problem's features [3].

Genetic algorithms (GA) was one of the first design optimization tool for industrial engineers capable of success. Back in the 1970's, GA was already used for

problem solving and it was proved to be a significant contribution for scientific and engineering applications. The main innovation was that GA does not implement a mathematically guided algorithm and optima is obtained by the evolution from generation to generation without demanding mathematical formulation. The search for the optimal by means of a population which evolves according to a stochastic, discreet and nonlinear process turned out to have such a great success to lead, in a few years, the development of another even more successful method: the particle swarm optimization [15].

### 2.2.1 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a metaheuristic algorithm based on the concept of swarm intelligence that is appropriate to optimize continuous nonlinear functions. The method comprises a very simple concept implemented in a few lines of computer code which requires only primitive mathematical operators; thereby, the algorithm is computationally inexpensive in terms of both memory requirements and speed [16].

The origins of the method date back to early of 1990s, when several studies regarding the social behavior of animal groups, such as flocks and shoals, were developed. These studies showed that some animals belonging to a certain group are able to share information among them [3]. Inspired by these works, James Kennedy and Russell Eberhart proposed in 1995 the Particle Swarm Optimization algorithm that emulates the interaction between members to share information [17].

PSO has roots in two main component methodologies. Perhaps more obvious are its ties to artificial life in general, in particular to swarming theory, but it is also related to evolutionary computation, having ties to both genetic algorithms and evolutionary programming [18].

Two paradigms, both implementing the swarm concept, have been proposed in that work: one globally oriented and the other one locally oriented, which were initially denominated respectively "Gbest" and "Lbest" algorithms by the two authors. The former was the most successful one by far but, to be precise, the one proposed in 1995 is the inertial version of the "Gbest" algorithm but, over the years, a lot of other forms have been proposed as variations of that classic formulation. For instance, the linear-decreasing of the inertia weight is nowadays the most common and robust formulation of the globally oriented version. Besides variations in the method itself, hybrid models or even quantum inspired approach optimization techniques have been applied to PSO [3].

Interest and research in PSO have been significant because since the outset tests found the implementation to be effective with several kinds of optimization problems and in combination with other existing algorithms [17].

In the following paragraphs, only the "Gbest" algorithm is take into consideration, being the state-of-the-art form of particle swarm optimization, namely the simplest one and the most basic to understand better other derivations [3].



## The Concept

This method performs the search of the optimal solution through particles whose trajectories are adjusted by a stochastic and a deterministic component [17]. The system is originally initialized with a random population of  $P$  particles, which are randomly assigned also the initial velocities. The particles are then “flown” through the domain and each keeps track of its coordinates in hyperspace which are associated with the best solution (fitness) it has achieved so far. The position vector with that fitness is stored and it’s called ***pbest***, while the best solution itself correspond to  $f(\mathbf{pbest})$ . Another “best” value is also tracked since the “global” version of the particle swarm optimizer keeps track of the overall best value; its location, obtained thus far by any particle in the population is called  $f(\mathbf{gbest})$ . The concept consists of, at each time step, changing the velocity (accelerating) each particle toward its ***pbest*** and ***gbest***. Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward ***pbest*** and ***gbest*** [16].

## The Structure

The diagram in Figure 6, which shows the flowchart of the algorithm, might help in order to understand the logical thread of the procedure. The whole population is

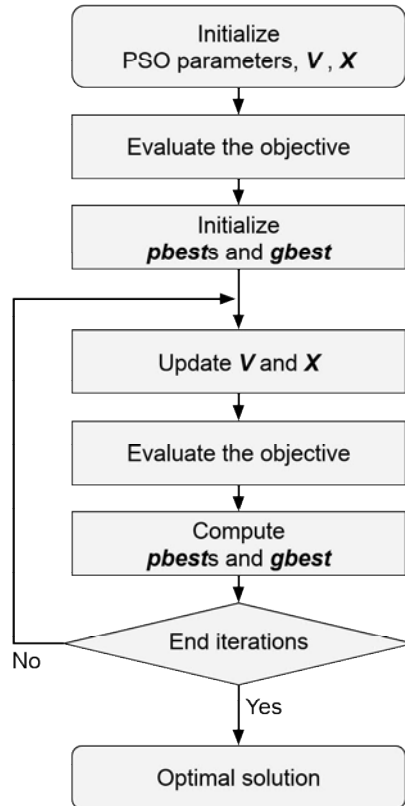


Figure 6: PSO flowchart.

initially made up by an array of  $P$  particles with random positions and velocities having  $n$  dimensions. For each particle the objective is evaluated and the ***pbest*** is initialized with a copy of the first population of particles. The ***gbest*** is initialized with a copy of the position which has the best fitness.

After this first evaluation, the iterative cycle starts and it goes on until a stopping criterion is satisfied. For each particle the velocities are updated according to

$$\mathbf{V}^{t+1} = w \cdot \mathbf{V}^t + C_1 \cdot rand_1 \cdot (\mathbf{pbest} - \mathbf{X}^t) + C_2 \cdot rand_2 \cdot (\mathbf{gbest} - \mathbf{X}^t). \quad (10)$$

The above equation denotes that there are three different contributions to a particle's movement affected by five different parameters.

The parameter  $w$  is named inertia weight and it reveals important for balancing the the overall motion of the population; in fact, we can say that the algorithm exploits a global search (also known as exploration) when higher values are set, while it turns into a local search (known as exploitation) when lower values are chosen. Velocity update equation's first term is a product between parameter  $w$  and particle's previous velocity, which is the reason it denotes a particles' previous motion into the current one. Hence, for example, if  $w = 1$ , the particle's motion is fully influenced by its previous motion, so the particle may keep going in the same direction. On the other hand, if  $0 \leq w < 1$ , such influence is reduced, which means that a particle rather goes to other regions in the search domain. Therefore, as the inertia weight parameter is reduced, the swarm may explore more areas in the searching domain, which means that the chances of finding a global optimum may increase. However, there is a price when using lower  $w$  values, which is the simulations turn out to have an higher computational time [3].

One may notice that the value of  $w$  is one of the main differences between the inertial formulation of PSO, where it is a positive constant value, and other versions derived from it. In the linear-decreasing formulation it is linearly decreased in the course of the iterations; starting from high values to low values it promotes a global exploration in the beginning while it turns into a local exploitation during the last iterations, which is quite logical because at the end of the routine particles should be converging towards the region in which the best solution is probably found [3].

The individual cognition term, which is the second term of Equation 10, is computed by means of the difference between the particle's own best position, ***pbest***, and its current position  $\mathbf{X}$ . One may notice that the idea behind this term is that as the particle gets more distant from the ***pbest*** position, the difference  $\mathbf{pbest} - \mathbf{X}^t$  must increase; therefore, this term increases, attracting the particle to its best own position. The parameter  $C_1$  existing as a product in this term is a positive constant and it is an individual cognition parameter, and it weighs the importance of particle's own previous experiences. The other parameter that composes the product of second term is  $rand_1$ , and this is a random value parameter with  $[0, 1]$  range. The role played by the random parameter is quite important, as it increases the most likely global optima avoiding premature convergences [3].

Finally, the third term is the social learning one. Because of it, all particles in the swarm are able to share the information of the best point achieved regardless of which particle had found it, for example, ***gbest***. Its format is just like the second

term, the one regarding the individual learning. Thus, the difference  $\mathbf{gbest} - \mathbf{X}^t$  acts as an attraction for the particles to the best point until found at some iteration. Similarly,  $C_2$  is a social learning parameter, and it weighs the importance of the global learning of the swarm. And  $\text{rand}_2$  plays exactly the same role as  $\text{rand}_1$  [3].

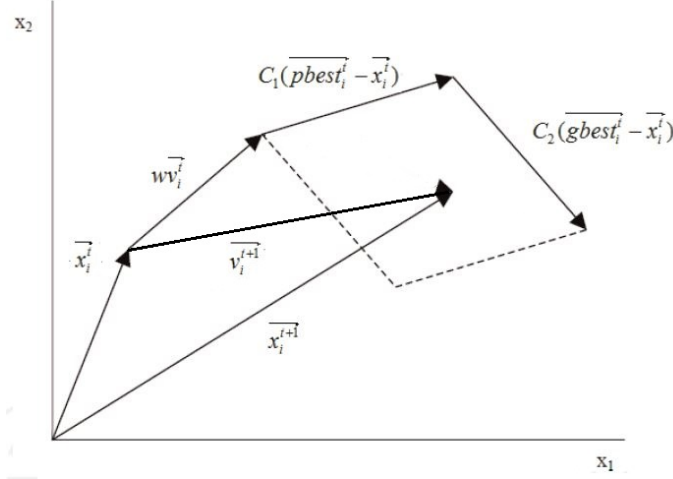


Figure 7: Each particle's position is updated according to the three contributions: 1)  $w \cdot \mathbf{V}^t$ ; 2)  $C_1 \cdot \text{rand}_1 \cdot (\mathbf{pbest} - \mathbf{X}^t)$ ; 3)  $C_2 \cdot \text{rand}_2 \cdot (\mathbf{gbest} - \mathbf{X}^t)$ .

After calculating the new velocities, each particle's position is easily updated as

$$\mathbf{X}^{t+1} = \mathbf{X}^t + \mathbf{V}^{t+1}. \quad (11)$$

Considering both Equation 10 and 11, Figure 7 illustrates the updating process of a particle's position in a bi-dimensional problem with variables  $x_1$  and  $x_2$ . The arrowhead of the new position vector  $\mathbf{X}_i^{t+1}$  has to stay inside the rhombus formed by the two vectors  $C_1(\mathbf{pbest}_i - \mathbf{X}_i^t)$  and  $C_2(\mathbf{gbest}_i - \mathbf{X}_i^t)$  because of the two random parameters with  $[0 \ 1]$  range.

After the positions have been updated, for each particle the objective is evaluated and compared with the previous best personal value; in this way, if  $f(\mathbf{X}_i^{t+1}) < f(\mathbf{pbest}_i^t)$  then it becomes  $\mathbf{pbest}_i^{t+1} = \mathbf{X}_i^{t+1}$ . The same evaluation is made with the previous global best: if in the present iteration there is a particle  $i$  in the position  $\mathbf{X}_i^{t+1}$  such that  $f(\mathbf{X}_i^{t+1}) < f(\mathbf{gbest}^t)$ , it will be  $\mathbf{gbest}^{t+1} = \mathbf{X}_i^{t+1}$  [16].

If the termination criterion of the iterations has been appropriately selected, once it has been achieved, the population of particles has converged into a small region of the domain where the best solution is located; the optimum found by the algorithm is given by  $f(\mathbf{gbest})$  and its position is stored in  $\mathbf{gbest}$ .

## The Constraints

Implementing an optimization problem often requires to consider some design constraints. Specific constraints on optimization variables are practically always present, but other general design constraints are growing in demand since design optimizations are becoming more and more articulated.

Constraints on variables are basically defined once the domain of the optimization routine is set. Most of the times in real engineering problems, each variable may require to be investigated within range that must be established in advance by the designer. Thus, the original PSO algorithm has to be modified to include dynamic maximum velocity function and bouncing technique to enhance particles to remain within the defined domain. Forcing the speed to remain within a maximum value is really useful for avoiding particles to get straight to the boundaries. On the other hand, the bounce method is used to bounce inside the domain again particles that have escaped from the boundaries [19]. This technique, which allows the algorithm not to loose any particle, includes more than one possibility to choose the position in which the particle has to be returned: randomly, by reflecting and by damping are three different bouncing techniques. These latter and other two possibilities are illustrated in Figure 8.

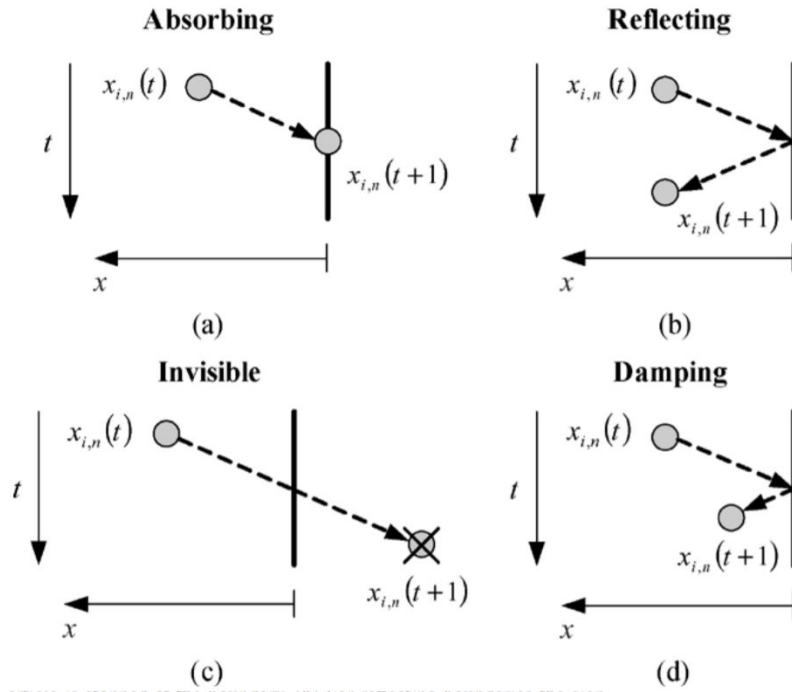


Figure 8: Different techniques to treat particles going beyond the boundary [20]: (a) for absorbing boundaries the velocity of the particle is zero and the particle is stopped at the boundary; (b) for reflecting boundaries the sign of the velocity is reversed and the particle is reflected back to the search space after the impact; (c) for invisible boundaries the particle is allowed to escape the boundary of the search space and is ignored by the fitness evaluator; (d) for damping boundaries part of the velocity is absorbed by the boundary during the impact and the particle is then reflected back with a lesser velocity of a reversal of sign.

When dealing with real engineering issues, the design might require, apart from more than one objective to be optimize, other constraints; they are particular conditions that have to be verified in order to obtain an acceptable solution. In that case, not only fitness functions must be taken into account during the optimization

algorithm. Considering what we have seen before, we are allowed to call simply boundaries the constraints on variables, while from now on constraints are the design specifications given by functions which must be evaluated without being optimized. In principle the optimization algorithm, even before finding optimal solutions, must take care that solutions satisfy the constraints; if they can't be met, no fitness function will matter [19].

In the specific case of PSO by making a small adjustment the algorithm will be able to perform a research method which considers the constraints. The two following rules are the key point of the modification that must be introduced in the algorithm [19]:

- in particles in which constraints are not satisfied, the constraints become part of the objective function; this means that the discrepancy between the evaluated value and the required limit value is numerically evaluated and minimized. In this way particles are moved towards regions which are closer to meet the constraints with respect to others. Basically it is the same concept applied when minimizing the real objective function.
- particles respecting constraints are better than ones which don't.

In particular, these two rules directly affect the way *pbest* and *gbest* are defined.

## 2.2.2 Multi-Objective Optimization

Multi-objective optimization is an integral part of optimization activities and has a tremendous practical importance, since almost all real-world optimization problems are, in addition to being nonlinear, ideally suited to be modeled using multiple conflicting objectives [21]. It has been applied to many fields of science and engineering; more commonly in the latter applications, designers are making decisions between clashing objectives, such as maximizing performance while minimizing fuel consumption and emission of pollutants of a vehicle. Similarly, in the field of electrical machinery, efficiency and cost represent two conflicting targets [22].

Due to lack of suitable solution techniques, the classical means of solving such problems were primarily focused on artificially converting multiple objectives into a single-objective optimization (SOO) problem. The difficulty of finding a single optimum solution arose because such problems give rise to a set of trade-off optimal solutions (known as Pareto optimal solutions). However, after that, the evolutionary assets have been to solve a multi-objective optimization problem as it is. This means that it becomes important to find not just one optimal solution, but as many of them as possible. In this way, any two such solutions constitute a trade-off between the objectives, namely, one is better in terms of one objective, but this betterment comes only from a sacrifice on the other objectives. In this sense, all such trade-off solutions are optimal solutions to a MOO problem and only the users, in the face of entirely practical considerations, will be in a better position to make a choice [21].

For better approach MOO problems, two types of plots are needed. The first one depicts the design space and it illustrates graphically the optimization variables; in

the simplest case, only two optimization variables affect the problem and the plane will be described by  $x_1$  and  $x_2$ , as depicted in Figure 9(a). It might be used to plot the isolines of the objective functions as well as to display the value of the variables yielding the Pareto front. Moreover, this chart can be useful to visualize any kind of boundary on the variables, as depicted by the dotted lines in Figure 9(a). On the other hand, criterion space, also called objectives space, can be depicted with axes identifying the objective functions. Again, in the easiest case represented by Figure 9(b), only two fitness functions are investigated and the plane will be described by the axes  $f_1$  and  $f_2$ . Constraints and boundaries of  $x_1, x_2, \dots, x_n$  can be translated into the criterion space as well, by solving them in terms of  $f_1, f_2, \dots, f_n$  even though this last step is often of no practical use [22].

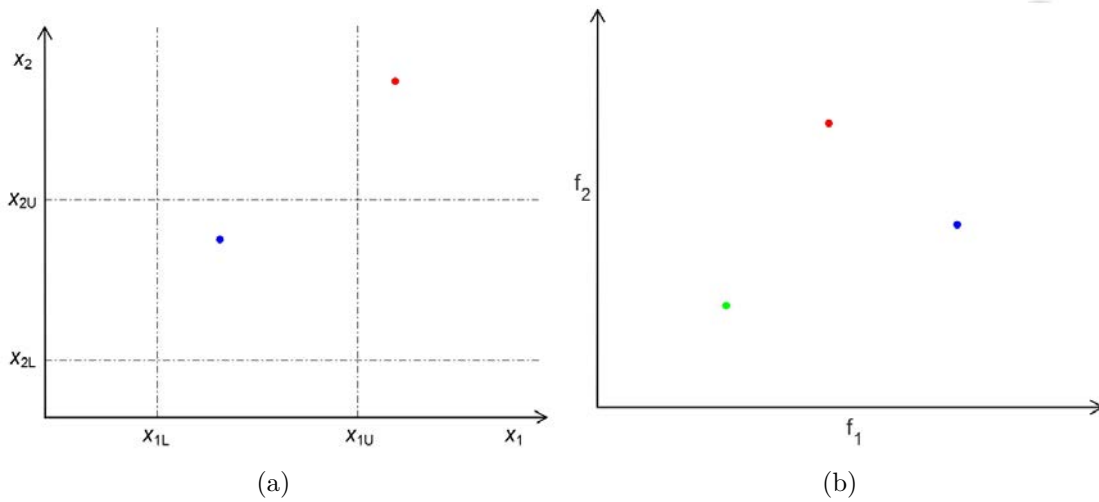


Figure 9: Multi-objective optimization work spaces: (a) design space, with upper (U) and lower (L) boundaries represented by the dotted lines; (b) criterion space.

The concept of correct design can thus be summarized through two elementary concepts [22]:

- **Attainability:** each point in the design space can be mapped to a point in the criterion space, nonetheless the opposite may not be true. In fact, we are only interested in the attainable points in the criterion space, namely, those which can be linked to a point in the design space. With regard to Figure 10, they are the blue and the red dots; on the contrary, the green dot cannot be linked to any couple of parameters in the design space.
- **Feasibility:** More easily, the feasibility of a design means that no constraint is infringed in the design space. Still looking at Figure 10, the blue dot is the only one which meets both the constraints, namely it stays within the design boundaries.

Before introducing the Pareto approach, a few methodologies of studying a MOO problem by turning it into a SOO one are briefly introduced.

## Methods with Articulations of Preferences

Since a primary goal of multi-objective optimization is to model a designer's preferences, these methods are categorized depending on how the designer articulates these preferences: methods with a priori articulation of preferences or methods with a posteriori articulation of preferences. As discussed earlier, by specifying some preferences, the MOO is converted to a SOO problem, leading to a single solution [22].

Articulating preferences a priori may be enunciated in terms of goals or the relative importance of different objectives. Most of these methods incorporate parameters, which are coefficients, exponents, constraint limits, and so on, that can either be set to reflect designer preferences and be continuously altered. The two most representative are illustrated below [22]:

- The weighted-sum method is the simplest and most common approach; it easily combines multiple objectives into one single-objective scalar function, as

$$f(\mathbf{X}) = \sum_{i=1}^N w_i \cdot f_i(\mathbf{X}), \quad (12)$$

with

$$\sum_{i=1}^N w_i = 1, \quad (13)$$

where  $w_i$  are the weight values, set by the decision maker, which generally reflect the relative importance of the  $N$  objectives. The designer may either set  $w_i$  to reflect preferences before solving the problem or methodically alter them to yield different optimal points; in fact, involving weights in this way, can be either used to generate a single solution or multiple solutions.

- Another possible approach comes from the possibility to minimize the “distance” to an ideal utopia point  $f(0)$  in the objectives space. The new objective  $f'$  to be minimize, can be represented as follows:

$$f'(\mathbf{X}) = \sum_{i=1}^N w_i \cdot (f_i(\mathbf{X}) - f_i(0)), \quad (14)$$

where  $f_i(0)$  is the utopia value for the  $i^{th}$  objective function.

Methods with a posteriori articulation of preferences are needed when it's difficult for the designer to express preference a priori. Several methods belong to this category and also, in a MOO problem, applying the genetic algorithm is often considered a posteriori method [22].

## Non-Dominated Solutions and Pareto-Optimal Solutions

The idea of Pareto optimality, summarized by the shape of the Pareto front itself, can be easily appreciate in the criterion space noticing that, for any point on the



front, an objective function cannot be further reduced without increasing the other objective function. Therefore, it becomes evident that a point in the feasible design space is called Pareto optimal if there is no other point in the set that reduces at least one objective function without rising another one. It is big to note that the Pareto optimal set is always on the boundary of the feasible criterion space [22].

Most multi-objective optimization algorithms use the concept of domination. In these algorithms, two solutions are compared on the basis of whether one dominates the other or not. Assuming any number of objective functions, the following definition covers mixed problems with minimization of some objective functions and maximization of the rest of them. A solution  $\mathbf{X}_1$  is said to dominate the other solution  $\mathbf{X}_2$  if both the following conditions are true [21]:

1. The solution  $\mathbf{X}_1$  is no worse than  $\mathbf{X}_2$  in all objectives;
2. The solution  $\mathbf{X}_1$  is strictly better than  $\mathbf{X}_2$  in at least one objective.

Let us consider a two-objective optimization problem with five different solutions shown in the objective space, as illustrated in Figure 10(a). Let us also assume that the objective function 1  $f_1$  needs to be maximized while the objective function 2  $f_2$  needs to be minimized. Since both objective functions are of equal importance to us, it is usually difficult to find one solution which is best with respect to both objectives. However, by using the above definition of domination, it is possible to decide which solution is better among any two given solutions in terms of both objectives. For example, comparing solutions 1 and 2, we observe that solution 1 is better than solution 2 in both objective function 1 and 2. Thus, both of the above conditions for domination are satisfied and we may write that solution 1 dominates solution 2. We take another instance of comparing solutions 1 and 5. Here, solution 5 is better than solution 1 in the first objective and solution 5 is no worse (they are equal) than solution 1 in the second objective. Thus, both the above conditions for domination are also satisfied and we may write that solution 5 dominates solution 1. Since the concept of domination allows a way to compare solutions with multiple objectives, most multi-objective optimization methods use this domination concept to search for non-dominated solutions [21].

Continuing with the comparisons in the previous section, let us compare solutions 3 and 5 in Figure 10(a). We observe that solution 5 is better than solution 3 in the first objective, while solution 5 is worse than solution 3 in the second objective. Thus, the first condition is not satisfied for both of these solutions. This simply suggests that we cannot conclude that solution 5 dominates solution 3, nor can we say that solution 3 dominates solution 5. When this happens, it is customary to say that solutions 3 and 5 are non-dominated with respect to each other because both objectives are of equal importance. For a given finite set of solutions, we can perform all possible pair-wise comparisons and find which solution dominates which and which solutions are non-dominated with respect to each other. At the end, we expect to have a set of solutions, any two of which do not dominate each other and also, for any solution outside of this set, we can always find a solution in this set which will dominate the former. Thus, this particular set has a property of dominating all



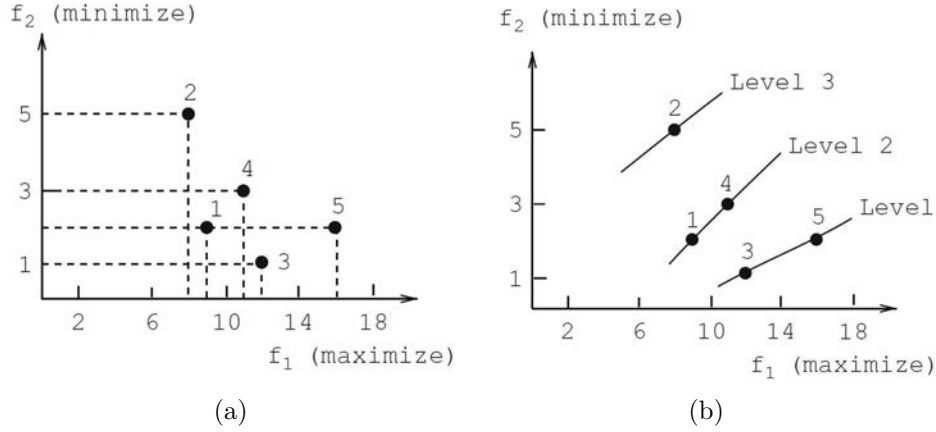


Figure 10: Criterion space of a two-objective optimization problem: (a) set of five solutions; (b) corresponding non-dominated fronts, or levels.

other solutions which do not belong to this set. In simple terms, this means that the solutions of this set are better compared to the rest of the solutions. This set is given a special name. It is called the non-dominated set for the given set of solutions. In the example problem, solutions 3 and 5 constitute the non-dominated set of the given set of five solutions. Thus, we define a set of non-dominated solutions as follows: among a set of solutions, the non-dominated set of solutions are those that are not dominated by any member of the set. When the set is the entire search space, the resulting non-dominated set is called the Pareto-optimal set. Like global and local optimal solutions in the case of single-objective optimization, there could be global and local Pareto-optimal sets in multi-objective optimization [21].

Finding the non-dominated set of solutions from a given set of solutions is similar in principle to finding the minimum of a set of real numbers. In the latter case, when two numbers are compared to identify the smaller number, a ' $<$ ' relation operation is used. In the case of finding the non-dominated set, the dominance relation can be used to identify the better of two given solutions. Here, we discuss one simple procedure for finding the non-dominated set (we call here the best non-dominated front). Many MOO problems, apart from requiring to find the best non-dominated solutions of a population, may require to sort a population according to different non-domination levels. When the above concept is applied to the five solutions of Figure 10(a), we obtain three non-dominated fronts as shown in Figure 10(b). From the dominance relations, the solutions 3 and 5 are the best, followed by solutions 1 and 4. Finally, solution 2 belongs to the worst non-dominated front. Thus, the ordering of solutions in terms of their non-domination level is as follows: ((3,5), (1,4), (2)) [23] [21].

## 2.3 Design of Experiments

Design of experiments (DOE) is a methodology that can be considered a branch of applied statistics, which deals with planning, conducting, analysing, and interpreting

controlled tests to evaluate the factors that control the value of a real function. DOE is a powerful data collection and analysis tool that can be used in a variety of experimental situations. DOE have been widely used to understand systems and improve designs. For many designed experiments, the available region is a known hyperspace (or hyperdomain) [24].

### 2.3.1 Sampling

Sampling is used worldwide in computer modeling related to performing safety assessments, manufacturing equipment and designing; in the latter case, it became particularly successful combined with optimization schemes [25].

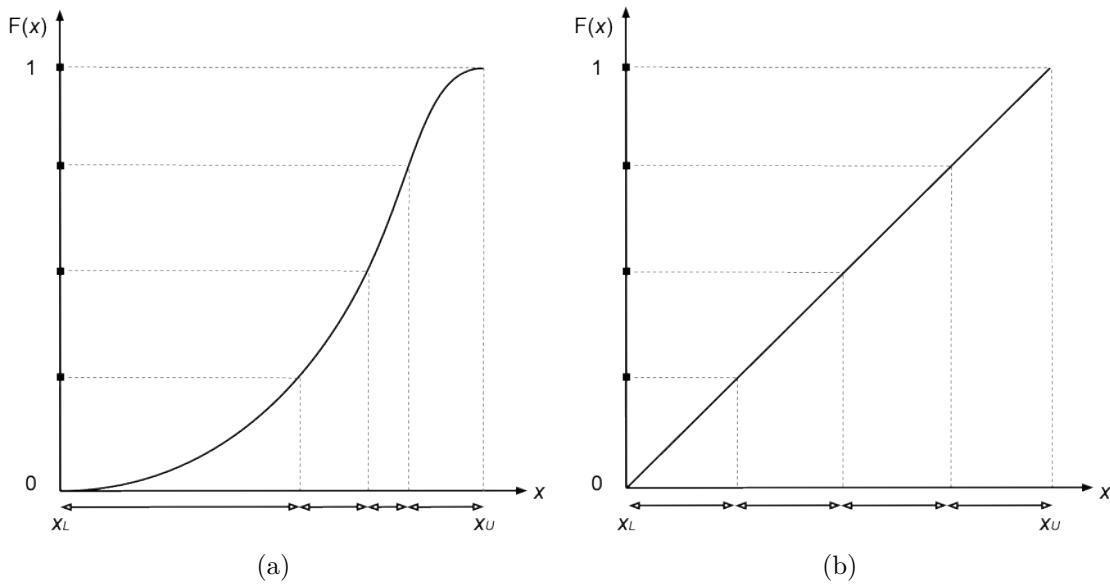


Figure 11: Cumulative distribution functions of a variable: (a) general type; (b) continuous uniform type.

### Latin Hypercube Sampling

The latin hypercube sampling (LHS) is a statistical method for generating a near-random sample of parameter values from a multidimensional distribution. It uses a stratified sampling scheme to improve the coverage of the input space. The stratification is accomplished by dividing the vertical axis on the graph of the distribution function  $F(x)$  of a random variable  $x_j$  into  $N$  non-overlapping intervals of equal length, where  $N$  is the number of computer runs to be made (number of samples). The distribution function describing an experiment where there is an arbitrary outcome that lies between certain boundaries which are defined by a maximum  $x_{jU}$  and a minimum  $x_{jL}$  value for each variable is of continuous uniform type. Through  $F^{-1}(x)$ , these  $N$  intervals divide the horizontal axis into  $N$  equiprobable, but not necessarily equal-length, intervals. Thus, the x-axis has been stratified into

Sample	Variable 1	Variable 2	Variable 3
1	0.1805	0.0020	0.9883
2	0.2774	0.3561	0.8703
3	0.6829	0.4889	0.4681
4	0.8772	0.9742	0.6576
5	0.0564	0.7591	0.5492
6	0.9689	0.6405	0.0914
7	0.7077	0.2738	0.2738
8	0.5570	0.8397	0.3199
9	0.3815	0.5289	0.1971
10	0.4095	0.1778	0.7071

Table 1: Values of 10 LHS samples of a three-variable domain with  $[0 \ 1] \ [0 \ 1] \ [0 \ 1]$  boundaries [26].

$N$  equiprobable and non-overlapping intervals, as represented in the two different cases of Figure 11. The next step in the LHS requires the random selection of a value within each of the  $N$  intervals on the vertical axis. When these values are mapped through  $F^{-1}(x)$ , exactly one value will be selected from each of the intervals previously defined on the horizontal axis [25].

The content of Table 1 represents an example of sample matrix  $\mathbf{S}$  resulting from the MATLAB function "lhsdesign"; it is a  $N \times n$  matrix whose columns contain the LHS for each variable  $x_j$  that are randomly chosen within the intervals. After that, a random process is required to ensure a random ordering of the values within each

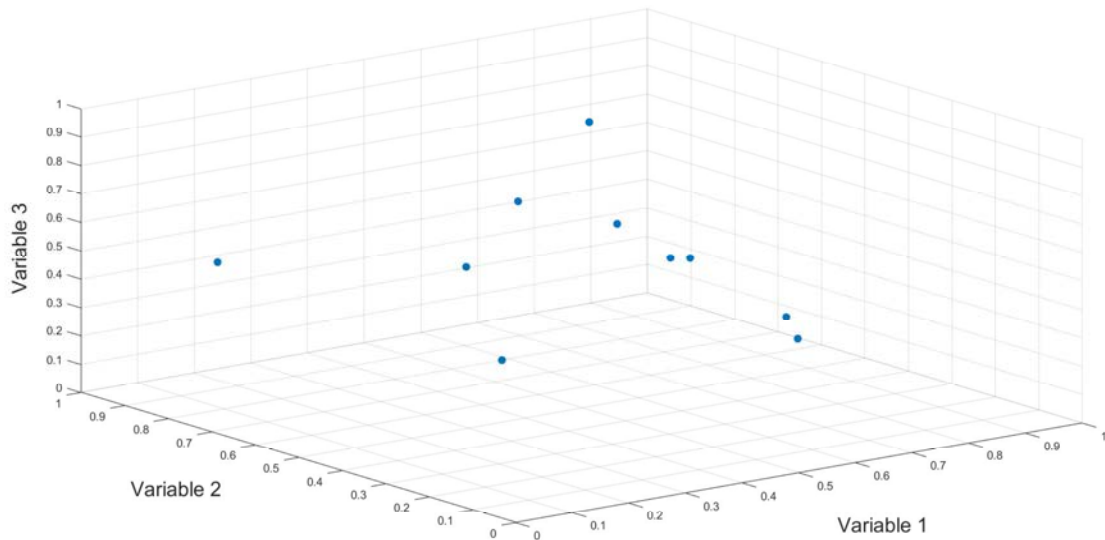


Figure 12: Three variable domain sampled with LHS method, according to Table 1 [26].

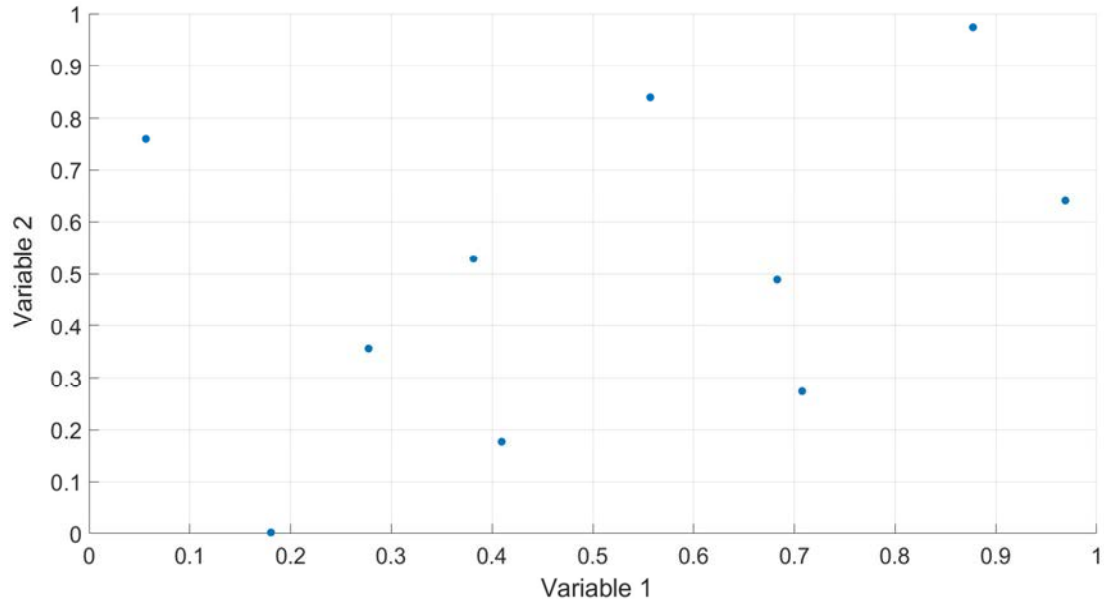


Figure 13: 2D section of the LHS distribution of variables 1 and 2 according to Table 1 [26].

column of the matrix. Figure 12 represent the sample distribution of Table 1 in a 3D space [26], while Figure 13 helps to figure out how the variables are homogeneously sampled.

This structure ensures that the entire range of every input variable dominating the model response, is completely covered. This means that a single sample will provide useful information when some input variables dominate certain responses while other input variables dominate other responses. By sampling over the entire range, each variable has the opportunity to show up as important. If an input variable is not important, then the method of sampling would be of little or no concern [25].

## Random Sampling and LHS

Simple random sampling enjoys widespread use in simulation applications, so it is important to compare it with LHS. One way to do this is to compare the variability of estimates obtained from the two procedures. Let  $x_1$  and  $x_2$  be two independent input variables for a computer model. Using order statistics, the expected probabilistic coverage of the joint input space  $x_1$  and  $x_2$  under random sampling for a sample of size  $N$  is given as

$$\left(\frac{N-1}{N+1}\right)^2. \quad (15)$$

On the other hand, LHS requires that one value be selected from each of the extreme intervals. Using the expected values from these intervals gives the expected probabilistic coverage of the joint input space for two variables under LHS for a

sample of size  $N$  as

$$\left(\frac{N-1}{N}\right)^2. \quad (16)$$

For  $n \geq 2$  the expression in Equation 16 is always greater than the expression in 15, so LHS can be expected to provide better probabilistic coverage of the input space than a simple random sample [25].

### 2.3.2 Surrogate Model

A surrogate is a function that approximates an objective function. Most engineering design problems require experiments or simulations to evaluate objectives and constraints of the design, which are both functions of the design variables. However, a single simulation or experiment can take hours, or even days to complete; in this way, a surrogate becomes useful because it takes little time to evaluate. For instance, an algorithm that looks for a point that minimizes a fitness function can be simply based on the evaluation of the surrogate on thousands of points. Thereby, the outgoing best value is an approximation of the real objective function and, for this reason, one must be careful that the achieved solution is enough accurate [27].

The most popular and easy modeling approach in order to construct the surrogate is polynomial response surface (PRS). The PRS uses a linear polynomial to approximate the objective and the coefficients are determined through an experimental evaluation. The general equation of a surrogate quadratic model with the PRS method is

$$f_s(\mathbf{X}) = \beta_0 + \sum_{i=1}^n \beta_i x_i + \sum_{i=1}^n \sum_{j=1}^n \beta_{ij} x_i x_j, \quad (17)$$

where  $x_i$  is the  $i$ -th component of the  $n$ -dimensional position vector and  $\beta_0$ ,  $\beta_i$  and  $\beta_{ij}$  are the parameters that have to be estimated which can be arranged in an appropriate order to form the column vector  $\boldsymbol{\beta}$  [28].

## Response Surface Methodology

One of the most commonly used experimental designs for constructing surrogates is the response surface methodology (RSM). The method was introduced by George E. P. Box and K. B. Wilson in 1951. The main idea of RSM is to use a sequence of designed experiments to obtain an optimal response of the surrogate function. More precisely, the RSM is a statistic technique to make an approximate response surface model with results obtained through experiment or simulation to express the experimental relation of design variables with the objective as a mathematical equation, namely the surrogate function [5].

Firstly, we have to choose the range of boundaries for the input variables, or factors, and it is wise not only to try extreme values. First of all, because in some cases extreme values result in runs that are not feasible; moreover, extreme ranges might move one out of a smooth area of the response surface into some jagged region, leading to low accuracy. Secondly, the number of levels of the experimental design must be defined. Experiments are run at different variable values, called levels. Each

run of an experiment involves a combination of the levels of the investigated factors. For instance, the standard layout for a 2-level design uses  $+1$  and  $-1$  notation to denote the "high level" and the "low level" respectively, for each variable as shown in Table 2. However, a two-level factorial design cannot estimate quadratic effects between two points as a three-level does, thus it easily becomes too rough for practical applications [29].

$x_1$	$x_2$
-1	-1
+1	-1
-1	+1
+1	+1

Table 2: Two-level full factorial design of a two-variables problem [29].

Quadratic response surfaces models, as described by Equation 17, can be calibrated using full factorial designs with three or more levels for each factor, but these designs generally require more runs than necessary to accurately estimate model parameters [30]. Table 3 shows how a three-level full factorial design quickly becomes unacceptable, in terms of runs, as the number of variables investigated increases. The last column in Table 3 shows the number of terms present in a quadratic model for each case, excluding the term  $\beta_0$ . This problem was the motivation for creating 'fractional factorial' designs [29].

Number of variables	Run combinations	Number of coefficients for quadratic model
2	9	6
3	27	10
4	81	15
5	243	21
6	729	28

Table 3: Three-level full factorial design [29].

The next two paragraphs discuss designs for calibrating quadratic models that are much more efficient, using three or five levels for each factor, but not using all combinations of levels, namely a 'fractional factorial' design [30].

The Box-Behnken design (BBD) is an independent quadratic three-level design, in which the run combinations are at the midpoints of edges of the process space and at the center. With reference to Figure 14, in a three-variable space, the geometry of this design suggests a sphere within the process space such that the surface of the sphere protrudes through each face with the surface of the sphere tangential to the midpoint of each edge of the space [29].

A Box-Wilson central composite design, commonly called central composite design (CCD), contains an embedded factorial or fractional factorial design with center

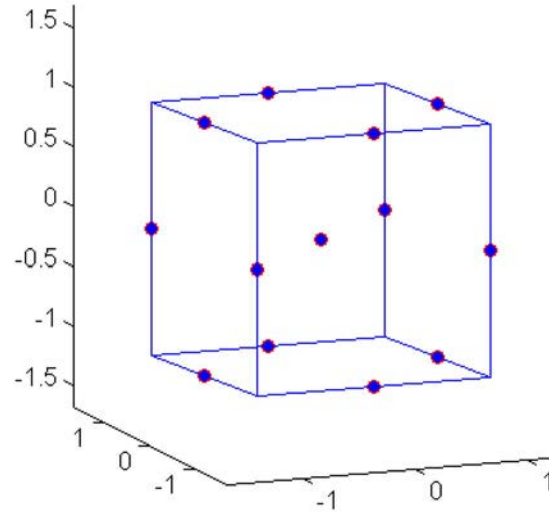


Figure 14: Run combinations of a three-variable BBD [31].

points that is augmented with a group of 'star points' that allow estimation of curvature. If the distance from the center of the design space to a factorial point is  $\pm 1$  unit for each factor, the distance from the center of the design space to a star point is  $|\alpha| > 1$ . The precise value of  $\alpha$  depends on certain properties desired for the design and on the number of factors involved. Depending on where the star points are placed, there are three varieties of central composite designs: circumscribed, inscribed and face centered (or face centered). Only the latter, the face centered design, is taken into consideration. In it, the star points are at the center of each face of the factorial space, so  $\alpha = \pm 1$ . This variety requires three levels of each factor.

Table 4 contrasts the structures of the methodologies above discussed. In both cases, the center points are run more times to allow for a more uniform estimate of

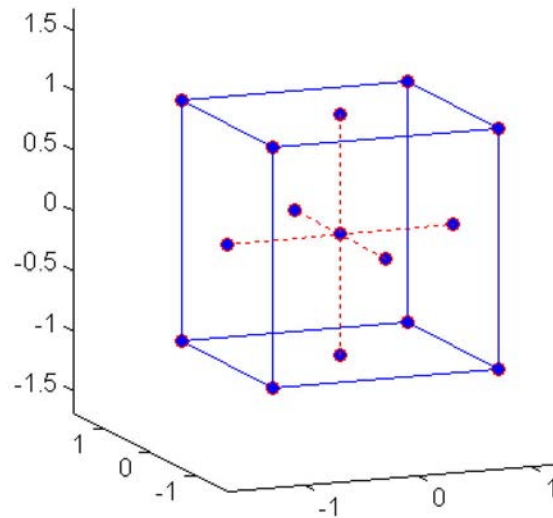


Figure 15: Run combinations of a three-variable CCD faced [31].

the prediction variance over the entire design space.

BBD			CCD		
$x_1$	$x_2$	$x_3$	$x_1$	$x_2$	$x_3$
-1	-1	0	-1	-1	-1
-1	+1	0	-1	-1	+1
+1	-1	0	-1	+1	-1
+1	+1	0	-1	+1	+1
-1	0	-1	+1	-1	-1
-1	0	+1	+1	-1	+1
+1	0	-1	+1	+1	-1
+1	0	+1	+1	+1	+1
0	-1	-1	-1	0	0
0	-1	+1	+1	0	0
0	+1	-1	0	-1	0
0	+1	+1	0	+1	0
0	0	0	0	0	-1
0	0	0	0	0	+1
0	0	0	0	0	0
			0	0	0
			0	0	0
			0	0	0
			0	0	0
			0	0	0

Table 4: Structures of three factors BBD (15 runs) and CCD (20 runs) [31] [32].

### 2.3.3 Clustering

Clustering is the task of grouping a set of items such that items in the same group (cluster) are more similar to each other than to those in other groups. It is commonly used for analysing data instances which do not include any class attribute, hence the need to group them together using the concept of "maximizing the intraclass similarity and minimizing the interclass similarity". Thus, this translates to the clustering algorithm which identifies and groups instances which have a higher measure of similarity, as opposed it separates instances which are much less-similar to one another. The analysis itself, in fact, involves applying one or more clustering algorithms with the goal of finding hidden patterns [33]. If such homogeneous groups are found, these may be named and their attributes be defined; one can choose either the group mean as the representative prototype of instances in the group or the possible range of attributes [34].

The easiest form of clustering is partitional clustering, which goal is to partition a given data set in such a way that a specific criteria are optimized. At this point, there are many clustering criteria that can be performed, but the most widely used



one is the clustering error criterion which computes the squared distance between each point and the corresponding cluster center, taking into account the sum of these distances for all the points. The most well-known method for minimizing the clustering error is the k-means algorithm, which is discussed in the next paragraph [35].

### The k-means Algorithm

The k-means algorithm finds locally optimal solutions with respect to the clustering error. It is a fast iterative algorithm that has been used in many clustering applications. It is a point-based clustering method that starts with the cluster centers (centroids) initially placed at arbitrary positions and proceeds by moving at each step the cluster centers in order to minimize the clustering error [35].

The procedure follows a simple and easy way to classify a given data set (generally called particles or points) through a certain number of clusters (assume  $k$  clusters) fixed a priori. The main idea is to define  $k$  centroids, one for each cluster. These centroids should be placed in a smart way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early grouping is done. At this point we need to re-calculate  $k$  new centroids as barycenters of the clusters resulting from the previous step. After we have these  $k$  new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated and two corresponding iterations are shown in Figure 16. As a result of the entire loop we may notice that the  $k$  centroids change their location step by step until no more changes are done. In other words, convergence is achieved when centroids do not move any more. Finally, we can say that this algorithm aims at minimizing an objective function, in this case a squared error function. The objective function  $J$ , also known as clustering error, is

$$J = \sum_{i=1}^k \sum_{j=1}^n ||x_i^{(j)} - c^j||^2, \quad (18)$$

where  $||x_i^{(j)} - c^j||^2$  is the squared Euclidean distance metric between a data point  $x_i$  and the cluster centre  $c_j$ ,  $n$  is the number of data points in the  $j^{th}$  cluster. Minimizing  $J$  simply means to assign each point to the cluster with the nearest centroid [36].

Thereby, the algorithm is composed of the following steps [36]:

1. Place  $k$  points into the space represented by the particles that are being clustered. These points represent initial group centroids;
2. Assign each particle to the group that has the closest centroid;
3. When all objects have been assigned, recalculate the positions of the  $k$  centroids;
4. Repeat steps 2 and 3 until the centroids no longer move. This produces a separation of the particles into groups from which the metric to be minimized can be calculated.

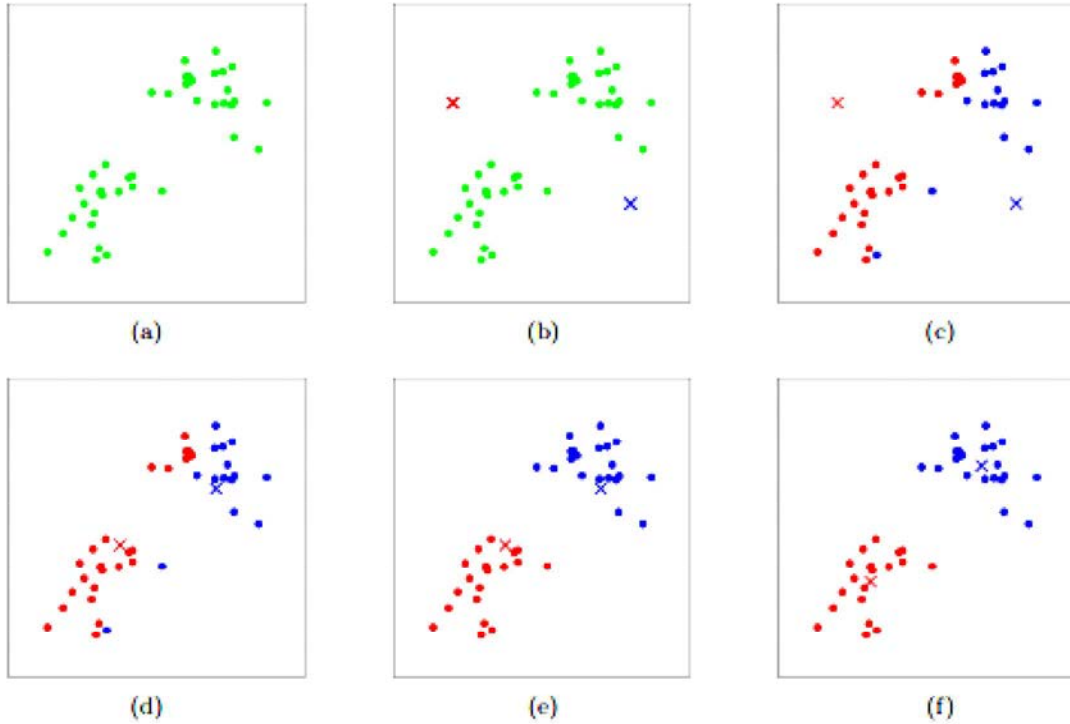


Figure 16: K-means algorithm. The data set of particles are shown as dots, and cluster centroids are shown as crosses: (a) Original dataset; (b) random initial cluster centroids; (c-f) illustration of running two iterations of k-means. In each iteration, we assign each particle to the closest cluster centroid (shown by "painting" the training examples the same color as the cluster centroid to which is assigned); then we move each cluster centroid to the mean of the points assigned to it [37].

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. In fact, the k-means algorithm is a local search procedure and it is well known that it suffers from the serious drawback that its performance heavily depends on the initial starting conditions. In other words, the main disadvantage of the method lies in its sensitivity to the initial randomly selected cluster centres. Therefore, in order to obtain near optimal solutions using the k-means algorithm several runs must be scheduled differing in the initial positions of the centroids [35] [36].

Choosing the number of cluster represents another fundamental aspect; given any  $k$ , the algorithm will always find  $k$  centers whether they are fixed or not. There are various way in order to establish the  $k$  number [34]:

- In some applications  $k$  is defined;
- The maximum allowed distance (namely the maximum clustering error per cluster or total) may also be set;
- Plotting data may be helpful to visualize the structure of data;

- Validation of the groups can be done manually by checking whether clusters actually code meaningful groups of the data or not.



### 3 Research Methodology

In this chapter, the entire design procedure is described step by step. Before introducing the content of each section, the logical thread is introduced according to the five stages diagram in Figure 17.

According to the design theory outlined in Section 3.1, starting from certain established characteristics and parameters, we can perform a theoretical design procedure and obtain a first geometry of the machine, as represented in the starting block.

Although the analytical modelling and design of SPMSM machines have been formalized by many authors in the last decades, such models are typically used during the preliminary design stage. They are obviously based on hypothesis and approximations in order to simplify their modelling and restrict the design space dimension. Consequently, the design procedure of such machines was conventionally divided in two steps: first an analytical model is used to carry out a preliminary design and then a Finite Element Analysis (FEA) is employed to evaluate the design aspects disregarded in the first stage. The number and the influence of such neglected aspects depend on the accuracy and on the initial hypothesis of the employed analytical model. But in recent works, the analytical design procedures assisted by FEA coupled with optimization algorithms have been presented as an alternative of the two steps design procedure. Those design strategy do not make use of any hypothesis and therefore have the advantages of considering all the design aspects during the performance evaluations [4].

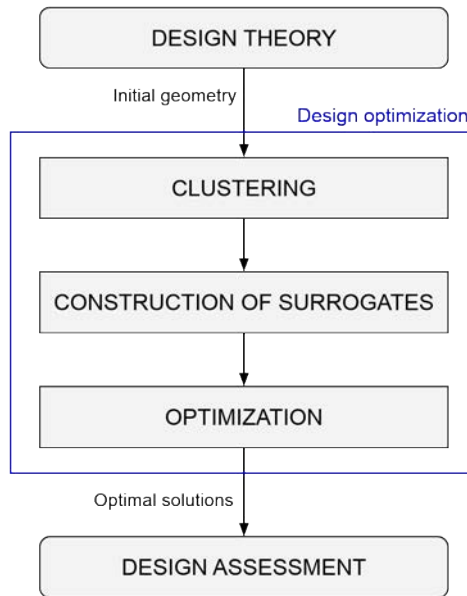


Figure 17: Flowchart of the entire full design procedure.

Therefore, the first approach might be to couple the optimization algorithm directly with the FEA but the weakness of such methodology lies on the computational burden required to carry out the optimization. The latter depends on two factors:

the computational time required to evaluate the performance of a single machine candidate and the number of functional evaluations needed by the optimization algorithm to converge to the optimal solutions. Several methodologies and techniques have been presented in literature in order to decrease the computational burden of the whole optimization, the key issue of design optimization [4].

In this work, the design of a 160 kW SPMSM is faced adopting an optimization methodology based on surrogate functions. In particular, the optimization design is conducted using the Box-Behnken design, which is a response surface method [5].

Moreover, this work proposes a clustered surrogate-assisted optimization in which several small clusters (or subdomains) are first created by the clustering algorithm and then local surrogate functions are built for each cluster. Finally, in each cluster, the local surrogates assists the optimization algorithm searching for optima. The clustering technique leads to certain important design benefits that will be further discussed; despite that, it will also cause an increase of the computational burden [38].

To wrap it all up, the analytical design procedures is assisted by a FEA which is coupled with a multi-objective optimization algorithm conducted using the surrogates built for every clusters. If we look at the flowchart, this proposed optimization design methodology, which comes after the preliminary design, is represented by the three rectangular action blocks inside the blue rectangle.

The set of procedures which aims to divide the initial domain into several small clusters, is investigated in Section 3.2. Section 3.3 describes the construction of the surrogate functions. After that, the implemented optimization algorithm is discussed in Section 3.4. Finally, optimal points coming from the design optimization are evaluated by means of the FEA and in Section 3.5, as meant by the ending block of Figure 17.

As previously anticipated, this work suggests an outset investigation of the entire starting domain in order to find the most suitable regions that can fulfill some specific project requirements and, as a consequence, lead the design to better results.

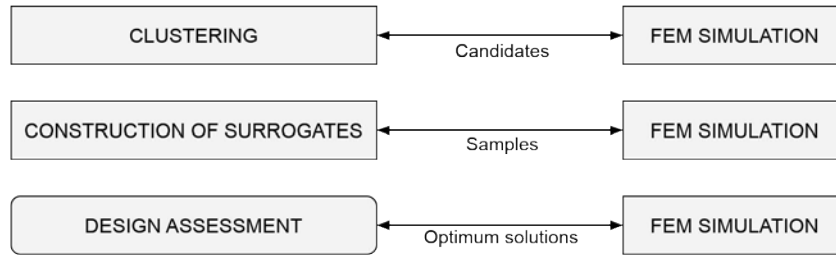


Figure 18: Overview on the design stages that need a FEM simulation.

As shown in Figure 18, three stages of the whole design process need finite element method (FEM) simulations; thus, some general information about the employed 2D FEM software and its routines are given in Section 3.6. You must always keep in mind that almost the whole computational time of the entire design stays in the finite element simulations.

In order to proceed with the FEM simulation, the software needs to receive as input a set of parameters (mostly concerning electrical and geometrical characteristics of the machine). These input data (or input parameters) are computed starting from a set of defined variables. Section 3.3.7, introduces these optimization variables and analyse the contents of the analytical routine implemented to calculate of parameters asked by the software.

From now on, as far as we are concerned, it is essential to distinguish between:

- Variables: They are the starting point of the analytical assessments; we may call them optimization variables, or even input variables. These are the variables employed in the optimization routine, thereby, surrogates need to be need to be built upon them.
- Parameters: They are the results assessed by the analytical calculations and, in this work, it is worthwhile to refer to them as input parameters or input data.

### 3.1 Design Theory of Surface Permanent Magnet Synchronous Machine

The step by step procedure outlined in this section is designed in particular for the analytical evaluation of a three-phase synchronous motor with rotor surface permanent magnets [14].

#### 3.1.1 Initial Data and Choice of Materials

The type of machine and construction are primarily defined (synchronous motor with rotor surface permanent magnets in our case), after that, the design of the machine begins considering some fundamental characteristics, which are [7]:

- Rated power.
- Rated speed.
- Rated frequency.
- Number of pole pairs.
- Rated voltage of the machine.

Different kind of circumstances impose the choice of all the mentioned features which are clarified in the outset and that will affect the further calculations without the possibility of being changed. The number of phases  $m$  and the rated line-to-line voltage  $U$ , for instance, usually depend on the features of the system in which the machine has to be installed. On the other hand, the rated power  $P$  together with the rated speed  $n$  represent two design specifications governed by the purpose that the machine must achieve. Rather, number of polar pairs  $p$  and frequency  $f$  deserve a separate discussion: in fact, since it would be necessary to adhere to the speed specification, choosing an adequate number of polar pairs could be not sufficient to

achieve a working frequency congruent to the one of the power supply network. They are directly correlated as stated in Equation 1 and for this reason it could become necessary to install a frequency converter to couple the machine with the grid. Even though a good frequency converter is fundamental to achieve the best sinusoidal shape of the voltage supply and to avoid harmonics injection in the net, it will not be discussed since it is irrelevant for the final purpose of this work [7].

The choice of the materials, which could be considered the first real trade-off between performances and cost, is governed by circumstances of every kind. Since the procedure later requires a FEM simulation, it is necessary that the materials selected for these calculations can be implemented within the software. Regarding this work, the two materials which need particular attention are the permanent magnet and the ferromagnetic material of the core.

According to what has just been stated, the chose permanent magnet is the Neorem model 495a, a sintered NdFeB magnet developed by the same name company [39]; the demagnetization curves for different temperatures are illustrated in Figure 19.

The laminated material for the core regions comes from the Cogent company and it is a fully processed electrical steel with a sheets thickness of 0.50 mm, called M700-50A [40]. The characteristic of the peak magnetic field strength with the peak magnitude polarization at 50 Hz is shown below in Figure 20.

The designer must make sure that they can work under certain conditions with particular attention to temperature, environmental conditions and mechanical stress. Undoubtedly, the study of the thermal regime represents one the most difficult but fundamental challenge of the design, in order to have temperatures as accurate as

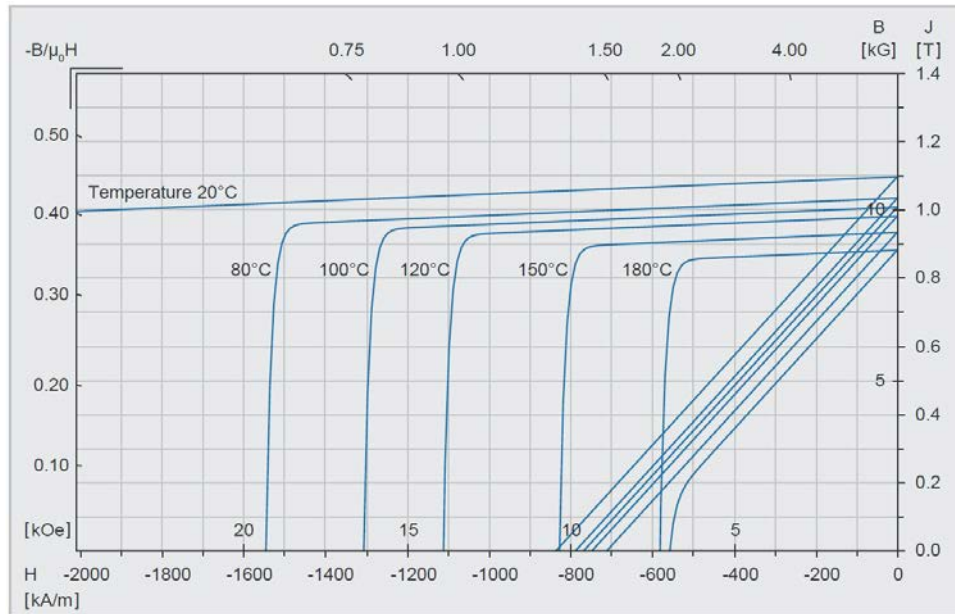


Figure 19: Typical demagnetization curves  $B(H)$  and  $J(H)$  at different temperatures [39].



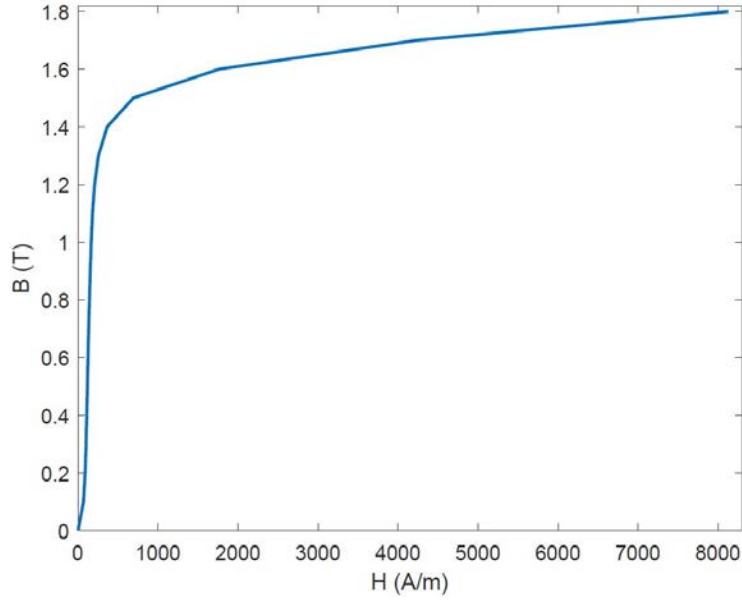


Figure 20: B-H curve of M700-50A at 50 Hz.

possible to make sure that materials work under the right conditions and also to develop a reliable design. Nevertheless, since these concerns do not fall within the goals of the research, the temperature of the rotor is fixed at 100 °C, whereas the stator, including the windings, it's supposed to work at the temperature of 90 °C.

### 3.1.2 Main Dimensions, Air Gap and Stator Windings

The term “main dimensions” refers to the equivalent core length  $l'$  and the air gap diameter (or inner stator diameter)  $D_s$ . Investigating the field diagram of the ends of the machine, like the one proposed in Figure 21, it is possible to figure out the influence of proximity to the edge in the flux density. This latter decreases gradually to zero along the axial coordinate but, in the first part, it remains approximately constant and it participates in the torque production; this is what the effective length of the machine takes into consideration. Moreover, since in the case of the SPMSM it's important to get a sufficient magnetic flux, it is recommended that the permanent magnet axial length  $l_{rPM}$  must be greater than the stator core one  $l$ . In order to achieve a good flux density even towards the end of the air gap  $\delta$ , it has been proved that it must be at least

$$l_{rPM} = l + 2\delta. \quad (19)$$

As far as the above considerations concern us, we're allowed to consider the equivalent core length of the machine equal to the effective axial length of the permanent magnets.

The tangential stress in the air gap  $\sigma_{Ftan}$  acting upon the rotor surface is the responsible of the torque  $T$  generation. It depends on the local linear current density  $\mathbf{A}$ , on the local flux density  $\mathbf{B}$  and on the phase shift between the two fundamentals of them. The tangential stress, which value has been selected according to Table 6.4

[7], gives the starting point for the design and it defines the size of the rotor, thus the rotor volume  $V_r$  in accordance with the desired torque  $T$

as evidenced by

By creating a system between

which is an empirical relationship for standard synchronous machines with more than one pole pair, and

it is possible to determine the equivalent core length and the outer diameter of the rotor.

The air gap width comes from practical considerations which, over the years, have been brought together to provide some empirical equations depending on the power and on the characteristics of the machine that, in our case, led to

After this first evaluation, for heavy duty machines, the air gap is usually increased about 60%. Since in this machine no cooling channel is planned, it is immediate to obtain the inner diameter of the stator with

$$D_s = D_r + 2\delta \quad (25)$$

and the effective length of the stator and rotor core with

$$l = l' - 2\delta. \quad (26)$$

Following, all parameters concerning the stator winding are determined. As a first remark, a two-layer integral slot it's chosen so that it is possible to take a winding pitch  $W$  multiple of the single stator slot pitch  $\tau_u$ , but shorter than the stator pole pitch  $\tau_p$ , as it is schematically depicted by Figure 22. This practical technique helps in order to reduce harmonic content of the air gap flux density, producing a more sinusoidal distribution of the current linkage.

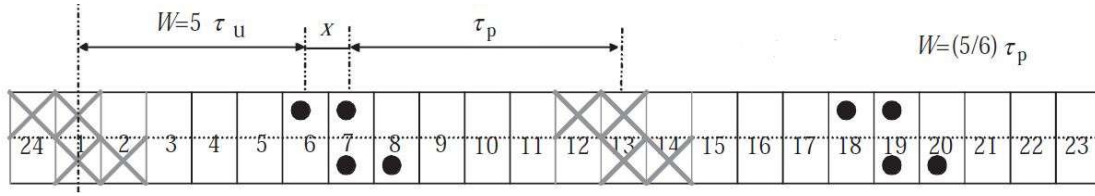


Figure 22: Winding step shortening [8].

Once the number of slots per pole and phase  $q$  is chosen, the total number of stator slots  $Q$  can be easily determined by

$$Q = 2pmq. \quad (27)$$

To get a first estimation of the number of coils turns in a phase winding  $N$  it's necessary to estimate the maximum value of the rectangular flux density  $B_{max}$ . Table 6.2 [7] provides some typical peak values of the fundamental air gap flux density  $B_{1peak}$ , which differs from the amplitude of the rectangular flux density given by the PMs. The latter is calculated in relationship to the Fourier series of a rectangular wave and basically it depends on the relative magnet width  $\alpha_{PM}$ ; as this last one increases the amplitude of the rectangular wave decreases in order to obtain the same fundamental value (first harmonic), in agreement with Figure 23 and the following equation:

$$B_{max} = \frac{\pi B_{1peak}}{4 \sin\left(\frac{\alpha_{PM}\pi}{2}\right)}. \quad (28)$$

The winding factor of the fundamental  $k_{w1}$  and, consequently, the number of coil turns in series in a phase winding are affected by the short pitching. It is reasonable to deduce that as the winding pitch decrease the winding factor decrease as well, which can be more precisely pointed out by the following formula:

$$k_{w1} = \frac{2 \sin\left(\frac{\pi}{2}W\right) \cdot \sin\left(\frac{\pi}{2m}\right)}{\frac{Q}{mp} \sin\left(\pi \frac{P}{Q}\right)}. \quad (29)$$

Reducing the winding pitch, also affects the number of coil turns, as evidenced by

$$N = \frac{\sqrt{2}E_{PM}}{\omega k_{w1} \alpha_{PM} B_{max} \tau_P l'}, \quad (30)$$

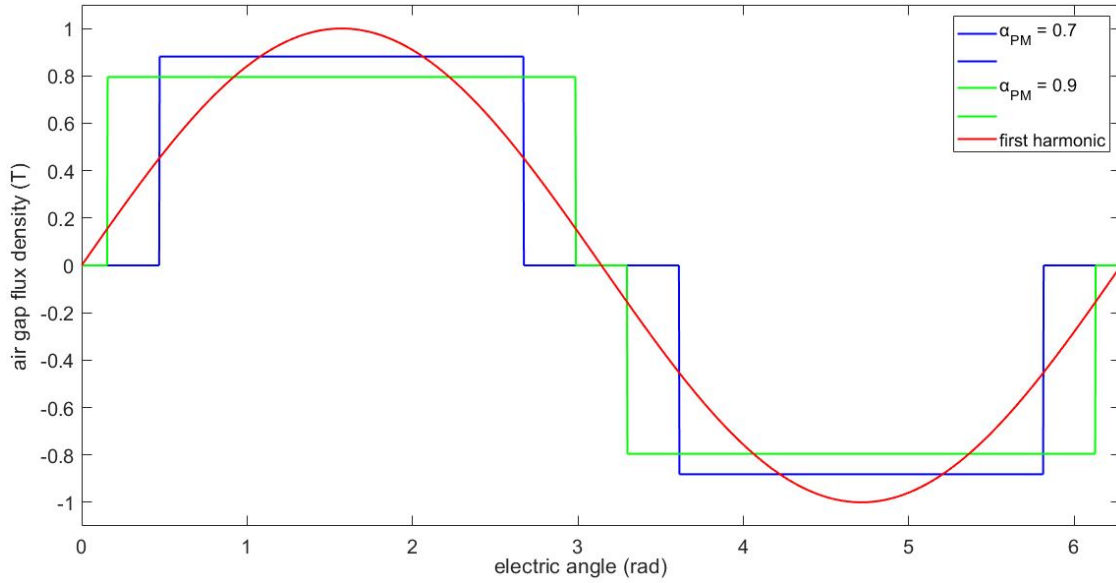


Figure 23: Different values of  $\alpha_{PM}$  require different amplitudes of rectangular flux densities to get an equal first harmonic.

where the induced voltage  $E_{PM}$  is assumed to be

$$E_{PM} = \frac{U}{\sqrt{3}}. \quad (31)$$

The last parameter about the stator winding to be figure out is the number of conductors in a slot  $z_Q$ , that has to be an even number in a two-layer winding; choosing a suitable number of parallel paths  $a$  can help, as it can be appreciated by

$$z_Q = \frac{2amN}{Q}. \quad (32)$$

The number that results must be rounded to the nearest even number and then used to calculate the new value of coil turns, which will consequently result integer as well.

### 3.1.3 Tooth Width and Stator Slot Dimensions

To keep the induced voltage  $E_{PM}$  unchanged, the  $B_{max}$  needs to be changed in accordance with the new number of coils turns in series, as can be established from Equation 31, by using

$$B_{max \text{ new}} = \frac{z_{Q \text{ non rounded}}}{z_{Q \text{ rounded}}} B_{max \text{ old}}. \quad (33)$$

The apparent flux penetrating the slot pitch  $\Phi_{sd}$  is

$$\Phi_{sd} = B_{max} l' \tau_u \quad (34)$$

and once the apparent tooth flux density  $B_d$  is given, with reference to Table 6.2 [7], it is possible to determine the tooth width  $b_d$  considering that between two sheets of

iron there's a little insulation layer, described by the space factor  $k_{Fe}$ . The equation is given by

$$b_d = \frac{\Phi_{sd}}{k_{Fe}lB_d} \quad (35)$$

and Figure 24 shows a cross-section of the stator tooth useful to appreciate the previous formulas. The real tooth width should be incremented by 0.1 mm in order to take into consideration that the punching process influences the crystal structure of the iron, decreasing the permeability near the cut.

For defining the stator slot dimensions, an estimation of the stator current  $I_s$  is needed

$$I_s = \frac{P}{m \frac{U}{\sqrt{3}} \eta \cos \phi}; \quad (36)$$

nevertheless, it can be quite coarse since both the values of the efficiency  $\eta$  and power factor  $\cos \phi$  are defined in the initial design stage. For calculating the area of one single conductor  $S_{cs}$  the following is used

$$S_{cs} = \frac{I_s}{aJ_s}, \quad (37)$$

in which Table 6.3 [7] can be employed to set the stator current density  $J_s$ . The value of the wound area of the stator slot  $S_{Cus}$  is now given by

$$S_{Cus} = \frac{z_Q S_{cs}}{k_{Cus}}, \quad (38)$$

where  $k_{Cus}$  is the copper space factor of the slot due to insulation between conductors and between conductors and iron.

The choice of the stator slot shape starts with having some fixed dimensions concerning precisely the insulation layer that separates the conductors from the iron. Afterwards, all the others dimensions are obtained according to geometrical formulas with some important cautions, that will be discussed in detail in the Subsection 3.7. The designer must manage to find a shape in accordance with number of conductors, number of layer winding, insulation layers and so on, especially taking into account the wound area previously calculated.

### 3.1.4 Height of Yokes and Permanent Magnets

The flux penetrating the air gap and the teeth section  $\Phi_m$  is

$$\Phi_m = \alpha_{PM} B_{max} l' \tau_P \quad (39)$$

and it is divided into two equal parts at the stator and rotor yokes. From the following two equations it is possible to deduce the height of the stator yoke  $h_{ys}$  and rotor yoke  $h_{yr}$  with

$$h_{ys} = h_r = \frac{\Phi_m}{2k_{Fe}lB_{yoke}}. \quad (40)$$

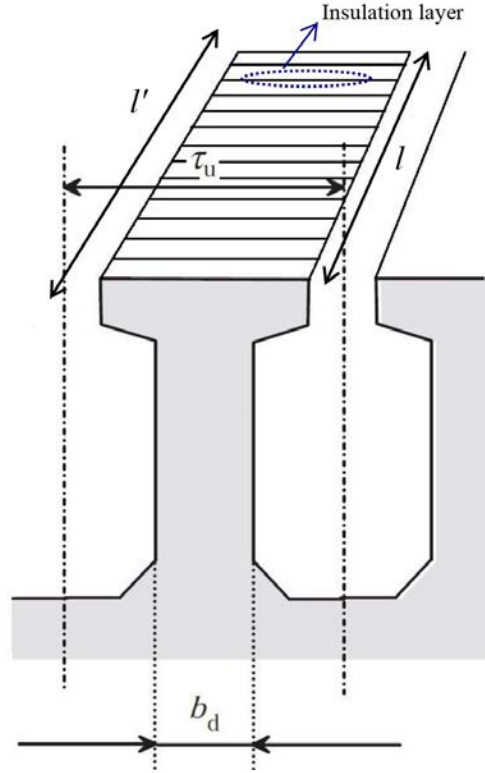


Figure 24: Cross-section of a stator tooth [8].

The maximum flux density of the yokes  $B_{yoke}$  can be selected in accordance with Table 6.2 [7]. When calculating the height of the permanent magnet, we must consider that the total magnetic voltage  $U_{mtot}$  must be equal to the current linkage of the permanent magnet  $H_c \cdot h_{PM}$ , as summarized by the following equation:

$$U_{mtot} = H_c \cdot h_{PM} = U_{m\delta} + U_{mts} + U_{mPM} + \frac{1}{2}U_{mys} + \frac{1}{2}U_{myr}. \quad (41)$$

Each component of the total magnetic voltage, illustrated in Figure 25, must be calculated separately by calculating the line integral of the magnetic field strength along the path. For instance, the magnetic potential difference  $U_{mys}$  over the whole stator yoke has to be determined by integrating between the two poles of the stator yoke integration path. Once all the heights are known, it is easy to determine the outer diameter of the stator  $D_{se}$  and the rotor core outer diameter  $D_{yri}$ .

All the geometrical dimensions of the machine have now been defined; next, electrical characteristics are investigated.

### 3.1.5 Resistance and Inductances Calculation

Windings are usually made of copper and the conductivity of commercial copper wire, at room temperature of 20 °C, is  $\sigma_{Cu} = 57 \cdot 10^6$  S/m and it varies with the rise temperature  $\Theta$  as follows

$$\sigma_{Cu} = \frac{\sigma_{Cu20^\circ C}}{1 + \Theta \alpha_{Cu}}. \quad (42)$$

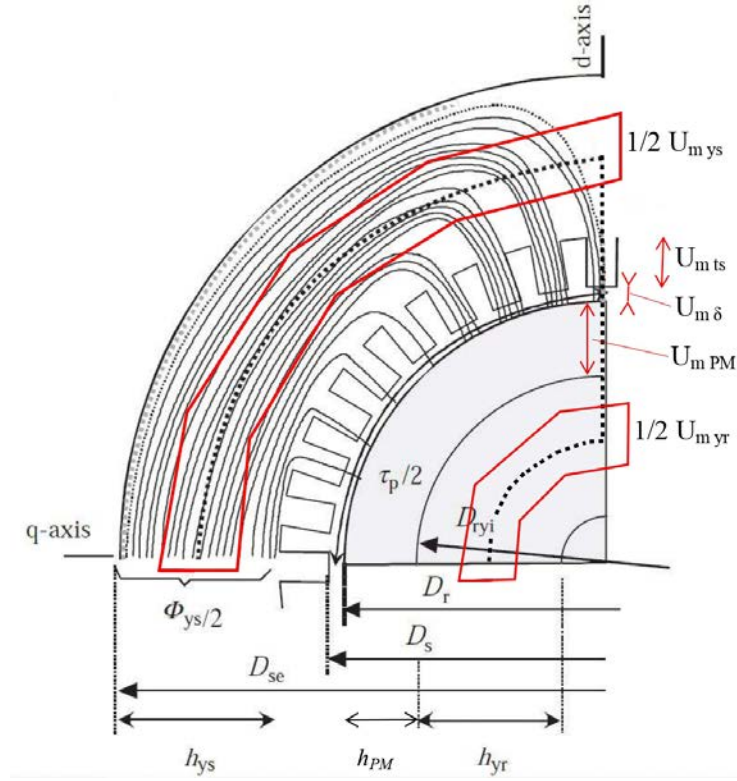


Figure 25: Ideal integration path line, indicated by the thick black dotted line, for the calculation of magnetic voltages in a two pole SPMSM [8].

The temperature coefficient of the copper resistivity is  $\alpha_{Cu} = 3.81 \cdot 10^{-3} \text{ K}^{-1}$ . An accurate definition of the winding length is a fairly difficult task but preliminary calculations can be made by applying empirical equations. The average length  $l_{av}$  of a coil turn of a slot winding of low-voltage machines is given approximately as

$$l_{av} = 2l + 2.4W + 0.1. \quad (43)$$

The DC resistance of a phase winding  $R$  can be finally estimated with

$$R = \frac{Nl_{av}}{\sigma_{Cu}aS_{cs}}, \quad (44)$$

which takes into account the effective section of the conductors together with the number of coils turns in series in a phase winding and the number of parallel paths.

Referring to the theoretical aspects underlined within the Subsection 2.1.3, in a multiphase winding the d-axis magnetizing inductance  $L_md$  of a three-phase machine can be calculated with the following:

$$L_{[md]} = \frac{m}{2} \cdot \frac{2}{\pi} \cdot \mu_0 \cdot l' \cdot \frac{1}{2p} \cdot \frac{4}{\pi} \cdot \frac{\tau_P}{\delta_{ef}} \cdot (k_{w1} \cdot N)^2, \quad (45)$$

where  $\delta_{ef}$  is the effective air gap, which considers also the reluctance of the iron. Due to the symmetrical structure of the machine in question, the q-axis magnetizing inductance  $L_q$  is equal to the d-axis one.

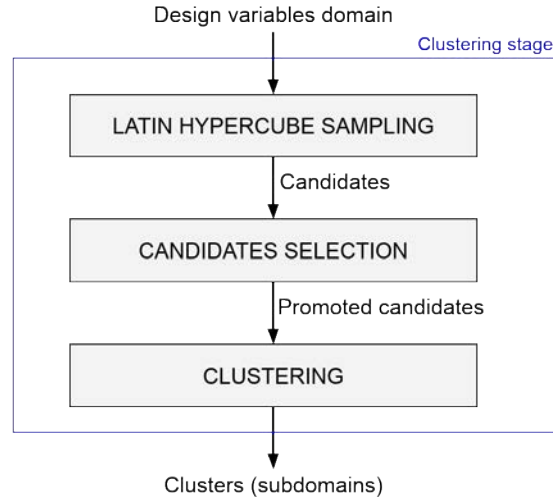


Figure 26: Clustering procedure illustrated step by step.

On the other hand, the leakage inductance of a machine can be calculated as the sum of the following partial different leakage inductances, as shown in Subsection 2.1.3:

- Air gap leakage inductance.
- Slot leakage inductance.
- Tooth tip leakage inductance.
- End-winding leakage inductance.

## 3.2 Clustering Process

Once the theoretical design is completed, all the geometrical aspects are defined so that the initial geometry of the machine can be characterised. Before involving the design optimization procedure, which starts with the clustering process here discussed, it becomes fundamental to choose the optimization variables, which will be fully examined in Section 3.7. For now, it is enough to know that, in order to proceed with this stage of the design, the optimization variables are required. In particular, the input information of the optimization process consist in the upper and lower boundaries of each variable; that means, an  $n$ -dimensional domain is inspected in the design, and final results can only stay within this one.

The clustering process is analysed with reference to the scheme outlined in Figure 26, which denotes the various steps to follow before getting the final clusters.

### 3.2.1 Latin Hypercube Sampling

The latin hypercube sampling ensures a homogeneous distribution of the candidates within the starting hyperdomain. What is executed in this step is schematically



shown in the two tables of Figure 27, which shows at first the inputs, namely the domain boundaries, and then the outputs of the LHS function, namely the candidates.

It can easily be deduced that the choice of the number of candidates  $c$  represents an important trade-off between carefulness and computational cost. In fact, a large number guarantees a denser sampling and consequently it is more likely to identify regions with better features; on the other hand, we should remember that for each candidate a FEM simulation is carried out. This makes the computational time of this step quite relevant; therefore, increasing too much the amount of candidates may lead to a depreciation of one of the most important matters in the proposed design methodology.

	variable 1	variable 2	variable ...	variable $n$
lower boundary	$X_{1L}$	$X_{2L}$	...	$X_{nL}$
upper boundary	$X_{1U}$	$X_{2U}$	...	$X_{nU}$
	↓ LHS			
	variable 1	variable 2	...	variable $n$
candidate 1	$X_{1\_1}$	$X_{1\_2}$	...	$X_{1\_n}$
candidate 2	$X_{2\_1}$	$X_{2\_2}$	...	$X_{2\_n}$
candidate ...	...	...	...	...
candidate $c$	$X_{c\_1}$	$X_{c\_2}$	...	$X_{c\_n}$

Figure 27: Inputs and outputs of latin hypercube sampling.

### 3.2.2 Candidates Selection

In order to proceed with the selection of candidates, it is fundamental to choose the subjects, or criteria, of this preliminary investigation within the domain. Basically, there have to be certain relevant features for the design that, in case they were not met somewhere inside the domain, would lead to the possibility to exclude a priori those regions from the search of optimums.

To make the concept easier, it is worthwhile to mention directly the criteria that have been chosen for this task. As one may also notice from Figure 28, these are the efficiency, the output power and the power factor. These operating conditions must pursue some project targets and, consequently, the design optimization could be made more effective by looking for optimal solutions only in the zones of the domain next to the promoted candidates.

### 3.2.3 Clustering

The candidates which pass the selection step are most probably scattered within the initial domain so it becomes necessary to group them before proceeding with the design. At this point, clustering means nothing more than grouping the promoted candidates and creating smaller domains.

The clustering algorithm takes into account the space distribution of the promoted candidates and, in a very illustrative way, one could say that the ones close to each other have more chances to be aggregated in the same cluster.

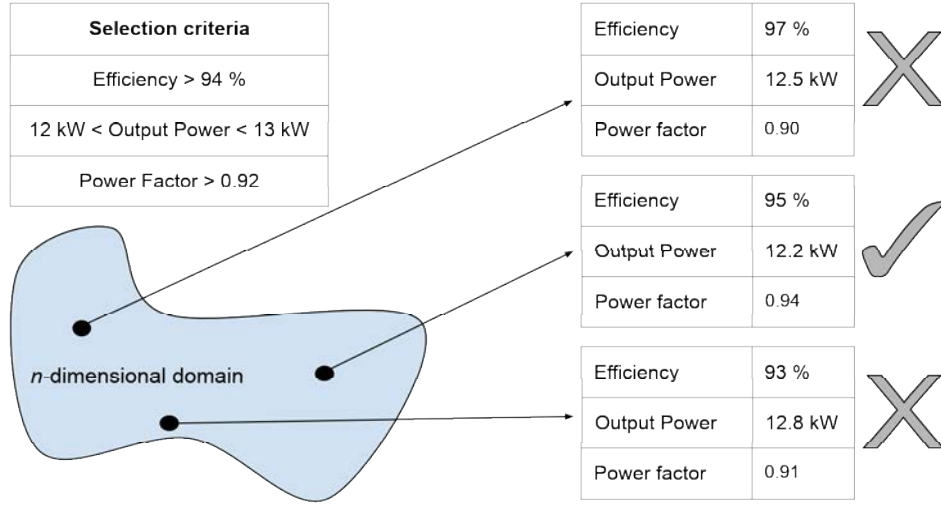


Figure 28: Example of a candidates selection in a  $n$ -dimensional domain based on certain specified selection criteria.

So far, it was said that employing LHS together with a selection step allows to create clusters that will focus the design towards better regions. But there is another fundamental advantage that deserves to be considered: in fact, clustering itself was firstly planned in order to increase the accuracy of the design. To figure out this point it is necessary to look at the the next stage: the construction of surrogate functions. Without waiting for the results, that will confirm what is about to be said, one might immediately understand that the smaller the domain, the more its surrogate functions will be able to represent accurately the functions they are called to replace.

So, the number of clusters represent another important trade-off, this time between accuracy and computational cost. Similarly as before, a high number of clusters implies an improved precision of surrogate functions because smaller subdomains will come up. Differently, the computation time would increase because each cluster demands the construction of its own surrogate functions and further a new optimum analysis.

### 3.3 Construction of Surrogates

Once the clustering has been carried out, to each of the cluster we must assign, in the best way we are granted, the functions that will replace certain "real" operating conditions implemented in the optimization routine. Obviously, we never know the real operating status of the machine and by "real" it is simply meant the ones arising from the FEM simulation. It is worth wile to mention, as an example, the functions required by this specific work: efficiency, output power and power factor.

Quadratic functions are chosen to characterize the evolution of the surrogates, as depicted in Equation 17. During the optimization process, starting from position vectors  $\mathbf{X}$  given by the particles, surrogates can be easily computed once  $\beta$  coefficients

are defined. Basically, building surrogates means to calculate a set of coefficients for each demanded function in each domain.

1	$x_{1\_1}$	$x_{1\_2}$	$x_{1\_3}$	$x_{1\_1} \cdot x_{1\_2}$	$x_{1\_2} \cdot x_{1\_3}$	$x_{1\_3} \cdot x_{1\_1}$	$x_{1\_1}^2$	$x_{1\_2}^2$	$x_{1\_3}^2$	$\beta_0$	$f_1$
1	$x_{2\_1}$	$x_{2\_2}$	$x_{2\_3}$	$x_{2\_1} \cdot x_{2\_2}$	$x_{2\_2} \cdot x_{2\_3}$	$x_{2\_3} \cdot x_{2\_1}$	$x_{2\_1}^2$	$x_{2\_2}^2$	$x_{2\_3}^2$	$\beta_1$	$f_2$
...	...	...	...	...	...	...	...	...	...	$\beta_2$	$f_3$
...	...	...	...	...	...	...	...	...	...	$\beta_3$	$f_4$
...	...	...	...	...	...	...	...	...	...	$\beta_4$	$f_5$
...	...	...	...	...	...	...	...	...	...	$\beta_5$	$f_6$
...	...	...	...	...	...	...	...	...	...	$\beta_6$	$f_7$
...	...	...	...	...	...	...	...	...	...	$\beta_7$	$f_8$
...	...	...	...	...	...	...	...	...	...	$\beta_8$	$f_9$
1	$x_{s\_1}$	$x_{s\_3}$	$x_{s\_3}$	$x_{s\_1} \cdot x_{s\_2}$	$x_{s\_2} \cdot x_{s\_3}$	$x_{s\_3} \cdot x_{s\_1}$	$x_{s\_1}^2$	$x_{s\_2}^2$	$x_{s\_3}^2$	$\beta_9$	$f_{10}$

Figure 29: Example of a matrix system with 10 samples and a three variables domain. Each variable is represented by the second number of the subscript, while the first number one refers to the sample:  $\mathbf{X}_s = [x_{s\_1}, x_{s\_2}, \dots, x_{s\_n}]$

It is done by defining a system of equations, which can be more easily seen as a matrix system, as illustrated in Figure 29. Each cluster is sampled accordingly with Box-Behnken methodology, which provides the group of samples within the defined boundaries; then, the sample matrix  $\mathbf{S}$  is built still according to Equation 17. Each sample is computed by the software and the arising results  $f$  are represented in the right side vector. At this point, the surrogation coefficients can be calculated by inverting the sample matrix and calculating

$$\beta = \mathbf{S}^{-1}f. \quad (46)$$

The number of lines in the sample matrix depends on the applied RSM because performing a different sampling method will also result in a different number of samples; on the other hand, columns reflect the grade of the surrogate functions. Both the number of lines and columns are obviously affected by the number of variables  $n$  of the problem.

### 3.4 Optimization

At this point, boundaries and surrogate functions are known, now it is time for the multi-objective optimization process itself, which will lead to a set of optimal solutions for each cluster.

In order to provide the final solutions, this stage employs:

- PSO, that is responsible for the way the population evolves looking for optiums.
- Pareto concept, that defines the way objectives are processed, namely how optimal solutions are chosen.

The design constraints, representing conditions that the machine must achieve, are considered both within the PSO and Pareto algorithms. First of all, we can say that they differ from objectives because they just need to be respected, without being optimized. Again, to simplify the analysis, let's take immediately into consideration the features which have been examined in this work.

Constraints are output power  $P$  and power factor  $\cos\phi$ , both evaluated by means of surrogate functions. It is clear that solutions must satisfy some requirements both in terms of  $P$  and  $\cos\phi$  which have to be known by the designer. For instance, it is reasonable to assume that an electric motor may require output power values within a certain range and, at the same time, a power factor above the minimum required by the power network.

Objective functions are efficiency and cost; the latter was chosen not to be evaluated by means of a surrogate. The choice is clearly evident both in terms of convenience and in terms of accuracy. First of all, one more surrogate function increases the complexity of the routine and the computational time; besides that, by using the same analytical calculations that will be illustrated in Section 3.7, it is possible to calculate it without requiring extra effort. This is achievable because, in this work, only materials are taken into account for the cost computation so, once all the geometrical aspects of the machine are found, the price required by each part of the machine can be estimated by multiplying its volume by the price per unit volume of that specific material. Some reliable prices of NdFeB magnets and electrical steel are given in Table 5.

Material	Price range in euro/m <sup>3</sup>	Price range in euro/kg
NdFeB magnet	235000 ÷ 253000	31.8 ÷ 33.7
Electrical steel	5210 ÷ 5550	0.668 ÷ 0.702
Copper		6 ÷ 7

Table 5: Price ranges of the three main materials employed in a SPMSM construction. The electric steel and the permanent magnet prices have been provided by some manufacturers [41] [42], while the price of copper can be found in the live stock market of materials [43].

### 3.4.1 Multi-objective Optimization Routine

The process carried out by the routine in the optimization stage is explained with reference to the diagram in Figure 30. Firstly, as depicted by the figure, there are some parameters characterizing the whole process that need to be established. These are:

- The PSO parameters (inertial, cognitive, social).
- The number of particles of the PSO population.
- The number of iterations of the routine.

- The dimension of the archive.

The number of particles and iterations represent a critical trade-off within the optimization stage. Excessively high values result in a significant increase in the execution time of the routine and, at the same time, may also prove to be redundant. However, a low number of particles and iterations can lead the optimization process not to converge.

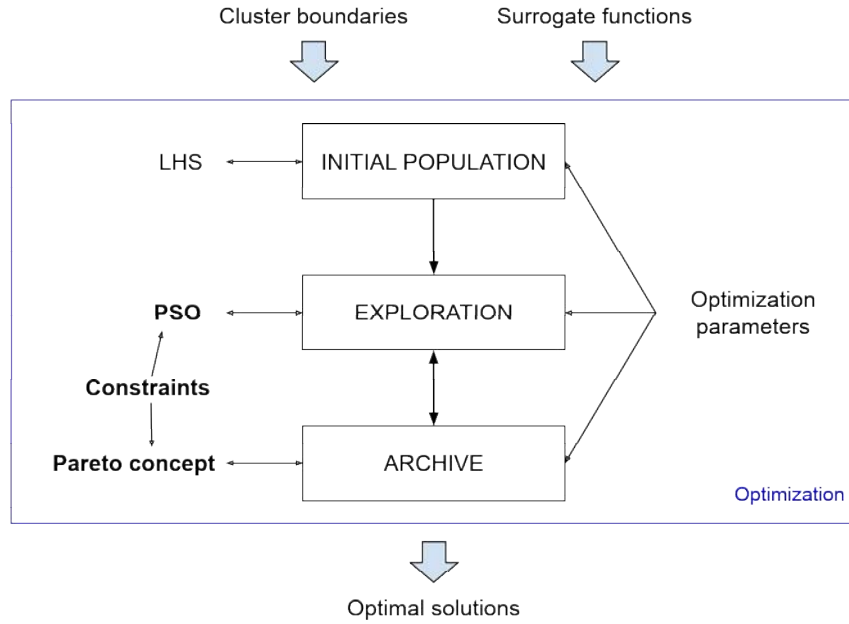


Figure 30: Optimization process scheme: the three main progression steps of the population evolution are shown in capital letters inside the three action blocks.

The archive is nothing more than the place where the best solutions so far are located, being updated iteration by iteration. Basically, a big archive result in a high number of solutions that will need to be checked after the optimization stage. The following paragraphs describe the evolution of the routine.

### Initial Population

At first, the whole population has to be initialized. A Latin Hypercube Sampling is performed to ensure a homogeneous distribution of particles within each cluster. This can help in order to avoid certain areas from remaining unexplored, improving the convergence of the algorithm. Instead, the first population of velocities is randomly chosen keeping in mind that speeds are assigned a maximum limit.

### Exploration

The population of particles is moved by means of the PSO algorithm. As we might know from the theory in Section 2.2.1, each particle moves according to three

components. The inertia coefficient is decreased is kept constant while the others two are not, according to the values originally set in the algorithm that was provided [19]. The personal best of each particle is chosen in agreement with the concept of domination, showed in Subsection 2.2.2; in addition, we must remember that every position respecting the constraints dominates all the ones that do not respect them, whatever the value of the objectives. The global best is the same for the entire population and it is randomly chosen within particles in the archive belonging to the first Pareto level. In this way, as shown in the scheme, the archive is exploited by PSO for the exploration [23].

## Archive

Since a finite number of solutions may be part of the archive, criteria must be given to determine how it has to be filled. First of all, particles that do not respect the constraints should not even be taken into consideration. Secondly, particles which stay in the lowest Pareto levels are given more importance, which means that particles that belonging to high levels are gradually discarded as the archive is updated.

## 3.5 Design Assessment

Once the entire optimization design (that includes clustering, surrogation and optimization) is concluded, a number of optimal solutions are obtained in the final archive, for each cluster. However, the Pareto fronts arising do not refer to FEM simulation directly because the design process employs surrogate functions. As a result, it is known that a design based on surrogate optimization is affected by error.

At this point, it is necessary to check the "real" machine's operating conditions of optimal solutions in the final archive. To have surrogate conditions verified, a FEM analysis is carried out and the same FCSMEK software is exploited.

In practice, in a real design situation, it is crucial to have the highest accuracy in predicting the operating conditions of the machine, since it might required to choose a solution for the production.

In this work, this stage is employed to appreciate the mismatch between surrogate results and FEM ones. Thus, the same surrogates which have been provided by a RSM and exploited during the optimization algorithm must be verified. Therefore, by considering this specif task, it is reasonable to proceed in this way:

- Firstly, the efficiency of each particle in the final archive resulting from the FEM simulation is analysed. Rather, there is no need to worry about the cost because it is certainly not affected by error. The mismatch between the surrogate efficiency and the one arising from the software may lead the designer not to consider certain points in the final archive, in case they do not belong to the Pareto front of the FEM solutions.
- Secondly, but not less important, power factor and output power arising from the software simulation have to be verified as well; optimal solutions that do

not meet these constraints have to be discarded.

The two items can also be applied in the opposite way, namely by first searching for points that meet the constraints and then taking the dominants.

Figure 31 illustrates these two considerations, with reference to a general process of surrogate Pareto optimization. Plotting the objectives space in the two cases, as illustrated in Figure 31(a), can help in order to visualize the discrepancy between the FEM outcomes and the surrogate Pareto front.

Moreover, different types of error analysis can be employed to have an idea about the goodness of the evaluation by means of the surrogate functions for each cluster. This is useful for a designer in order to characterize the accuracy of the design optimization methodology; such methods for the error computation are presented in Appendix A.

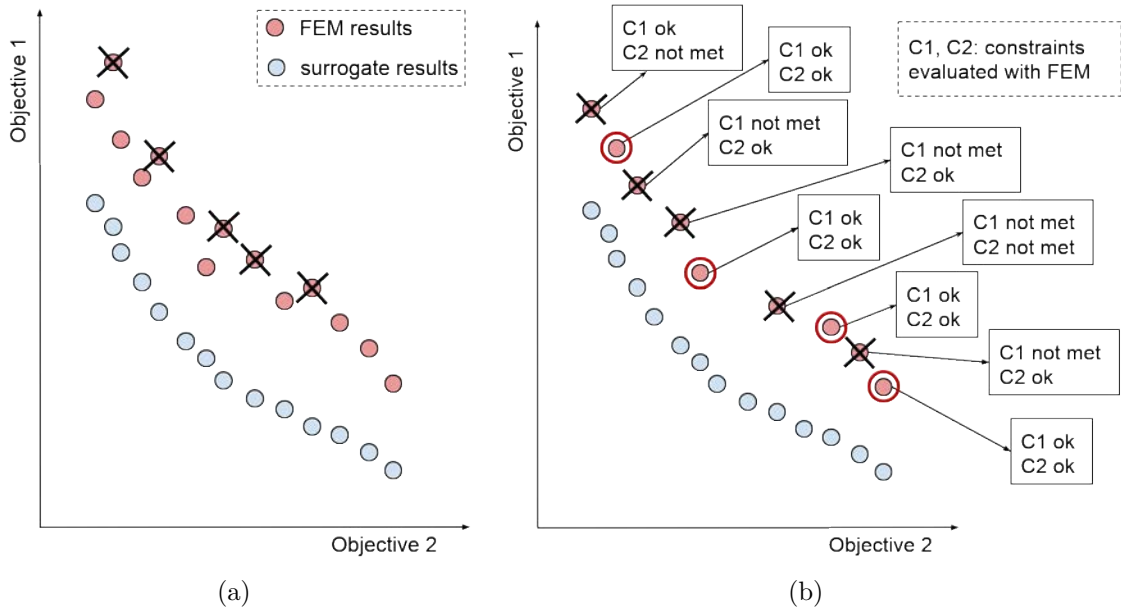


Figure 31: Design assessment of a surrogate multi-objective optimization with a FEM software. Blue solutions come from surrogate optimization, while red ones correspond to their FEM simulations. Two passages are required: (a) points crossed with "X" have to be discarded because, applying Pareto, they are dominated; (b) points crossed with "X" do not respect both the design constraints, thereby only points circled in red pass the final design selection.

### 3.6 Introduction to FCSMEK

FCSMEK is a collection of routines for 2D finite element electromagnetic analysis of synchronous or asynchronous radial flux machines, based on the solution of the magnetic vector potential. The tool, programmed in FORTRAN, was developed as a result of the research conducted by Antero Arkkio in his thesis [44]. Some of the

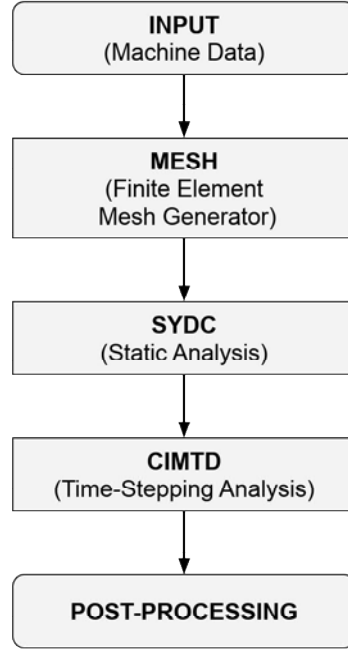


Figure 32: Flowchart of the FCSMEK routines execution.

most significant routines are briefly introduced below, in the same order in which they are performed for this work, as shown in Figure 32.

The software demands a set of machine input parameters, which can be easily computed by means of a MATLAB script and then, after being converted in the appropriate data format, copied in the data file folder which is read by the software at the beginning of the simulation.

MESH generates a 2D mesh with finite elements for the cross-sectional geometry of the machine. The program itself will deduce the smallest symmetry sector; on the other hand, the user has the possibility to choose the order of the finite elements as input from terminal [45][46].

In SYDC, the calculation of the operating characteristics of the synchronous machine is based on the solution of a DC field. The circuit equations for the stator winding are constructed based on a two-axis model solved together with the field equation. From the static FE solution, the program is able to obtain the initial state for the time-stepping simulation. In order to provide all the necessary for the FEM computation, also the temperature of the rotor and stator are requested as input parameters of this routine [45][46].

CIMTD correspond to the time-stepping finite element analysis for the synchronous machine. The circuit equations for the stator winding, field winding and damping cage are solved together with the field equation in each time step; the rotor rotates during the simulation. The routine requires as input data the information concerning the time-stepping [45][46].

Once the finite element analysis is completed, a post-processing study can be conducted by means of four routines [45][46]:



- teho: program for computing effective and average values for operating characteristics from the results of the time-stepping program CIMTD.
- curplot: program for plotting machine characteristics computed by the CIMTD routine as function of time.
- cimplot: program for generating plots of flux, flux-density and iron-loss distributions.
- cimpiir: program for plotting finite element meshes created by program MESH.

In this task, the computation of the electromagnetic losses is performed through the time-stepping simulation, thus the optimization design methodology uses data outcoming from the CIMTD routine.

However, losses computed from a SYDC analysis, which computationally takes much less time, must not deviate too much from the time-stepping ones and they could be helpful to get a first check of the machine characteristics.

### 3.7 Analytical Calculations

The purpose of the analytical calculations, performed within the homonym routine, is to assess all the parameters of a SPMSM, both geometrical and electrical, starting from the defined set of optimization variables. These calculation steps represent a fundamental crossroads between the design theory of electric machines and the optimization.

Even though it is not directly visible in the diagram of Figure 17, this routine is engaged in several circumstances as shown in Figure 33. As we can see, it is mostly employed to generate input data required by the software for the simulation.

Observing the figure, begin blocks suggests the "status" of the variables, i.e.: candidates arising from LHS, samples employed by RSM and final archive solutions. On the other hand, end blocks in the right side mean directly the stage in which analytical calculation are required; in fact, they match the left side blocks of the diagram of Figure 17.

The routine is exploited also during the optimization stage; We have seen that the cost only depends on materials and geometry of the machine which, in fact, can be determined by the same calculations. Thus, when particles are moved through the cluster, analytical calculations allow to determine the value of the second objective function.

Before analysing the procedure with the analytical calculations it is worth to make some considerations on the collection of parameters required by the software employed for this task.

Table 6 shows all the parameters that need to be defined for the FEM simulation, namely input parameters, in the specific case of this work. As may be seen, during the entire design process parameter can be established at different times and in different ways. As reported in the right column, parameters can be:

Parameter	Type
Machine type	Outset
Number of poles	Outset
Supply frequency	Outset
Number of phases	Outset
Length of the stator core	Design choice
Length of the rotor core	Design choice
Outer diameter of the stator	Design choice
Filling factor of a stator slot	Design choice
Number of parallel paths	Design choice
Stator winding layers	Design choice
Winding pitch	Design choice
Stator slots shape	Design choice
Rotor poles shape	Design choice
Stator core material	Design choice
Stator slot wedges material	Design choice
Permanent magnet material	Design choice
Rotor core material	Design choice
Rotor shaft material	Design choice
Stator winding material	Design choice
Air gap width	Variable
Stator tooth width	Variable
Inner diameter of the stator	Variable
Length of permanent magnets	Variable
Height of permanent magnets	Variable
Relative width of permanent magnets	Variable
Rotation angle of the rotor	Variable
Stator winding temperature	Estimated
Rotor temperature	Estimated
Effective length of the machine	Calculated
Number of stator slots	Calculated
Number of conductors in a stator slot	Calculated
Stator slot dimensions	Calculated
Outer diameter of the rotor	Calculated
Inner diameter of the rotor	Calculated
Average length of a coil	Calculated
Resistance of a stator phase	Calculated
End-winding reactance	Calculated
Moment of inertia of the rotor	Calculated

Table 6: Input data collection required by the software; all the possible missing dimensions of the machine depend and can be obtained from the above ones.

- Fixed in the outset: they are characteristics defined by the project requirements or by the plant in which the machine has to be installed.
- Free: they are the variables on which the entire design methodology is based.
- Fixed for design choice: they depend on choices made by the designer depending convenience and availability.
- Estimated: only the temperatures in our case are estimated, since no study about the thermal regime will be carried out.
- Calculated: these parameters depend on the design methodology, namely on the theoretical procedure used as reference, implemented in the analytical routine and they are derived from the calculations.

In machine design, there are a considerable number of possible free parameters and, when looking for an optimal solution, the choice becomes extremely complicated [7]. Free parameters, in design optimization, become no other than the optimization variables. Increasing their number, besides raising the computational time, can lead to some geometrical or electrical incongruities because certain parameters dependent on each other; this issue is the reason why theoretical calculations are needed.

### 3.7.1 Routine for Analytical Calculations

Starting from outset parameters, choices made by the designer, temperature estimations and optimization variables the routine allows to define the whole geometry of the machine. The procedure refers to the same design theory illustrated in Section 3.1.

Firstly, all the initial characteristics and materials are implemented in the algorithm, as in Section 3.1.1. Apart from that, some geometrical parameters are chosen to be fixed design choices in order to meet a secondary target which was, in this specific task, the space occupied by the machine. They are stator outer diameter and machine length, both provided by the theoretical design calculations.

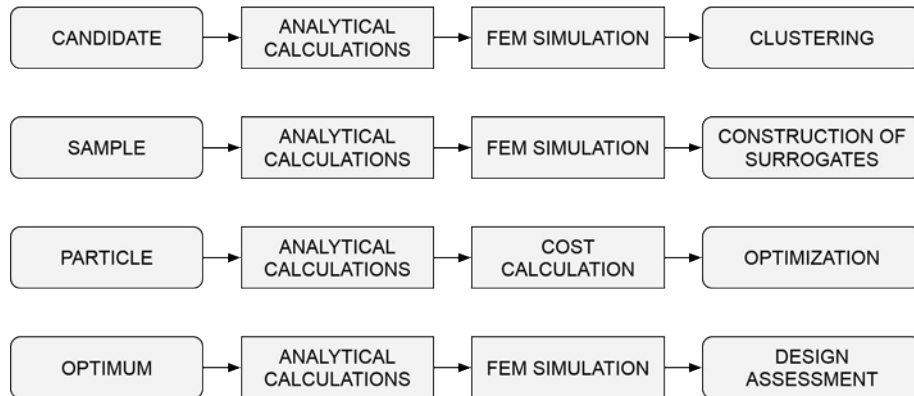


Figure 33: Stages where analytical calculations are employed.

After some outset assessments, the routine computes the main dimensions of the machine and the winding parameters with the same equation given in Section 3.1.2. The main challenge in this step is to properly approximate the number of conductors in a stator slot, since in a two-layer winding it has to be an even number.

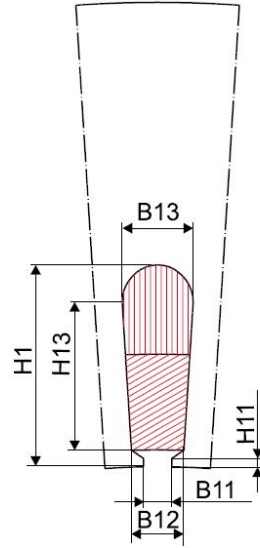


Figure 34: Shape of the stator slot.

After that, all the stator slot dimensions are figured out by taking into consideration the observation made in Section 3.1.3. In order to build a correct geometry, it is necessary to respect the limit of the stator current density, which is determined in accordance with the required stator current that is, in turn, set with regard to the chosen working conditions. The cross section area of the copper is consequently

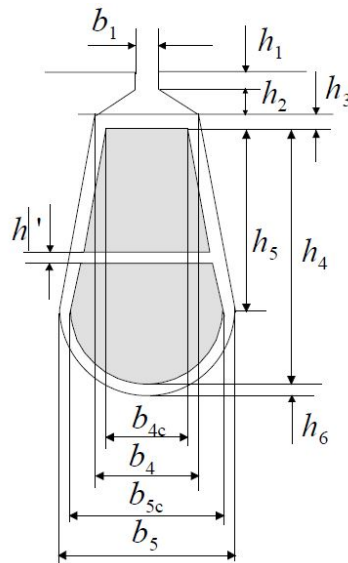


Figure 35: Main dimensions in the stator slot.

constant and represents the geometrical starting point of the slot design; it is also reasonable to fix it if we consider to use the same wire whatever the final geometry of the machine is. The chosen shape of the stator slot is illustrated in Figures 34, while Figure 35 shows in details all the stator slot dimensions that have to be determined. Certain dimensions are fixed a priori and they cannot change during the optimization process. For instance, with reference to Figure 35, the size of the insulating layer between conductors and iron  $h_6$  is kept constant and the slot separator  $h'$  is assumed to be null. The parameters which define the shape of the last part of the slot between the air gap and the conductors,  $h_1$ ,  $h_2$  and  $b_1$  are preferred constant as well. At first, the procedure allows to calculate  $b_{4c}$  and then iterates in order to find the values of  $h_5$  and  $b_{5c}$  which give the same round value of the wound area previously computed. After that, all the other dimensions and the total area of one slot are computed.

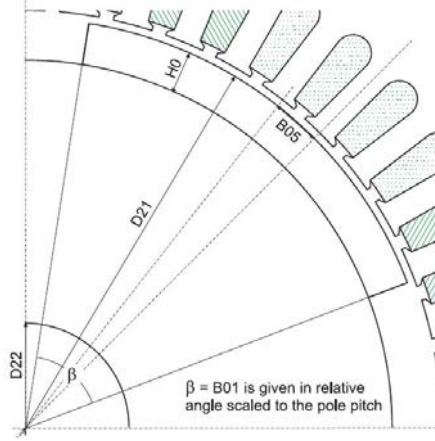


Figure 36: Shape of the rotor slot.

The size of the rotor yoke is computed as described in 3.1.4 and considering the shape chosen for the rotor pole, illustrated in Figure 36. Since the initial estimation of the number of turns was correlated to the maximum value of the rectangular flux density, it is necessary to calculate the new value of  $B_{max}$  derived from the new number of conductor per slot. After that, the height of the rotor yoke is determined.

The latest calculation are intended to compute the resistance and the end-winding leakage reactance, as depicted in Section 3.1.5.



## 4 Results and Discussion

This chapter justify step by step the reasons why the methodology was developed through graphs and practical analysis of the achieved results. Section 4.1 makes some initial observations about the computational burden and the achievement of convergence in the algorithm. Section 4.2 presents the improvements obtained by applying clustering within the proposed methodology. Finally, Section 4.3 makes some practical considerations which need to be taken into account when employing the method.

### 4.1 Computational Burden and Convergence

The computational burden always represent one of the main issue for any kind of design. As depicted by the previous chapter, the proposed methodology employs the FEA, which constitutes the majority of computational burden in this work. The aim of this section is to figure out how a different number of FEM simulations affects the time required by the design.

In this chapter, three simple names will be used to identify as many different MOO methods that one could perform, in order to characterize the different concepts. We will simply call them method 1,2 and 3; the numbers indicate increasing complexity. They can be summarized as follows:

1. PSO directly assisted by FEA;
2. PSO based on FEM assisted BBD (Subsection 4.2.1);
3. PSO based on FEM assisted clustered BBD (Subsection 4.2.2).

Obviously, the Pareto concept is considered to be employed in the same way. This means that all the three methods use the same MOO routine with the same number of iterations  $ITE$ , particles  $P$  and solutions in the archive  $D_{ARC}$ . Thus, different design approaches leads to different number of runs as depicted in Table 7, in which:  $N_{CLU}$  is the number of clusters,  $S$  is number of run combinations (samples) of a  $n$ -variable BBD (Table 8) and  $C$  is the number of the initial LHS candidates.

Method	Number of computer runs
1	$ITE \cdot P$
2	$S + D_{ARC}$
3	$C + N_{CLU} \cdot (S + D_{ARC})$

Table 7: Number of computer runs required by the three methods. With reference to 2 and 3.

Apart from the formulas, it is also worth looking at the examples given in Table 9. In the latter numbers are given in the case of a time-stepping finite element analysis (CIMTD) executed with FCSMEK. With the computer that was made available, every single CIMTD routine required around 30 seconds. We must be aware that,

Number of variables	Run combinations
3	15
4	27
5	46
6	54
7	62
8	120

Table 8: Number of runs depending on the number of variables, or design factors,  $n$  for a BBD.

although this time can change from machine to machine, the computational load will vary in a proportional way for the three methods.

Regarding Table 9, some values have to be discussed so that we can say that the choice of parameters in the left column is not random. The first row takes into account a four-variable MOO, while the second one a six-variable MOO. Obviously, the higher the number of optimization variables the more expensive the problem becomes. For instance, in the second example it is reasonable to expect that an higher number of particles is needed to find the region of convergence. In addition, a large number of variables may also require more candidates for the initial LHS exploration, more clusters and more solutions to be included in the archive.

As one may notice, coupling FEM directly with PSO becomes rather an expensive demand in terms of running time; hence the choice is definitely to base the optimization design on surrogate functions, as suggested by methods 2 and 3. In second place we might notice that employing clustering causes a slight increase in the computational burden. However, the computational burden of the third still remains largely sustainable compared to the first one. The following section will discuss the advantages of clustering despite the small increase in time.

Example	Method 1	Method 2	Method 3
$ITE = 1000; P = 50;$ $S = 46; D_{ARC} = 50;$ $C = 200; N_{CLU} = 3$	17 days	50 minutes	4 hours 5 minutes
$ITE = 1000; P = 100;$ $S = 62; D_{ARC} = 100;$ $C = 400; N_{CLU} = 4$	34 days	1 hour 25 minutes	8 hours 45 minutes

Table 9: Two different cases of computational burden required by a time-stepping FEA performed with CIMTD routine in FCSMEK: first row with  $n = 4$  and second row with  $n = 6$ .

One key issue that deserves a little comment is the number of iterations. According to theory, we must know that a non-dominated Pareto algorithm is able to establish the level of each non-dominated front. Hence, it is reasonable to be satisfied once a



good portion of the points in the archive belong to the first level. In this regard, it is necessary to employ an algorithm able to show us the progress of the optimization process, namely the level in which each particle is situated within the non-dominated fronts. Obviously, the larger the size of the archive the longer it will take to have a completely first level front in the same design space with the same routine. Consequently, also the choice of the dimension of the archive is important. If it is too large, it will slow down the algorithm and might also prove useless because several solutions at the end would remain out of the first non-dominated front (unless we choose a very high number of iterations). On the other hand, too few solutions in the archive cannot cover properly the whole design space and the MOO study would not be exhaustive.

Regarding the convergence of the algorithm, a further aspect which deserves to be mentioned concerns the fact that the entire set of optimal solutions arising from an optimization process in one single domain should turn out to have close values of the optimization variables. If this does not happen, it might be that the particles have not reached convergence in the design space. This has always occurred during the work whatever the design domain was clustered or not and whatever was the number of clusters. In practice, this means that all the optimum of the same final archive figure out machines having very similar features. On the other hand, it is quite obvious that each final archive is related to a region of convergence different from all the others. This confirms that the employed multi-objective optimization algorithm can be applied in every domain (or subdomain) and it will always converge if a sufficient number of iterations is set.

## 4.2 Optimization by Box-Behnken Design in a SPMSM

In this section the achieved results are shown by first observing the objective space. In particular, the most interesting analysis comes from the comparison between the Pareto front that emerges from the surrogate optimization and the FEM results arising from the same solutions. In fact it is evident from the very beginning that it may be too inaccurate to rely the conclusions exclusively on surrogate results, which is basically the reason why a FEM assessment of optimal solutions is proposed in the last stage of design. In this chart, the x-axis represents the cost which is directly related to a specific geometry of the machine. In fact, when a solution is set in the final archive its values of optimization variables are saved; they unequivocally distinguish one and only one machine built from the variables by means of the analytical calculation routine. In this way, the two points (one surrogate and the other assessed with FEM) referring to the same machine lie vertically to each other because they have the same price, which depends exclusively on the geometry of the SPMSM. Therefore, we are allowed to think that the cost identifies the machine (this consideration is due mainly to the fact that the cost is not surrogate).

After that another analysis must be carried out on constraints, which have to be checked since they are surrogate as well. Still, the key point of the representation is to notice how much surrogate optimal solutions differs from the FEM assessment outcomes. It is no longer necessary to take the cost into account for this purpose, so

each value in the x-axis simply represents a different machine.

#### 4.2.1 Method Without Clustering

According to the previous section, we know that employing a RSM and build surrogates allows to reduce very significantly the computational time. The first idea that might to be taken into consideration is to apply the Box-Behnken design (which is the chosen RSM four out task) to the entire domain of optimization variables. Figure 37 illustrates both the Pareto front arising from the surrogate optimization and the corresponding actual values of the FEM assessments. You can immediately notice the discrepancy in values between the red and the blue curve, which is clearly more than 1% in terms of efficiency. A second consideration can also be made: the blue curve is no longer a Pareto front at all, being almost flat. Thus, from the blue points of Figure 37 the designer could only consider the machine represented by the first point on the left. In fact all the others, despite being more expensive, appear to have more or less the same efficiency which basically means that many solutions turn out to be no longer in the final Pareto front. As far as we are concerned, we are allowed to call "final front" the one represented by the FEM assessed results. Therefore, two rather negative aspects emerge in the objective space:

- The design is not accurate;
- The design is not worthwhile.

Moreover, it is not enough to look only at the efficiency given by FEM assessment because the design requires also the machine to respect the constraints. Figures 38(a) and 38(b) shows what happens respectively in the case of the output power

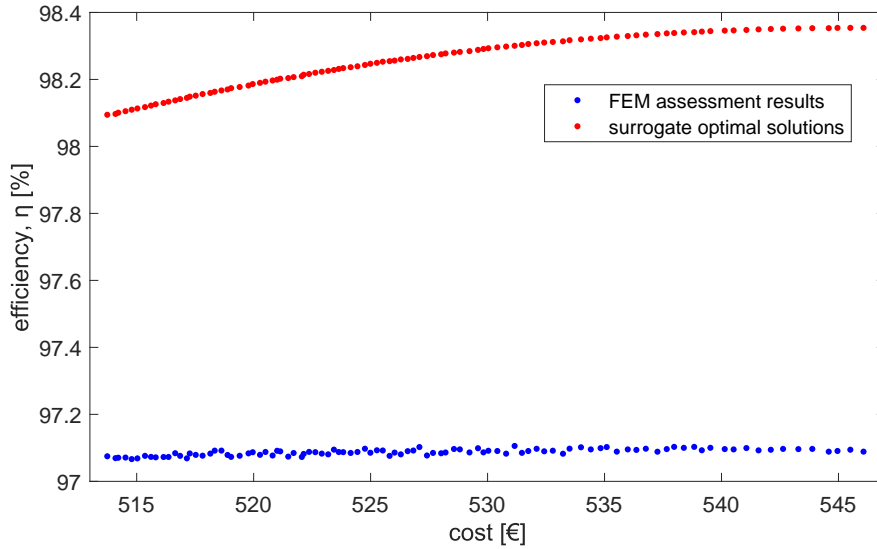
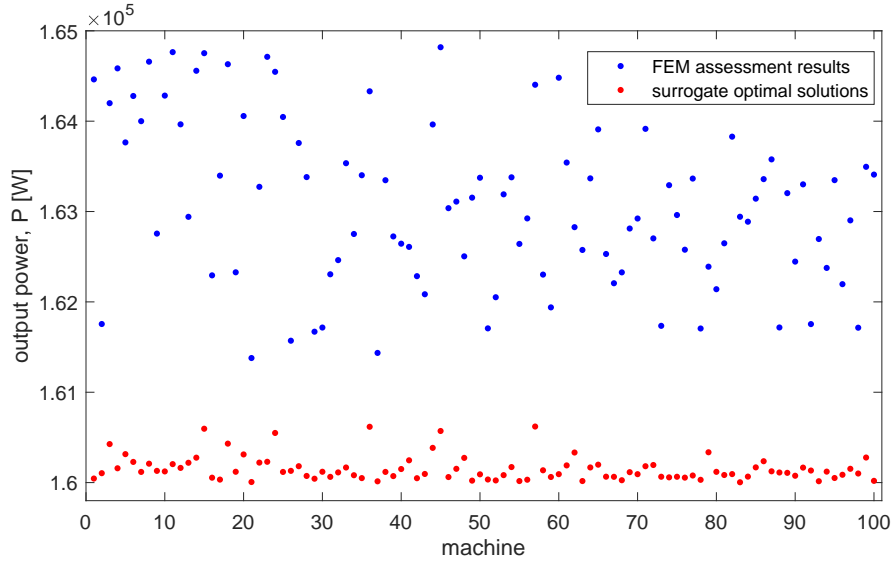
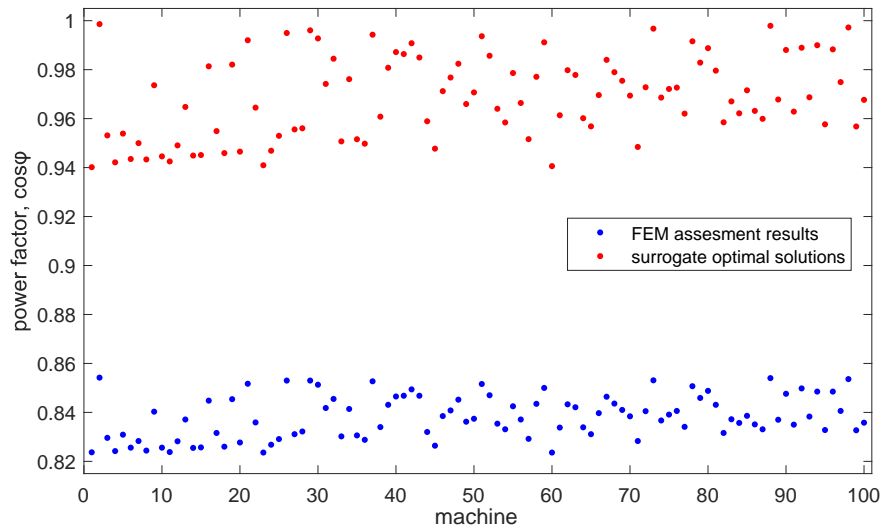


Figure 37: Comparison between the surrogate Pareto front and the results given by FEM in the case of BBD applied in the entire design space.



(a)



(b)

Figure 38: Constraints comparison between the surrogate solutions and the results given by FEM with BBD applied in the entire design space: a) output power; b) power factor.

and power factor. With regard to the output power, FEM computations are not too divergent compared to the surrogate results and we can just notice that the latter are a little lower. On the other hand, the power factor is really bad evaluated by surrogate functions. In fact, it turns out that almost all the optimal solutions have a  $\cos\phi$  between 0.82 and 0.86, which is quite different if compared to the surrogate values which stays between 0.94 and 1.

Now, we should be aware that the last stage of the methodology requires to discard all the points which do not meet the constraints. For this reason, it becomes

practical at this point knowing which values could be required by the project. Just for exemplification purpose, we can take into considerations the last row of Table 12 and by looking at Figure 38 we can see that there would be no suitable solutions for the design; in particular, no point assessed with FEM respects constraints on the power factor which was required to be at minimum 0.92. On the other hand, the output power is evaluated by the surrogate with more accuracy and all the assessed points stay within the required range of  $P$ , which was for this task.

In conclusion, it is evident that applying Box-Behnken design in the entire variables domain, apart from being inaccurate and unprofitable for performing a surrogate optimization study, could even reveal to be faulty. Thus, in general we could see that this happens when applying a RSM in a design space that is too wide. In fact, what happened with method 2 is that surrogate functions of  $\cos\phi$  was so much inaccurate that it did not lead to acceptable solutions on the entire final front. It has to be clear that there are some project requirements on the machine (the design constraints) which must necessarily be fulfilled to make it feasible.

#### 4.2.2 Clustering-Based Method

The outcomes of the previous paragraph suggest the need to develop a different methodology that can somehow restrict the design space in which the BBD has to be applied. The first approach that one might perform is to split the starting design domain in order to obtain smaller subdomains, which joined together form the full domain. Then, in each of these subdomains a surrogate BBD followed by a surrogate MOO study should be performed, leading to a number of final archives equal to the number of subdomains. It can be deduced from Table 10 that this approach would become quite expensive in terms of computational burden. Moreover, it would certainly not be easy to manage too many solutions coming from the archives.

Number of variables ( $n$ )	Subdomains dividing each variable in 2 parts ( $2^n$ )	Subdomains dividing each variable in 3 parts ( $3^n$ )
2	4	9
3	8	27
4	16	81
5	32	243
6	64	729
7	128	2187
8	256	6561

Table 10: Number of resulting subdomains dividing every optimization variable by two and three.

Instead, by simply making use of clustering it can be obtained a design much more selective and simplified. LHS followed by a selection of the candidates provide the k-means algorithm a set of suited particles to be clustered. In this task, the k-means clustering algorithm has been carried out four times in order to observe what can be achieved with a different number of clusters, from 2 to 6.

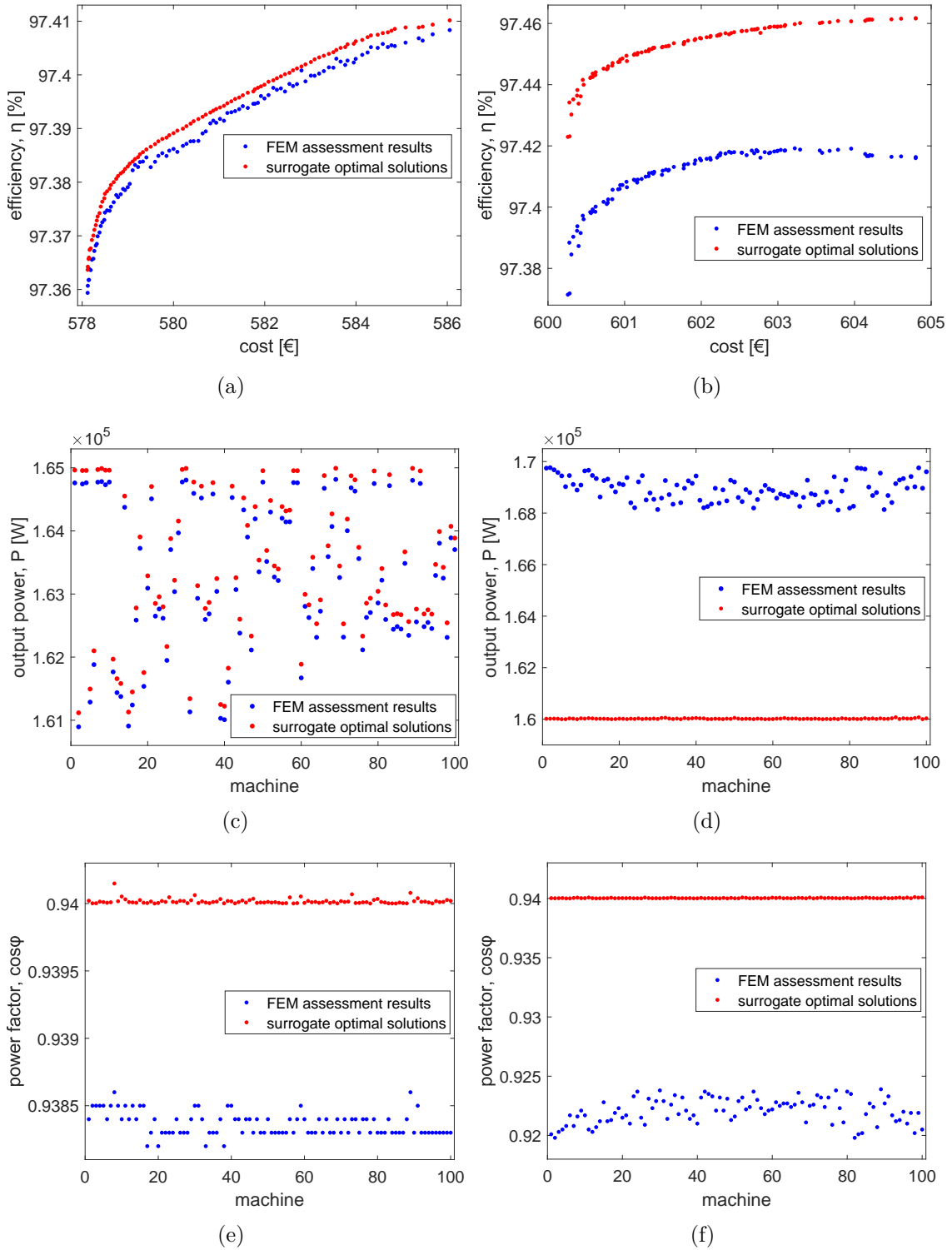


Figure 39: Surrogate optimal solutions in the final archive and corresponding FEM of a six-cluster surrogate BBD. Objective functions in cluster number: a) 2; b) 4.  $P$  in cluster number: c) 2; d) 4.  $\cos\phi$  in cluster number: e) 2; f) 4.

Figure 39 shows the same kind of charts as Figures 37 and 38 but this time, as it has just been said, the optimization process is based on a BBD applied in different clusters. Only the results arising from two out of six clusters are depicted. To be precise, the results presented in the left column are taken from the second cluster, while the right column refers to the fourth cluster. The most visible consequence that one might notice is that the curves of the results arising from FCSMEK better maintains the desired "Pareto shape", thereby the multi-objective study is worthwhile and the target is reached. Moreover, by applying the RSM in a smaller design space the error decreases, which implies that the working conditions are evaluated more accurately by means of the surrogates. For instance, in Figure 39(a) the difference between the two performed efficiencies is almost negligible, while Figure 39(b) shows a discrepancy less than 0.1% which is still much better than the divergence of 1% in method 1. In addition, also the design constraints are better evaluated with clustering. In particular, Figures 39(c) and 39(e), which both refer to the same cluster of 39(a), have very accurate surrogates since both  $P$  and  $\cos\phi$  evaluated with FCSMEK differ notably less than 1%. On the other hand, Figures 39(d) and 39(f) confirm that the surrogate functions built on the fourth cluster are slightly more inaccurate, in accordance to what occurs in Figure 39(b).

Conducting similar inspections to the ones above in all the clusters, whatever it is the set number of clusters, it was observed that efficiency and power factor were always evaluated at higher values by surrogates than the actual (FEM) ones, while it did not happen systematically with output power.

#### 4.2.3 Comparative Analysis of the Accuracy

At this point, in order to obtain a more detailed analysis, it is necessary to link the accuracy of the surrogates to a certain feature of the clusters. One might immediately think about comparing the size of the clusters, which aspect was actually already anticipated in the observations that led to the development of the third method. In particular, in each cluster the relative hypervolume has been computed. The concept, which is explained in Appendix B, allows to get an idea of the size of the subdomain compared to whole design domain.

Table 11 shows the comparison between the mean relative error (see Appendix A) in the evaluation of surrogates efficiency, output power and power factor. One might immediately notice that the hypervolume of the cluster heavily affects the error. So now we have results proving that the dimension of the cluster is the most important feature which affects the accuracy of Box-Behnken surrogate functions within the cluster.

However, using the k-means algorithm we are not able to set a limit in the dimensions. Thus, the only thing that we can do is to change the number of clusters and by doing that we can be sure that also the dimensions of the the clusters will change. It is quite logical to think that by choosing a higher number of clusters it will be much more likely that they will be smaller on average; this is ensured by the fact that promoted candidates that participate in clustering are always the same.

Note that in Table 11 the case study with five clusters presents only four results.

In fact, it might happen that the optimization algorithm cannot reach any suitable solution within one cluster. This occurs when in that specific subdomain no particle was ever in the condition of respecting both the constraints during the whole optimization process, thus the archive has remained empty.

To better appreciate how the accuracy of surrogates improves using clustering let's take a look at the trends in Figure 40, where the standard deviation (STD) is shown in relation to the relative hypervolume. As we can see from Figures 40(a) and 40(c), both the  $\eta$  and the  $\cos\phi$  demonstrate a great enhancement in the application of the methodology, if we compare the STD computed without clustering (which is the one having relative hypervolume equal to 1). On the other hand, the surrogate output power was already quite good accurate even without employing clustering, as one might notice from Figure 40(b). Despite this result might contrast a little our thesis, we can still figure out a positive trend observing Figure 41, which illustrates the  $STD_P$  without considering the non-clustered solutions. In fact, the chart distribution shows that the possibility of obtaining an accurate surrogate function for  $P$  decreases very quickly as the hypervolume increases. The same type of behavior can be observed by zooming the bottom left part of graphs about  $STD_\eta$  and  $STD_{\cos\phi}$ .

Number of clusters employed	Relative hypervolume	MRE		
		$\eta$	$P$	$\cos\phi$
1	1	0.011978	0.01793	0.15582
2	0.0096434	0.0014267	0.082046	0.056262
	0.061697	0.0013388	0.11881	0.062877
3	0.0010169	0.00070619	0.069102	0.032988
	0.012132	0.0021882	0.11367	0.08373
	0.016796	0.0013786	0.10777	0.063842
4	9.1543e-07	0.00045885	0.025683	0.015805
	9.4327e-06	0.00043031	0.05291	0.018511
	0.00015375	0.0011096	0.069941	0.043503
	0.013604	0.0014332	0.074238	0.050499
5	3.5531e-05	0.0010818	0.063628	0.042457
	0.0015208	0.00051552	0.048239	0.023518
	0.0027262	0.0013444	0.10662	0.061672
	0.0049471	0.0015088	0.086538	0.066827
6	8.4813e-09	2.9994e-05	0.0012071	0.0016961
	9.4327e-06	0.00043	0.052718	0.018438
	1.0225e-05	3.4078e-05	0.0001819	0.00029961
	0.0003225	0.0010498	0.05749	0.037945
	0.0011033	0.0021456	0.10475	0.073802
	0.0027155	0.001365	0.10968	0.063619

Table 11: Relative error of the three surrogate function in each cluster. Within each case study, clusters are ordered in ascending order of relative hypervolume.

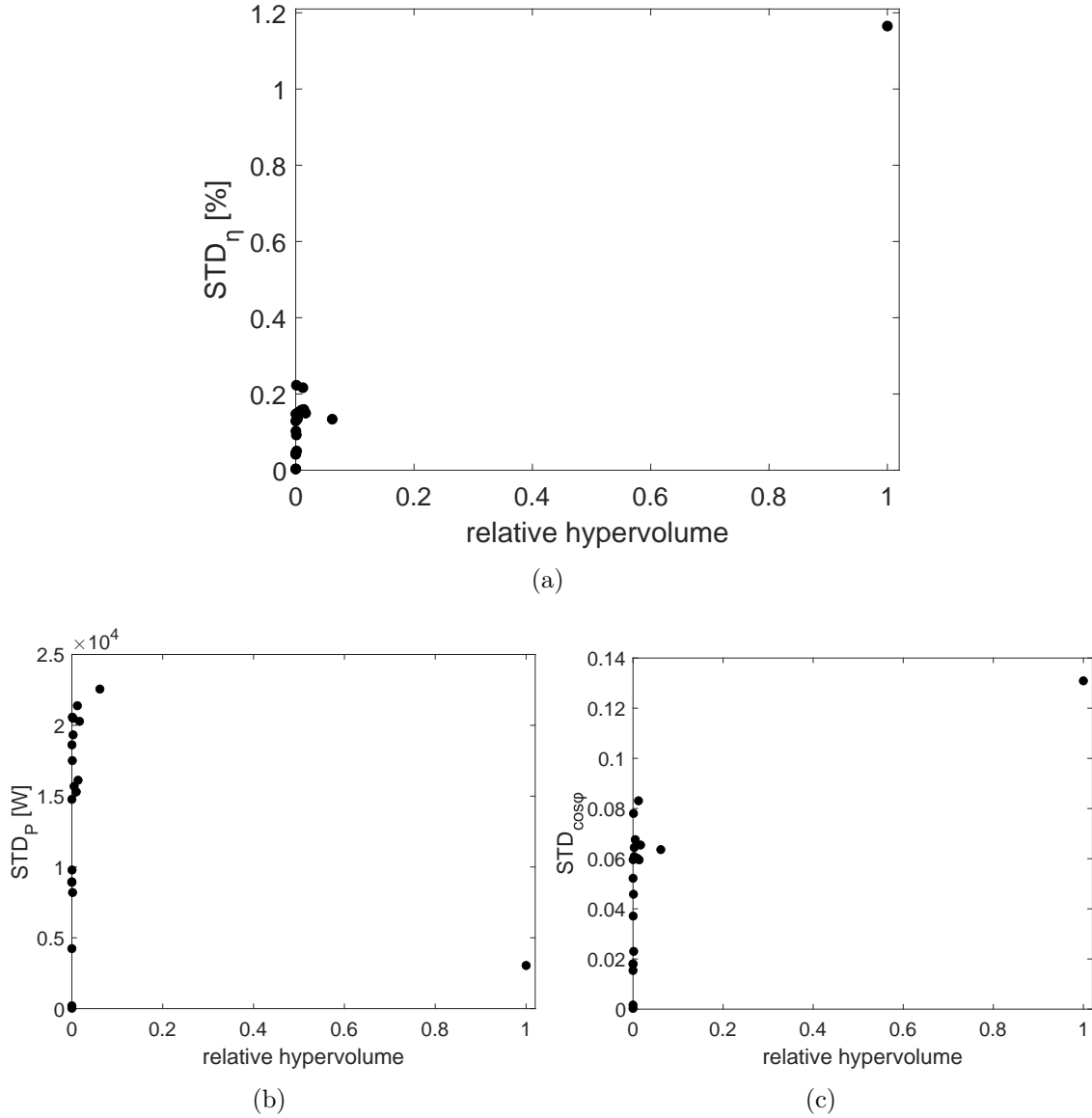


Figure 40: Surrogate optimal solutions in the final archive and corresponding FEM of a six-cluster surrogate BBD. Objective functions in cluster number: a) 2; b) 4.  $P$  in cluster number: c) 2; d) 4.  $\cos\phi$  in cluster number: e) 2; f) 4.

#### 4.2.4 Achievements using the Clustering

In the first part of this paragraph we will try to figure out if, in addition to greater accuracy, clustering has also revealed to find better optimal solutions. For this purpose we must observe Figure 4.2.4 very carefully. The four charts present the results arising from the five clusters with the same color. At first, in 42(a) surrogate clustered solutions (green) are compared to the ones arising without employing clustering (orange) in the criterion space. One might notice that most of the solutions given by the BBD applied in the entire design space dominate the ones arising by employing clustering. However, this does not happen in Figure 42(b) where this time the same



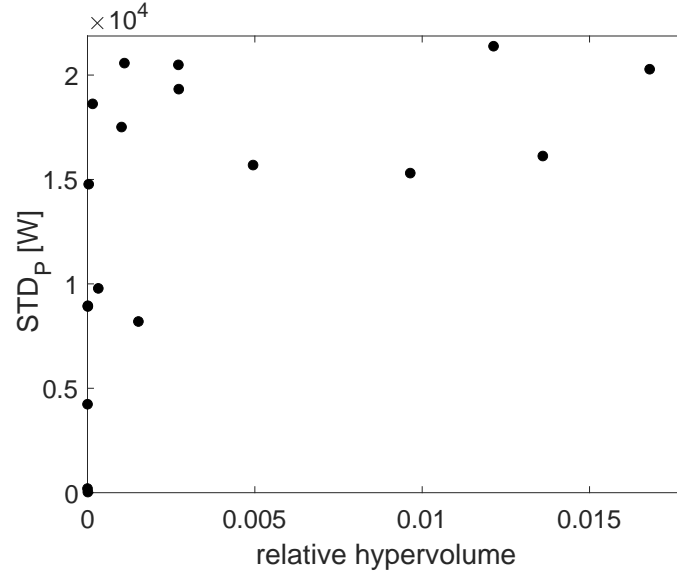


Figure 41: Detail of Figure 40(b): STD of the surrogate output power that refers to the design with clusters.

optimal solutions as before are represented with the results coming from the FEM simulations. By having set the same scale in the two y-axis, one might notice at first the greater reliability of a clustering assisted design. Apart from this, Figure 42(b) shows that using clustering leads to better solutions, especially in terms of efficiency. This last observation is important and allows us to say that employing clustering within a surrogate optimization BBD leads not only to more accurate evaluations, but might also help to find better results.

Despite everything, one should also reply that by observing Figure 42(b) there is no reason to exclude all the orange solutions because some of them, thanks to the low cost, are not dominated by the blue ones. Nevertheless, we might also remember the observations already made at the end of the paragraph which analyses the method number 2: most of the orange points do not belong to the Pareto front at all and none of them meet the design constraints once assessed with FEM.

Now, the weak points of the surrogate design must be considered. Thus, a simple but efficient procedure, resuming with what has been said at the end of 3.5, is illustrated in the last two charts in Figure 4.2.4. Starting by applying Pareto, the designer must take into account only the solutions situated in the first non-dominated front, namely the Pareto optimums. To make it easy, Figure 42(c) has taken from Figure 42(b) only the points included in the first Pareto level, by simply applying the concept of domination. After that, we know that only the FEM assessed solutions meeting the required values of  $P$  and  $\cos\phi$  will be finally considered. Therefore, from Figure 42(c) to Figure 42(d) other points have been excluded.

In this task, once the whole design process has been performed, we have found a set of optimal machines which satisfy the two project requirements of  $P$  and  $\cos\phi$ . The range of price goes approximately between 560 and 600 EUR, while the efficiency stays between 97.4 and 97.7 %. When choosing the final solution for the machine,

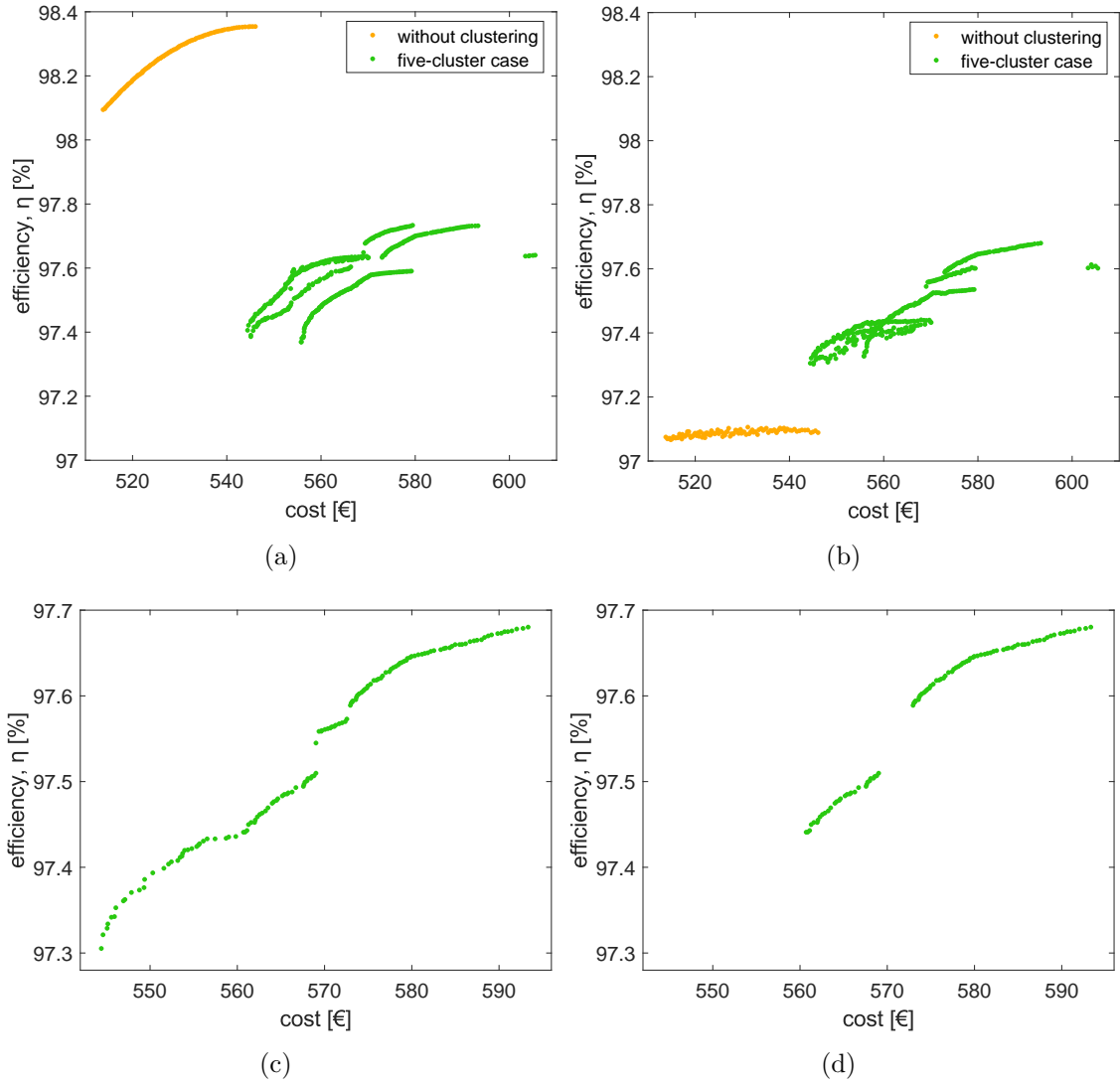


Figure 42: Comparison in the objective space of optimal solutions between not clustered and five-cluster design with: a) surrogate efficiency; b) FEM efficiency. Procedures which allow to achieve the final optimal solutions by considering the FEM assessed results: c) removing solutions that do not belong to the first non-dominated Pareto front; d) removing solutions which do not respect the design constraints.

it is the designer's job to make posteriori evaluations on results depicted in Figure 42(d):

- If more concerned with efficiency, the choice would fall on the most right point;
- If more concerned about the price, the choice would fall on the most left point;
- If efficiency and cost still have no reason to be preferred to each other, a point in the middle of the final front could be appropriate.

### 4.3 Further Observations

This chapter illustrates some choices that have been made during the execution of the routines.

During the process, several criteria must be defined, as depicted in Table 12. As we might remember from Subsection 3.2.2 in the candidates selection within the clustering process, also the efficiency was a criterion but it is not shown in the table which refers only to constraints. The reason why the criteria in the initial stage were made less restrictive is that particles which pass the selection are only supposed to be located in regions with better features than others. Setting excessively restrictive conditions in the first stage could lead to the exclusion of potentially very good regions. In fact, it could be that a region of space which contains optimal points is discarded because only low-performance samples within it are selected by the LHS. In this way, one may agree that the more densely the initial domain is sampled, the more restrictive the conditions can be imposed.

Stage	Output power [kW]	Power factor
Clustering	$154 < P < 171$	0.90
Optimization	$160 < P < 165$	0.94
Design assessment	$160 < P < 170$	0.92

Table 12: Constraints which have been chosen within the entire process.

Afterwards, still looking at Table 12, it is quite reasonable to assume that constraints in the last stage are not the same as in the optimization one. It is worth to give less prohibitive values in the optimization stage, expecting that FEM results can turn out to be slightly different because of surrogation. It is important to understand that the minimum required values that the final design cannot fail to respect are the ones in the final assessment. For example, if we take into account the results of the previous paragraphs, which arise from a five-cluster and six-cluster surrogate design, no solution would satisfy a minimum power factor of 0.94 once assessed with FEM.

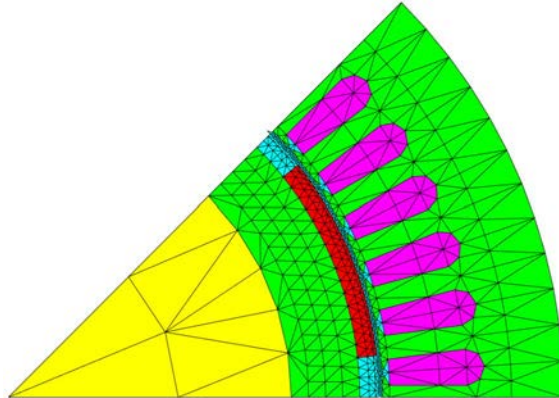


Figure 43: Triangular mesh of the machine arising from the preliminary calculations.

During the execution of the routine, many other numbers and parameter was chosen. Regarding the FEA, a triangular mesh and linear elements were chosen. The arising mesh is illustrated in Figure 43. Four periods were taken for the time-stepping simulation, each one divide into three hundred time steps.

For the optimization process, around sixty thousand iterations were required in order to obtain a first level Pareto archive with one hundred solutions.

## 5 Conclusions

In this thesis, a clustering-based surrogate multi-objective optimization technique has been developed. The method has been implemented in the design of a surface permanent magnet synchronous machine. The operating conditions of the machine were performed in CIMTD routine incorporation with the FEM solver software FCSMEK provided by Aalto University. All scripts, functions and plots were developed in MATLAB, starting from the analytical design of the machine. Certain parameters have been chosen to be the optimization variables of the design and, according to the analytical results, the boundaries of the design space have been set. A latin hypercube sampling was performed within the initial domain and only the candidates which adhered the established design constraints were taken to the next step, namely clustering. The latter was the main improvement, proved to have very positive and significant consequences within surrogate optimization design. After the promoted candidates have been clustered, a Box-Behnken design (which a response surface methodology) was carried out for each cluster. BBD allowed to get different surrogate functions within them: efficiency, output power and power factor. A multi-objective particle swarm optimization algorithm, based on surrogates, was applied to solve this problem by searching for the Pareto front. The two objectives of the task were efficiency and cost (calculated based on materials), but within the process the design. After having obtained a Pareto front for each cluster the optimal solution are run in FCSMEK, giving rise to the actual results.

The application of a RSM has significantly reduced the computational burden of the design. In fact, if we had coupled the optimization algorithm with FEM it would have taken days to get the solutions every time.

The comparison between actual results and evaluations made by the surrogate functions in the case of clustering-based design and non-clustered design allowed us to draw many valuable conclusions, listed below:

- The accuracy increases when applying clustering. This has been explained by the relationship between the surrogate error and the relative hypervolume of the design space in which the RSM is applied. Also, we have seen how accuracy of the surrogates might reveal crucial especially in the evaluation of the design constraints; in the worst case, no feasible solution is reached.
- Good maintenance in the shape of the actual Pareto front when clustering is applied. This leads to many more solutions being considered acceptable once analysing the criterion space of actual results.
- The clustering-based design give rise to better actual results, which is still mainly due to the greater accuracy.

In conclusion we must say that employing clustering within surrogate MOO allows to have a design which is:

- Affordable in terms of computational burden.
- Accurate in the evaluation made by surrogate.

- Worthwhile and feasible with respect to the final actual achievements.

Finally, there are several things that can be done to further develop the method. For instance, to improve the accuracy and the reliability of it, one might go into deep with clustering methods. In fact, apart the k-means algorithm, many other methodologies exist and could help in order to optimize the grouping and the number of clusters. Other than that, the central composite design might also be tested in the same way we have done with BBD.

Besides the method itself, there are many other relevant aspects that should be enhanced to take this kind of design to the next level. For instance, it would be possible and interesting to perform a multi-objective optimization study where both the objective functions are surrogate. For this purpose, by employing FCSMEK one may consider to build a surrogate of the torque ripple and minimize it, which is such a demanded target in this kind of machines. Just to give some more ideas, a further analysis that may be added in the design of a SPMSM is the analytical computation of temperature of the permanent magnet. The thermal analysis always represent on of the main issue in engineering and it is not really treated in FCSMEK, which just ask for it as input parameter. Thus, since we have kept the temperature constant, the next step should be to compute the temperature of the permanent magnet within the analytical calculations and add it as design constraint.

## References

- [1] G. Neidhöfer. The evolution of the synchronous machine. *Engineering Science and Education Journal*, 1(5):239–248, 1992.
- [2] P.C. Sen. *Principles of Electric Machines and Power Electronics*. Wiley Global Education, third edition, 2013.
- [3] B. Seixas Gomes de Almeida and V. Coppo Leite. Particle Swarm Optimization: A Powerful Technique for Solving Engineering Problems. In Javier Del Ser, Esther Villar, and Eneko Osaba, editors, *Swarm Intelligence*, chapter 3. IntechOpen, Rijeka, 2019.
- [4] M. Degano, M. Di Nardo, M. Galea, C. Gerada, and D. Gerada. Global Design Optimization Strategy of a Synchronous Reluctance Machine for Light Electric Vehicles. In *8th IET International Conference on Power Electronics, Machines and Drives (PEMD 2016)*, pages 1–5, 2016.
- [5] S. Lee, Y. Kim, K. Lee, and S. Kim. Multiobjective Optimization Design of Small-Scale Wind Power Generator With Outer Rotor Based on Box–Behnken design. *IEEE Transactions on Applied Superconductivity*, 26(4):1–5, 2016.
- [6] M. Billah Md. Iron loss computation in a synchronous machine from a static field solution. Master’s thesis, 2020.
- [7] J. Pyrhönen, T. Jokinen, and V. Hrabovcová. *Design of Rotating Electrical Machines*. John Wiley & Sons, Ltd, second edition, 2013.
- [8] J. Pyrhönen, T. Jokinen, and V. Hrabovcová. *Design of Rotating Electrical Machines*. John Wiley & Sons, Ltd, 2008.
- [9] Electrical Concepts. Understanding of Direct and Quadrature Axis Synchronous Reactance, 2018.
- [10] A Hughes. *Electric Motors and Drives: Fundamentals, Types and Applications*. Elsevier Science, third edition, 2006.
- [11] T. Mohanarajah, J. Rizk, A. Hellany, and M. Nagrial. Design of Permanent Magnet Assisted Synchronous Reluctance Machines with Rare-Earth Magnets. In *2019 International Conference on Frontiers of Information Technology (FIT)*, pages 122–126, 2019.
- [12] SKF. General catalogue from SKF. Catalogue 4000/IV E, 1994.
- [13] W. Schuisky. *Berechnung elektrischer Maschinen*. Springer-Verlag GmbH, 1960.
- [14] J. Pyrhönen, T. Jokinen, and V. Hrabovcová. *Design of Rotating Electrical Machines*. Three-phase permanent magnet motor with rotor surface magnets, a two-layer integral slot winding and open-circuit cooling.

- [15] K. F. Man, K. S. Tang, and S. Kwong. Genetic Algorithms: Concepts and Applications. *IEEE Transactions on Industrial Electronics*, 43(5):519–534, 1996.
- [16] R. Eberhart and J. Kennedy. A New Optimizer Using Particle Swarm Theory. In *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43, 1995.
- [17] C. M. Martinez, X. Hu, D. Cao, E. Velenis, B. Gao, and M. Wellers. Energy Management in Plug-in Hybrid Electric Vehicles: Recent Progress and a Connected Vehicles Perspective. *IEEE Transactions on Vehicular Technology*, 66(6):4534–4549, 2017.
- [18] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 4, pages 1942–1948 vol.4, 1995.
- [19] F. Martin. Particle swarm optimization algorithm.
- [20] T. Huang and A. Sanagavarapu Mohan. A Hybrid Boundary Condition for Robust Particle Swarm Optimization. *Antennas and Wireless Propagation Letters, IEEE*, 4:112–117, 02 2005.
- [21] K. Deb. Multi-objective Optimization. In Edmund K. Burke and Graham Kendall, editors, *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, chapter 15, pages 403–449. Springer US, Boston, MA, 2014.
- [22] K. H. Chang. *e-Design: Computer-Aided Engineering Design*. Elsevier Science, 2015.
- [23] P. Alotto and P. Bettini. Domination & Non-Dominated Sorting Pareto Archive.
- [24] S. Mu, J. Yin, J. Yuan, and S. H. Ng. Design of Experiments for Simulation Models with Stochastic Constraints. In *2009 IEEE International Conference on Industrial Engineering and Engineering Management*, pages 2094–2098, 2009.
- [25] R. Iman. Latin hypercube sampling. pages 408–411, 1999.
- [26] MathWorks. Response Surface Designs - MATLAB lhsdesign.
- [27] J. W. Bandler, Q. S. Cheng, S. A. Dakroury, A. S. Mohamed, M. H. Bakr, K. Madsen, and J. Sondergaard. Space mapping: the state of the art. *IEEE Transactions on Microwave Theory and Techniques*, 52(1):337–361, 2004.
- [28] P. Jiang, Q. Zhou, and X. Shao. *Surrogate-Model-Based Design and Optimization*, pages 135–236. 2020.
- [29] NIST/SEMATECH. Process improvement. In *e-Handbook of Statistical Methods*, chapter 5. 2012.



- [30] MathWorks. Response Surface Designs - MATLAB & Simulink.
- [31] MathWorks. Box-Behnken design - MATLAB bbdesign.
- [32] MathWorks. Central composite design - MATLAB ccdesign.
- [33] M. Johnsen. *The Future of Artificial Intelligence in Digital Marketing: The next big technological break*. CreateSpace Independent Publishing Platform, 2017.
- [34] E. Alpaydin. Clustering. In *Introduction to Machine Learning*, Adaptive Computation and Machine Learning, chapter 7, pages 161–183. MIT Press, Cambridge, MA, third edition, 2014.
- [35] A. Likas, N. Vlassis, and J. Verbeek. The global k-means clustering algorithm. *Pattern Recognit.*, 36(2):451–461, 2003.
- [36] Polimi. A Tutorial on Clustering Algorithms.
- [37] C. Piech. K Means.
- [38] Y. Liu, J. Liu, T. Zheng, and Y. Yang. A Surrogate-Assisted Clustering Particle Swarm Optimizer for Expensive Optimization Under Dynamic Environment. In *2020 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8, 2020.
- [39] Neorem Magnets. Neorem 4945a.
- [40] Cogent Power Ltd. Electrical Steel Non Oriented Fully Processed, 2008.
- [41] GRANTA EduPack. Neodymium magnet N50M, 2019.
- [42] GRANTA EduPack. Soft magnetic alloy, 2.5 Si-Fe, free machining, 2019.
- [43] Il Sole 24 Ore. Mercati: Materie Prime, 2020.
- [44] A. Arkkio. *Analysis of induction motors based on the numerical solution of the magnetic field and circuit equations*. G4 monografiaväitöskirja, 12 1987.
- [45] A. Arkkio and E. Dlala. *FCSMEK, Part D: Reference Manual*. Helsinki University of Technology - Department of Electrical Engineering - Team of Electromechanics, 09.
- [46] A. Arkkio and E. Dlala. *FCSMEK, Part D: User's Guide*. Helsinki University of Technology - Department of Electrical Engineering - Team of Electromechanics, 09.



## A Errors in Surrogates Functions

All the three types of errors that are illustrated in A.3, A.1 and A.2 are employed in this work in order to find the discrepancy between the surrogate values and the actual value of the function being represented by the surrogates. The first two are adopted in a classical manner, while the third one needs to be observed in further detail as its formulation deviates from the conventional criteria. They are all computed referring to the optimal machines resulting in the final archive, which includes  $D_{ARC}$  optimum.

### A.1 Mean Absolute Error

The mean absolute error (MAE) measures the average magnitude of the errors in a set of evaluations, without considering their direction. It's the average over the test samples of the absolute differences between surrogate and actual function where all individual differences have equal weight. It is expressed by

$$MAE_i = \frac{1}{D_{ARC}} \sum_{i=1}^{D_{ARC}} |f_i - f_{i,s}|, \quad (A1)$$

where  $f_i$  is the actual  $i^{th}$  function and  $f_{i,s}$  is its surrogate.

### A.2 Mean Relative Error

The mean relative error (MRE) measures the relative magnitude of the errors in a set of evaluations, without considering their direction. It's the average over the test samples of the relative differences between surrogate and actual function where all individual differences have equal weight. It is expressed by

$$MRE_i = \frac{1}{D_{ARC}} \sum_{i=1}^{D_{ARC}} \left| \frac{f_i - f_{i,s}}{f_i} \right|. \quad (A2)$$

### A.3 Standard deviation

The standard deviation (STD) is a statistic concept and is a measure of the amount of variation (or dispersion) of a data set over a reference, such as the arithmetic mean or their estimation. In this work we are more concerned with the second concept, as we are interested in the measure of dispersion between the actual function value and the surrogate value of each sample. The STD is given in the same units as the function which is considered and it is computed as

$$STD_i = \sqrt{\frac{1}{D_{ARC}} \sum_{i=1}^{D_{ARC}} (f_i - f_{i,s})^2}. \quad (A3)$$

## B Relative Hypervolume

The concept of relative hypervolume turned out to be helpful in this work to quantify the size of clusters. First of all, let's clarify when and why considering the relative one might be the only suitable choice. It becomes necessary to employ the relative quantification if not all the employed optimization variables are described by the same physical quantity. Apart from that, it also provides a more practical relationship between the hyperspace being occupied by one cluster and the entire design hyperspace. As first approach, it is fundamental to realize that if the boundaries of a subdomain coincide with those of the entire design space, its relative hypervolume would be 1. Dealing with  $n$  design variables, it can be easily computed as:

$$hypervolume_j = \left( \frac{x_{1U,j} - x_{1L,j}}{x_{1U} - x_{1L}} \right) \cdot \left( \frac{x_{2U,j} - x_{2L,j}}{x_{2U} - x_{2L}} \right) \cdot \dots \cdot \left( \frac{x_{nU,j} - x_{nL,j}}{x_{nU} - x_{nL}} \right), \quad (\text{B1})$$

where  $x_{iU,j}$  and  $x_{iL,j}$  are the upper and the lower boundaries of the  $i^{th}$  design variable in the  $j^{th}$  cluster, while  $x_{iU}$   $x_{iL}$  are the boundaries of the same variable in the entire design space.