UNIVERSITA' DELLA CALABRIA

Dipartimento di Ingegneria Meccanica, Energetica e Gestionale – DIMEG

Scuola di Dottorato

"Pitagora"

Dottorato di Ricerca in Ingegneria Meccanica

CICLO

XXVIII

Titolo della tesi

Process Design Optimization Based on Metamodeling and Metaheuristic Techniques

Settore scientifico disciplinare: MAT/09 – Ricerca Operativa

Coordinatore: Prof. LEONARDO PAGNOTTA

Supervisore: Prof.ssa GIUSEPPINA AMBROGIO

Dottorando: Dott. CLAUDIO CIANCIO

Anni 2012/2015

Contents

List of Figures **ix**

List of Tables **xi**

1 Introduction **1**
- 1.1 Research statement . . . 2
- 1.2 Scope of this thesis . . . 2
- 1.3 Thesis structure . . . 3

2 Machine Learning Techniques for Manufacturing Applications **5**
- 2.1 Introduction . . . 5
  - 2.1.1 Kriging . . . 7
  - 2.1.2 Response Surface Methodology (RSM) . . . 8
  - 2.1.3 Artificial Neural Network (ANN) . . . 9
  - 2.1.4 NeuroFuzzy . . . 11
  - 2.1.5 Support Vector Regression . . . 12
  - 2.1.6 Comparison between experimental model/optimisation techniques . . . 14
- 2.2 Design of a high performance predictive tool for forging operation 15
  - 2.2.1 Computational test . . . 16
  - 2.2.2 Conclusions . . . 18

3 Heuristic Techniques to Optimize Neural Networks in Manufacturing Applications **21**
- 3.1 Neural network application on manufacturing processes . . . 22
- 3.2 Neural Network Architecture Optimization . . . 23

3.3 Selection of the neural network architecture 25
3.3.1 Parameters Description 25
3.3.2 Genetic Algorithm 28
3.3.2.1 Representation of the neural network 29
3.3.2.2 Selection 29
3.3.2.3 Crossover and mutation 30
3.3.2.4 Final Solution 30
3.3.3 Tabu Search 31
3.3.4 Taguchi Method 33
3.3.5 Decision Trees 34
3.4 Additional heuristic approaches to optimize Neural Network accuracy 35
3.4.1 Backpropagation Limits 36
3.4.2 Neural Network Initialization 36
3.4.3 Simulated Annealing 40
3.5 Proposed Algorithm 42
3.6 Computational Tests 43
3.6.1 Numerical Simulations 43
3.6.2 Optimization techniques Initialization 45
3.6.3 Extrusion 46
3.6.4 Rolling 48
3.6.5 Shearing 49
3.6.6 Robustness analysis 51
3.7 Conclusions 51

4 Metamodelling-based process design for homogeneous thickness distribution: an application to Incremental Sheet Forming 53
4.1 Introduction 53
4.2 Methodology 54
4.3 Application to Incremental Sheet Forming problem 55
4.3.1 Experimental Campaign 56
4.3.2 Process Optimization 59
4.4 Discussion of the results 60
4.5 Conclusion 61

5 Manufacturing Processes Modeling: an Application to Temperature Prediction on Incremental Sheet Forming 63
5.1 Introduction . . . 63
5.2 Mathematical Model . . . 64
5.3 The Experimental Campaign . . . 69
5.4 Model Applications . . . 72
5.4.1 Model Tuning . . . 73
5.4.2 Verification and Validation . . . 74
5.5 Conclusions . . . 78

6 Adaptive KPIs prediction for manufacturing processes: an application to remote laser welding 81
6.1 Methodology . . . 82
6.1.1 Definition of the input variables . . . 84
6.1.2 Sampling strategy starting problems . . . 84
6.1.3 Define lower and upper process window . . . 85
6.1.4 DoE Phase 1 . . . 86
6.1.5 Surrogate Model Preliminary analysis . . . 86
6.1.6 Surrogate Model . . . 87
6.1.6.1 Model Optimisation . . . 88
6.1.7 DoE Phase 2 . . . 89
6.1.8 Stopping criteria . . . 90
6.1.9 Multi Objective Optimisation . . . 90
6.2 Case Study: Remote laser welding . . . 91
6.2.1 Literature Review . . . 94
6.3 Experimental Test and Results . . . 96
6.3.1 Definition of input Variables . . . 96
6.3.2 Sampling strategy starting problems . . . 97
6.3.3 Define lower and upper process window . . . 98
6.3.4 DoE Phase 1 . . . 99
6.3.5 Surrogate model preliminary analysis . . . 100
6.3.6 Surrogate model . . . 101
6.3.7 DoE Phase 2 . . . 101

6.3.8 Stopping criteria . . . 103
6.3.9 Multi-Objective analysis . . . 103
6.3.10 Verification and Validation . . . 104
6.4 Conclusion . . . 107

7 Multi-Objective Optimization 109
7.1 Introduction . . . 109
7.2 Multi-Objective problem optimization . . . 110
7.3 Benchmark techniques . . . 111
7.3.1 Weighted sum method . . . 111
7.3.2 ϵ-Constraint method . . . 112
7.4 Method . . . 112
7.4.1 GDE3 Algorithm . . . 114
7.5 Case Study . . . 118
7.5.1 Porthole Die Extrusion . . . 118
7.5.2 Analysis of the results . . . 121
7.6 Conclusion . . . 123

8 Conclusions 125

References 129

List of Figures

2.1 NeuralNetwork 10
2.2 Support Vector Regression 12
2.3 Impression-die forging process 15
2.4 Impression-die forging: a) Maximum forging load at the end of the process b) Maximum load when the punch begins to compress the billet 17

3.1 Representation of the solution space S of a generic problem 31
3.2 Tabu search algorithm 32
3.3 Training error 36
3.4 Local and global minima 37
3.5 Training with a local minimum 37
3.6 Extrusion process 47
3.7 Rolling process 48
3.8 Shearing process 50
3.9 Machine learning approaches to optimize neural network architecture robustness 52

4.1 Incremental Sheet Forming: the investigated shapes and the analysed Lateral Section (LS) AA' and Diagonal Section (DS) BB'. . . 57
4.2 Incremental Sheet Forming: comparison between Single Slope and Decremental Slope strategy. 57
4.3 The comparison between experimental (EXP) and predicted (MOD) thickness. 62

5.1 Nomenclature Incremental Sheet Forming model. 65

LIST OF FIGURES

5.2 Flow chart Incremental Sheet Forming model. 66
5.3 Representation of the temperature prediction model. 66
5.4 Thermo-camera placed inside the lathe working volume (Incremental Forming). 71
5.5 Intermediate step of the test carried out on Ti6Al4V by using V = 50m/min and p= 0.3mm. 72
5.6 Thermal properties Ti6Al4V (Incremental Forming). 75
5.7 Numerical and experimental temperature trends for both the analysed materials. 76
5.8 Temperature trend for different tool velocity (maximum value reached in each point). 77
5.9 Temperature trend for different tool pitch (maximum value reached in each point). 77

6.1 Optimisation procedure (Remote Laser Welding) 83
6.2 ϵ-tube 90
6.3 S-value (Remote Laser Welding) 92
6.4 Penetration (Remote Laser Welding) 93
6.5 Top Surface Concavity (Remote Laser Welding) 93
6.6 Bottom Surface Concavity (Remote Laser Welding) 94
6.7 Remote Laser Welding: Process window 99
6.8 Remote Laser Welding: Efficient frontiers 104
6.9 Remote Laser Welding: Sensitivity analysis 106

7.1 Extrusion Perspective 119
7.2 Porthole extrusion process: I section 119
7.3 Porthole extrusion process: training error 121
7.4 Tensile test equipment 121
7.5 Porthole extrusion process: neural Network prediction 122
7.6 Porthole extrusion process: Efficient frontiers 123

List of Tables

2.1 Comparison of common machine learning model/optimisation techniques 14
2.2 Impression die forging: Average percentage error (energy) 17
2.3 Impression die forging: Average percentage error (load) 18

3.1 Heuristic techniques to optimize neural network architecture . . . 25
3.2 Gentic algorithm: chromosome encoding 29
3.3 Machine learning approaches to optimize neural network architecture initialization 45
3.4 Average percentage error (Extrusion original dataset) 46
3.5 Average percentage error (Extrusion noisy dataset) 47
3.6 Average percentage error (Rolling original dataset) 49
3.7 Average percentage error (Rolling noisy dataset) 49
3.8 Average percentage error (Rolling original dataset) 50
3.9 Average percentage error (Shearing noisy dataset) 50

4.1 Incremental Sheet Forming: Categorical Variables 58
4.2 Incremental Sheet Forming: Continuous Variables 58
4.3 Incremental Sheet Forming: Validation tests (Input data) 60
4.4 Incremental Sheet Forming: Validation tests (Output results Thinning minimization) 61
4.5 Incremental Sheet Forming: Validation tests (Output results Thinning minimization) 61

5.1 Incremental Sheet Forming: thermal properties of the analyzed materials 72

LIST OF TABLES

5.2 Incremental Sheet Forming: process set-up 73
5.3 Incremental Sheet Forming: flow Stress datasets 74
5.4 Incremental Sheet Forming: maximum temperature of the sheet (AA-5754) 75
5.5 Incremental Sheet Forming: maximum temperature of the sheet (Ti6Al4V) 75
5.6 Incremental Sheet Forming: statistical criteria to define the model accuracy. 76

6.2 Remote Laser Welding: process parameters 97
6.3 Remote Laser Welding: stack-up features 98
6.4 Remote Laser Welding: polynomial model p_3^{2kW} results 98
6.5 Remote Laser Welding: polynomial model p_3^{4kW} results 99
6.6 Remote Laser Welding: DoE Phase 1 100
6.7 Remote Laser Welding: surrogate model (Preliminary analysis) . . 101
6.8 Remote Laser Welding: surrogate model (Best Similarity Function Accuracy) 101
6.9 Remote Laser Welding: DoE Phase 2 102
6.10 Remote Laser Welding: Model Accuracy (Phase 2) 102
6.11 Remote Laser Welding: comparative analysis Test 1(minimum number of experiments) 105
6.12 Remote Laser Welding: comparative analysis Test2 (maximum accuracy) 105

8.1 Dissertation Contributions. 126

1

Introduction

Manufacturing companies have to constantly improve the process performances to comply with high competitive markets. Improving the performance of processes leads to enhancement of the quality of the final manufactured products. Major challenges involved in quality improvement can be enumerated into three categories:

1. identification of product (KPCs - Key Product Characteristics) to process (KCCs - Key Control Characteristics) relationships;

2. selection of most crucial parameters (and/or characteristics) affecting the overall quality of the final product;

3. optimization of process performances to achieve improved/optimized product quality.

Real-world manufacturing problems often contain nonlinearities, combinatorial relationships and uncertainties that are too complex to be modeled analytically. In these scenarios, the identification of product-to-process relationships becomes a complex task which involves integration of physical-driven (or even simulation-driven) experiments and data mining algorithms. This task is costly and time consuming. Traditional optimization methods have been unable to cope with real world problems due to the limitation of resources available to perform such a complex analysis. The aim of this thesis is to present benchmark and new methodologies to perform all three task previously described in a automated way

minimizing the necessary resources by integrating response surface models, sampling strategies and optimization algorithms. In order to prove the effectiveness of the proposed methodologies several industrial case studies on different manufacturing processes have been utilized.

1.1 Research statement

The research objective of this thesis is to provide methods that allow undersanding, control and optimize manufacturing processes by applying machine learning techniques.The goal of having deeper knowledge about processes is two-sided: to provide scientific knowledge, that is to "to develop laws and theories" that explain, predict, understand, and control phenomena" , and to enable practical applications to have a solid base to develop systems that efficiently support and control manufacturing processes. From the research objective, we derived the following main research questions:

1. What and how much data can be useful/necessary for modelling manufacturing processes?
2. What kind of processes can be modeled from past process executions?
3. Is it possible to extract process models from data?
4. How to determine which process input parameters affect the analyzed responce?
5. Is it possible to use the metamodel to define the optimal process configuration according to user need?

1.2 Scope of this thesis

In this thesis we show that machine learning techniques can be used successfully to understand a process on the basis of data, by means of a combined use of metamodel, sampling and optimization approaches. Several approaches have been developed in the past to cope with similar problems but each technique presented

is characterized by strengths and weaknesses that limits its applicability. The scope of this thesis is to provide to the readers a procedure to understand which factors have to be analyzed and how problem peculiarities can be used to develop customized techniques that produce better results. Different features have to be taken into account before use/develop a suitable methodology:

- Number of available data;
- Input factors typology (discrete, continuous);
- Number of output parameter (global optimization, multi-objective optimization);
- Data reliability;
- Analysis required (response surface, feature selection, clustering, optimization).

Finally, through the applications on different case studies, we want to show that the results achieved using our methods are significantly better than benchmark approaches.

1.3 Thesis structure

The outline of the thesis is the following: Chapter 2 provides a review and a comparative analysis of the most used machine learning approaches in manufacturing applications. Chapter 3 presents several heuristic techniques to optimize the prediction accuracy of neural network. A kriging based metamodel to solve problem with continuous and discrete variables is presented in Chapter 4. An alternative approach based on the use of data mining techniques that does not require experimental data is proposed in Chapter 5. This method was used to estimate the temperature variation of the sheet during an Incremental Sheet Forming process (ISF). A new metamodel techniques based on geometrical projection of responces surface through similarity function is presente in Chapter 6. Multi objective algorithm based on metaheuristic techniques are finally described in Chapter 7. Two

different applications are reported showing the benefits achievable compared to traditional formulations.

2

Machine Learning Techniques for Manufacturing Applications

2.1 Introduction

The fundamental idea of the metamodeling concept is to find an empirical approximation model, which can be used to describe the typically unknown relation between input variables and response values of a process or product. To adjust the chosen analytical formulation to the problem under investigation, the original response values are evaluated at some selected input variable settings, the so-called sampling points. Based on these input-output pairs, free parameters in the model formulation are fit to approximate the original (training) data in a best possible way. The approximation models can then be used to predict the behavior of the original system at "untried" input variable settings. Originally, the primary application for this technique was the analysis of physical experiments. In the optimization context, approximations either allow for the analytical determination of the approximate optimum or they replace the original functions within one or in the course of several iteration steps. The use of metamodel techniques can play a role to solve the following problems:

- Model approximation. Approximation of computation-intensive processes across the entire design space, or global approximation, is used to reduce computation costs.

- Design space exploration. The design space is explored to enhance the engineers' understanding of the design problem by working on a cheap-to-run metamodel.

- Problem formulation. Based on an enhanced understanding of a design optimization problem, the number and search range of design variables may be reduced; certain ineffective constraints may be removed; a single objective optimization problem may be changed to a multi-objective optimization problem or vice versa. Metamodel can assist the formulation of an optimization problem that is easier to solve or more accurate than otherwise.

- Optimization support. Industry has various optimization needs, e.g., global optimization, multi-objective optimization, multidisciplinary design optimization, probabilistic optimization, and so on. Each type of optimization has its own challenges. Metamodeling can be applied and integrated to solve various types of optimization problems that involve computation-intensive functions.

A number of experimental techniques are currently used in manufacturing applications. These are based on mathematical models that help to understand the correlations between the process parameters and quality measurements of performance. Traditionally, regression and response surface methods have been two of the most common metamodeling approaches. In recent years, artificial neural networks (ANN) have gained increased popularity, as this technique requires fewer assumptions and less precise information about the systems being modeled. In the last two decades, Design of Experiments (DoE) has been the preferred technique, although evolutionary algorithms and computational network have also gained importance. Several authors have used these and other techniques in different manufacturing processes. The purpose of this chapter is to present a comprehensive review of the literature that underpins the methodologies proposed in this dissertation.

2.1.1 Kriging

The kriging interpolation technique is quite different from the classical interpolation approaches (1). In classical interpolation, the data are assumed to be sampled from a function, that is reconstructed from the data under some assumption on the nature of the interpolating function $\hat{z}$. Typically, for classical interpolation it is assumed that the function $\hat{z}$ is polynomial. In general, interpolation algorithms (inverse distance squared, splines, radial basis functions, triangulation, etc.) estimate the value at a given location as a weighted sum of data values at surrounding locations. Almost all assign weights according to functions that give a decreasing weight with increasing separation distance. Kriging assigns weights according to a (moderately) data-driven weighting function (2). Consider N points $x_i, \;\; i = 1, ..., N$ in the vector space R^d. At these locations, data $z_i \;\; i = 1, ..., N$ are assumed to be known. These data are interpreted as the values of a field z, whose value depends on the position in space. In general, the points x_i will be scattered disorderly in space, rather than aligned on a regular grid. Furthermore, the data are assumed to be affected by some uncertainty, due either to measurement error, or to the fact that the quantity z is dependent on some unpredictable physical process, or both. Given the N points $x_i, \;\; i = 1, ..., N$ and the uncertain data $z_i \;\; i = 1, ..., N$, the kriging interpolation technique attempt to:

- predicting the most appropriate value z_0 for the quantity z at a point x_0, different from the points associated to the available data;
- estimating the uncertainty of the prediction z_0 as a function of the uncertainty on the available data $z_i \; i = 1, ..., N$ and of their correlation structure.

The basic form of the kriging estimator is the following:

$$z^*(x) - m(x) = \sum_{\alpha=1}^{n(x)} \lambda\alpha[z(x_\alpha) - z(x_\alpha)] \tag{2.1}$$

with:

- x, x_α: location vectors for estimation point and one of the neighboring data points, indexed by α

- $n(x)$: number of data points in local neighborhood used for estimation of $(z^*(x))$
- $m(x)$ $m(x_\alpha)$ expected values of $z(x)$ and $z(x_\alpha)$
- $\lambda_a(x)$ kriging weight assigned to datum $z(x_\alpha)$ for estimation location x; same datum will receive different weight for different estimation location
- $z(x)$ is treated as a random field with a trend component, $m(x)$, and a residual component, $R(x) = z(x) - m(x)$. Kriging estimates residual at x as weighted sum of residuals at surrounding data points. Kriging weights, λ_a , are derived from covariance function or semivariogram, which should characterize residual component.

The goal is to determine weights, λ_α , that minimize the variance of the estimator. Some advantages of the kriging metamodel technique are:

- Helps to compensate for the effects of data clustering, assigning individual points within a cluster less weight than isolated data points (or, treating clusters more like single points)
- Gives estimate of estimation error (kriging variance), along with estimate of the variable, z, itself
- Availability of estimation error provides basis for stochastic simulation of possible realizations of $z(x)$

2.1.2 Response Surface Methodology (RSM)

Response surface methodology is a technique (also referred as meta-modelling or surrogate function) that is used to acquire relevant information from experiments that are expensive or take a great amount of time, with the purpose of determining functional relationships between the input parameters and the responses of an experiment (3). In many different situations the relationship between input and output of a process is not known exactly but can be estimated using a model of the form $y = f(x_1, x_2, ..., x_n) + \varepsilon$ where ε represents the error in the system. Usually the function f is a first-order or a second-order polynomial model. If

the output variable can be defined using a linear function of input variables the approximating function is a first order model. A first order model with n different input variable can be written as:

$$y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \varepsilon_i \tag{2.2}$$

A second order model can be written, instead, as:

$$y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \sum_{i=1}^{n} \sum_{j=1}^{n} \beta_{ij} x_i x_j + \sum_{i=1}^{n} \gamma_i x_i^2 + \varepsilon_i \tag{2.3}$$

Models with a higher degree are not recommendable because they have a high overfitting probability. Moreover even in the second order model usually not all the terms are necessary. One criterion that is often used to find the optimal value of the parameters in 2.2 and 2.3 is the minimization of the least square error. This method chooses the β vector so that the sum of the squares of the errors ε_i is minimized. The least square function is therefore determined using the following equation:

$$L = \sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{k} \beta_j x_{i,j} \right)^2 \tag{2.4}$$

To measure the robustness of the model different test can be performed (4). However these test are valid under the hypothesis that errors are normally and independently distributed with zero mean and variance σ^2. The methods which are usually used for this purpose are the analysis of variance (ANOVA), the sum of squares and the F-test. One crucial aspect to define the most accurate function is given by the design o the experiment (DOE). There are different ways to collect the required information, in order to create the model choosing the most relevant data. This means that there is not a unique technique to explore the design space but many different options known as space filling strategies.

2.1.3 Artificial Neural Network (ANN)

Neural networks are powerful tools for classification and regression problems (5). A neural network based on backpropagation is a multilayered architecture made

up of one or more hidden layers placed between the input and the output layer (6). The performances of a neural network are affected by many factors such as the network architecture, the type of activation function and the training algorithm.

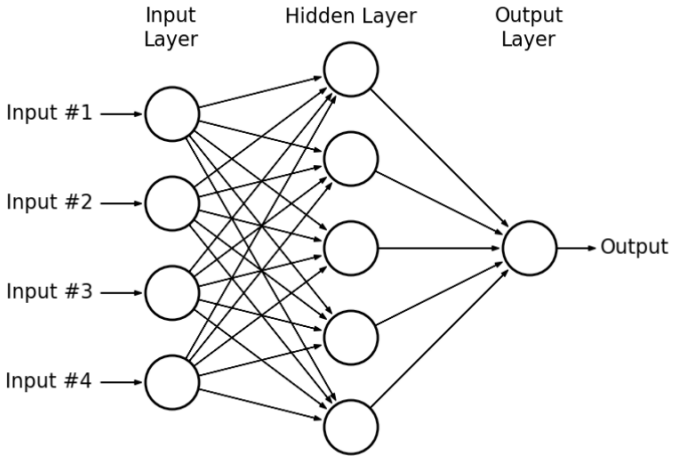


Figure 2.1: NeuralNetwork

In the simplest form the first step of the back-propagation algorithm is to calculate the value of the error function E with the initial weight matrix and its gradient.

$$\frac{\partial E}{\partial W} = \left[\frac{\partial E}{\partial w_{1,1}^{1}} \; \frac{\partial E}{\partial w_{1,2}^{1}} \; \cdots \; \frac{\partial E}{\partial w_{N_{z-1},N_z}^{z-1}} \right] \tag{2.5}$$

The error is then back-propagated to the output and the hidden layer for several iterations and the new weights of this layer are calculated. The idea of the algorithm is to distribute the error function across the output and hidden layers corresponding to their effect on the output. In the generic k-th iteration the value of the weights will be updated in the following way:

$$w_{i,j}^{l}(k+1) = w_{i,j}^{l}(k) - \gamma \frac{\partial E}{\partial w_{i,j}^{l}} + \mu(w_{i,j}^{l}(k) - w_{i,j}^{l}(k-1)) \tag{2.6}$$

$$b_{i}^{l}(k+1) = b_{i}^{l}(k) - \gamma \frac{\partial E}{\partial b_{i}^{l}} + \mu(b_{i}^{l}(k) - b_{i}^{l}(k-1)) \tag{2.7}$$

It is crucial to identify a suitable value for the learning rate since it can affect significantly the time to get convergence. If a small value is chosen the number

of iterations required to find a good solution will be generally greater. However, a large value of this parameter is usually related with higher perturbation in the error function during the training. The momentum parameter is instead usually introduced to reduce oscillation in the error function during the iterations (7). One of the main disadvantages of the back-propagation algorithm to train feed-forward neural network is the elevated number of iterations required to find a good solution. This drawback can be overcome using heuristic approaches. A more in depth discussion about these techniques will be presented in Chapter 3.

2.1.4 NeuroFuzzy

Fuzzy logic is a form of many-valued logic or probabilistic logic. It deals with reasoning that is approximate rather than fixed and exact (8). Compared to traditional binary sets, fuzzy logic variables may have a truth value that ranges in degree between 0 and 1. Fuzzy logic has been extended to handle the concept of partial truth, where the truth value may range between completely true and completely false. According to Zadeh the term *Fuzzy logic* has two different meanings. In a "*narrow sense*" fuzzy logic is a logical system which aims at a formalization of approximate reasoning. In its "*wide sense*" is a synonymous of the fuzzy set theory, which is the theory of classes with unsharp boundaries (9). In the past few years, various neuro-fuzzy systems have been developed. The neuro-fuzzy is a technique that combines the natural language description of fuzzy systems and the learning properties of a neural network. Neural networks are powerful tool for recognizing patterns but they are not good to explaining how they reach their decisions. Fuzzy logic systems, which can reason with imprecise information, are good at explaining their decisions but they cannot automatically acquire the rules. Hybrid systems that combine fuzzy logic and neural networks have proved their effectiveness in a wide variety of real-world problems. Neural fuzzy systems use neural networks to produce a fuzzy system. In the training process, a neural network adjusts its weights in order to minimize the mean square error between the output of the network and the desired output. The weights of the neural network represent the parameters of the fuzzification function, the fuzzy rules and the defuzzification function. Several fuzzy sets are used to represent

linguistic concepts such as low, medium or high and are often employed to define states of a variable. Such a variable is usually called a linguistic variable. Fuzzy logic can encode directly expert knowledge using rules with linguistic variables. However this process is usually time consuming and require a long trial and error approach to tune the membership functions which quantitatively define these linguistic variable. In the Neuro-Fuzzy the use of a neural network guarantees the automation of this process reducing the computational time and improving the performance (10).

2.1.5 Support Vector Regression

Support vector machines are analytical tools that can be used for both classification and regression but their use with kernel is often time consuming (11). In fact, classical kernel based algorithms typically have memory and computational requirements of $O(N^2)$. The basic idea behind constructing nonlinear SVR is to map the training data from the original space into a higher dimensional space called feature space, and compute an optimal linear regression function in this feature space.

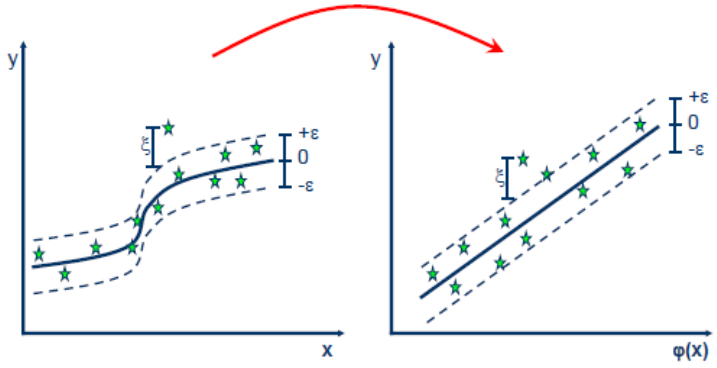


Figure 2.2: Support Vector Regression

Given a set of training data, a linear SVR tries to find a model, so that the quantity $w^t x_i + b$ (predicted value of the configuration i) is close enough to the target

value y_i for all the training data. The optimal value of w^T is found by solving the following optimization model:

$$\min_T f(w) = \frac{1}{2} w^T w \tag{2.8}$$

$$\text{subject to} |w^T x_i + b - y_i| \leq \epsilon \tag{2.9}$$

The assumption of this problem is the existence of a function f that approximates all pairs (x_i, y_i). However, sometimes this problem could be infeasible. Therefore in some cases it is necessary to use a different formulation of this problem that allows errors larger than ϵ:

$$\min_T f(w) = \frac{1}{2} w^T w + C \sum_{i=1}^{n} \xi_\epsilon(w; b; x_i, y_i) \tag{2.10}$$

where C is a penalty coefficient and $\xi_\epsilon(w; b; x_i, y_i)$ is called ϵ-intensive loss function and its value is equal to:

$$\xi_\epsilon(w; b; x_i, y_i) = \max\left(|w^T x_i + b - y_i| - \epsilon, 0\right). \tag{2.11}$$

In most cases this optimization problem can be solved more easily in its dual formulation using a Kernel function (12). The main advantage of this technique is that it can be used to solve nonlinear problems preprocessing the training data x_i by a map K into a features space $\Im(X \to \Im)$ in which is possible to use the standard SVR algorithm. The dual formulation of the problem is reported as follows

$$\max_{\alpha, \alpha^*} f(\alpha) = \frac{1}{2}(\alpha - \alpha^*)^T Q((\alpha - \alpha^*) + \epsilon \sum_{i=1}^{T_r}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{T_r} z_i(\alpha_i - \alpha_i^*) \tag{2.12}$$

$$\text{subject to: } e^T(\alpha - \alpha^*) = 0 \tag{2.13}$$

$$0 \leq \alpha_i, \alpha_i^* \leq C \qquad i = 1, ..., T_r \tag{2.14}$$

where $Q_{ij} = K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$

2.1.6 Comparison between experimental model/optimisation techniques

After reviewing different approaches it is possible to conclude that there is remarkable interest in optimisation techniques to predict responses and select the optimal process parameters. RSM was found to perform better than all the other techniques in cases where it is not possible to conduct a great amount of experiments, which is generally the case in industry. However, in some cases this tool is not the most appropriate to map highly not-linear product-to-process relationships. Table 2.1 shows strength and weaknesses of the previous analyzed approaches. It is clear that does not exist one approach that appears to be better

Table 2.1: Comparison of common machine learning model/optimisation techniques

	ANNs	**GA**	**RSM**	**Kriging**	**SVR**
Computational time	Long	Very long	Short	Medium	Short
Experimental domain	Regular or irregular	Regular or irregular	Regular only	Regular or irregular	Regular only
Model developing	Yes	No	Yes	Yes	Yes
Optimisation	Through model	Straight	Through model	Straight	Through model
Understanding	Moderate	Difficult	Easy	Normal	Easy
Availability in software	Available	Available	Available	Available	Available
Optimisation accuracy	High	High	Very High	Normal	Very High
Application	Frequently	Rarely	Frequently	Frequently	Frequently

than all the others under all the point of views considered in the table. This is the main reason why all these approaches are still used by many researchers in different applications. However, to maximize the information achieved form the available data it is necessary to choose the most suitable technique based on the peculiarity of the problem under analysis. The next section presents a forging applications in which these techniques have been used to define a response surface.

2.2 Design of a high performance predictive tool for forging operation

The forging process is a massive forming process, characterised by the application of high compressive load which generates plastic strain of the billet (13). This process is used to manufacture crankshafts, connecting rods, gears, turbine blades, disks and other components for mechanical industry. In this study, the attention is focused on the impression-die forging variant (14) 2.3.

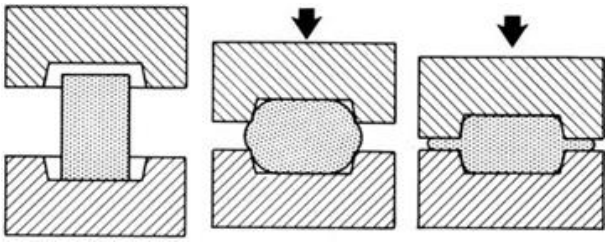


Figure 2.3: Impression-die forging process

In this process the billet assumes the dies cavity shape, due to the compressive stresses on the billet during the closing phase. At the end of this phase, the dies are not mutually in contact; consequently, part of the material may flow radially outwardly of the cavity, forming flash. This flow must be hampered to guarantee the complete dies cavity filling and to obtain the desired shape. The friction at the die-workpiece interface plays a fundamental role in the final quality of the forged part. The design of the flash channel typically requires that the gap between the dies is 3% of the maximum thickness of the forged part. The flash channel length is five times larger than its thickness (15).Other important parameters are the radii of the die that must be large enough to facilitate the plastic flow of the material avoiding stress concentration that may reduce the useful life of the tools. The impression-die forging processes are aimed at the production of components with complex geometric shapes. This process still appears as one of the most applicable processing methods in the machine building industry due to the high quality of the forging parts and low production costs. Several studies have been executed to investigate and validate the Finite Element Method

(FEM) capability as predictive tool for the considered process (16, 17). Starting by this result, the FEM analysis was used for the study purposes. The model is implemented by a commercial finite element code, DEFORM 2D. Naturally, implementation of periodic remesh to produce accurate plasticity solutions and good convergence is set into the model. The simulations will be axisymmetric exploiting the shape of the investigated geometry. The software package DEFORM 2D is applied here aimed at improving the process and billet design that ensure the complete dies cavity filling minimizing flash waste, energy and material consumption. In the specific case the manufactured component is a flywheel, which results an industrially interesting case.

2.2.1 Computational test

This section shows the results of the previously described algorithms for a forging process. Two different tests were carried out. In the first test only the billet dimensions were considered as input variables while in the second case also the friction between the billet and the molds was taken into account. For the first test a dataset of 25 input-output pairs was used. These data were collected through numerical simulations. The volume of the raw material is increased of 1% with respect to the 3D profile. The shape of the initial billet is a hollow cylinder and the parameters to optimize are the billet dimensions: inner diameter, outer diameter and height. In the second test-set, a variation of the friction coefficient is considered. The simulations are axisymmetric exploiting the shape of the investigated geometry. The height of the flash channel is 1.8mm and its length is 6.5mm. The material considered is a steel alloy, DIN-C35. The initial temperature of the billet is 1100C and the dies temperature is 1000C. The temperature is constant and the punch speed is 1mm/s. Constant-shear model is applied as boundary condition and a friction coefficient equal to 0.1 is fixed, according to the range quoted by Kalpakjian for hot forging (18). First of all, a series of tests were performed to identify the ranges of the geometrical dimensions and to define the correct workability area. The following ranges were identified for the investigate dimensions:

- internal diameter 40-80mm;

- external diameter 180-260mm

The height is calculated as a function of the previous two parameters. The proper shape volume has been determined with software Pro-Engineer. The two analyzed output variables are the maximum load during the process and the total energy absorbed. The analysis of the results show that the maximum load reached increases during the process. In most cases, maximum load occurred at the end of the process but in other cases it occurs when the punch begins to compress the billet. The dataset is split in two parts: 20 pairs are used as training set

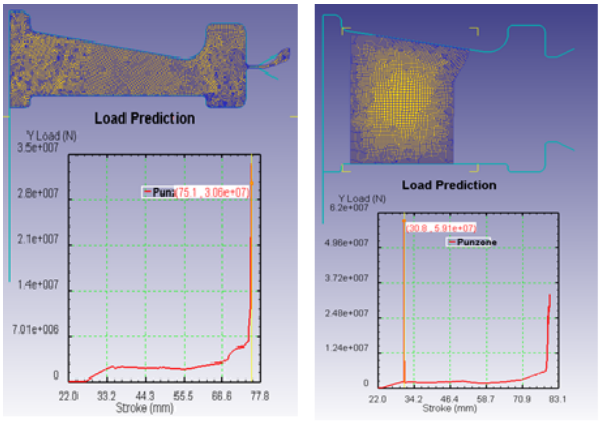


Figure 2.4: a) Maximum forging load at the end of the process b) Maximum load when the punch begins to compress the billet

and 5 as test set. Furthermore, in order to verify the real performance of these techniques 5 input-output pairs are used as validation set. For each technique has been calculated the average percentage error for training, test and validation test.

Table 2.2: Average percentage error (energy)

	Kriging	**ANN**	**NNFuzzy**	**RSM**	**SVR**
Training set	1.85%	4.21%	2.73%	3.20%	3.10%
Test set	2.83%	3.11%	3.25%	1.28%	1.36%
Validation set	1.23%	3.47%	1.97%	2.04%	1.67%

Table 2.3: Average percentage error (load)

	Kriging	**ANN**	**NNFuzzy**	**RSM**	**SVR**
Training set	0.99%	1.12%	1.81%	1.99%	1.96%
Test set	1.98%	1.98%	3.28%	1.78%	1.56%
Validation set	2.12%	1.71%	0.53%	0.62%	0.28%

As shown in the tables, the best performances are obtained using the GA-NN, the SVR and the NN-Fuzzy techniques. However the average percentage error is less than 5% for each illustrated technique. Not all the input configurations are feasible according to quality constraints. The construction defects can be classified as microcrack and macrocrack, depending on the position of the billet on which they occur. The microcracks are welding defects that occur on the upper side of the material; the macrocracks are bending and welding defects that occur on the lateral side of the material when the dies are approaching. To identify the set of feasible solutions the Matlab library LIBSVM has been used. The probability to obtain a feasible solution (class 1) is high for configurations with small inner diameter and high outer diameter.

2.2.2 Conclusions

This chapter provided a quick introduction of the most used machine learning techniques for manufacturing application. The approaches discussed are respectively the kriging metamodel, the response surface method, the artificial neural network, the fuzzy logic and the support vector regression. The chapter start with a brief discussion of strengths and weakness of each methodology trying to highlight the situation in which the use of each approach is suggested. The chapter ended with an application on a case study in which 2 output responses (energy and load) have been controlled for a forging process analyzed through 2D numerical simulations. As expected, the maximum load reached during the process depends on the billets size. The load in fact increases when the height of the billet is small. Despite, the energy reduces his value in the same conditions. The change in friction factor affected the maximum load in the impression die-forging

process. This was because during the flash formation, which can be thought of as extrusion of metal through a constriction that circumscribes the workpiece, the dominant mechanical process is friction. All the artificial intelligence techniques described have shown good results and can therefore be a useful tool to assist business users to optimize a mechanical process. The best results have been achieved using an Artificial Neural Network. However, in the last years have been pointed out that the neural network performance can be significantly improved through the used of heuristic and meta-heuristic techniques to tune the network architecture and improve the learning algorithm. The use of these approaches will be more accurately discussed in Chapter 3.

2. MACHINE LEARNING TECHNIQUES FOR MANUFACTURING APPLICATIONS

3

Heuristic Techniques to Optimize Neural Networks in Manufacturing Applications

Neural networks have been often used in manufacturing applications as the use of this technique is a useful tool for the decision maker due to the reduced efforts to acquire the knowledge required to make the correct process decisions. Many manufacturing companies execute the process analysis by using an hybrid approach based on numerical simulation (19) and a prediction tool. The use of this approach offers several benefits to the company due to the reduced number of experimental tests and the possibility to analyze the process behaviour under different process conditions (material, equipment, etc).

Neural networks learn by examples. A set of data are given to the network and then the training algorithm changes the network's weights so that, when training is finished, the networks returns the desired output for a given input. In the last decades, feed-forward neural networks have become a standard technique for regression problems (5). Backpropagation is the most used algorithm for training neural networks (6). The main disadvantages of this algorithm are the possibility of being trapped in local minima and the elevated time required to get convergence. In some cases, researches have used the Levemberg-Marquardt method to train the multilayer perceptron because has better convergence properties but it always require calculations of order $O(N^2)$ (20). To overcome this disadvantage

many variants of the backpropagation algorithm have been developed in the last years using new techniques to set some parameters such as momentum and learning rate. However it is not possible to know accurately the optimal value of these parameters, so in order to find a neural network with good prediction abilities usually different configurations must be tried. In this work different heuristics to improve the performance of a feed-forward neural network will be presented. The performance improvement obtained with these techniques have been measured on different datasets trying in this way to generalize the results. The prediction abilities of the neural networks were measured using the *MSE* (Mean Square Error) criterion (21). The remainder of this chapter is organized as follows: in the first section it is show the current state of the art of neural network application in the manufacturing field; subsequently the learning algorithms used to compare the quality of the proposed approach will be briefly presented highlighting the state of the art and the problems related to their use. Heuristic techniques based on architecture selection, weight initialization, combination of training algorithms, and manipulations of training data will be also analyzed. Finally, in the last section computational results on three case studies, in which different processes have been mapped using numerical 2D simulations, are reported and discussed.

3.1 Neural network application on manufacturing processes

Neural networks have been widely introduced to the analysis of increasing number of problems in the manufacturing field (22). Different applications are reported in literature for both strategic and operational issues. Yang (23) proposed a NN metamodeling method to generate the cycle time - throughput for single/multi-product manufacturing environments. In his work a progressive model-fitting strategy was developed to obtain the simplest-structured NN that is adequate to capture the required relationships. Unal et al (24) used a genetic algorithm optimized neural network for fault diagnosis of rolling bearings. Their results show that the optimized NN is able to reach a better accuracy compared to one with a reference NN architecture. Venkatesan et al. (25) implemented also the

combined use of genetic algorithm and NN to optimize the weight initialization phase; more in particular, in their approach genetic algorithm is used as a complementary tool for weights optimization that serves in turn for NN to perform robust prediction of machining process performance. Always for machining process, Yao and Fang (26) used NN to predict the development of chip breakability and surface finish at various tool wear states. Moving from machining processes to forming operations, NNs result widely utilized to predict both quantitative and qualitative variables: Ambrogio and Gagliardi (27) designed a customized toolbox combining NN and the Design of Experiments (DoE) for predicting opposite performance in porthole die extrusion process; Ozcan and Figlali (28) proposed the NN for estimating the cost of stamping dies, as alternative and only way to analytical and conventional methods; finally, Saberi and Yussuff (29) introduced NN to evaluate advanced manufacturing technologies implementation outcomes and predict company performance as high, low, or poor in technology adoption. Even if many successfully applications of NN in the manufacturing field are reported in the literature, the possible benefits deriving from the use of this tool are not always completely achieved due to the use of random network parameters or trial and error approaches.

3.2 Neural Network Architecture Optimization

Neural networks consist of some components such as architecture structure (30), activation functions of each layer and training algorithm. These components can greatly affect the performance of a neural network and even if several approaches have been proposed in recent years, the process of selecting an adequate value of these parameters is still a controversial issue. The interest towards algorithms able to optimize NN performance is significantly increased in the last years. However, the results of the different proposed techniques are often contradictory and do not provide a clear vision of the benefits and drawbacks of using such approaches. Initially, the focus of the research was mainly on how to determine the number of neurons of a neural network with one hidden layer; nevertheless these methods did not take into account the features of the dataset but only its size (31). As example, Baum et al. (32) obtained some bounds of the number of

neurons, for a network with a linear activation function, related to the number of training example. Camargo et al. (33) solved the same problem implementing a function using Chebyshev polynomial. Wangn and Shao (34) proposed an architecture selection algorithm based on an extended localized generalization error model which considers the generalization capability and costs less time than cross validation. Another method similar to the genetic algorithm approach was proposed by Lirov (35) and is often used as a reference point for new techniques. The state of the art about the influence of the architecture on the generalization abilities is in agreement with the result that the number of optimal nodes in a neural architecture for obtaining valid generalization depends mainly on two factors (36): the number of available training samples and the complexity of the function. A different approach often used to select the features of the network is related to the use of information criteria like the Aikake's information criterion (AIC) and the Bayesian's information criterion (BIC). These two criteria can be expressed through the following equation:

$$AIC = \log(\hat{\sigma}^2_{MLE}) + \frac{2m}{T} \tag{3.1}$$

$$BIC = \log(\hat{\sigma}^2_{MLE}) + \frac{m^d \log(T)}{T} \tag{3.2}$$

$$\hat{\sigma}^2_{MLE} = \frac{SSE}{T} \tag{3.3}$$

where $\hat{\sigma}^2_{MLE}$ is the maximum likelihood estimate of the variance of the residual term, m and T are respectively the number of parameters in the model and the number of data available. Finally d is a fixed constant usually greater than 1. As example, Berger (37) and Bernardo and Smith (38) used an approach in which the neural network architecture is selected based on the maximization of the expected utility criterion. Also in these approaches, however, the technique is able to give only some indications regarding the number of units in the hidden layers. Finally, the literature overview clearly stated a lack of homogeneity for the advantages and disadvantages of different techniques, as well as it is not possible to identify the most promising NN research approach for manufacturing problems, typically characterized by data insufficiency. In the next section four different approaches

to optimize NN architecture based on the use of genetic algorithm, tabu search, taguchi and decision trees will be discussed. In general, all the techniques used in this work can provide some insights regard the interactions between the considered network parameters. At the same way all these approaches have some drawbacks that discourage their use under particular conditions. The next table summarize the main property of these algorithm highlighting their strength and weakness.

Table 3.1: Heuristic techniques to optimize neural network architecture

Optimization Technique	**Advantages**	**Disadvantages**	**References**
Genetic Algorithm	High optimization accuracy. Good ability to fit non-linear neural network features interactions.	Extensive computational time required. Low model understanding	(39, 40, 41, 42, 43).
Tabu Search	Exhaustive search of the solutions space. Ability to escape from local minima.	Optimization accuracy is highly related to the choice of the neighborhood structure. Optimization of complex problems requires a lot of storage memory.	(44, 45)
Taguchi	Maximize information from a small database.	Poor estimation of neural networks features interactions.	(46, 47, 48)
Decision trees	Good model understanding. Low computational time.	Low optimization accuracy.	(49, 50, 51, 52, 53)

3.3 Selection of the neural network architecture

Four different methods widely present in the literature overview to determine the best neural network architecture have been proposed and compared. These techniques are based on the use of genetic algorithm, Taguchi, Tabu search and decision trees. The goal of these algorithms is to find, in a small computational time, the features of the neural network that allow the prediction error to be minimized.

3.3.1 Parameters Description

The parameters taken into account in this analysis are the number of hidden layers, the number of hidden neurons and the activation function of each layer

and the training algorithm. The choice of the first two parameters is related to the ability of the network to approximate complex functions. Regarding the number of hidden layers to use there is a consensus about the performance differences: the situations in which performance improves by adding a hidden layer are very small. One or two hidden layers are sufficient for the large majority of problems. In some cases neural networks with more hidden layers seem to work better but the performances are improved only on the training set while in the validation set the results could significantly deteriorate. The most suitable activation function of each layer is determined by the characteristics of the input-output relationship. Using a good mix of these functions it is possible to approximate any kind of problem. In this work four different activation function were considered:

1. Linear function
2. Log-Sigmoid function
3. Tan-Sigmoid function
4. Radial basis function

The linear function is the most simple transfer function. It is mostly used for problem with a limited complexity. However the combination of a linear with a non linear function results able to approximate any kind of function if a sufficient number of hidden unit is available (54). The linear transfer function can be represented through the following equations:

$$l(x) = x \tag{3.4}$$

However, in the large majority of cases to represent efficiently complex problems is necessary the use of not linear activation functions. The sigmoidal function (55) is a real function in the range (0,1) and is probably the most popular activation function used in the back-propagation. The sigmoid is defined by the following expression:

$$s(x) = \frac{1}{1 + e^{-x}} \tag{3.5}$$

A transfer function similar to the sigmoid but defined over the region (-1,1) is the hyperbolic tangent sigmoid. One properties of this activation function is the anti-symmetry. This means that for each value of x the following relation holds:

$$f(-x) = -f(-x) \tag{3.6}$$

The anti-symmetry properties is often associated to better and faster learning capabilities (56). This transfer function is expressed by these formula:

$$t(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{3.7}$$

The last function considered is the radial basis (57) retained as one of the best function to approximate different models. The transfer function is calculated in the following way:

$$r(x) = e^{-x^2} \tag{3.8}$$

Another factor that can affect the generalization ability of a neural network is the training algorithm. In this work four different algorithms were analyzed: Levemberg-Marquardt, Quasi-Newton, gradient descent with momentum and adaptive learning and resilient backpropagation. The Levemberg-Marquardt algorithm is the most widely used optimization algorithm. It outperforms simple gradient descent and other conjugate gradient methods in a wide variety of problems. It is often the fastest backpropagation algorithm and highly recommended as first choice among supervised algorithms (58). In this training algorithm the weight-update is based on the following formula:

$$w(k+1) = w(k) - (J^T J + \lambda I)^{-1} J^T \varepsilon(k) \tag{3.9}$$

where ε is the difference between the actual and the required value of the network output and J is the Jacobian matrix. The Quasi-Newton and the gradient descent with momentum and adaptive learning rate algorithm offer more sophisticated exploitation of the gradient information compared to simple gradient descent methods but they are usually more computationally expensive (59). Different variants of the back-propagation algorithm have been developed in the last years in which not only the local gradient is considered but also the recent change of

the weight matrix. According to these variants at each iteration the new value of the weights will be equal to:

$$w_{i,j}^{m}(k+1) = w_{i,j}^{m}(k) - \gamma \frac{\partial E}{\partial w_{i,j}^{m}} + \beta(w_{i,j}^{m}(k) - w_{i,j}^{m}(k-1)) \tag{3.10}$$

Where $w_{i,j}^{m}$ is the weight of the connection between the neuron i of the layer $m-1$ and the neuron j of the layer m whereas β is called momentum and is usually introduced to reduce oscillation in the error function during the iterations. The last training algorithm analyzed is the Resilient Propagation (60, 61). The main difference between this approach and the other techniques previously described is that the weight update is not blurred by the gradient behaviour. For each weight $w_{i,j}^{m}$ is determined a parameter $\Delta_{i,j}^{m}(t)$ that influences the size of the weight-update. This parameter evolves during the iterations according the behaviour of the partial derivative during two successive steps. The choice of learning algorithm has a double impact on the performance of the neural network because it affects both the quality of the solution found and the computational time to reach the convergence (62).

3.3.2 Genetic Algorithm

Genetic algorithm (GA) is an optimization technique that tries to replicate the evolution process, in which individuals with the best features have more possibilities of surviving and reproducing (39, 40, 41). The first step of this method is to encode the features of the neural network into specific chromosomes. A chromosome is a sequence of bits with value 0 or 1. GA undertakes to evolve the solution, during its execution, according to the following basic pattern:

1. Random generation of the first population of solutions
2. Application of a fitness function to the solutions belonging to the current population
3. Selection of the best solutions based on the value of the fitness function
4. Generation of new solutions using crossover and mutation

5. Repetition of Step 2-3-4 for n iterations

6. Selection of the best found solution

3.3.2.1 Representation of the neural network

The first step of this method is to define the input variables and the chromosome structure. In this work five input variables using chromosomes with 11 genes are considered. The first two genes represent the number of hidden layers in the network with a search range from 1 to 3. The next four genes are used to represent the activation function of the hidden (HL) and output layers (OL). The next three genes are used to represent the number of neurons (HN) in each hidden layer. The range of these features is fixed from 2 to 9. Finally the last two genes are used to represent the training algorithm by which the network learns. The whole domain of the chromosomes generation is summarized in Table 3.2.

According to this table 541184 possible network configurations can be created.

Table 3.2: Chromosome encoding

Hidden Layers	HL activation	OL activation	HN		Training algorithm
00=1	00=Linear	00=Linear	000=2	100=6	00=Levemberg-Marquardt
01=2	01=Log-Sigmoid	01=Log-Sigmoid	001=3	101=7	01=Quasi Newton
10=3	10=Tan-Sigmoid	10=Tan-Sigmoid	010=4	110=8	10=Gradient Descent
	11=Radial Basis	11=Radial Basis	011=5	111=9	11=Resilient

3.3.2.2 Selection

The selection of the chromosomes to produce a new generation is an extremely important step of the algorithm. The most promising chromosomes will be included in the next generation and will be used as "parents" in the crossover operations. A chromosome will be selected if the value of its correspondent fitness function is low. The fitness function implemented consists of two terms. The first term is the sum of the absolute error on the training set using a specific neural network, whereas the second term is measured on the test set. Both terms are multiplied by appropriate weights. Finally, a penalty term based on the number of hidden

layers is introduced to avoid the increase in this value. The fitness function is therefore calculated in the following way:

$$fitness(n) = \left(K_1 \sum_{i=1}^{Tr} (y_i - y_i^n) + K_2 \sum_{j=1}^{Te} (y_j - y_j^n) \right) \sum_{k=1}^{3} H_k z_k \tag{3.11}$$

where Tr and Te are respectively the size of training and test set, y_i and y_i^n are the real and predicted value of the output of $i-th$ configuration, z_K is a binary variable with value 1 if the network consists of k hidden layer, 0 otherwise, while H_k is the corresponding penalty term.

3.3.2.3 Crossover and mutation

After the selection of the most promising chromosomes, a new population is generated using the crossover technique that allows exploring new areas of the feasible region. In the genetic algorithm, crossover is a genetic operator used to create new chromosomes from one generation to the next. Two different crossover operations are used in this work: n-point crossover (42) and SBX crossover (43). Mutation is another genetic algorithm used to maintain genetic diversity from one generation of a population to the next. This operation consists of randomly altering the value of one element of the chromosome according to a mutation probability.

3.3.2.4 Final Solution

Selection, crossover and mutation are repeated iteratively until one of these conditions is satisfied:

1. the processing time exceeds a maximum time;
2. the number of iterations performed exceeds a maximum number;
3. the fitness function of the best found solution is lower than a given threshold.

Finally, the chromosome with the best fitness function is decoded and a neural network with those features is built.

3.3.3 Tabu Search

Tabu search (TS) is a metaheuristic technique that is used to find a solution for several kinds of optimization problems. One of the main property of this technique is the ability to escape from local minima allowing to exhaustively exploring the solutions space.

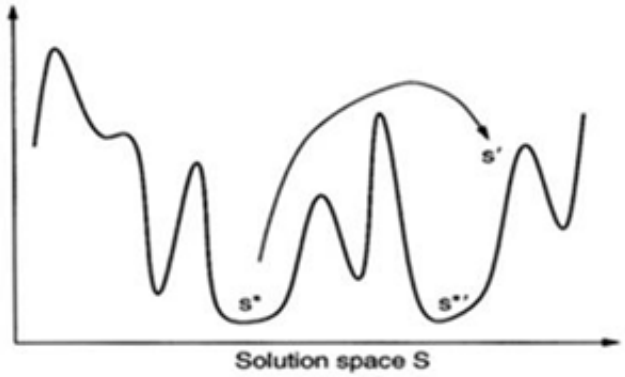


Figure 3.1: Representation of the solution space S of a generic problem

Fig. 3.1 highlights the advantages of this method. s^* is a local optimum of the problem. However, s^* is slightly better. If s^*' is not part of $N(s^*)$, this solution will never be reached using local search algorithms. This might happen instead using the TS method. In fact s^*' can be reached by moving along s'. Another advantage of this technique is the possibility of keeping in memory the latest moves made in order not to repeat the same exchanges (44). These memory structures form what is known as the tabu list, a set of rules and banned solutions used to filter which solutions will be admitted to the neighborhood to be explored by the search. Hybrid approach based on tabu Search and neural network were developed in the past to improve the performance of the learning algorithm (45). In this work a similar technique is proposed to find the optimal neural network architecture by means of tabu search. The main steps are summarized in Fig. 3.2. First of all, the architecture of a neural is chosen randomly. Then, the neighborhood of the current solution s' is generated at each iteration. Three types of neighborhood were considered in this phase to establish the effect of each investigated factor. The first contains all the networks that can be obtained by adding

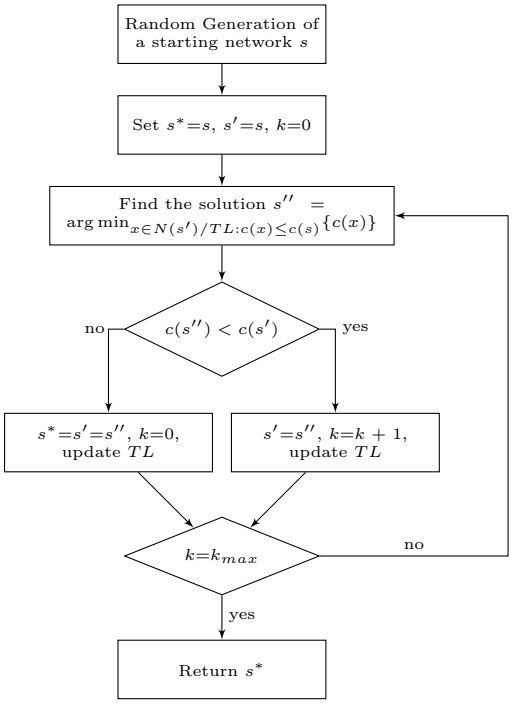


Figure 3.2: Tabu search algorithm

or removing a neuron from one or more hidden layers. The networks belonging to the second neighborhood have the same architecture except for the activation function of one layer. Finally, the last neighborhood considers different learning algorithms. These three neighborhoods can be considered simultaneously at each iteration, or alternatively. To reduce the processing time it is possible to use a diversification strategy. Subsequently, the best solution of the neighborhood, $N(s)$, that does not belong to the Tabu list, TL ,or that belong to TL and is better than the solution s* is selected. Afterwards the Tabu list is updated adding the solution s". If this solution is better than s* the value of this variable is set equal to s", otherwise the counter k is increased by one and only the variable s' is updated. This procedure is repeated iteratively until the value of k is less of a maximum threshold k_{max}. The algorithm is repeated several time in order to explore different areas of the feasible region. The starting network is chosen randomly only in the earliest iterations. In fact, during the execution of the algorithm statistical information on the neural network analyzed are stored and used to explore areas not yet investigated. The following approach was used to

calculate the probability which an attribute of the network is chosen with. Let $m_i^k \; i = 1, ..., M, \; k = 1, ..., K$ the number of times that the feature k of a network assume the value i and let N the number of neural network analyzed until that moment. Each parameter of the network can assume only one value, so:

$$\sum_{i=1}^{M} m_i^k = N \quad \forall k = 1...K \tag{3.12}$$

In contrast, if $N - m_i^k$ is the number times that the feature k of a network do not assume the value i:

$$\sum_{i=1}^{M} (N - m_i^k) = (M-1)N \quad \forall k = 1...K \tag{3.13}$$

The probability p_i^k that the feature k of a new network assume the value i is then calculated in the following way:

$$p_i^k = \frac{(N - m_i^k)}{(M-1)N} \quad \forall k = 1...K \tag{3.14}$$

The procedure illustrated in Fig. 3.2 is repeated iteratively, using this approach, until a maximum number of iterations have been carried.

3.3.4 Taguchi Method

The Taguchi method was developed as an optimization technique by Genuchi Taguchi during the 1950's. The method is based on the statistical analysis of data and offers a simple means of analysis and optimization of complex systems. Taguchi method uses a special design of orthogonal arrays to analyze the solutions space with a small number of experiments. This method has been used to optimize the performances searching the best architecture of the neural network (46, 47, 48). Generally the number of experiments, which compose a fully orthogonal experimental plan is equal to $N = L^F$ where F is the number of factor and L the number of levels for each factor. Five factors and three levels for each factor are considered in this work. The number of required experiments according to fully orthogonal experimental plan should be 3^5 to investigate all the possible solutions; however, with Taguchi it is possible to use a reduced set of 3^3 network

configurations. The choice of the optimal network is based on the value of the S/N ratio that is a measure of the performance variability in presence of noise factors. S/N ratio is a performance criterion where S stands for mean and is called signal and N stands for standard deviation and is called noise. The idea is to maximize the S/N ratio and thereby minimizing the effect of random noise factors. The S/N ratio is measured with the following equation:

$$S/N = -10 \log\left(\sum_{i=1}^{n} y_i^2/n\right) \tag{3.15}$$

where y_i is calculated using the formula (3.11). For each feature of the neural network the value with the smallest average S/N ratio is selected.

3.3.5 Decision Trees

Decision trees are one of the most used learning methods. The main reasons why this technique is frequently used for classification and regression problems are its ease of use, the low computational time and the possibility of quickly analyzing the results. In the beginning decision trees were used only for classification problems (49), but in 1998 Neville explained how to use decision trees to create stratified regression problems models by selecting different slices of the data population for in-depth regression modeling (50). Decision trees have also some shortcomings. When the connections between inputs and outputs is too complicated a simple tree may simply excessively this relationship and even if the tree shows good accuracy this does not mean that the generalization ability is elevated because a completely different set of data might give a different explanation to the same problem (51, 52). An important issue in the use of decision trees are the branching criteria (53). For this purpose, a measure of goodness of the model is calculated at each partition and for each variable. Let $t_r(r = 1, .., s)$ be the number of child groups generated by segmentation and p_r the proportion of observations that are placed in each child node, with $\sum_r p_r$. The criterion can be expressed by:

$$\phi(s, t) = I(t) - \sum_{r=1}^{s} I(t_r)p_r \tag{3.16}$$

Where I is a function of impurity. For regression problem a suitable impurity function can be expressed using this formula:

$$I_v(m) = \frac{\sum_{l=1}^{n_m}(y_{lm} - \hat{y}_m)^2}{n_m} \tag{3.17}$$

where y_{lm} is the real accuracy of the neural network l belonging to the node m while $\hat{y}_m$ is the predicted value of all the networks in this node. Other important factors are the choice of the rules of pruning and stopping. The stopping criteria are used in each node to determine whether further ramifications are necessary. There are essentially two reasons why an excessive branching is unwise: the generalization ability (since a decision tree with too many nodes is exceedingly tied to the values of the training set but may have worse performance when applied on the test set) and the facility to interpret the classification rules (since it decreases by the increasing of the tree ramifications). To train the decision trees, a set of neural networks with different properties are initially tested by measuring the average error rate obtained on the validation set. The leaf node with the best performance is then selected. This leaf could contain different neural networks architecture. In this case all the neural networks that belong to that particular leaf are tested searching to find the one with the smallest average percentage error. This step also highlighted the importance of the stopping and pruning criteria. In fact, if a decision tree too big usually overfit the data, an excessively small tree could be useless if the number of possible solutions in the optimal leaf is elevated.

3.4 Additional heuristic approaches to optimize Neural Network accuracy

In this section different heuristic techniques to improve the performance of the neural network reducing the time to get convergence and increasing at the same time the prediction ability of the tool will be presented. More in particular, these techniques will describe different methods that explain how to set the initial conditions of the network and execute the learning phase using an hybrid algorithm.

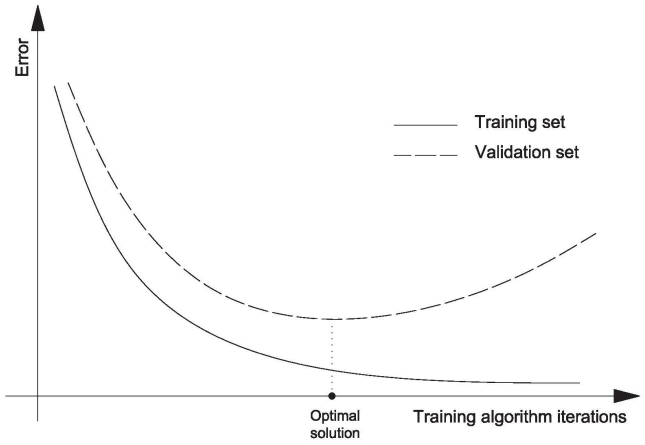


Figure 3.3: Training error

3.4.1 Backpropagation Limits

The training of a neural network can be divided in two phases as represented in Fig. 3.3. In the first phase the prediction error decreases both for the training and the validation set. Then, in the second phase, if the algorithm is not arrested, the network starts to be overtrained and the error starts to increase for the validation set. To overcome this problem the weight configuration selected is usually the one that minimize the *MSE* for both the set of data. Another problem is the presence of local minima (Fig. 3.4). In many cases, the training algorithms described are not able to escape from these minima and therefore the global optimum will never be reached. Furthermore, even if the network is able to escape from a local minimum a high number of iterations will be performed for small changes around it (Fig. 3.5). This problem affects the performances of all the algorithms described previously.

3.4.2 Neural Network Initialization

As stated before, one of the main disadvantages of the back-propagation algorithm to train feed-forward neural network is the elevated number of iterations required to find a good solution. This problem could be reduced by using some techniques that allow to start the algorithm from a good solution (63). For this reason weight

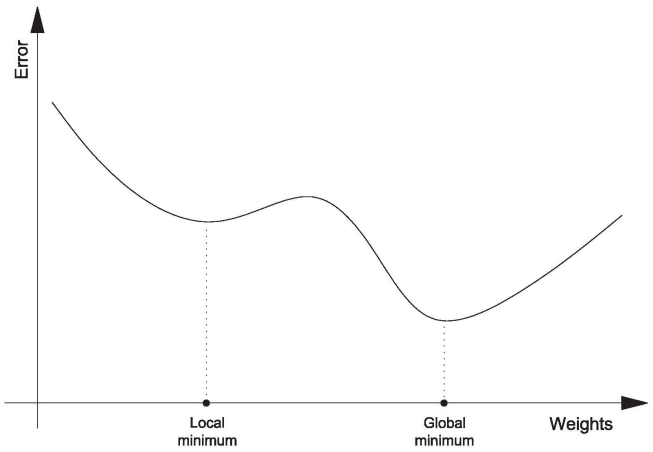


Figure 3.4: Local and global minima

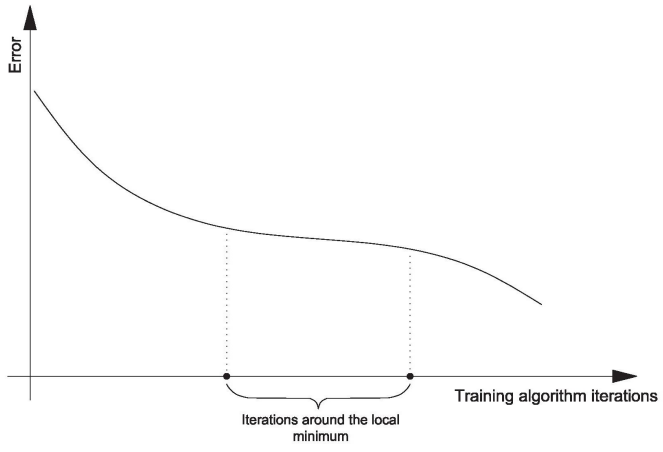


Figure 3.5: Training with a local minimum

initialization has been widely recognized as one of the most effective approaches to speeding up neural network training. Many weight initialization techniques have been developed in the past (64, 65). The main idea of these techniques is to choose the weights randomly with a distribution probability in a way that the expected value of the error function is minimized and its derivative function maximized. Normally to solve this task it is sufficient to choose random weights with a small value. This approach is particularly appropriated for some types of activation function such as the hyperbolic tangent sigmoid. This function can be divided in two parts: a saturation area and a flat area (active region). If we chose the weights so that the function works for many data in the saturation area, the local gradient will be very small and consequently the network will start to learn slowly. Assuming that the training set has been normalized a first approach suggested by LeCun et al. (56) is to generate the initial weight from a distribution with zero mean and standard deviation σ_w proportional to the number of connection that enter in the neuron. Another weight initialization approach was proposed by Nguyen-Widrow (66). In this method the weights are fixed in a way that the region of interest is divided into small intervals. The main idea of our proposed approach is to create a starting weight matrix so that the activation function will work as much as possible in its active region. In this paper, the active region is assumed to be the region in which the derivative of the activation function (f^{-1}) is greater than 20% of the maximum derivative (67). More in particular the threshold was fixed equal to:

$$\min f^{-1} = 0.05 \text{ for the sigmoidal function} \tag{3.18}$$

$$\min f^{-1} = 0.2 \text{ for the linear and the hyperbolic tangent sigmoidal function} \tag{3.19}$$

$$\min f^{-1} = 0.1716 \text{ for the radial function} \tag{3.20}$$

Given a neural network with M layers and a dataset of P training data we can formalize this approach using the following formulation:

$$\min z = \sum_{p=1}^{P} \sum_{m=1}^{M} \sum_{n=1}^{N} k_{m,j}^{p} \tag{3.21}$$

$$n_{1,j}^{p} = x_{1,j}^{p} \quad \forall p \in 1, ..., P \quad \forall j \in 1, ..., N_m \tag{3.22}$$

$$n_{m,j}^{p} = \sum_{i=1}^{N_{m-1}} (w_{i,j}^{m-1} \tilde{Y}_{m-1,i}^{p} + b_{m-1}) \quad \forall p \in 1, ..., P \quad \forall m \in 2, ..., M \quad \forall j \in 1, ..., N_m \tag{3.23}$$

$$\tilde{Y}_{m,j}^{p} = f_m(n_{m,j}^{p}) \quad \forall p \in 1, ..., P \quad \forall m \in 1, ..., M \quad \forall j \in 1, ..., N_m \tag{3.24}$$

$$Ok_{m,j}^{p} \geq \max\{\tilde{Y}_{m,j}^{p} - U_m, 0\} \quad \forall p \in 1, ..., P \quad \forall m \in 1, ..., M \quad \forall j \in 1, ..., N_m \tag{3.25}$$

$$Ok_{m,j}^{p} \geq \max\{L_m - \tilde{Y}_{m,j}^{p}, 0\} \quad \forall p \in 1, ..., P \quad \forall m \in 1, ..., M \quad \forall j \in 1, ..., N_m \tag{3.26}$$

$$k_{m,j}^{p} \in \{0, 1\} \quad \forall p \in 1, ..., P \quad \forall m \in 1, ..., M \quad \forall j \in 1, ..., N_m \tag{3.27}$$

The objective function minimizes the number of times that the activation function works in the saturation region. To measure this quantity a binary variable has been introduced for each training data and for each neuron of each layer. The net input of the neurons of the input layer in the configuration p is equal to the correspondent element of the vector X^p (4.1) while the net input of a neuron j in a hidden layer is equal to the sum of the product between the output value of the neurons i in the previous layer and the weight of the connections (i, j) (3.23). The output of a neuron is calculated applying the activation function of the correspondent layer on the net input (3.24). For each activation function has been fixed an upper bound and a lower bound describing the range of the

active region. The value of the binary variable $k^p_{m,j}$ will be set equal to 1 if the correspondent variable is out of this region (3.25) (3.26). In these equation the value of $Ok^p_{m,j}$, where O is a sufficient great scalar, can be equal to 0 ($k^p_{m,j} = 0$) only if $L_m \geq \tilde{Y}^p_{m,j} \geq U_m$. Generally the starting solution found through this model allows the backpropagation iterations to be reduced keeping the same performance. However if the network have to handle a huge amount of data this model can not be solved exactly in a short computational time because the size of the model increases quickly. However, usually, the optimal solution of this model is not unique and an optimal solution, or at least a good solution, can be found in a very short time using metaheuristic techniques. In this work in the numerical tests reported in the next chapters this model has been solved using a genetic algorithm with a small population and few generations.

3.4.3 Simulated Annealing

Simulated annealing is a probabilistic meta-heuristic that is used to find a good approximation of the global optimum of a given function (68). The name of this technique is inspired by the annealing in metallurgy, a technique involving heating and controlled cooling of material. The main characteristic of the Simulated Annealing algorithm is the possibility to move along worse solutions during the research avoiding of being trapped in local minima. The simulated annealing algorithm for neural network training require the initialization of some parameters like the starting weight of the synapses chosen using the method previously described and the temperature T. During the training phase a worst solution is accepted if this condition is satisfied:

$$p' \leq p \tag{3.28}$$

$$p = e^{\left(\frac{f-f'}{T}\right)} \tag{3.29}$$

Where f and f' are the objective function of the current and the new solution to evaluate. The value p' is a random number with a uniform distribution between [0,1]. At each iteration, a new candidate solution is found by applying a

perturbation on the current solution using the following formula:

$$w_{z,j}^{m} = w_{z,j}^{m} + rv_{z,j} \tag{3.30}$$

$$b_{j}^{m} = b_{j}^{m} + rv_{j} \tag{3.31}$$

Where r is a random number with a uniform distribution between [0,1] and v_j is a coefficient that is used to determine the max perturbation that could be applied on the value of the weight element. Usually the parameters T and v are not static but their value decrease during the iterations (69). Moreover it is important to find the starting value of T because a too much high value of this parameter has not practical purpose because too many bad solution will be accepted. In contrast, a too small value would limit the capability of the algorithm to explore different area of the space solution losing the main advantage of the simulated annealing algorithm. An appropriate starting value of the parameter T would be chosen so that the initial probability p is included in the range of [0.6,0.9]. In this work the starting temperature has been calculated so that the p value will be how much close as possible to the value 0.8 using the following approach. Initially a number of K iterations have been done setting the parameter T equal to 0 and with an elevated value of v. These iterations are useful to collect statistical information regarding the standard deviation of the *MSE* (σ_{MSE}).

$$\mu_{MSE} = \sum_{k=1}^{K} \frac{MSE_k}{K} \tag{3.32}$$

$$\sigma_{MSE} = \sum_{k=1}^{K} \sqrt{\frac{(MSE_k - \mu_{MSE})^2}{K-1}} \tag{3.33}$$

$$T = \frac{\sigma_{MSE}}{\log 0.8} \tag{3.34}$$

The temperature parameter during the iterations will be then updated in the following way:

$$T_{new} = T_{old} - T_0 \frac{N_T}{IT_{max}} \tag{3.35}$$

$$v_{new} = v_{old} - v_0 \frac{N_v}{IT_{max}} \tag{3.36}$$

where N_T and N_v are the number of iterations done without update the correspondent parameters. The idea of this formula is to decrease temperature and step size according to the number of maximum iteration IT_{max}. In this way in the first iterations the algorithm will explore different area of the space solution and found the most promising zones whereas in the lasts will be applied just few perturbation trying to find the local minimum of this specific area.

3.5 Proposed Algorithm

The proposed algorithm is based on the combined use of the heuristic techniques previously described. More in particular the step of the algorithm are the following:

1. manipulation of the training set;
    - mean cancellation;
    - variance equalization;
2. architecture selection;
3. weight initialization;
4. mixed use of backpropagation algorithm with momentum and simulated annealing.
5. denormalization of the data

The ratio of the third point is the following: the backpropagation algorithm, as described before, is usually faster than simulated annealing but has worst ability to escape from local minima. So during the training the weight update will be based prevalently on gradient information while simulated annealing will be used only after k consecutive iterations without an improvement of the error function less than a fixed parameter δ. In this case to explore different regions of the space solution a little perturbation will be applied on the weight matrix using the simulated annealing. The temperature used in the different iterations will

be modified according to the residual computational time or iterations using the following formula:

$$T = \min\{T_{it}, T_{time}\} \tag{3.37}$$

$$T_{it} = T_0 - T_0 \frac{IT_{current}}{IT_{max}} \tag{3.38}$$

$$T_{time} = T_0 - T_0 \frac{t_{current}}{t_{max}} \tag{3.39}$$

Where IT_{max} and t_{max} are respectively the maximum iterations and time to train the neural network while $IT_{current}$ and $t_{current}$ are the current value of these parameters.

3.6 Computational Tests

In order to generalize the results, the above introduced methods were used to find the best neural network configuration for three different manufacturing processes. Each dataset is formed by 100 input-output pairs and is split in three parts: 80 pairs are used as training set and 10 as test set. Furthermore, in order to verify the real performance of these techniques 10 input-output pairs were used as validation set by using numerical models already validated. Four different tests were performed for each problem. The original datasets was used for the first test while a random noise with zero mean and standard deviation of one, two and five per cent was added to the outputs value in the other tests. The aim of these additional tests is to verify the ability of the network to filter out the noise and find the real link between the inputs and outputs. All the techniques were set up in the $MATLAB\ R2011b$ version except for the decision tree which was adapted by using the existing library *classregtree*.

3.6.1 Numerical Simulations

By taking advantage of the already introduced hybrid approaches (19) finite element (FE) models were performed to generate the process datasets. The numerical simulations have been always more utilized to reduce the number of the

experimental investigations whose equipment construction and trial execution generate great time and money wasting. Nowadays, the results which are possible to obtain by numerical simulations, if properly set, are in good agreement with the reality, and therefore, the FE codes can be considered as a reliable tool useful for process optimization. Different investigations can be carried out highlighting: 1) local variables, i.e. strain, strain rate and stress distributions, in the whole formed part, and 2) global greatnesses as the forces which are generated on the equipment during the forming phase. Concerning to the proposed work, the investigated processes can be properly analyzed by using two dimensional analyses by using specific boundary conditions. In fact, both the extrusion and the hydro-blanking are characterized by a symmetrical shape while the rolling can be accurately analyzed by using plane strain consideration. Taking advantage of that, the commercial finite element code, DEFORM 2DTM (70) was utilized for the numerical campaign; this is an useful simplification which has been chosen to reduce the computational time ensuring, at the same time, the quality of the obtained results. By the way, it has to be taken into account that the numerical investigations used for the work here proposed, analyze manufacturing processes with different peculiarities; due to that, suitable attentions had to be taken into account for the proper setting of each simulation case. More in detail, the extrusion is considered as a bulk process which is usually characterized by high deformation; this means that, the strong reduction in area deeply impacts on the remeshing-rezoning code capability, and this is a possible reason of result inaccuracy. The simulation convergence, in this type of process, can be reached, step by step, through a tidier mesh obtained by numerical interpolation from the old to the new discretization. The material deformation during the rolling, on the other side, is reduced compared to the extrusion; in this case, special attention has to be given for the condition setting between the formed material and the rollers. Finally, the hydro-blanking, being a sheet forming process, is characterized by a small dimension of one of the specimen sizes, its thickness. According to that, the mesh has to be defined for having at least three elements along the thickness; this is a necessary condition which has to be complied for a correct analysis of the thinning which the sheet is subject to. Anyway, all the simulations, were carried out modelling, as billet material, an AA6061 whose plastic behaviour was derived

from the FE code library; the different tools of each simulation were modelled as rigid bodies to reduce the computational time. Mechanical analyses were executed by fixing the material temperature for each case. Tetrahedral elements were utilized for the discretization of the formed part; the different tools, instead, were considered rigid bodies to reduce the computational time. The friction conditions, moreover, were properly set for each process according to literature (71). Moreover, the outputs which were taken into account for the process optimization were chosen according to the main variables which have to be investigated for appropriately establishing the process effectiveness (71). In conclusion, 300 numerical simulations, 100 for each case, by using two-dimensional analyses were executed according to a full orthogonal plane. The computational time can be found in less than one hour for each simulation using a PC-Dual- Xeon 2,8 GHz with 4-GB RAM.

3.6.2 Optimization techniques Initialization

The proposed optimization techniques require the initialization of different input parameters. According to our preliminary analysis (and/or making use of standard techniques setting proposed in the works cited in Table 3.1 we performed the design of each technique according to the rules defined in Table 3.3.

Table 3.3: Techniques Initialization

Genetic Algorithm		**Tabu Search**	
Responce to be optimized	fitness function	Responce to be optimized	fitness function
Population size	25	Neighborhood strategy	diversification
Mutation Probability	0.05	Neighborhood probability	0.33
Crossover Probability	0.5	k value	10
Stopping criteria	Max time (1 minute)	Stopping criteria	Max time (1 minute)
Taguchi		**Decision trees**	
Responce to be optimized	S/N	Responce to be optimized	impurity function
Number of factor	5	Branching strategy	binary
Number of level	3	Max depth	6
Type of design	L27	Dataset size	50
		Stopping criteria	Max time (1 minute)

This design will be used for all the analysis reported in the next sections in which

the results of each technique have been compared with the one obtained using a neural network architecture randomly selected (the average percentage error obtained using 10 networks selected according to the range defined in Table 3.2.

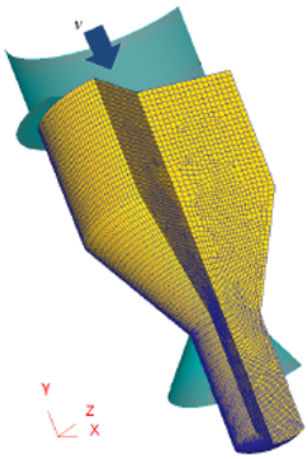
3.6.3 Extrusion

The extrusion process (27) is one of the most used technology in the manufacturing scenario and allows to obtain bars, tubes, profile shapes with different sections, (open, hollow, or filled) and numerous other elements. In its simplest configuration the process consists in a reduction of diameter: by action of horizontal compression, the movement of the punch forces the material (a solid cylinder) to flow and then to extrude through a shaped opening (matrix), whose section is equal to that of the product that the user wishes to achieve. In this study, punch velocity, extrusion ratio and temperature were considered as main parameters for the process decision maker. Typical technology constraints, which can penalize the process feasibility are the maximum punch load and the pressure in the critical area of the die, such as it is strategic to determine their variations for changing the temperature, the process velocity and the extrusion ratio. According to this aim, the introduced algorithms were set up to search the optimal neural network that minimize the prediction error of the highlighted response variables. Both factors and investigated ranges and a sketch of the numerical model for simulating extrusion process are reported in Fig 3.6.

As already specified, the described techniques were initially tested on the original dataset. Table 3.4 lists the mean absolute percentage error (MAPE) for both the output responses and the selected neural network architecture.

Table 3.4: Average percentage error (Extrusion original dataset)

	Genetic Algorithm	**Tabu Search**	**Taguchi**	**Decision Trees**	**Random Network**
MAPE Die Pressure	0.69	0.71	1.51	2.79	7.67
MAPE Punch Load	0.57	0.64	1.27	2.98	9.81
Hidden Layers	2	2	2	1	
Hidden Neurons	7-6	6-8	5-5	8	
Activation Function	Lol-Lin-Log	Lin-Lin-Log	Lin-Lin-Log	Rad-Log	
Training Algorithm	LM	LM	LM	LM	



Factor	Range
Punch speed, v	$[1-10]$mm/sec
Temperature, T	$[350-450]$ °C
Extrusion Ratio, ER	$[5-15]$

Figure 3.6: Extrusion process

As the table showed the best performance are obtained using the genetic algorithm procedure. However the error obtained using the Tabu search is at roughly the same. The average percentage error obtained by using the Taguchi method or decision trees is slightly higher but this error could be relevantly greater trying to solve this problem with a neural network with a random architecture. The results of all the other tests with a perturbation of different magnitudes are summarized in Table 3.5.

Table 3.5: Average percentage error (Extrusion noisy dataset)

	Perturbation of 1%		Perturbation of 2%		Perturbation of 5%	
	Die Pressure	**Punch load**	**Die Pressure**	**Punch load**	**Die Pressure**	**Punch load**
Genetic Algorithms	0.68	0.98	2.24	2.13	2.98	2.56
Tabu search	0.81	1.02	2.03	1.98	2.69	3.12
Taguchi	1.81	1.69	3.12	3.64	5.23	6.12
Decision Trees	3.18	2.92	4.84	4.55	6.21	5.65
Random network	10.89	9.22	12.56	14.87	13.38	17.56

Also in this case, the best performances are reached using the Tabu search and the genetic algorithm. Both of these techniques show a good ability to filter out the noise, so to map only the real inputs-outputs relationship. In fact, the performances of the neural network found with a noise of 5% have in both cases an

average percentage error less than 3%. In all the tests, the neural network obtained by using the proposed algorithms is composed of one or two hidden layers and the best training algorithm is always Levemberg-Marquardt. The magnitude of the perturbation does not seem to significantly affect the structure of the optimal network.

3.6.4 Rolling

The rolling process (72) is the second problem analyzed. Rolling is a plastic deformation process that allows to reduce the cross section of a block, through successive operations of reduction in thickness, in a tape with certain metallurgical and mechanical properties. In this study, an optimal neural network was found with the aim to understand how the rolling ratio, the process velocity and the temperature can affect the value of the normal pressure and the maximum load that typically penalize the process feasibility. A sketch of the used numerical model and of the investigated variables range is reported in Fig 3.7.

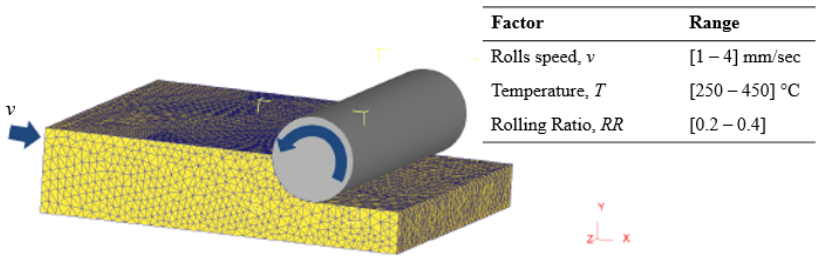


Figure 3.7: Rolling process

Also in this case the first test was made on the original dataset (Table 3.6) while the others introducing a perturbation (Table 3.7). As it is possible to see from the first table, the Tabu search and genetic algorithm procedure have the smallest error (less than 1%) for both the output but also the Taguchi method shows good outcomes. Compared to the extrusion process tests, the architectures of the neural networks selected to describe the rolling performances are characterized

by a smaller structure; in fact, the optimal architecture is always composed of one hidden layer with a number of neuron in the range [5-8].

Table 3.6: Average percentage error (Rolling original dataset)

	Genetic Algorithm	**Tabu Search**	**Taguchi**	**Decision Trees**	**Random Network**
MAPE Normal Pressure	0.91	0.78	1.14	2.56	8.97
MAPE Maximum Load	0.26	0.24	0.98	1.78	9.89
Hidden Layers	1	1	1	1	
Hidden Neurons	8	8	5	7	
Activation Function	Lin-Log	Log-Lin	Lin-Lin	Rad-Log	
Training Algorithm	LM	LM	LM	LM	

Table 3.7: Average percentage error (Rolling noisy dataset)

	Perturbation of 1%		**Perturbation of 2%**		**Perturbation of 5%**	
	Normal Pressure	**Maximum Load**	**Normal Pressure**	**Maximum Load**	**Normal Pressure**	**Maximum Load**
Tabu search	1.22	0.65	2.56	1.58	2.95	2.12
Genetic Algorithms	1.13	0.75	2.15	1.86	3.02	2.14
Taguchi	2.07	1.69	3.58	3.02	5.79	5.23
Decision Trees	2.88	2.03	4.02	3.91	7.21	6.85
Random network	11.23	8.22	13.23	12.87	18.24	20.24

3.6.5 Shearing

The last process analyzed is the shearing one (73) assisted by a counter pressure on the bottom of the blank. The shearing process consists of cutting of a sheet, a plate or a tubular component to derive individual components by subjecting the material to shear stresses, using a punch and a matrix. In the present study, the hydro assisted variant has been considered. More in detail, the punch speed was kept constant according to the high values typically used in industrial applications, while the effect of a fluid pressure on the bottom side of the blank was introduced to improve the quality of the sheared part (74). The main variables in the blanking process are the radii, the clearance between the punch and the die and the counter pressure. The two outputs examined in this test are the max

normal pressure and the punch load. For completeness, a sketch of the used numerical model for simulating the shearing process and the investigated factor are observable in Fig 3.8. As already shown for the other two processes, the perfor-

Factor	Range
Counter pressure, p	$[0-20]$ Mpa
Clearance, c	$[0{,}2-1]$ mm
Radii of the punch, r	$[0-0.9]$ mm

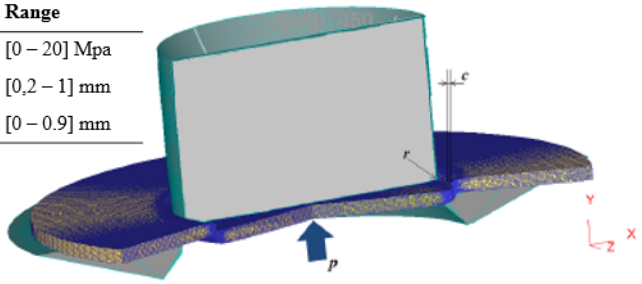


Figure 3.8: Shearing process

mances are measured on the validation set, using a training set with or without perturbation. The results are reported in Tables 3.8. and 3.9. below.

Table 3.8: Average percentage error (Rolling original dataset)

	Genetic Algorithm	Tabu Search	Taguchi	Decision Trees	Random Network
MAPE Normal Pressure	1.51	1.61	1.76	2.86	7.11
MAPE Punch Load	0.18	0.2	0.36	1.32	4.24
Hidden Layers	1	1	2	2	
Hidden Neurons	8	8	5-5	7-6	
Activation Function	Lin-Lin	Tan-Lin	Tan-Lin-Lin	Rad-Tan-Lin	
Training Algorithm	LM	LM	LM	LM	

Table 3.9: Average percentage error (Shearing noisy dataset)

	Perturbation of 1%		Perturbation of 2%		Perturbation of 5%	
	Normal Pressure	Punch load	Normal Pressure	Punch load	Normal Pressure	Punch load
Genetic Algorithms	2.03	0.56	2.84	1.02	5.01	1.98
Tabu search	1.7	0.54	2.45	1.36	4.37	2.31
Taguchi	2.89	0.73	3.78	1.36	5.42	3.38
Decision Trees	4.12	2.52	5.03	3.78	7.75	4.12
Random network	11.96	6.06	16.53	8.79	26.27	10.94

Also in this final test, the methods that seem to have best generalization ability are the Tabu search and the genetic algorithms. The results obtained for the normal pressure with a perturbation of 5% gives even more the idea of how very

important it could be to determine a suitable neural network architecture. In fact, the value of the average percentage error changes from a value of 26.27% to a value of 4.37%.

3.6.6 Robustness analysis

A last test was performed to evaluate the robustness of each technique. In this case we solved only one problem performing 100 runs for each technique. A technique will be considered robust if is able to provide the same results with a good consistency. The evaluation of the punch load and the die pressure for the extrusion problem was selected for this analysis. The following graph shows the frequency with which each neural network architecture is selected in each technique. How it is possible to see from the graph the technique characterized by the lowest consistency is the one based on the decision trees. This means that the results achieved using this algorithm are highly affected by the starting dataset (neural network architectures tested) used to build the analytical model. Obviously the same results are obtained for all the runs using the Taguchi technique since it does not use of parameter randomly generated during its execution. Based on the results of all the analysis reported in this chapter it is clear that the best trade off between accuracy and robustness of the algorithm is obtained using the genetic algorithm approach.

3.7 Conclusions

In this chapter different heuristic approaches to optimize the performance of a neural network improving the learning capabilities and reducing the time to get convergence have been discussed. These approaches are based on the use of simulated annealing, weight initialization, architecture selection using non-linear activation functions. Four different algorithms for the architecture selection of a neural network have been proposed and compared. The same methods have been tested and validated taking into account three of the most used process in the manufacturing industry. The algorithms illustrated are not only simple but showed a good efficiency to automatically determine the number of hidden

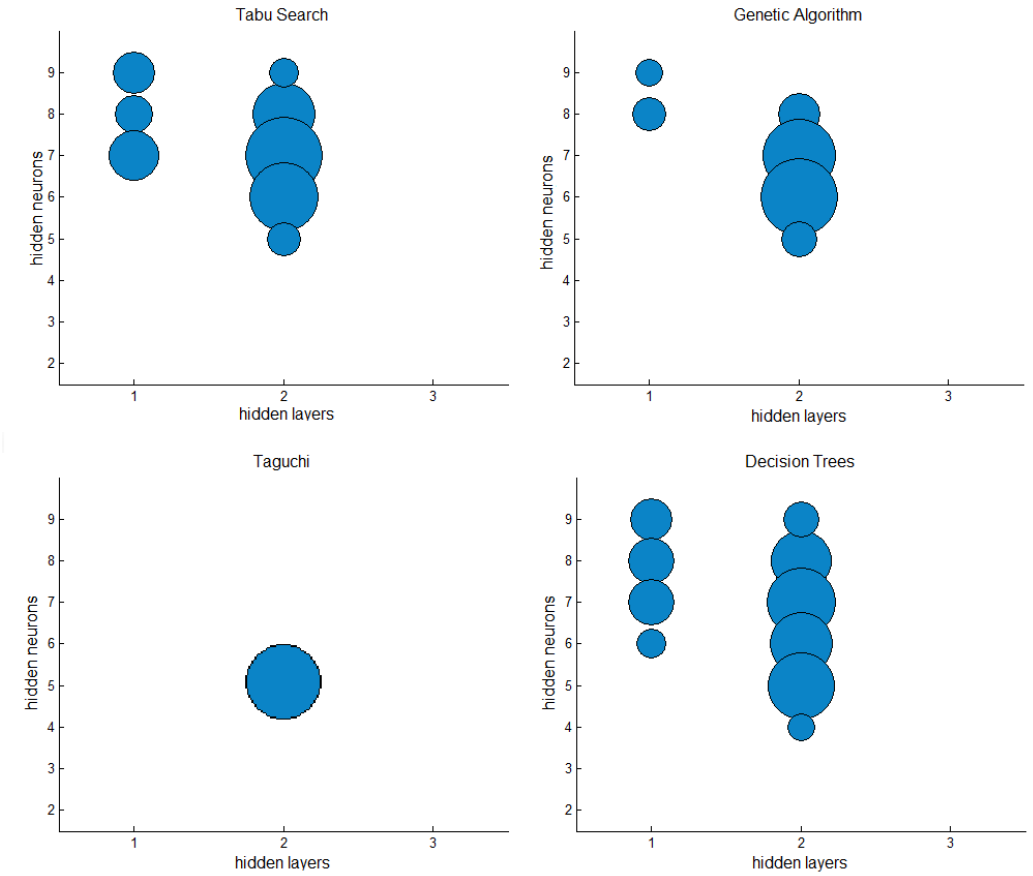


Figure 3.9: Techniques robustness

layers, neurons and other features that guarantee good generalization abilities. The numerical results demonstrate that the methods based on tabu search and genetic algorithms are able to select a neural network architecture which has a higher average testing accuracy. However the GA seems to be more reliable compared to TS and less affected by the parameters randomly selected during its execution. For all techniques, as is easily predictable, the performance decreases by increasing the magnitude of the perturbation in the dataset, but the way in which these results deteriorate seems better for these two methods.

4

Metamodelling-based process design for homogeneous thickness distribution: an application to Incremental Sheet Forming

4.1 Introduction

The optimal design of processes and products requires a deep understanding of influences that achieve desirable performance. As shown in the previous chapter metamodeling technique can be a useful tool to obtain the required knowledge to understand how a process can be optimized according to a particular criterion. However many of the techniques previously discussed are not able to deal properly with problems characterized by both continuous and discrete variables. This drawbacks is particularly relevant in manufacturing applications in which the impact of many discrete parameters (i.e. material, shape, tools) can usually not be neglected. The chapter provides an overall description of optimization problem classes with a focus on problems with continuous variables. A based kriging metamodel to solve mixed discrete/continuous variables is predsented in the next sextion. Finally an applications on a Incremental sheet forming process (ISF) is discussed.

4.2 Methodology

Kriging is an effective metamodeling technique used in many manufacturing applications to approximate the response surface of mechanical processes, as detailed in (1) and (2). Actually, ordinary Kriging can be used to manage processes characterized by only continuous variables. Many manufacturing applications however need mathematical tools able to deal with both quantitative and categorical factors (i.e. materials, geometrical conditions). Suppose that the response Y under analysis is affected by the factors $w = (x^t, z^t)^t$ where x^t is a set of quantitative input variables $x = (x_1, x_2, , x_d)^t$ and z^t is a set of qualitative input variables $z = (z_1, z_2, z_J)^t$. In this work we analyze a general case, with J qualitative factors $z = (z_1, z_2, z_J)^t$ where z_j has m_j levels, denoted by $1, , m_j$, for $j = 1, .., J$. Let denote as M the number of possible combination of qualitative factors $M = \prod_{j=1}^{J} m_j$ enumerated as $c = \{c_1, c_2, .., c_M\}$. The response $y(w)$ of the input value w can be modelled as:

$$y(w) = \beta^t f(w) + \varepsilon(w) \tag{4.1}$$

where $f(w)$ is a vector of p pre-specified functions, $\beta = (\beta_1\beta_2, .., \beta_m)^t$ is a vector of unknown coefficients and $\varepsilon(w)$ is the residual vector with 0 mean and variance σ^2. A valid correlation function for $\varepsilon(w)$ needs to be defined in the space involving both qualitative and quantitative factors while typical correlation function usually assumes that all the input data are quantitative. This problem was assessed by Qyan and Wu that built a Gaussian process models able to incorporate both factors (75). According to them a correlation function for $\varepsilon(w)$ can be constructed as:

$$cor(\varepsilon(w_1), \varepsilon(w_2)) = \tau_{c_1c_2} \exp\big(-\sum_{i=1}^{I} \phi_i(x_{1i} - x_{2i})\big) \tag{4.2}$$

where $\tau_{c_1c_2}$ is the cross correlation between the combinations of categories of input w_1 and w_2 and the $M \times M$ matrix $T = \{\tau_{c_1c_1}, \tau_{c_1c_2}, .., \tau_{c_1c_M}, .., \tau_{c_M,c_M}\}$ is a positive definite matrix with unit diagonal elements. A more in depth discussion about the correlation function and the hypersphere decomposition used to define the matrix T can be found in (75) and (76). To use equation 4.1 it is necessary to estimate β, σ^2 and ϕ. To this aim we use the maximum log-likelihood and denote

as $\hat{\beta}$, $\hat{\sigma^2}$ and $\hat{\phi}$ the resulting estimators. The natural logarithm likelihood (log-likelihood) is defined as:

$$\ln L(\theta, w, y) = -\frac{n}{2} \ln 2\pi\sigma^2 - \frac{1}{2\sigma^2}(y - w\beta)^t(y - w\beta) \tag{4.3}$$

The maximum likelihood $\hat{\theta}_{MLE}$ is then calculated as follows:

$$\hat{\theta}_{MLE} = \max_{\theta} \ln L(\theta, w, y) \tag{4.4}$$

The accuracy of $\hat{\theta}_{MLE}$ depends on the curvature of the log-likelihood function near $\hat{\theta}_{MLE}$. If the curvature of the log-likelihood (measured by its second derivative) is very high around $\hat{\theta}_{MLE}$, then θ will be precisely estimated. Finally, after all the necessary parameters have been estimated, equation 4.1 can be used to predict the output response $y(w)$ of new data w not included in the starting dataset.

4.3 Application to Incremental Sheet Forming problem

The kriging approach described in Section 2 was tested to predict and optimize the thickness distribution of different products formed through ISF process. This manufacturing technology is characterized by a localized deformation due to progressive punch movement on a sheet which is clamped around its perimeter (77). Kitazawa and Nakajima estimated the final thickness of the sheet through the sin law which a prediction of the thinnest part of the sheet t is made in only through geometrical considerations. However, it is well known that the sin law offers a poor estimation of the final thickness since it neglects the effects of different process parameters. In this research, the analysis was properly set to overcome the above highlighted limitations designing, at the same time, an optimization tool for the definition of the punch trajectory. To this aim, an experimental campaign was performed to implement a kriging metamodel explained in Section 2. The proposed procedure can be summarised as follows:

1. Problem modelling phase: the process variables that affect the final thickness of the sheet are identified. A lower and an upper bound are defined

for each variable according to the available equipment and mechanical considerations.

2. Collection phase: a set of data are collected for training and testing.

3. Training phase: the kriging metamodel is used to analyse the data and to fix a transfer function between the set of controlled process parameters and the final thickness distribution.

4. Design of the optimization model: selection criterion and a set of constraints are established.

5. Testing and validation: the metamodel and the mathematical formulation are used to select the optimal process configuration for a specific case study.

4.3.1 Experimental Campaign

Selecting an experimental design is a key issue in building an efficient metamodel. The factorial relationship between process variables and samples size, however, reduce the level of applicability of this kind of techniques when a huge number of process parameters is involved. Therefore, it is necessary to reduce as much as possible the problem complexity to obtain a robust model. To this aim some secondary factors were fixed and neglected from the analysis. More in detail, the product height was fixed equal to 40mm for each test, as well as the tool depth step among two subsequently punch loops along the trajectory was kept constant to 1mm. In the same way, the forming tool diameter was equal to 12mm and its velocity was equal to 2m/min. The experiments were performed using sheets with a 400x400mm size.Usually the first step to define a DoE strategy is to select the investigated process parameters and a set of levels for each factor. Afterwards, a fraction of all the possible combinations is tested. However, not all the possible configurations are feasible according to the process behaviour. To generalize the results of the proposed technique three different frustum shapes were analysed, i.e. circle, square and flower (Fig. 4.1). The choice was done because with these different geometries all the possible effects on a deformed components can be reproduced (78). Three materials were investigated, namely an aluminium alloy

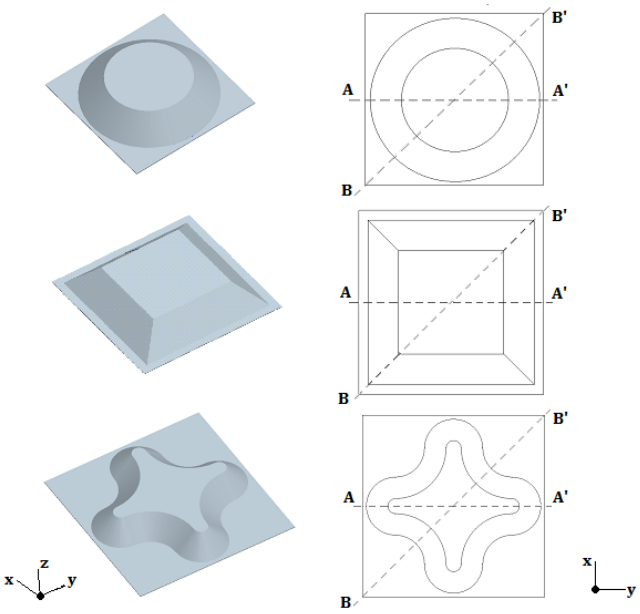


Figure 4.1: The investigated shapes and the analysed Lateral Section (LS) AA' and Diagonal Section (DS) BB'.

(AA1050), a pure brass and a high speed steel (HSS); the selection was made by choosing materials with different mechanical properties. The maximum slope angle $\alpha_{\max}$ that can be performed on each material was taken into account, while nominal slope angle α was encoded in the DoE as the percentage of $\alpha_{\max}$. Three α levels were so defined inside the range [65%-95%] of $\alpha_{\max}$. According to previous study (77) $\alpha_{\max}$ was set equal to 67.5° for the AA1050, 40° for the brass and 65° for the HSS. The decremental slope strategy was followed by introducing two artificial angles, α_1 and α_2, Fig. 4.2 for the punch trajectory optimization. The

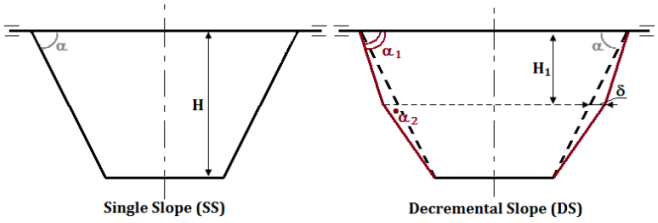


Figure 4.2: The comparison between Single Slope and Decremental Slope strategy construction.

investigated levels of each categorical factor and the bound of each continuous

variable are reported in Tables 4.1 and 4.2. Five process parameters were

Table 4.1: Categorical Variables

Parameters	Levels		
Material (M)	AA1050	BRASS	HSS
Shape (S)	Square	Cone	Flower

Table 4.2: Continuous Variables

Parameters	Lower Bounds	Upper Bounds
Thickness (t_0)	1mm	2mm
α_1	$65\%\alpha_{\max}$	$65\%\alpha_{\max}$
α_2	$0.9\alpha_1$	$1.1\alpha_1$

therefore taken into account in the analysis. A full factorial DoE is not feasible since 3^5 (243) experiments would be necessary. Moreover, many standard design techniques are not suitable when both qualitative and quantitative factors are present. The DoE permits the analysis of the interaction between qualitative variables and response surface, allowing in a robust way to determine if a model is adequate. Two criteria often used to build kriging metamodel are the IMSPE and the entropy criteria (79). The equispaced design, the one with equidistant point, is optimal for both criteria. The set of points to analyse was therefore selected trying to maximize the distance d_tot between each point and its closest.

$$\max d_tot = \sum_{t \in T} \min_{t' \in \{T/t\}} \sum_{i=1}^{n} \frac{|x_{i,t} - x_{i,t'}|}{x_i^{up} - x_i^{lo}} \tag{4.5}$$

where $|T|$ is the dataset size, $x_{i,t}$ is the value of the $i-th$ factor on the $t-th$ experiment and n is the number of analysed factors. The distance between each couple of points is then normalized in the range [0;n] according to the bounds x_i^{up} and x_i^{lo} of each factor i. 75 experiments were selected using this criterion. To reduce the scientific campaign, one third of these tests were experimentally carried out and repeated three times for avoiding any result uncertainty; the obtained data

were utilized to set and validate a numerical model, which was subsequently used to complete the dataset. The thickness sections were experimentally measured by a 3D Coordinate Measuring machine, while the numerical simulations were executed with the Ls-Dyna solver V.971 whose reliability for this type of applications was widely validated. For each test the thickness was measured on two different sections (see Fig. 4.1) chosen to cover as much as possible the results variability. Two sections were taken into account also concerning the frustum of cone, even if the geometry is axisymmetric. This was done because the blankholder was not used during the forming phase and, therefore, the distance between the clamped sheet and the working area is not constant.

4.3.2 Process Optimization

When all the experimental data were collected, the kriging metamodel was used to define a function $t_z = f(M, S, t_0, \alpha_1, \alpha_2, z, d)$ that links the final thickness of the sheet on the section d at the height z. The accuracy of the model was tested using a leave-one-out cross validation (80) using the R^2 as performance criterion. The achieved results were satisfying reaching a value equal to 0.926. The following model was used to optimize the tool trajectory (α_1 and α_2) according to the worked material, the initial thickness, and final shape of the sheet. A suitable tool trajectory was found by the optimization model for products characterized by a nominal slope angle α. Therefore, not all the possible tool trajectories can be accepted, but only the ones that approximate the desired final shape with sufficient accuracy. Possible tolerances on the final shape of the product were taken into account adding the following constraint on the model:

$$H_1 \tan \alpha_1 - \tan \alpha \leq \delta \tag{4.6}$$

$$H_1 \tan \alpha_1 + H_2 \tan \alpha_2 = H \tan \alpha \tag{4.7}$$

The criterion used to select the optimal value of α_1 and α_2 is the maximization of the thinnest area of the sheet ($t_{\min}$). In this case all the sections $d \in D$ in which the profile thickness was measured have to be taken into account:

$$\max t_{\min} = \min_{z \in Z, d \in D} f(M, S, t_0, \alpha_1.\alpha_2, z, d) \tag{4.8}$$

Also in this case, safe working conditions have to be selected to avoid the sheet breaking. As a consequence it is necessary to put a threshold on the value of each slope angle used during the process.

$$\alpha_1, \alpha_2 \leq 0.95\alpha_{\max} \tag{4.9}$$

The quality of the selected solution was evaluated as the percentage increment of $t_{\min}$ compared to the value reached using a single slope trajectory. In many practical applications, however, the interest of the designers is not limited to the value of $t_{\min}$ but the objective to pursue is the realization of a homogenous thickness profile. The following equation was finally used to define this new performance criterion.

$$\min_{\alpha_1,\alpha_2} \Delta t = \frac{1}{|Z||D|} \sum_{d\in D} \sum_{z\in Z} |f(M, S, t_0, \alpha_1, \alpha_2, z, d) - t_0 \sin(\frac{\pi}{2} - \alpha)| \tag{4.10}$$

in which the thickness estimated using the sin law is used as a reference value.

4.4 Discussion of the results

Three target products were defined to validate the results of the developed model which mix all the materials and shapes investigated. The main features of these products are reported in Table 4.3. Material, shape and thickness of the blank can not be used as decision variables, but their values are fixed according to the specific application. Therefore, the only variable that can be designed to optimize the thickness profile is the tool trajectory. Even in this case, the solution space is constrained by equations 4.6, 4.7 and 4.9. The tool trajectory was optimized

Table 4.3: Validation tests (Input data)

Test	Material	Shape	Slope angle	Thickness[mm]
1	AA1050	Cone	60°	1
2	BRASS	Square	36°	1.5
3	HSS	Flower	59°	2

for each investigated product by using Eq. 4.8. The quality of the results were

compared to the minimum thickness using single slope strategy, $t^s_{\min}$. The results are reported in Table 4.4. Analyzing the results of all the investigated specimens

Table 4.4: Validation tests (Output results Thinning minimization)

Test	α_1 [°]	α_2 [°]	H_1 [mm]	$t^s_{\min}$ [mm]	$t_{\min}$ [mm]	thinning reduction [%]
1	63.3	58.2	12.7	0.405	0.518	28.0
2	38.6	35.1	9.9	0.912	0.988	8.3
3	62.7	57.9	16.1	0.628	0.693	10.3

it is possible to deduce that an increment up to 28% of the thickness limit can be achieved on the target product using the first optimized tool trajectory. A second test was performed using the same material and geometrical conditions reported in Table 4.3 but applying Eq. 4.10 as optimization criterion. The results of this second test are reported in Table 4.5 where Δt^s is the average thickness variation using single slope trajectory. Also in this case, it is possible to see, that the use of

Table 4.5: Validation tests (Output results Thinning minimization)

Test	α_1 [°]	α_2 [°]	H_1 [mm]	Δt^s [mm]	Δt [mm]	thinning reduction [%]
1	65.9	56.1	9.4	0.136	0.103	24.3
2	38.6	35.1	10.0	0.178	0.121	47.1
3	53.3	64.2	18.5	0.211	0.167	24.9

an optimization tool can produce several benefits on the thickness distribution of the final product. Finally, the model was definitively experimental validated for both the criteria. The comparison between experimental (EXP) and predicted (MOD) thickness for test #1 along the lateral section is reported in Fig. 4.3.

4.5 Conclusion

In this chapter is presented an efficient kriging metamodel technique that allows to manage manufacturing applications with both continuous and categorical process

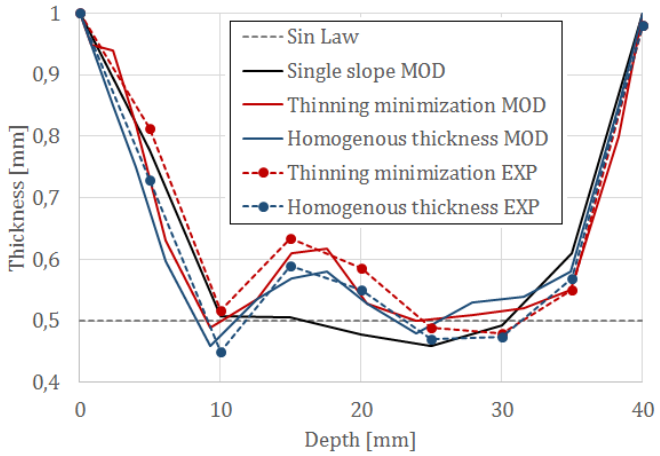


Figure 4.3: The comparison between experimental (EXP) and predicted (MOD) thickness.

variables. The proposed metamodel is quite general and can be adapted for different processes taking into account their specific features. An application of the model was shown in which the algorithm was used to define the relationship that links a set of ISF parameters to the final thickness distribution of the sheet. Mathematical models were designed to minimise the sheet thinning or to obtain more homogenous thickness distribution for ISF process.

5

Manufacturing Processes Modeling: an Application to Temperature Prediction on Incremental Sheet Forming

5.1 Introduction

Building a model able to analyze the relationship between a set of process parameters and a set of process responses is the first and yet critical step for design optimization. The accuracy of this model will directly affects the feasibility, cost, and effectiveness of optimization. Even if metamodeling techniques can be an extremely useful tool for the decision maker it is clear that their application is not always feasible due to lack of resources to collect the data to build the model. The possible reason of this scenario are usually three:

1. lack of tools, machines, materials to perform the analysis;

2. features of the problem: the process is too complicated and affected by a huge number of parameters. A metamodel can not be build due the exponential relation between number of investigated parameters and required data;

3. elevated variable cost to perform a single experiment.

To overcome these drawbacks a different strategy can be used if a deep knowledge of the process mechanics is available. The finite element method it is considered a powerful numeric method for finding good approximate solutions for systems of partial differential equations and it can be a good alternative to metamodel techniques. This method has been successfully applied to a wide field of engineering problems however also in this case the use of this approach is not always feasible due to the elevated computational time to perform a certain number of simulations or the difficulty to implement the model. As it will be shown in the next sections a simpler approach can be used if it is possible to map only the required information through practical formulae denying other features of the process. One disadvantage of this method is that the model is not general as metamodeling and FE techniques but is customized for the required application. However, it can be a suitable alternative since usually none or few data are necessary to build the model if the required knowledge is available In this chapter it is presented a model for the temperature prediction for an incremental sheet forming process. An experimental campaign was carried out for properly set the model; the punch velocity and the step depth are the process variables, which were changed because these ones are the main responsible for the temperature variation. Two lightweight alloys, i.e. AA-5754 and Ti6Al4V, were investigated due to their different thermal properties.

5.2 Mathematical Model

This section presents the mathematical model aimed at predicting the temperature trends of materials based on the values of different ISF process parameters. In this paper the model is used for ISF processes where the final shape of the sheet is the frustum of a cone; however, the model can be adapted to create more sophisticated shapes by adjusting the values in the formulae. Moreover, the use of a conical shape makes it possible to build the model by keeping the values of wall angle and process velocity constant; this is not possible for other geometrical shapes where mathematical formulations have to be set accordingly. The model

N	Set of nodes	t_j	time required to perform l_j
L	Set of loops	T_{melt}	melting temperature of the material
i, i'	index set of nodes	$\sigma_{i,j}$	flow stress on node n_i during l_j
j	index set of loops	$F_{i,j}$	load applied on n_i during loop l_j
$\dot{\varepsilon}_j$	average strain rate during the loop l_j	$A_{i,j}$	sectional area perpendicular to $F_{i,j}$
v	average speed of the process	μ	friction coefficient
T_r	room temperature	$M_{i,j}$	mass of the contact surface $A_{i,j}$
T_{pj}	temperature of the punch during l_j	r_p	punch radius
T_{ij}	temperature of node n_i during l_j	$Q_{i,j}$	heat generated on n_i during the loop l_j
p	depth step	$q^j_{i,i'}$	heat flow generated by conduction between n_i and $n_{i'}$ during l_j
v	average speed of the process	$q^j_{i,p}$	heat flow generated by conduction between n_i and the punch during l_j
h_f	final height of the sheet	c_p	specific heat of the material
r_j	radius of the loop l_j	λ	% of energy transformed to heat
r_p	radius of the punch	δ	density of the material
s_0	initial thickness of the sheet	K_m	thermal conductivity of the material
s_j	thickness of the sheet during l_j	K_p	thermal conductivity of the punch
α	wall angle	$d_{i,i'}$	distance between n_i and $n_{i'}$

Figure 5.1: Nomenclature Incremental Sheet Forming model.

uses an iterative approach in which the temperature variation is updated for each loop. To better understand the mathematical notations a nomenclature 5.1 of each symbol used and a flowchart of the algorithm 5.2 are reported below.

The approach used in this work involves the partitioning of the sheet being formed in points called nodes. The number of nodes analyzed $|N|$ is equal to the number of loops $|L|$ performed on the sheet during the process 5.3 and it is a function of the depth step (p) and of the final height of the sheet (h_f),

$$N = \ulcorner \frac{h_f}{p} \urcorner \tag{5.1}$$

The distance $d_{i,i+1}$ between two consecutive nodes is equal to the depth step along the y-axis while it is equal to the radius reduction along the x-axis:

$$d^x_{i,i+1} = p \tag{5.2}$$

$$d^y_{i,i+1} = p \cot \alpha \tag{5.3}$$

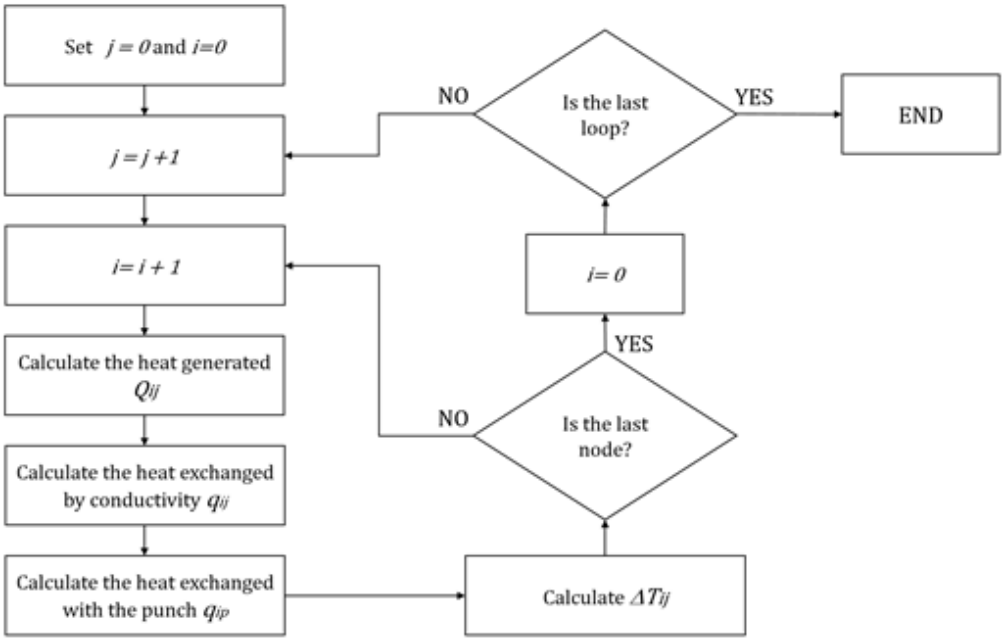


Figure 5.2: Flow chart Incremental Sheet Forming model.

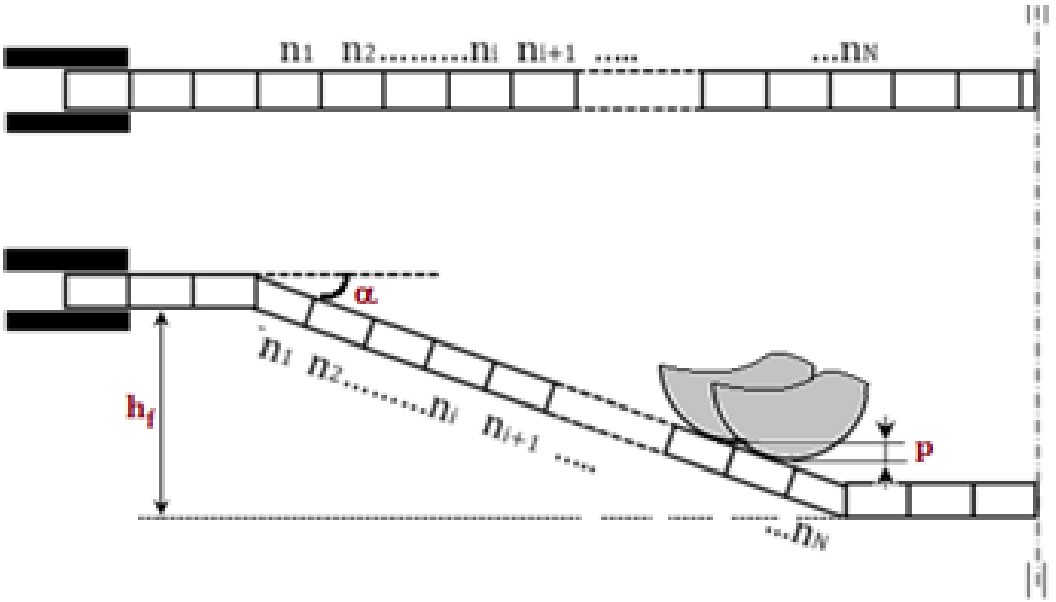


Figure 5.3: Representation of the temperature prediction model.

The values of the parameters in the model will always refer to a particular node. The temperature variation ($\Delta T_{i,j}$) is calculated for each node n_i and for each loop l_j using the following equation:

$$\Delta T_{i,j} = \frac{Q_{i,j} + q^j_{i,p} \sum_{n_{i'} \in N} q^j_{i,i'}}{c_p M_{i,j}} \tag{5.4}$$

Where $Q_{i,j}$ is the heat generated on the node n_i during the loop l_j, $q^j_{i,p}$ and $q^j_{i,i'}$ are respectively the heat exchanged by n_i with the punch and the other nodes, c_p is the specific heat of the material used, and $M_{i,j}$ is the mass of the volume $V_{i,j}$ in contact with the punch, measured as:

$$M_{i,j} = V_{i,j}\delta \tag{5.5}$$

being δ the material density. The volume $V_{i,j}$ is measured as the product of the current thickness of the sheet s_j and the area $A_{i,j}$ of the section perpendicular to the force applied by the punch:

$$s_j = s_0 \cos\alpha \tag{5.6}$$

$$A_{i,j} = 2\pi r_p(\dot{r_p}(1 - \cos(\alpha/2))) \tag{5.7}$$

The thickness of the sheet is calculated using the sine law, which is frequently used in this process (81). Sine law is based on the hypothesis that the process is started using an horizontal flat sheet and the material is in a plane strain state in the direction of the tool movement. A more in depth explanation of the sine law and its accuracy can be found in (82). The thickness predicted through sine law is usually a good approximation of the one measured after real tests, even if, the equation does not consider the effect of different process parameters and material properties that could play a relevant role. In 5.4, the heat is instantly generated for each loop only on the nodes that are in contact with the punch. The number of heated nodes for each punch pass depends on the size of the punch radius (r_p), the inclination angle (α) and the depth step (p):

$$Q_{i,j} = \begin{cases} \lambda V_{i,j} \int_0^{\varepsilon_f} \sigma_{i,j} d\varepsilon + \frac{F_{i,j}\mu p}{\cos\alpha} & \forall n_i \in N : i \in [j, j + \frac{r_p \alpha \cos\alpha}{p}] \\ 0 & \text{otherwise} \end{cases} \tag{5.8}$$

where λ is the percentage of energy generated through plastic deformation transformed to heat, $\sigma_{i,j}$ is the value of the flow stress on the node n_i during the loop l_j, $F_{i,j}$ is the load applied on n_i during loop l_j, μ is the friction coefficient between punch and sheet ε, is the strain value and $[j, j + (r_p \alpha \cos\alpha)/p]$ are the index of the nodes in which the heat is generated. As concerns the value of λ, (83) suggest that when working with AA5754 85% of the work converts to heat. Furthermore, they suggest that the conversion of work to heat is only 60% for Ti6AI4V. As previously mentioned the heat generated during the process is mainly due to plastic deformation and friction forces. Both components are affected by the flow stress equation of the material. The relevant effects of these two heat factors on the formability of an AA5086 sheet has been studied by (84). They pointed out that the effect of strain rate and temperature on the mechanical behaviour of the material is extremely important. This is particularly true for alloys where flow stress decreases with the increasing of the temperature; this results in a completely different behaviour when the temperature is inferior or superior to 200C. For this reason the flow stress equation is usually estimated using the Johnson and Cook model in this kind of application because it considers both strain rate hardening and thermal softening effects. The Johnson and Cook model is expressed as follows:

$$\sigma_{i,j} = [A + B\varepsilon_{i,j}^{n}]\left[1 + Clog\left(\frac{\dot{\varepsilon}_{i,j}}{\dot{\varepsilon}_0}\right)\right]\left[1 - \left(\frac{T_{i,j} - T_r}{T_{melt} - T_r}\right)^{m}\right] \tag{5.9}$$

where A is the initial yield strength of the material at room temperature, $\dot{\varepsilon}_0$ is a reference strain rate used to normalize $\dot{\varepsilon}$, T_r is the room temperature and T_{melt} is the melting temperature of the material. The strain rate is estimated for each punch pass using the following expression:

$$\varepsilon_{i,j} = \frac{(1 - \cos\alpha)v}{2\pi r_j l_j} \tag{5.10}$$

This equation was derived considering the thinning of the sheet, with the final thickness estimated by the sine law, and the forming time, calculated as the ratio between the length of each loop and the punch velocity. The heat generated by friction forces is usually less significant due to the low friction coefficient between punch and sheet (84). Heat is also transferred by conduction from the nodes

at higher temperature to those at lower temperature. This exchange has been examined using Fouriers law:

$$q^{j}_{i,i'} = -K_m \cdot A_{i,j} \frac{T_{i,j} - T_{i',j}}{d_{i,i'}} \cdot \Delta t_{i,j} \tag{5.11}$$

where K_m is the thermal conductivity of the material, $d_{i,i'}$ is the distance between the nodes and $\Delta t_{i,j}$ is the time required to perform two consecutive loops. If the time required to perform two consecutive loops is excessively elevated the accuracy of this formula decreases. Therefore, in order to guarantee a good approximation of the heat exchange a reference time (t) of 0.05s was updated iteratively for small time variations. Finally, heat transferred by conduction between the punch and the node is estimated by taking into account the energy conservation; thus, the following equation was used to calculate $q^{j}_{i,p}$ in 5.4:

$$q^{j}_{i,p'} = \frac{T_{p,j} - T_{i,j}}{\frac{s_j}{K_m A_{i,j}} + \frac{s_p}{K_p A_{i,j}} + \frac{1}{h_c A_{i,j}}} - K_m \cdot A_{i,j} \frac{T_{i,j} - T_{i',j}}{d_{i,i'}} \cdot \Delta t_{i,j} \tag{5.12}$$

being K_p the thermal conductivity of the punch and h_c the thermal conductance coefficient between the materials in contact. Based on Eq.(12), the punch is heated during the process and, according to its thermal properties, the value of $q^{j}_{i,p}$ gives a positive or negative energy contribution to the temperature variation 5.4.

5.3 The Experimental Campaign

An experimental campaign, using a MazakTM QTurn 1000 CNC lathe, was carried out, by varying both material properties and process variables. Due to the machine tool peculiarities, the used geometry is the frustum of a cone, which represents a benchmark profile for ISF. The lathe was chosen to increase the range of strain rate to be investigated (85). Three different orders of magnitude (velocity) were analysed starting from 5 m/min, then 50 m/min, and ending with 500 m/min. Specific G codes were utilised in the writing of the ISO programs with the aim of linking the spindle rotation to two process characteristics. Firstly, linking the spindle rotation to the radius of the performed circumference, in order to maintain the relative linear velocity between punch and sheet for the whole

process constant. Secondly, linking the spindle rotation to the punch movement, along the z direction, to properly check the coil pitch which is the second analysed variable. With regard to the second point, the tool pitch ranges between 0.1 mm and 0.5 mm for the whole experimental investigation. The equipment used was custom designed in order to set the sheet inside the workspace. Circular rings were utilised to block the sheet (240 mm x 240 mm) with bolts on a frame, which was subsequently placed axially inside the spindle of the machine tool 6.8. A hemispherical punch, 15 mm in diameter, was inserted inside the tool holder. The working trajectory is therefore similar to the chips removal configuration but, here, the punch movement along the x-z plane is used to plastically shape the sheet. Finally, three bearings were placed between the punch and the holder to allow the punch rotation which takes place because of friction forces that are generated during the forming phase. The temperature trend during the process was measured by a thermal camera, which was properly set and placed inside the working area 5.5. The software FLIR Tool+, which allows for continuous tracking of the temperature variation, was utilised for the monitoring and the recording of temperature variation during the whole process phases. In order to properly record the measurements, the distance, as well as the inclination angle between the thermal camera and the working zone, have to be given as datum on the software interface. The emissivity of the worked surface is another important value to set for an appropriate temperature evaluation; according to this, it has to be highlighted that Molykote (MoS2) was brushed on the sheet both for lubricating and surface blackening. In general, the emissivity value of the black body is about one; taking this into account during the setting phase, the emissivity of the worked surface was set at about one and the obtained temperature values were compared with punctual data extracted by using a thermocouple. This device was located on the sheet immediately after the forming phase. By comparing the obtained temperature values to punctual data extracted by using thermocouple, an emissivity value equal to 0.95 was chosen as a proper coefficient settable for temperature monitoring. The investigation focused on the behaviour of two lightweight alloys widely utilised in the automotive and aerospace industries. Aluminum AA5754 and titanium Ti6Al4V were chosen because these materials are characterized by different thermal properties (Table 5.1).

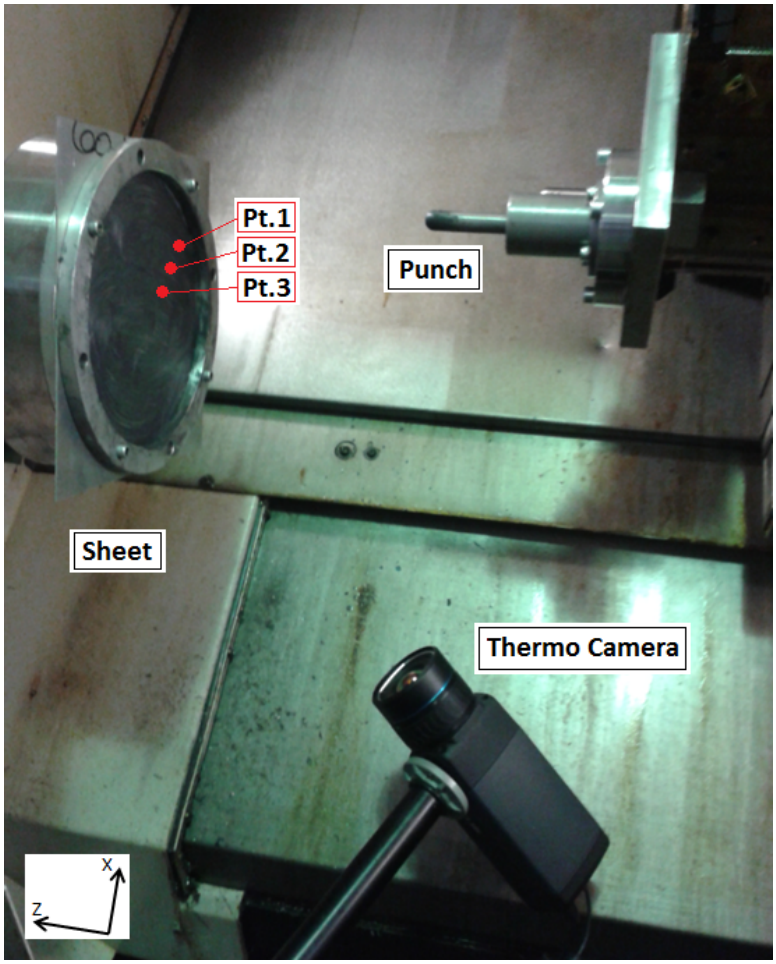


Figure 5.4: Thermo-camera placed inside the lathe working volume (Incremental Forming).

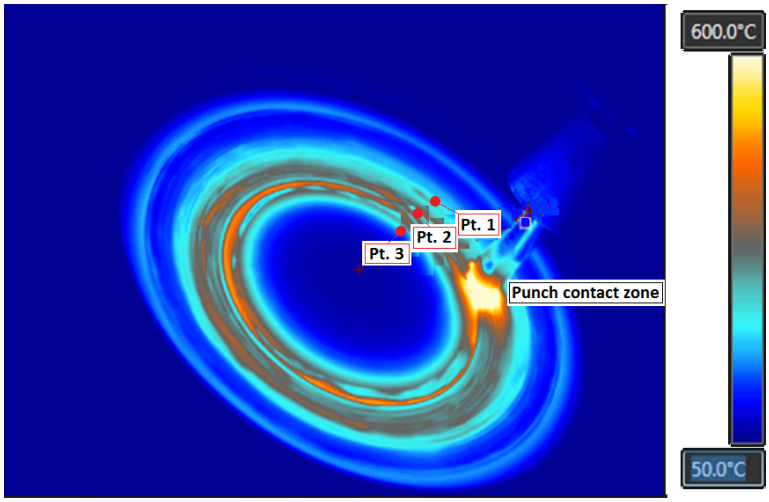


Figure 5.5: Intermediate step of the test carried out on Ti6Al4V by using V = 50m/min and p= 0.3mm.

Table 5.1: thermal properties of the analyzed materials (Incremental Forming)

	Unit	AA 5754	Ti6Al4V
Thermal Conductivity	[W/(m*K)]	125.00	6.70
Specific Heat Capacity	[J/g-C]	0.90	0.5263

ISF is not a stationary process; because of this the monitored temperature starts from room temperature values and it increases, coil by coil, for each punch pass. Three equally spaced points, also called nodes in the model definition, 6.8 were mapped on the work area to extract curves, which reach the maximum temperature at different times and are characterized by different trends. Finally, the maximum temperature, recorded for each point, was obtained at the punch passage 5.5; therefore, these maximum values can be considered as the forming temperature of the process.

5.4 Model Applications

To evaluate the quality of the proposed model the predicted results were compared with the ones measured through experimental tests in the laboratory. The

model was validated using two datasets for both investigated materials and under different process configurations. More in detail two process variables (tool pitch and tool speed) have been investigated keeping a fixed set-up for other process variables. Both the materials were formed with a slope angle respectively equal to 30 for Ti6Al4V and 35 for AA5754, which correspond to the 85% of the maximum one α_{max}, equal respectively to 35 and 42.

Table 5.2: Process set-up

Parameters	Value
Sheet Thickness	1.5 mm
Punch Radius	7.5 mm
Final Height	30 mm
Initial Diameter	180 mm
Tool Pitch	[0.1 0.5] mm
Tool Velocity	[5 500] m/min

5.4.1 Model Tuning

The model presented in section 5.2 requires a proper design of different constant parameters to reach a satisfactory accuracy. Some of these parameters were estimated according to the proposed literature overview; for instance, according to (86), the friction coefficient between punch and sheet was fixed equal to 0.2. The flow stress equation of the two materials, in agreement with Johnson and Cook formulation reported in Eq. 5.9, was estimated using 240 data (84) for the Ti6Al4V and 60 for the AA-5754 in which the interested parameters (strain, strain rate, temperature) are analyzed in the range summarized in Table 5.3:
The value of the constant parameters A,B,C,n and m, has been defined using an optimization model minimizing the average absolute percentage error obtaining the following equations:

$$\sigma_{AA-5754} = [270 + 154.3\varepsilon^{0.221}]\left[1 + 0.13\log\left(\frac{\dot{\varepsilon}}{597.2}\right)\right]\left[1 - \left(\frac{T_r - 20}{T_m - 20}\right)\right]^{1.34} \quad (5.13)$$

Table 5.3: Flow Stress datasets

Parameters	AA-5754	Ti6Al4V
Strain	[0 0.5]	[0 0.5]
Strain rate	[0.001 10]	[0.001 10]
Temperature	[20 280] C	[20 1000] C

$$\sigma_{Ti6Al4V} = [962.3 + 1041\varepsilon^{0.017}]\left[1 + 0.398\log\left(\frac{\dot{\varepsilon}}{31.2}\right)\right]\left[1 - \left(\frac{T_r - 20}{T_m - 20}\right)\right]^{0.78} \quad (5.14)$$

where T_m is equal to 581 for the AA5754 and equal to 1660 for the Ti6Al4V. Equations 5.13 and 5.14 were able to approximate the available data with a percentage error equal to 6.1% for the AA5754 and 2.3% for the Ti6Al4V. The lower accuracy reached using the Johnson and Cook model to fit the available data for the AA5754 can be explained based on two different factors:

1. the smaller dataset size;
2. the use of constant parameters is not suitable since the material shows different behaviors depending on temperature and strain rate as explained in previous paragraphs (87). Stress sensitivity to strain rate is important only at higher temperatures while could be neglectable at low temperatures (88).

Another important property that has to be defined is the thermal behavior of the material. The parameters necessary for the model are the specific heat and the thermal conductivity. Both values were kept constant for the AA5754 (89). The Ti6Al4V shows instead a different behavior under different working temperature (90); therefore, two regression curves 5.6 were defined using available data for different temperatures (84).

5.4.2 Verification and Validation

The proposed model was tested and validated comparing model and experimental results for eighteen process conditions (9 tests for each material). The maximum

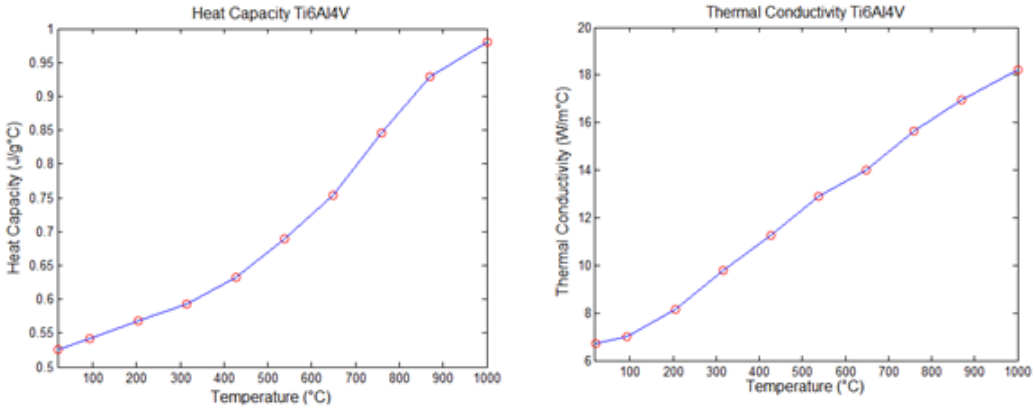


Figure 5.6: Thermal properties Ti6Al4V (Incremental Forming).

temperature achieved during the process for each experimental test is reported in Tables 5.4 and 5.5.

Table 5.4: Maximum Temperature of the sheet (AA-5754)

Velocity	5 m/min		50 m/min		500 m/min	
Tool Pitch	Test	Model	Test	Model	Test	Model
0.1 mm	40.2	56.11	99	107.65	172.73	186.76
0.3 mm	38.11	47.39	75.04	75.11	95.98	96.33
0.5 mm	36.04	31.31	58.19	53.48	93.04	68.87

Table 5.5: Maximum Temperature of the sheet (Ti6Al4V)

Velocity	5 m/min		50 m/min		500 m/min	
Tool Pitch	Test	Model	Test	Model	Test	Model
0.1 mm	187	202	582	567	585	617
0.3 mm	172	183	425	399	435	499
0.5 mm	162	166	271	322	431	417

The tables show that the mathematical model is able to predict the maximum temperature of the sheet for different process configurations. Table 5.6 lists the

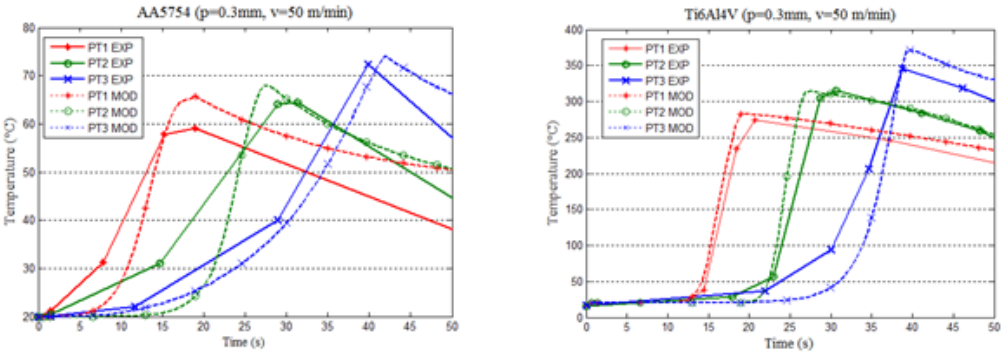


Figure 5.7: Numerical and experimental temperature trends for both the analysed materials.

value of different statistical criteria, which the model accuracy was evaluated with.

Table 5.6: Statistical criteria to define the model accuracy.

	AA5754	Ti6Al4V
R^2	0.93	0.95
MAD	$7,1$	23.1
$MAPE$	10.12%	7.28%

Similar results could probably be achieved by using machine learning techniques such as ANN, Kriging or SVR. However, the main advantage of the proposed model is the possibility of predicting the temperature without the need of a starting dataset. This characteristic is particularly relevant for this process since it would mostly be used for small batch production. Therefore, costs related to the execution of a few tests could substantially affect the process profitability. A different test was performed to verify the ability of the model to predict not only the final temperature but the entire behaviour during the process. Experimental and model results are shown in Fig. 5.7. The measurements were recorded for each material and they were taken on three different points located on the first third, at the center and on the last third of the worked area (see Fig. 5.5).

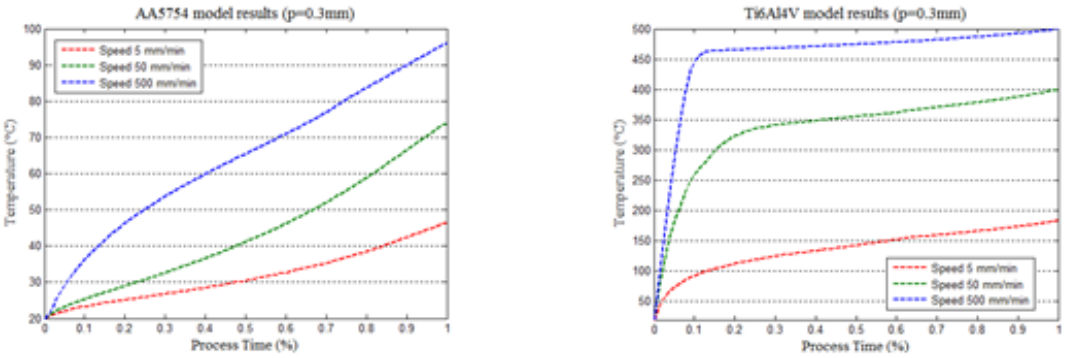


Figure 5.8: Temperature trend for different tool velocity (maximum value reached in each point).

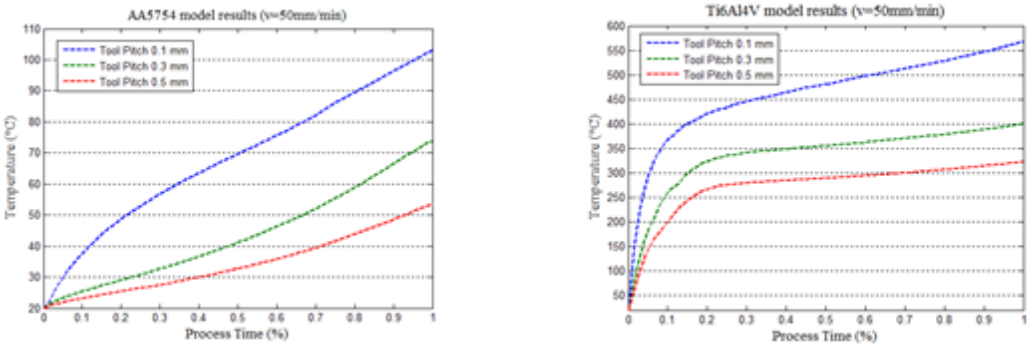


Figure 5.9: Temperature trend for different tool pitch (maximum value reached in each point).

In both cases, the same process configuration with a depth step equal to 0.3mm and a tool velocity of 50m/min was experimentally and numerically analysed. The graphs show the different temperature trends of the two materials. In order to understand the impact of different process factors on the final temperature of the sheet, different tests for each parameter were performed by only changing its value and keeping all the other process variables constant. The first test was performed to analyze the two variables investigated in the experimental campaign. The following graphs show the trend of the temperature variation on the sheet due to different process velocity (Fig. 5.8) and tool pitch (Fig. 5.9).

The maximum temperature increases by speeding up the process while maintaining the step depth constant. The reason for this relationship is attributed to the fact that there is less time available to dissipate the generated heat; the reduction of the heat loss is directly linked to the monitored temperature growth. Furthermore, using a higher speed, the temperature reaches its maximum value after a small number of punch passes remaining approximately stable for the rest of the process. (Fig. 5.8). As shown in Fig. 5.9, the maximum temperature significantly increases by raising the step depth; this is because the number of loops that involves one node is greater if smaller step depth values are set. Different behaviour between the two analysed materials can be observed as well; the titanium curves tend to converge after a smaller number of punch passes with respect to the aluminum ones. The main reason for this material behaviour is given by the different thermal conductivity that is significantly lower for Ti6Al4V; the heat exchanged between nodes with different temperature can be therefore neglected in this case. On the other hand, the maximum temperature will constantly increase specially for AA5754 due to the greater starting temperature of the internal nodes.

5.5 Conclusions

The use metamodeling and FE techniques to map complicated and costly manufacturing processes can be subjected to some limitaions. In some scenarios alternative simpler customized techniques can be defined to manage the relationship between the analyzed inputs and outputs. These techniques usually does not require experimental data so it can be a free decision support tool to reduce the elevated costs for process optimization. A mathematical model designed to solve the problem of temperature prediction of the sheet worked through Incremental Sheet Forming was then presented . An experimental campaign was performed for two materials. The proposed model was calibrated using the experimental data collected by using AA5754 sheets and, then, it was validated using new data for Ti6Al4V. This was done to verify the accuracy achieved for different thermal and mechanical properties proving the generality of the model. Satisfactory results,

in terms of R^2 and mean average percentage error ($MAPE$), were obtained as reported in Table 5.6, suggesting the suitability of the proposed thermo-mechanical model to predict the temperature distribution during the ISF process. The proposed model can be utilised for industrial applications introducing mathematical formulations instead of constant values, such as wall angle and process speed, arising from the chosen profile geometry. At the same time, the model as it is can be utilised to predict the temperature of the workpiece for a wide range of process conditions, as well as for different blank materials.

5. MANUFACTURING PROCESSES MODELING: AN APPLICATION TO TEMPERATURE PREDICTION ON INCREMENTAL SHEET FORMING

6

Adaptive KPIs prediction for manufacturing processes: an application to remote laser welding

Quality is considered a key factor of success for manufacturing companies especially in the last decade where customer needs must be met by fulfilling tolerances always lower (91). On the other hand, any company must obtain a sufficient profit on their products by minimizing its business costs. These two goal are usually in conflict, therefore, the process design have to be performed trying to find the right trade-off between different objectives. The first step necessary to pursue these objectives is to understand the mechanisms that link the process parameters under control and the performance indicators that the companies want to optimize. Different machine learning or statistical techniques, can be used to understand the process behavior changing different process parameters. Statistical analysis and data mining techniques are in fact particularly useful in the engineering world for improving the understanding of manufacturing processes reducing the efforts involved to acquire the knowledge required to make the correct process decisions. In this chapter is presented a new technique to define input-output relationships for manufacturing applications minimizing the number of required experiments taking into account process peculiarities and making

use of historical data on similar problems, if available. ANN and GA are prediction tools that can be used to map highly not linear relationships. However, both the techniques usually require a large number of experiments to reach a satisfactory accuracy. On the other hand, linear regression and fuzzy logic can be used to manage different parameters with a smaller number of data. Nevertheless both the techniques have several disadvantages that discourage their usage in this domain. Linear regression, in fact, can not map accurately the complex relationship that regulates the process, while fuzzy logic make usually use of linguistic variables based on subjective opinions. In general, all the techniques can provide some insights into different process phenomena and interactions between process parameters. However, to reduce the number of total experiments it can be used a different approach that allow using iteratively the information about the process behaviour previously collected. The ratio of this methodology is based on the hypothesis that response values for different materials or different process conditions follow common rules that allow introducing sequentially new parameters inside the analysis, reducing the number of required experiments to reach the same prediction performance (or equivalently, improve the accuracy using the same size of data) taking into account all the data collected in the past. In this work each key performance indicator y_k (henceforth referred as KPI) of the predicted responces vector $Y = \{y_1, y_2, .., y_m\}$ for a given process configuration $X = \{x_1, x_2, .., x_n\}$ is estimated using a weighted average transformation of previously observed similar values. We assume, so, that y_k is determined by the proximity to a similarity-weighted average for a certain similarity function. The remainder of this cahpter is organized as follow: a general introduction of the proposed methodology to perform the parameters selection process is presented in section (6.1); a case study and an application of the method is then reported in section (6.2); results and conclusions are also provided in section (6.3).

6.1 Methodology

Given a problem characterized by n input and m output parameters we want to identify an accurate transfer function $y_j = f_j(x_1, x_2, .., x_n)$ for each output $y_j \in Y$ minimizing the number of experiments to reach a good level of accuracy. The

research work is carried out using the main steps reported in the flow chart.

Each step of the proposed methodology will be discussed in this section. To sake

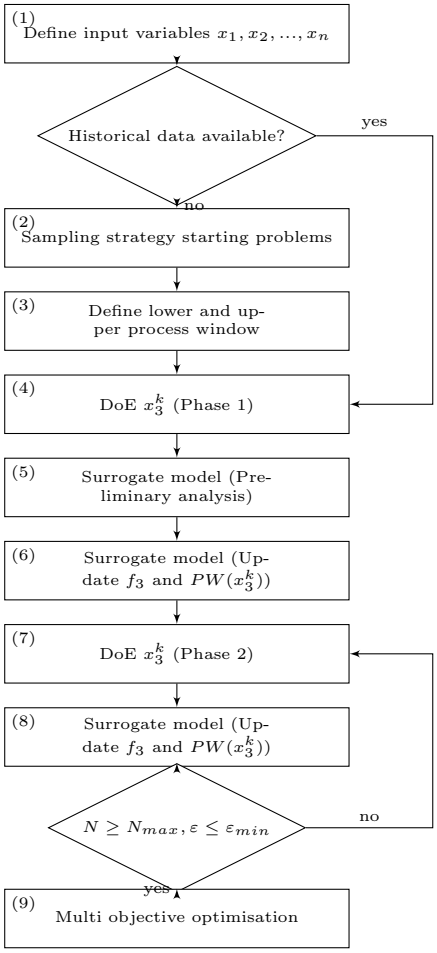


Figure 6.1: Optimisation procedure (Remote Laser Welding)

of clearness a nomenclature of all the symbols used in this section is here reported.

Nomenclature

X	set of input parameters
Y	set of output parameters
T	dataset size
t	dataset index
X_t	input parameters of the t-th configuration
Y_t	set of output parameters of the t-th configuration

$x_{i,t}$	i-th element of the X vector of the t-th datum.
$y_{j,t}$	j-th element of the Y vector of the t-th datum.
$\overline{y}_{j,t}$	predicted value of the j-th element of Y of the t-th datum.
x_i^{lo}	minimum allowed value of the i-th element of the X vector
x_i^{up}	maximum allowed value of the i-th element of the X vector
x_i^k	fixed value of the i-th element of the X vector
Ω	solution space
Ω_f	process window
E	prediction error
d_w	weighted Euclidian distance
f_w	similarity function

6.1.1 Definition of the input variables

First of all the main control parameters are determined. The experimental ranges for these parameters are selected trough preliminary tests and/or engineer's experience.

6.1.2 Sampling strategy starting problems

Selecting an experimental design is a key issue in building an efficient metamodel. The learning approach presented in this work is developed in a sequential way. At each iteration the set of parameters under analysis is split in three subsets:

1. S_1: parameters already investigated;
2. S_2: current parameter under analysis;
3. S_3: parameters not yet investigated.

At each iteration of the algorithm the following conditions holds:

$$S_1 \cap S_2, S_1 \cap S_3, S_2 \cap S_3 = \emptyset \tag{6.1}$$

$$S_1 \cup S_2 \cup S_3 = X = \{x_1, x_2, ..., x_n\} \tag{6.2}$$

During the first iteration the parameters are sorted according to their importance and these three set are initialized as follows

$$S_1 = \{x_1, x_2\} \qquad S_2 = \{x_3\} \qquad S_3 = \{x_4, ..., x_n\} \tag{6.3}$$

Two problems are then investigated. In the first problem S_1 variables are analyzed inside their range while S_2 variable is set equal to its lower value. The second problem has the same format but this time S_2 variable is set equal to its upper bound. S_3 variables are kept constant. Since no information about the process behaviour are available at this point a full factorial DoE with 3 levels and 2 factors is performed for both the problems.

6.1.3 Define lower and upper process window

The process window Ω_f is a subregion of the solution space Ω that satisfies a set of constraints introduced to monitor the final quality of the product or to avoid selecting incompatible solutions according to the available equipment. More in general a limitation may be fixed for each input/output parameter defining a lower and/or upper allowed value.

$$\Omega_f \subseteq \Omega : \begin{cases} x_i \leq b_i & i \subseteq n \\ x_j \geq b_j & j \subseteq n \\ y_z \leq b_i & z \subseteq m \\ y_k \geq b_j & k \subseteq m \end{cases} \tag{6.4}$$

The data generated with the DoE for the problems previously defined can be used to build a polynomial model and define the correspondent process windows. For the rest of the paper the notation $f_{x_i^k}$ and $\Omega_{x_i^k}$ will be used to represent the fitting model and the process window of the problem p_i^k defined as follows:

$$S_1 = \{x_1, x_2, ..., x_{i-1}\} \qquad S_2 = \{x_i\} \qquad S_3 = \{x_{i+1}, ..., x_n\} \tag{6.5}$$

where the value of the parameter x_i is set equal to k.

6.1.4 DoE Phase 1

To maximize the information about the most promising area of the solution space the DoE of a new problem should be designed trying to maximize the number of experiments selected inside the process window (keeping at the same time good space filling properties). In the early stage the only information about the response surface to estimate using a fitting model $f_{x_i^k}$ are given by the results achieved using the model $f_{x_i^{lo}}$ and $f_{x_i^{up}}$ setting the variable x_i to its lower and upper allowed value. A first process window is estimated defining a new fitting model for this problem through a linear combination:

$$f_{x_i^k} = \lambda f_{x_i^{lo}} + (1 - \lambda) f_{x_i^{up}} \tag{6.6}$$

$$\lambda = \frac{x_i^k - x_i^{lo}}{x_i^{up} - x_i^{lo}} \tag{6.7}$$

The assumption made to estimate the process window $\Omega_{x_i^k}$ is not so strong and reliable but it can be used as a starting point to define a first set of experiments. New experiments will be then performed according to the results achieved with a first surrogate model. The first set of experiments will be selected maximizing the intersite distance inside the process window. A different approach can be used if historical data on similar problem are available. This situation happen frequently in manufacturing applications (i.e. use of new materials with better mechanical properties, new machines etc.). In this case, without no further information, the process window is assumed to be identical to the one identified for the previous process conditions. Possible changes will be identified using the subsequent steps of the algorithm.

6.1.5 Surrogate Model Preliminary analysis

A preliminary analysis is used to identify the complexity of the function to approximate. This phase is performed to introduce a regularization parameter (or a penalty) to avoid over-fitting. Given a model $f_{x_1^k}$ so that one output component y_j of the problem p_i^k can be predicted through the equation $y_j = f_1(x_1, x_2, .., x_n)$

we want to identify the optimal value of a vector:

$$\beta = \{\beta_0, \beta_1, ..., \beta_{n+1}\} \tag{6.8}$$

that allow minimizing the error obtained on a second problem p_i^h using a model $f_{x_i^h}$ defined as follows:

$$y_j = \beta_0 f_1(x_1 + \beta_1, x_2 + \beta_2, .., x_n + \beta_n) + \beta_{n+1}. \tag{6.9}$$

The error E_0 obtained on the training data by this model will be used as an input parameters to define a risk coefficient. This parameter is estimated using a leave-one-out cross validation (80) and the R^2 criterion.

6.1.6 Surrogate Model

In this phase we develop an alternative, general theory of learning with similarity functions. The idea of our algorithm is based on the computation of a linear regression function in a high dimensional feature space where the input data are mapped via a nonlinear function. In the nonlinear setting, therefore, the optimization problem corresponds to finding the flattest function in the feature space. The methodology developed in this section can be considered a generalization of the kernel method (92). The theory behind kernel functions is based on the fact that many standard algorithms, such as SVR (93) and Perceptron algorithm (94), can be written in such a way that the only way they interact with their data is via computing dot-products on pairs of examples. Usually, kernel functions are used to fit a set of data (training data) trying to build a model that can be generalized for a second set of data (validation data). However, the problem addressed in this phase is different. Given two problems p_i^k and p_i^h for which two sets of data $X^k = \{X_1^k, X_2^k, ..., X_n^k\}$ and $X^h = \{X_1^h, X_2^h, ..., X_m^h\}$ are available we want to define a surrogate model $f_{x_i^h}$ through a kernel transformation (denoted henceforth as similarity trasdormation) on a second model $f_{x_i^k}$ previously defined. This approach was used as it allows obtaining the following benefits compared to classic kernel methods:

- ability to map complex shapes using simple transformations (overfitting risk reduction);

- use of historical data on similar problems.

Informally, a similarity function can be said to be good if points with similar labels are closer to each other than points with different labels (95). The similarity function can be any function that is estimated from the data, or that is chosen to fit the data. However, the similarity function should be as simple as possible to reduce the overfitting risk. Simple similarity transformations can be built considering a weighted Euclidean distance:

$$d_w(X_t, X_{t'}) = \sqrt{\sum_{i=1}^{n} w_i (x_{i,t} - x_{i,t'})^2} \tag{6.10}$$

The similarity function is expected to increase decreasing the value d_w reaching the value 1 for d_w=0. So typical similarity function can be:

$$\text{Exponential function: } f_w = e^{-d_w + d_0} \tag{6.11}$$

$$\text{Linear function: } f_w = \frac{1}{1 + d_w + d_0} \tag{6.12}$$

$$\text{Logarithmic function: } f_w = \frac{1}{\log(1 + d_w + d_0) + 1} \tag{6.13}$$

$$\text{Logarithmic function: } f_w = \cos(d_w + d_0) \tag{6.14}$$

where d_0 is used to map translational changes of the response surface.

6.1.6.1 Model Optimisation

Given a similarity function the output of a new point X_j^h is estimated trough the following equation:

$$y_{j,t}^h = \sum_{t'=1}^{T'} \alpha_{t'} f_w(X_t^h, X_{t'}^k) y_{j,t'}^k \tag{6.15}$$

The optimal value of the vector $\alpha = \{\alpha_1, \alpha_2, .., \alpha_{T'}\}$ is determined through a mathematical model solved by means of genetic algorithms:

$$\min z = \lambda E + \sum_{t'=1}^{T'} \alpha_{t'}^2 \tag{6.16}$$

$$\overline{y}_{j,t}^h = \sum_{t'=1}^{T'} \alpha_{t'} f_w(X_t^h, X_{t'}^k) Y_i^k \qquad \forall t = \{1, ..., T\} \tag{6.17}$$

$$e_{j,t}^h = \overline{y}_{j,t}^h - y_{j,t}^h \qquad \forall t = \{1, ..., T\} \tag{6.18}$$

$$E = \sum_{t=1}^{T} (e_{j,t}^h)^2 \tag{6.19}$$

The parameter λ used in equation (6.16) is inversely proportional to the risk coefficient E_0 determined in section 6.1.5. Moreover E_0 is used to tune a parameter α_{max} that determines the maximum number of variables which may assume any value other than 0.

6.1.7 DoE Phase 2

The similarity function technique will be used to make a new estimation of the response surface of the problem defining a new fitting model $f_{x_i^k}$. The correspondent process window $\Omega_{x_i^k}$ should provide a better approximation compared to the one defined in section 6.1.4. However, the prediction abilities of the new tool could be not yet satisfactory due to the lack of data in some area of the process window. This is particularly true in case of a significant deviation between the predicted and the real process window. A new set of experiments will be defined to reduce this localized weakness of the tool. Each new experiment is designed using the results of the current metamodel until the accuracy of this tool can be considered statistically satisfactory. More in particular, at each iteration a new experiment is performed considering new samples localized in a, so called, ϵ-tube (96). The ϵ-tube is composed by all the possible solution $X^\epsilon \in \omega : |P(x \in \Omega_{x_i^k}) - P(x \notin \Omega_{x_i^k})| \leq 2\epsilon$. Also in this case, the new set of points to analyze will be selected trying to maximize the distance d_{tot} between the new

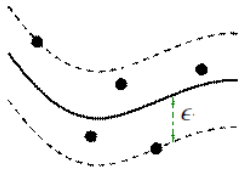


Figure 6.2: ϵ-tube

point and the previous training data.

$$\max d_{tot} = \min_{t' \in T} \sum_{i=1}^{n} \frac{|x_{i,t} - x_{i,t'}|}{x_i^{up} - x_i^{lo}} \tag{6.20}$$

where $x_{i,t}$ is the value of the i-th input parameter of the experiment t. The ϵ value will be gradually reduced during the iterations. The motivation of this choice is related to the fact that the accuracy of the prediction about the process window boundaries will gradually increase so the subsequent iterations will lead to small changes.

6.1.8 Stopping criteria

According to our preliminary testing, we found profitable to use two criteria to stop the algorithm. The algorithm is arrested if:

1. the number of experiments performed exceeds a maximum threshold;
2. the accuracy of the predictive tool is greater than a given threshold ε_{min}.

6.1.9 Multi Objective Optimisation

The response surfaces obtained using the regression technique described in the previous chapter can be used to model a mathematical formulation of the problem according to the user needs. The parameters selection is usually a problem that require a good trade-off between different objectives. The use of a multi objective algorithm is therefore necessary. One of the most popular algorithms to solve multi objective problems is the NSGAII. This algorithm is characterized by a $O(MN^2)$ complexity, where M is the number of objective functions and N the

population size. This new algorithm incorporates also elitism ensuring the survival of the best solutions. The algorithm can be briefly described as follow: first of all a random population is generated. This population is then sorted according to non-domination in several fronts. The first front is totally non-dominated in the current population while second front is dominated only by the first front. Each individual in each front has a value of rank (fitness) based on the front to which it belongs. A function denoted as *crowding distance* is also considered in the fitness value. The crowding distance is a measure of the distance of an individual from its neighbors. Large average crowding distance create a population with a greater diversification. The variant of the NSGAII algorithm used in this work make use of the simulated binary crossover (43) for the crossover phase and polynomial mutation (97) for the mutation phase. One of the most important aspects of this technique is given by the choice of two parameters: the population size and the number of the generations. A too high value of these two parameters leads to an increase of the computational time. However, decreasing the number of generations the quality of the solutions found deteriorates while decreasing the size of the population the maximum number of optimal solutions found is lower.

6.2 Case Study: Remote laser welding

The proposed technique was used to optimize a set of process control parameters for remote laser welding. Laser welding is a promising technology to weld aluminum alloys or steels because of its potential high productivity, manufacturing flexibility and the ease of automation (98). As a consequence, it has been increasingly utilized in industrial sectors with different applications in the automotive, marine and electronic field. Currently, resistance spot welding is the most used joining process in the automotive field. However, it has been proved that laser welded components offer better mechanical properties (99). Furthermore, the desired user specifications can be reached with a superior repeatability. Despite of the fact that laser welding offers a great flexibility it is also true that the amount of process parameters that have to be monitored is higher. Laser welding process parameters play a crucial role on the final quality of a weld joint. A comprehensive analysis which takes account of all the interactions is not viable due the

factorial relationship between accuracy and number of analyzed parameters. The proposed methodology tries to reduce this trend, however, to keep a sufficient prediction capabilities the following steps still needed to:

1. select the most important process parameters to analyze;
2. define a fixed value for the other process parameters;
3. define a response surface to define the joint quality after different process conditions.

The joint quality can be evaluated in terms of properties such as weld bead geometry and/or mechanical properties (100). These mechanical properties should be controlled in order to obtain good welded joints. It is not always immediate to define a set of quality indicators since the evaluation of a welded joints could be different depending on the final customer and/or the specific application. Therefore, according to the particular context it is necessary to:

- define a set of key performance indicators (KPIs);
- define a desired value (or a range) of each KPI.

Many studies use the s-value or interface weld width as a relevant characteristic of the weld geometry (101, 102). Therefore, the s-value serves as one of the measurements that translate strength needs into product features that are easily measured through metallographic images.

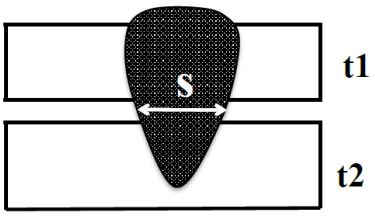


Figure 6.3: S-value (Remote Laser Welding)

The minimum allowed s-value is a function of the thickness of the thinnest sheet

being welded. According to Ford Standards (103), interface weld width should be equal to or greater than 90% of the thickness of the thinnest sheet.

$$s \geq 0.9 \times t_{min} \tag{6.21}$$

Similar to the s-value, penetration is also a relevant quality criterion in laser welding (104). In actual fact, laser welding has become popular due to its deep penetration capabilities (105) and as part of the weld bead geometry, it is vital to determine the mechanical properties of the weld (106).

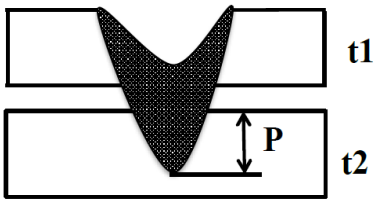


Figure 6.4: Penetration (Remote Laser Welding)

The minimum penetration limit has been defined as 30% of the lower sheet thickness.

$$P \geq 0.3 \times t_2 \tag{6.22}$$

Top surface concavity, defined as TSC in Fig. 3, is a depression of the welds top surface that extends below the top surface of the upper sheet (107). If present, it could lead to a reduction of the s-value, which will decrease the mechanical properties in terms of strength, reliability and durability. Furthermore, it could negatively impact the appearance of the weld. This concavity might happen as a result of gravity (108), reduction of surface tension during welding (103), or a large gap between the two mating parts (109).

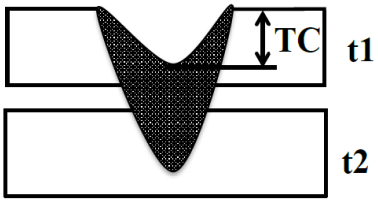


Figure 6.5: Top Surface Concavity (Remote Laser Welding)

Although the target is not to have top surface concavity along the weld seam, the maximum accepted value (TC) is 50% of the upper sheet thickness.

$$TC \leq 0.5 \times t_1 \tag{6.23}$$

Similar to top surface concavity, bottom surface concavity might also occur. Even though it is not considered by industry standards, laser welding projects (e.g TSB Project) and international standards have found that it can be present along with top surface concavity as seen in Fig. 4.

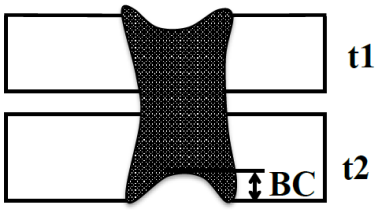


Figure 6.6: Bottom Surface Concavity (Remote Laser Welding)

Although the limit for bottom surface concavity has not been defined in industry standards, therefore in this work will be considered the same negative effects on strength and appearance of the weld of TSC. Hence,the same 50% limit is applied.

$$BC \leq 0.5 \times t_2 \tag{6.24}$$

6.2.1 Literature Review

The research of the optimal weld input parameters for new welded products satisfying the required specifications is usually a time consuming task. Therefore, the use of a trial and error approach is not practicable due to the amount of resources necessary in terms of materials and labour time. To solve this problem, several numerical models have been developed to find mathematical relationships that can be used to select the optimal process configuration to obtain the desired quality level. In the last years, different models have been successfully used to study the laser partial penetration welding process. Benyounis and Olabi (110) showed how to perform the parameters selection process using various optimization methods

reducing time, materials and labor effort. M. Khan (111) presented an experimental design approach to process parameters optimization for the laser welding of martensitic AISI 416 and AISI 440FSe stainless steels in a constrained overlap configuration. Casalino (112) used Taguchi and artificial neural network to perform a statistical analysis of the effects of welding parameters on the shape of the welded area and the hardness variation in Ti6Al4V alloy. A similar approach (113) was used to find out the optimal levels of welding speed, laser power and focal position for CO_2 keyhole laser welding of carbon steel. Reisgen et al (114) proposed a three factor three level Box-Behnken statistical design to optimize and evaluate the main effects of laser power, welding speed and focal position on the weld penetration, weld width, tensile strength and welding operation costs. Through this analysis has been pointed out that welding speed and laser power are the most critical parameters that affect the weld penetration. This result is due to the fact that an increase on the power level leads to an increase in the heat input. An opposite approach can be defined for the welding speed. Therefore to achieve the maximum penetration the laser power has to be maximized while the welding speed minimized. Kim (115) conducted an analysis to verify the feasibility of applying laser welding on alluminium alloy to a car body under various welding conditions. The radiation angle of the laser incidence angle, laser power, and welding speed were used as control variables. Analysis of variance (ANOVA) was conducted to identify the effect of the process variables on the tensile shear strength. A final model was then developed using different regresion techniques. Palani (116) showed an application of the Taguchi method to find the optimal welding process parameters for cladding of austenitic stainless steel. According to Galantucci (117) defects in welds are mainly induced by material characteristics and can be strongly influenced by the choice of process parameters. The defects taken into account in his analysis are pores, undercut and excessive sags. According to this analysis laser power and welding speed are the most significant parameters that have to be monitored to reduce these defects since a concentrated energy input and the high speed of the process can reduce pore formation. The mathematical relationship between process parameters has been detected using a feedforward neural network. A quality index was then defined to select the

optimal process parameters.

6.3 Experimental Test and Results

In this section, we present the results of numerical studies we performed to prove the effectiveness of the proposed algorithm. The application of the different algorithm steps are discussed highlighting the intermediate results. The algorithm was implemented as a single thread code in Matlab and all tests are performed on a desktop computer equipped with an Intel Core i3 processor with 2.2 GHz, 6 GB RAM, and running Windows 7 Professional. The above introduced method was used to find the optimal laser welding process configuration for different combination of materials. More in particular, the technique was used to:

- develop a model between RLW process parameters an quality performance indicators
- select the optimal process parameters for different combinations of thickness and material of the two mating parts;

In order to generalize the results making several tests under different conditions, we found profitable to use artificial data that properly describes the domain under analysis. To this aim a response surface developed in a previous work (118) was used as a black box.

6.3.1 Definition of input Variables

As previously described there is a huge number of parameters that can affect the quality of a welded joint. It would be impossible to define an accurate model with such a great number of controlled parameters. Moreover, some of them are measurable but not controllable in practice. According to the literature overview and the preliminary analysis undertaken, four independently controllable process parameters, which have a significant influence on the key performance indicators defined in section 1, have been selected:

Table 6.2: Process parameters

Process Parameter	Units	Lower Value	Upper Value	Id variable
Speed	m/min	1	4	x_1
Gap	mm	0.05	0.5	x_2
Laser Power	kW	2	4	x_3

- *Laser Power* (kW): the time rate at which energy is emitted, transferred, or received;
- *Welding Speed* (m/min): speed at which the weld is created over the work-piece;
- *Gap* (mm): vertical distance between mating parts;
- *Stack-up*: combination of materials and thicknesses.

The range identified for each process parameters is reported in Table 4: The lower value for the laser power was chosen according to the minimum level of power required to reach a satisfactory level of penetration. The upper value is instead constrained by the maximum value achievable with the available equipment. The welding speed range was chosen according to the laser power range in such a way to reach a suitable level of energy. The allowed ranges for the part-to-part gap it was determined based on the fact that a gap value of 0.1 mm to 0.3 is usually sufficient to overcome porosity, but if this range is exceeded it could lead to quality issues (119). To undesrtand clearly this trend we expanded the analyzed rang to 0.05-0.5. Five different stack-ups (id variable x_4) have been analyzed in this work. The main features of each stack-up are reported in Table 5.

6.3.2 Sampling strategy starting problems

This step is performed only once at the first iteration. The three set S_1, S_2 and S_3 have been defined in the following way:

$$S_1 = \{Speed, Gap\} \qquad S_2 = \{Power\} \qquad S_3 = \{Stack - up\}$$

Table 6.3: Stack-up features

	Upper Part		Lower Part	
	Material	**Thickness**	**Material**	**Thickness**
Stack-up 1	DX56D+Z	0.75	DX52D+Z	1
Stack-up 2	DX56D+Z	0.75	DX54D+Z	1.8
Stack-up 3	DX56D+Z	1	DX54D+Z	0.7
Stack-up 4	DX56D+Z	1	DX54D+Z	1
Stack-up 5	DX56D+Z	1.2	DX53D+Z	1.8

Two problem (p_3^{2kW} and p_3^{4kW}) are defined using the following set of variables. In the first problem the laser power is set equal to 2kW while in the second is set equal to 4kW. The upper value of this variable has been chosen according to machine constraints while the lower value has been defined according to a qualitative analysis since with a lower value of these parameters it would be difficult to reach a sufficient level of penetration. Both the problem have been analyzed using a full factorial DoE.

6.3.3 Define lower and upper process window

The process window of both the problem has been defined using a polynomial model. The degree of the model was chosen maximizing the R^2 using a leave-one-out cross validation. Moreover the irrelevant variables interaction have been neglected using the ANOVA techniques. As example, the results achieved for the prediction of the first KPI (penetration) are reported in the following tables:

Table 6.4: Polynomial model p_3^{2kW} results

Degree	**Number of terms**	**Intercept**	R^2
1	2	yes	0.442
2	5	yes	0.851
3	8	yes	0.913

Table 6.5: Polynomial model p_3^{4kW} results

Degree	Number of terms	Intercept	R^2
1	2	yes	0.523
2	5	yes	0.867
3	9	yes	0.929

These models have been used to define the correspondent process windows based on the following set of constraints.

$$\Omega_{x_3}^{2kW}, \Omega_{x_3}^{4kW} \subseteq \Omega : \begin{cases} penetration \geq 0.3 \\ s - value \geq 0.675 \\ TC \leq 0.375 \\ BC \leq 0.5 \end{cases} \tag{6.25}$$

6.3.4 DoE Phase 1

The response surfaces of these two power values have been used to define the design of experiments to estimate the new response using a laser power equal to 3kW. A first prediction of the process window of this problem was defined using a linear combination between the two response surfaces available.

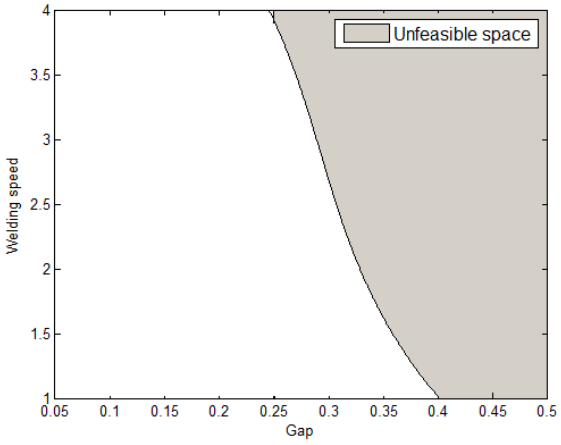


Figure 6.7: Process window

A first set of 8 experiments was designed inside this process window according to

the following rules.

1. Feasibility constraints: gap level equal to a multiple value of 0.05.

2. Maximum intersite distance.

In practice, the feasibility constraint is added because the control of the part-to-part gap solution is achieved through the use of dimples or shims with a pre-determined thickness to force the gap between the mating parts. The process configuration analyzed and the correspondent results are presented in Table 8.

Table 6.6: DoE Phase 1

Speed	Gap	Penetration	S-value	Top Concavity	Bottom Concavity
3.62	0.05	0.65	1.34	0.02	0.3
4	0.25	0.16	0.41	0.06	0.02
1	0.35	0.04	1.34	0	0.46
1	0.05	0.45	1.33	0	0.46
2.81	0.1	0.52	1.45	0.03	0.44
2.48	0.25	0.3	1.55	0.06	0.44
3.29	0.2	0.03	0.32	0.11	0.16
1.62	0.15	0.87	1.55	0.06	0.44

6.3.5 Surrogate model preliminary analysis

The surrogate model of the problem p_i^{2kW} was used as reference point to develop a new model for the problem p_i^{3kW} using a combination of simple transformations. The only changes allowed in this phase are translations along the x and y axes using a different magnitude level.
As explained in section 6.1.6 this model will not be used for prediction purposes both only to tune some parameter of the surrogate model.

Table 6.7: Surrogate model (Preliminary analysis)

KPI	R^2 **Phase2**
Penetration	0.517
S-value	0.421
Top Concavity	0.623
Bottom Concavity	0.578
Average	0.534

6.3.6 Surrogate model

The results achieved using the preliminary analysis was used to define the λ parameter in the model (6.16)-(23) defined in section 6.1.6.1. Different similarity transformations were tested on this problem trying to map the relationship between the problems analyzed maximazing the value of the information available. The results of this phase are summarized in the next table:

Table 6.8: Surrogate model (Best Similarity Function Accuracy)

KPI	R^2 **Phase2**	Similarity Function
Penetration	0.721	Exponential
S-value	0.741	Exponential
Top Concavity	0.753	Exponential
Bottom Concavity	0.664	Exponential
Average	0.772	Exponential

As it is possible to see, the best results are achieved using the exponential function for all the KPIs. This model will be used to develop a new DoE strategy.

6.3.7 DoE Phase 2

The results obtained were used to define a second set of experiments. The experiments have been selected iteratively around an ϵ-tube (ϵ=0.05) until the accuracy

was greater than a predetermined treshold. The ϵ value will be iteratively modified at each iteration t according to the residual number of maximum experiments available using the following formula:

$$\epsilon_t = \epsilon_0 - \epsilon_0 \frac{T_{current}}{T_{max}} \tag{6.26}$$

Where $T_{current}$ and T_{max} are respectively the current and the maximum number of performed experiments. Also in this case the selected process configurations and the KPIs value are presented through a table.

Table 6.9: DoE Phase 2

Speed	Gap	Penetration	S-value	Top Concavity	Bottom Concavity
3.58	0.15	0.20	1.27	0.08	0.13
2.2	0.15	0.16	1.55	0.04	0.45
2.48	0.05	0.33	1.34	0.04	0.46
2.56	0.2	0.19	1.48	0.12	0.40
3.05	0.15	0.41	1.54	0.08	0.36
2.85	0.15	0.42	1.55	0.08	0.41
3.21	0.1	0.57	1.45	0.04	0.38

This second set of data was used to refine the response surfaces and therefore the process window. The accuracy of the model before and after this second phase is reported in the following table:

Table 6.10: Model Accuracy (Phase 2)

KPI	R^2 Phase2	R^2 Phase2
Penetration	0.819	0.902
S-value	0.831	0.941
Top Concavity	0.718	0.798
Bottom Concavity	0.913	0.965
Average	0.820	0.902

6.3.8 Stopping criteria

Two stopping criteria have been used in this analysis as described in section 6.1.8. The accuracy of the tool was measured using the R^2. The threshold value of this parameter was fixed equal to 0.9. To set the maximum number of experiments we used the polynomial method with full factorial DoE as a benchmark. The number off required experiments to achieve an average R^2 greater than 0.9 was equal to 320 experiments. The following procedure was repeated for all the combination of thicknesses and materials. However the starting point of the new analysis was performed using the following sets:

$$S_1 = \{Speed, Gap, Power\} \qquad S_2 = \{Stack - up\} \qquad S_3 = \{\emptyset\}$$

where the process window of the previous stack-up was used to select the first set of experiments (DoE phase 1).

6.3.9 Multi-Objective analysis

The surrogate model defined in the previous section was finally used to determine the optimal process configuration for each stack-up. The major objectives defined are given by the minimization of the operative costs that are affected by the laser power and the welding speed (labour cost). Therefore the following mathematical formulation has been defined.

$$\min \text{Power} \tag{6.27}$$

$$\max \text{Speed} \tag{6.28}$$

$$\text{Penetration} \geq 0.3t_1 \tag{6.29}$$

$$\text{S-value} \geq 0.9t_{min} \tag{6.30}$$

$$\text{Top Concavity} \leq 0.5t_1 \tag{6.31}$$

$$\text{Bottom Concavity} \leq 0.5t_2 \tag{6.32}$$

$$P([\text{Speed,Gap,Power}]) \in \Omega) \geq 0.9 \tag{6.33}$$

This model have been solved for each stack up using the NSGAII algorithm. The different efficient frontiers are represented in the following figure:

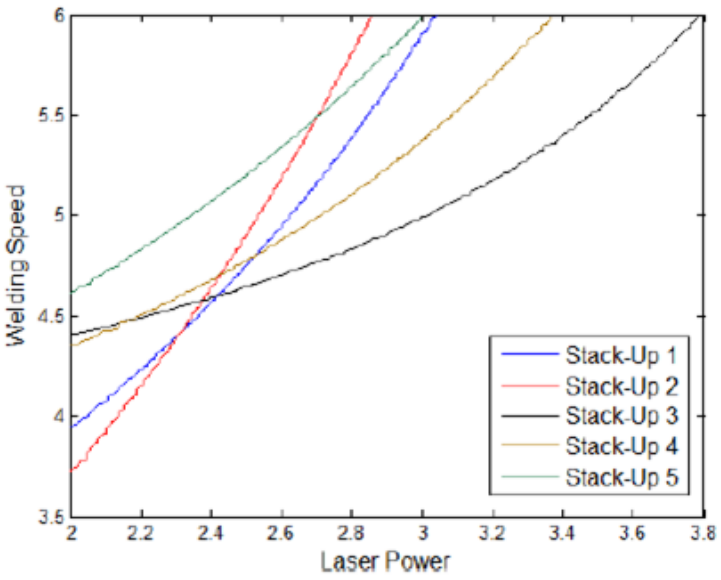


Figure 6.8: Efficient frontiers

6.3.10 Verification and Validation

Once the model has been built, estimating its prediction abilities is an important issue. To prove the capabilities and the effectiveness of our proposed methodology we compared our results with other benchmark approach usually used in machine learning applications. Two comparative tests have been performed. In the first one we set a minimum accuracy level ($R^2 \geq 0.9$) and we measured the minimum number of required experiments to satisfy this constraints. The results

of this test are reported in Table 13.

Table 6.11: Comparative analysis Test 1(minimum number of experiments)

	NN	LinReg	RSM	Kriging	NFuzzy	SF
FF	320	null	320	320	320	246
LH	243	null	296	267	269	208
CCD	null	null	null	null	null	null
Rd	292	null	331	256	312	213
AS	210	null	224	245	267	178

As it is possible to see from the table the best overall results are obtained using the proposed approach based on the similarity function method as prediction tool and the adaptive sampling technique to collect the data to analyze. A different approach was used for a second test. We set a constant number of 30 experiments for each stack-up and we measured the correspondent R^2. Also in this case this value is calculated as the average for the different KPIs. The results of this second test are highlighted in Table 14.

Table 6.12: Comparative analysis Test2 (maximum accuracy)

	NN	LinReg	RSM	Kriging	NFuzzy	SF
LH	0.791	0.410	0.757	0.783	0.729	0.879
CCD	0.771	0.412	0.795	0.756	0.708	0.845
Rd	0.789	0.411	0.791	0.747	0.712	0.897
AS	0.846	0.409	0.812	0.856	0.873	0.912

The table shows that also in this case the tool with the best prediction abilities is the one proposed in this work. The main advantage of the proposed algorithm is given by the possibility to use a response surface of a similar problem previously developed to reduce the number of required experiments. The ability of this algorithm is therefore surely affected by the relationship between the two problems under analysis. A sensitivity analysis has been developed to understand strength and possible limitations of this methodology. To this aim, for each input $x_i \in X$

the following couple of problems have been analyzed:

$$S_1 = \{x_1, x_2, ..., x_{i-1}, x_{i+1}, ..., x_n\} \qquad S_2 = \{x_i^{lo}\} \qquad S_3 = \{\emptyset\} \tag{6.34}$$

$$S_1 = \{x_1, x_2, ..., x_{i-1}, x_{i+1}, ..., x_n\} \qquad S_2 = \{x_i^{up}\} \qquad S_3 = \{\emptyset\} \tag{6.35}$$

For each problem we measured the R^2 achieved for different dataset size comparing the results with the ones obtained with a polynomial model as shown in the next graph:

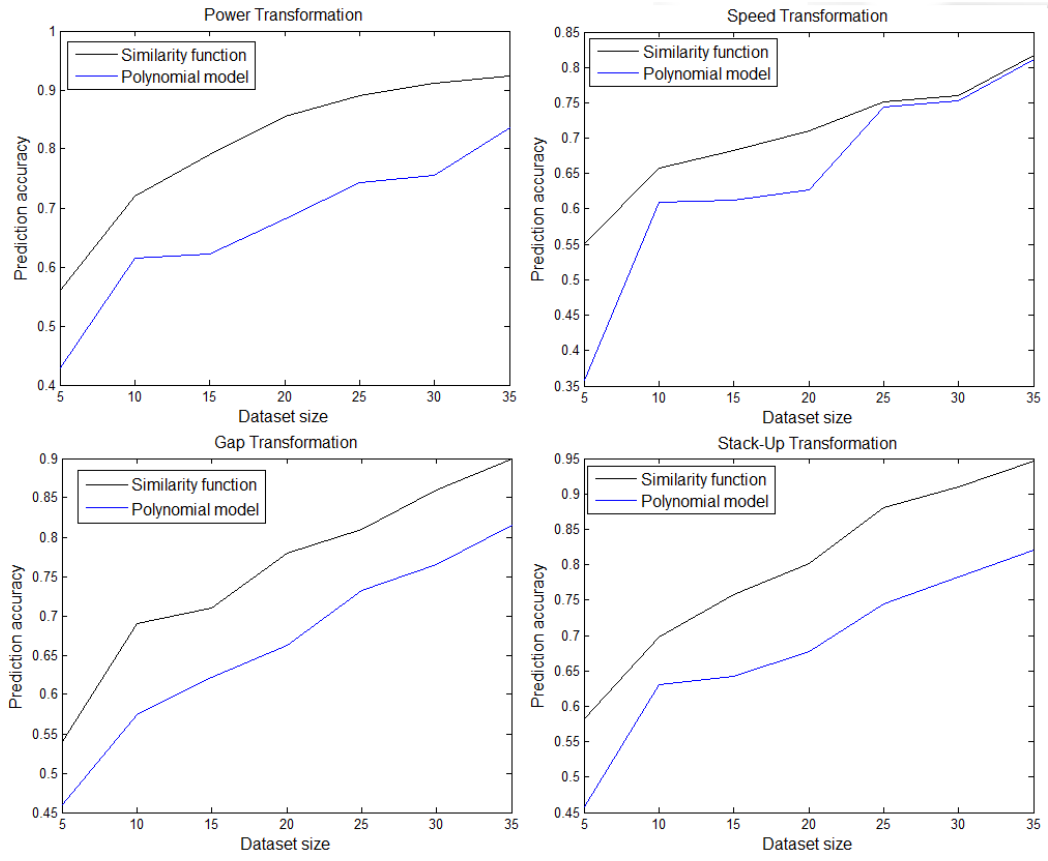


Figure 6.9: Sensitivity analysis

It is possible to see that in each case the proposed algorithm reach a superior level of accuracy for any dataset size. However, the improvement achieved using a response surface with a different speed value is significantly lesser. A possible interpretation of this result is given by the fact that in case of excessive significant

interactions the response surface of the previous problem can not be used as a suitable starting point.

6.4 Conclusion

In this chapter has been presented a new algorithms to automatically determine input-output relationships in manufacturing applications. This method was tested and validated on a laser welding problem taking into account three input and five output variables. The experimental results demonstrate that the proposed method based on an adaptive sampling DoE and a prediction tool based on shape similarities between a set of surfaces lead to better results compared to benchmark approaches used for this kind of problems. The proposed methodology could be used to solve problems on different domains. However, this technique is particularly effective if at least one of this conditions is satisfied:

- the influence of some input parameters is significantly greater;
- restricted interest in a particular subarea of the solutions space;
- some interactions between input parameters are negligible.

All these conditions are exploited in order to obtain the desired information reducing the number of required experiments which is directly related to the cost and time necessary to perform the parameter selection process.

6. ADAPTIVE KPIS PREDICTION FOR MANUFACTURING PROCESSES: AN APPLICATION TO REMOTE LASER WELDING

7

Multi-Objective Optimization

7.1 Introduction

In the last years evolutionary algorithms have created more interest among researchers and manufacturing engineers for solving multiple-objective problems. The objective of this chapter is to provide comprehensive understanding and also to give a better insight into the applications of solving multi-objective problems using evolutionary algorithms for manufacturing processes. Many papers have focused on the combination of FE or phisical experiments and optimization technology to solve different problems in forming or machining applications and it has been found that these processes are a typical multi-objective problem (MOP) with conflicting relationships between multiple objective functions.

The method of previous studies used to solve the MOP by single-objective optimization method, such as the weight coefficients approach which combined multiple objective functions into one. Unfortunately, it is a big trouble for users to determine the value of these coefficients, although these coefficients are crucial for the results of optimization. What is more, another problem is each set of coefficient combination can only acquire one optimal solution, and it is hard to make sure whether the solution achieved is an optimal one. However, in engineering design domains, more and more attentions have been paid on multi-objective evolutionary algorithm, which mimics the natural selection process in which a superior creature evolves whilst inferior creatures fade out from their population as generations go by. Many advantages of MOGA are very attractive, such as the

capability of exploring a large design space and the merit of none gradients information is needed. But, the most important one is MOGA can compute multiple independent objective functions simultaneously in one optimization run without converting multiple objective functions into a single objective by weighted linear combination. For these reasons, MOGA can be used for higher non-linear MOPs which often arise in manufacturing applications. The most important feature of evolutionary algorithms is that it can successfully find globally optimal solutions without getting restricted to local optima. Another advantage of these algorithm is that it can be combined with sophisticated metamodeling techniques without significantly affect the required computational time. As it will be shown in the next sections this is not possible using other benchmark technique that require a formal mathematical description of the problem. Among the evolutionary algorithms, the third version of the generalized differential evolution (GDE3) has emerged as most efficient algorithms for solving multi-objective problems in manufacturing processes. A new methodology based on this algorithm is deeply described and analyzed. The chapter concludes by showing an application of the proposed evolutionary algorithms on the porthole extrusion process in which multiple objectives has been considered.

7.2 Multi-Objective problem optimization

Multi-objective optimization refers to the optimization of problems with a formulation that involves at least two objective functions. The solution of these problem is usually not unique but is composed of a set of non dominated solutions because there is not a solution that minimizes all the objective function simultaneously. A solution of this set is called Pareto optimum. A multi objective optimization problem can be described as follows:

$$\min \{f_1(x), f_2(x), ..., f_n(x)\} \tag{7.1}$$

$$x \in \Omega \tag{7.2}$$

where $n \geq 1$ is the number of objective function and Ω is the objective space of the problem. The objective space Ω is given by the set of solution that satisfied all the constraints of the problem:

$$\Omega = \{x \in R^m : h(x) = 0, g(x) \leq 0\} \tag{7.3}$$

where $h(x)$ is the set of equality constraints and $g(x)$ the set of inequality constraints. An important concept in the multi-objective problems is the dominance criterion. A solution x_1 dominates a solution x_2 if $f_i(x_1) \leq f_i(x_2)\ \forall i \in \{1, ...n\}$ with at least one inequality. The dominance criterion is used to express the definition of Pareto optimum solution. A Pareto optimal solution can be:

1. *weak Pareto optimum*: a solution x_w is weak efficient if there is no $x \in \Omega$: $f_i(x) < f_i(x_w)\ \forall i \in \{1, ...n\}$

2. *strict Pareto optimum*: a solution x_s is strict efficient if there is no $x \in \Omega$: $f_i(x) \leq f_i(x_s)\ \forall i \in \{1, ...n\}$ with at least one inequality.

The set of Pareto optimal solutions is called efficient frontier or Pareto front.

7.3 Benchmark techniques

7.3.1 Weighted sum method

The weighted sum method is the most simple multi-objective optimization techniques and probably the most used. In this approach all the objective functions are multiplied for different weights w_m, usually in the range [0,1] with $\sum_{m=1}^{M} w_m = 1$, and combined into a single function. To assign the same value. To assign the same importance to the different objectives, the functions have been normalized in the range [0,1]. In this problem the new objective function will be therefore:

$$\min\ z = \sum_{m=1}^{M} w_m f_m(x) \tag{7.4}$$

It has been prove that the solution to the problem presented is Pareto-optimal if the weight is positive for all the objectives. However this not means that all

the optimal solution can be reached using a positive weight vector. The main disdvantages related to this technique are connected with the fact that a uniform distribution set of weights is not always connected to a uniform distribution set of Pareto optimal solutions and that diffent combinations of weighs can produce the same solution.

7.3.2 ϵ-Constraint method

In the ϵ-Constraint method only the model is modified assuming of a single objective optimization problem. The other objective functions are transformed into constraints that limit their value to a specific upper bound ϵ. Like for the weighted sum method the unique solution of the ϵ-constraint problem stated is Pareto-optimal for any given upper bound vector. Given the value of ϵ the correspondent model obtained assume the following formulation;

$$\begin{aligned} \min z = \quad & f_1(x) \\ & f_2(x) \leq \epsilon_2 \\ & f_3(x) \leq \epsilon_3 \\ & \dots \\ & f_m(x) \leq \epsilon_m \end{aligned} \tag{7.5}$$

The value of the parameter ϵ have to be chosen in the range between the minimum and maximum value that the objective function can reach. Like for the weighted sum method the found solution strongly depends on the value of the parameter but it is difficult to chose a set of values to represent exaustively the entire pareto front.

7.4 Method

The proposed method is based on the combined use of a machine learning tool with a multi-object optimization algorithm. More in particular to study the behaviour of the process changing the input parameters has been used a neural network. One of the main disadvantages of the neural network is that is usually used like a black box due to the difficulty to formalize correctly the mathematical relationship between input and output. Therefore in the second phase where the

information obtained with the neural network have to be used to optimize the process configuration is not suitable the use of the classical multi-object optimization techniques like $\varepsilon - constraints$ or the *weighted sum method* because it would be necessary to formalize the equations to put in the model. The model to optimize has in fact the following configuration:

$$\min\ z_1 = F_x \tag{7.6}$$

$$\min\ z_2 = G_x \tag{7.7}$$

$$Lb \leq x \leq Ub \tag{7.8}$$

where x is a vector that represents the data of a process configuration F_x and G_x are the estimated results of the output variables through the neural network. These functions could be defined using the information in the weights matrix and the activation functions of the hidden layers of the network using the following formula:

$$F_x = f_o(b_{H-1} + \sum_{i=1}^{N_H} X_i^{H-1} w_i) \tag{7.9}$$

where f_o is the activation function of the output layer, H is the number of hidden layers in the neural network, b_h and N_h are the weight assigned to the bias node and the number of neurons in the layer h and X_i^h is the input of the neuron i of the layer h. The value of X_i^h is calculated with a recursive approach. However the expressions obtained can not be manageable in a short computational time because the activation functions are usually not linear. To solve this problem has been used an evolutionary algorithm called GDE3. The reason of this choice is that evolutionary algorithms are able to deal with problems with objective functions that are non-linear, non-differentiable and non-convex finding a set of solutions in a very short time. In brief, the outline of the approach used to find the set of solutions in the efficient frontier is the following:

1. Definition of the input and output variables of interest;

2. Definition of the dataset by tests;

3. Training of neural networks to estimate the input-output relations;
4. Implementation of the multi-object optimization algorithm;
5. Resolution of the model using the GDE3 algorithm.

The different step will be analyzed more in details in the following sections. The quality of the implemented algorithm will be compared in the last section on a case study with the previously cited standard multi-objective algorithms where the function between input and output parameters has been approximated using a regression technique.

7.4.1 GDE3 Algorithm

In the last few years the number of applications of the evolutionary algorithm is growing due to their fast convergence and their good distribution of the found solutions along the Pareto front. One of the most used evolutionary algorithms is the GDE3. It is has been proved that this algorithm outperforms other evolutionary algorithms in the large majority of benchmark problems. The GDE3 Algorithm is the third version of the Generalized Differential Evolution. In this algorithm has been introduced a new concept of dominance called constraint domination. A solution x_1 constraints-dominates a solution x_2 if one of these conditions is satisfied:

- x_1 is feasible and x_2 is not
- x_1 and x_2 are both feasible and x_1 dominates x_2
- x_1 and x_2 are both not feasible and x_1 dominates x_2 in the constraint function violation space.

Given a problem with H equality and G inequality constraints, the constraint function violation space is calculated in the following way:

$$\max(g_k(x), 0) \ \forall k \in G \tag{7.10}$$

$$\max(|h_k(x)|, 0) \ \forall k \in H \tag{7.11}$$

The first step of the algorithm is to generate a starting population. Each elements of these population is initialized randomly from a uniform distribution. Then like in the most part of the evolutionary algorithms the solutions evolve using combination, mutations and selection operators. Different variants of the GDE3 algorithm have been used in literature using different equation for the combination and mutation operators. In this work the combination phase has been done using the Simulated Binary crossover (SBX). The *SBX crossover* is known as one of the best crossover operators in term of search power. The search power is defined as the ability to generate any arbitrary solution from two parents. In this operator given two parent solutions p_1, p_2 two child solutions c_1, c_2 are generated based on the value of a parameter named spread ratio. The spread ratio β is expressed trough the following equation:

$$\beta = \left| \frac{c_1 - c_2}{p_1 - p_2} \right| \tag{7.12}$$

Starting from this parameter it is possible to calculate an approximated probability to generate a solution with the desired value of β using the following formula.

$$P(\beta) = \begin{cases} 0,5(n+1)\beta^n, & \text{if } \beta \leq 1 \\ 0,5(n+1)\frac{1}{\beta^{n+2}}, & \text{otherwise} \end{cases} \tag{7.13}$$

Where n is a positive real number. Using a large value of n gives there will be an high probability to create solutions similar to the parent, whereas with a small value of n even very distant points can be selected as children solutions. In this work the value of n is not static but evolve decreasing during the iterations. The idea of this choice is that in the earliest iteration the algorithm should to examine how many zones of the space solutions as possible but in the latest iterations the algorithm should converge examining only solutions near the actual estimated efficient frontier. In the form described the SBX operator is able to generate any solution in the interval $[-\infty, +\infty\ ]$. However in our problem, each variable x_i is defined in the interval $x_i^{lb} \leq x_i \leq x_i^{ub}$ so it could be useful to modify the algorithm limiting the research of new solutions in the region of interest. The probability to extract a solution outside the bound will be therefore set to 0. To do so, the spread factor and the cumulative distribution probability will be calculated for

both the bounds using the following equations:

$$\beta^{lb} = \frac{p_1 + p_2 - 2x_i^{lb}}{|p_2 - p_1|} \qquad \beta^{ub} = \frac{2x_i^{ub} - p_1 - p_2}{|p_2 - p_1|} \tag{7.14}$$

$$P_1 = \int_0^{\beta^{lb}} P(\beta)\, d\beta \qquad P_2 = \int_0^{\beta^{ub}} P(\beta)\, d\beta \tag{7.15}$$

Two new spread factors are then calculated using a distribution probability of $P(\beta)/P_1$ and $P(\beta)/P_2$. In this way the algorithm do not produce any solution out of the bound because the new probability distributions is scaled using the range of each variable. At each iteration the outline of the SBX operator The algoritm proceed as follow. First a random number u in the range [0,1] is generated. This number is used to find the correspond value of β so that the cumulative probability is equal to u;

$$\int_0^{\beta} P(\beta)\, d\beta = u \tag{7.16}$$

Subsequentially the formula 7.14 and 7.15 are used to calculate the value of β_1 and β_2 and finally two children solutions are generated using the following equations:

$$c_1 = 0.5[(p_1 + p_2) - \beta_1 |p_1 - p_2|] \tag{7.17}$$

$$c_2 = 0.5[(p_1 + p_2) - \beta_2 |p_1 - p_2|] \tag{7.18}$$

The mutation phase has been done using the approach proposed by Rechenberg in the 1973. In this approach each gene g_i is mutated with a give probability p_m as follows:

$$g_i^{t+1} = g_i^t + N(0, \sigma) \tag{7.19}$$

where $N(0, \sigma)$ is a normal distribution with zero mean and standard deviation σ. One of the most important aspects of the evolutionary algorithms is given by the choice of two parameters: the size of the population and the number of the generations. A too high value of these two parameters leads to an increase of the computational time. However, decreasing the number of generations the quality of the solutions found deteriorates while decreasing the size of the population the number of optimal solutions found is lower. The algorithm proceeds iteratively

until the computation time exceeds a maximum time or the id of the current generation is greater than a predetermined upper bound. The outline of the used GDE3 algorithm is therefore the following:

1. Set $G_{current} = 0$ and $T_{current} = 0$;
2. generate a random population of NP individuals;
3. evaluate the value of each objective function and each constraint function for each solution of the starting population;

while $G_{current} \leq G_{max}$ and $t_{current} \leq t_{max}$

4. apply the mutation and crossover operators;
5. apply the selection operator on the current population;
6. update the value of $G_{current}$ and $t_{current}$;

end while

7. return the current population.

The pareto front obtained with the GDE3 has been proved to approximate better the set of optimal solutions of the problem compared with the standard multi-objective algorithms. To prove this point, the case study of this paper has been solved also with these techniques where the objective functions of the problem have been approximated using the response surface method. The response surface methodology is a collection of mathematical and statistical techniques for empirical model building. The goal of this method is to optimize a response (output variable) that is influenced by many independent variable (input variable). Generally the true response function f is unknown and the research of a good approximation starts with a low-order polynomial function. If the output variable can be defined using a linear function of input variables the approximating function is a first order model. A first order model with n different input variable can be written as:

$$y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \epsilon_i \tag{7.20}$$

Where β is a vector of n unknown constant coefficients referred to the input variables, β_0 is the bias term and ϵ is a random experimental error assumed to have zero mean. The second order model can be written, instead, as:

$$y = \beta_0 + \sum_{i=1}^{n} \beta_i x_i + \sum_{i=1}^{n} \sum_{j=1}^{n} \beta_{ij} x_i x_j + \sum_{i=1}^{n} \gamma_i x_i^2 + \epsilon_i \tag{7.21}$$

Models with a higher degree are not recommendable because they have a high probability of overfitting. Moreover even in the second degree model usually not all the term are necessary. In this work the relevant parameter have been selected using the ANOVA technique and the correspondent vectors β and γ are found using the software $Stat - Ease\ Design - Expert\ v7.0$.

7.5 Case Study

7.5.1 Porthole Die Extrusion

Due to its significant relevance in the field of metal forming, the multi-objective optimization will be implemented on the Extrusion process. The last is one of the most important and common industrial bulk-forming processes used to transform raw materials into semi-finished or finished products. Since the end of the second world war, with the diffusion of heating systems, the extrusion process takes place in the industries thanks to its versatility. The diffusion and the large utilization of this forming process motivate scientists to invest in research on applications and advantages, trying to find properly solutions to its limits. In this paper, the attention will be concentrated on porthole die extrusion, a technique useful to produce cable product as tube or whole profile in general, for which a billet is forced to flow first in a more complex die, named porthole, and then in the welding chamber, before passing through the die bearing to form a hollow geometry (120). In particular, an I section (Fig. 7.14) was investigated in order to observe the functions behavior for changing of both geometrical and process variables: i.e. profile thickness (t_p) and process velocity (V). The value of the b section remains unvaried. Accordingly, an experimental set up composed by nine configurations

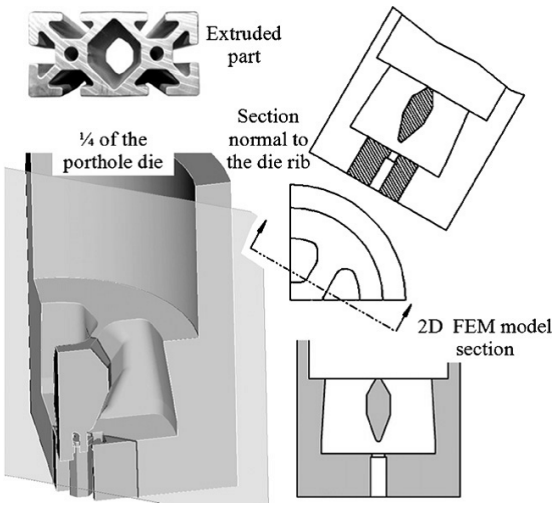


Figure 7.1: Extrusion Perspective (72)

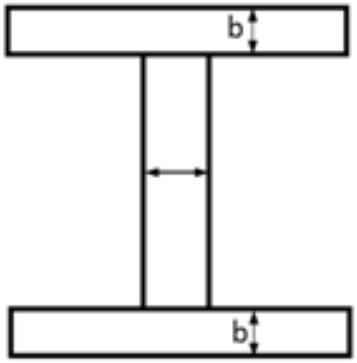


Figure 7.2: Porthole extrusion process: I section

has been created, changing the t_p and V values in the following way:

$$t_p[mm] = 0,5; 1; 2. \tag{7.22}$$

$$V[mm/s] = 1; 5; 10. \tag{7.23}$$

The material flow from larger sections to more restricted one could generate profile distortions and wrappings which compromise the product integrity and quality. Inside the die, the material is firstly split into two streams which join in a 5mm height welding chamber. The working material was an AA6060-T6 aluminum alloy, firstly subjected to an annealing process and subsequently heated in an electrical furnace up to 500C and maintained at this temperature enough time to its core homogenization. It was supplied in cylindrical D30x20mm billets. The tests were carried using a sequence of three billets for each configuration. The optimized functions: quality and sustainability. In order to obtain the best possible combination of output quality maximization and extruded manufacturing environmental impact minimization, it has been necessary to choose two suitable response functions to optimize. The extruded manufacturing environmental impact is quantified towards the energy consumption of the machine, obtained by integrating the load F [kN] over a specified displacement s [mm] (Fig. 7.3), defined as follows:

$$E = \int_{\gamma} F ds. \tag{7.24}$$

In this estimation, additional aliquots are neglected because the analysis focuses on the strictly manufacturing process without considering the entire production line. The designed specimens were thought in order to obtain a fracture on the welded tongue. So, as regards the extruded product quality measurement, according to Donati and Tomesani (121), the best quality estimation of the welding line is the strain to failure. Fig. 7.4 shows the tensile test equipment used to estimate quality.

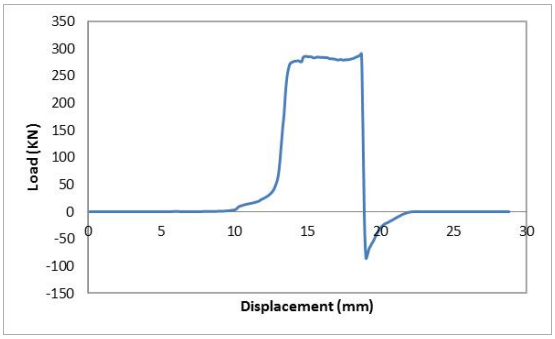


Figure 7.3: Training error

Figure 7.4: Tensile test equipment

7.5.2 Analysis of the results

The multiobjective technique illustrated in the previous section has been used to optimize the process parameters of a porthole die extrusion considering contrastant objective functions. More in particular the objectives considered are the quality of the final product Q and the maximum load to execute the process F. The process parameters taken into account are the tickness T_C of the profilate and the process velocity V. The range considered for these two variables are respectively of [0.5mm,2mm] and [1mm/s,10mm/s]. The material used in these test is the AA6060. As described previoulsy to map the input-output relationship with a neural network the first step is to built an adequate dataset trought test in the lab. The number of test considered is 27. More in particular for each test have been made 3 complete test analysis of a 3^2 where 3 is the number of level for each variable and 2 is the number of considered variables. The 3 level are set to 1mm, 1.5mm and 2mm for the tickness and 1mm/s, 5mm/s and 10mm/s for the process velocity. A sketch of an example of the results obtained are illustrated in the next figure: The neural network used in this analysis has

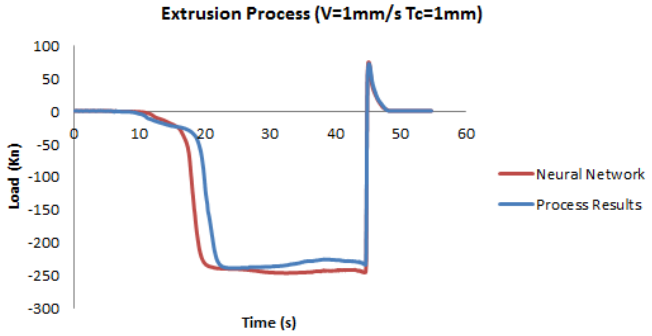


Figure 7.5: Neural Network prediction

been created in the Java environment using the *ENCOG* library (122). Then the multi-objective problem has been solved using The GDE3 algorithm. This algorithm has been implemented in Java making the necessary adjustment on the $MOEAframework$. In the next figure is illustrated the efficient frontier obtained setting a maximum time of 5 minutes and, a maximum number of generations of and a population size equal to. Like already said in the previous chapters the pareto front found with the GDE3 has been compared with the the classical multi-objective algorithm approximating the input output relationship with the responce surface method. The input-output functions has been found with the regression teechinque previously described. The results obtained are the following:

$$Q = -0.115 + 0.781T_c - 0.022V - 0.003T_cV - 0.292T_c^2 + 0.001V^2 \qquad (7.25)$$

$$F = -208.166 - 14.017V - 69.452T_c - 0.639T_cV + 0.912V^2 + 44.502T_c^2; \quad (7.26)$$

In the next figure is reported the pareto front obtained for this problem using the weighted sum method and the ϵ-Constraint method.

It is possible to see that the pareto front extimated is completely the same for the the weighted sum and the ϵ-Constraint methods. This result was expected since the two models are based on the same regression equations and characterized by a very simple structure since are present only bound constraints for the

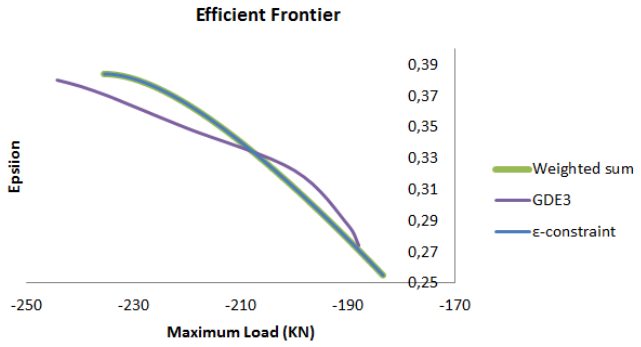


Figure 7.6: Efficient frontiers obtained with the different algorithms

two variables. It is therefore easy to provide a good approximation of the efficient frontier for both methods. The value of the two objective functions for the solution in the Pareto-front found with the $GDE3$ is instead slightly different. However for all the techniques the results show that the optimal point are in the same zone. In fact, all the solution found with the three methods are characterized by a minimum value of the process velocity and a value of tickness in the range [0.6mm,1.7mm] The results obtained with the GDE3 algorithm seem more reliable than the other illustrated methods because the prediction error obtained with the neural network is less than a third of the error obtained with the regression technique. Moreover the number of point found is extremely greater and the computional time lower.

7.6 Conclusion

In this chapter it has been discussed the use of multi-objective techniques for manufacturing applications. Many different studies that makes use of multi-objective algorithm to optimize complicated processes with contrastant objectives are available in the literature. Recently the attention of the researcher has moved around the development of evolutionary algorithm that show greater performance compared to benchmark approach use in the past. A new methodology based on the GDE3 algorithm has been discussed in this chapter. The technique is based on the use of neural networks and evolutionary algorithms. Finally a case study on a porthole die extrusion process for an allumin AA6060 has beeen discussed. The

analysis involves the evaluation of a quality and an economic function altering the value of different process parameters. The results show that is not possible to find a unique solution that allow both the function to be minimized at the same time. Therefore an efficient frontier has been mapped representing different efficient solutions. How it is possible to see from the results the technique proposed in this work allow to predict and optimize the process using few experiments. Moreover this approach is very general and could be used to evaluate the impact of different parameters or material or even for different manufacturing processes.

8

Conclusions

This dissertation explores the use of mathematical and statistical tools to analyze, control and optimize manufacturing processes. Table 8.1 summarizes the topics analyzed and the contributions of this dissertation.
The main topic discussed in this thesis is related to the features that have to be taken into account to select, according to the analyzed process:

1. the metamodel technique;
2. the sampling strategy;
3. the optimization algorithm

All these problems were analyzed from two different point of views. Attacking the problems from a Computer Science angle has led to the development of a general version of the methodologies. In contrast, it is also crucial to analyze the process from a mechanical point of view trying to detect peculiarities that may simplify the computational effort to solve the problem making use of the available knowledge.

Chapter 2 provides a brief introduction of the most used approaches to define input-output relationships. It is pointed out that each technique is affected by many limitations that could significantly affect the accuracy of the model under certain conditions.

Table 8.1: Dissertation Contributions.

Chapter	Theory	Applications
Chapter 1	Introduction, research statement and scope of the thesis	
Chapter 2	Machine learning techniques introduction	Impression die forging
Chapter 3	Heuristic technique to optimize neural network performance	Extrusion, rolling and shearing
Chapter 4	Kriging metamodel for mixed continuous/discrete problems	Incremental sheet forming: thickness distributon
Chapter 5	Manufacturing processes problem modeling	Incremental sheet forming: temperature prediction
Chapter 6	Adaptive KPI prediction based on response surface projection through similarity function	Remote laser welding
Chapter 7	Multi Objective Techniques. Development of a GDE3 based algorithm.	Porthole extrusion

Chapter 3 discusses the use of heuristic algorithms to solve these limitations. Different techniques were developed to improve the performance of a neural network metamodel. In particular three heuristics were developed and used to:

1. select the network architecture;
2. select the starting weights;
3. reduce the training time through an hybrid algorithm (simulated annealing+backpropagation)

Chapter 4 presents a novel kriging metamodel to solve problem characterized by both continuous and discrete variables. The model was coupled with a customized

sampling strategy to reduce the number of experiments to reach a required accuracy. Typically, a specific DoE method is most suitable in combination with each individual metamodel formulation. The proposed designs try to maximize a space-filling property. This feature assures a balanced predictive performance of the approximation model throughout the investigated model space. To collect training data efficiently, the locations for sampling points have to be chosen systematically thus assuring a maximum gain in information with minimal effort. In particular the space filling criterion was considered only around a feasible region denoted as process window.

Chapter 5 discuss the use of customized model based on prior knowledge of the process.

According to that chapter 6 presents a new methodology with which a metamodel is developed making use of qualitative knowledge and/or historical data on similar problems. The metamodel try to iteratively develop new response surfaces through a geometrical projection based on a similarity function.

Finally chapter 7 discuss the use of evolutionary algorithm to solve multi objective problems making use of the previous developed metamodel.

To conclude, it is believed that this dissertation explores machine learning and optimization techniques for manufacturing from many angles, and that several of the ideas presented here will be useful both in practice and for theoretic studies

References

[1] Xi Chen, Kai Wang, and Feng Yang. **Stochastic kriging with qualitative factors**. In *Proceedings of the 2013 Winter Simulation Conference: Simulation: Making Decisions in a Complex World*, pages 790–801. IEEE Press, 2013. 7, 54

[2] Bruce Ankenman, Barry L Nelson, and Jeremy Staum. **Stochastic kriging for simulation metamodeling**. *Operations research*, **58**(2):371–382, 2010. 7, 54

[3] George EP Box and Norman Richard Draper. *Empirical model-building and response surfaces*, **424**. Wiley New York, 1987. 8

[4] Raymond H Myers and Christine M Anderson-Cook. *Response surface methodology: process and product optimization using designed experiments*, **705**. John Wiley & Sons, 2009. 9

[5] Pospichal J Svozil D, Kvasnicka V. **Introduction to multi-layer feed-forward neural networks**. *Chemometrics and intelligent laboratory systems*, **39**(1):43–62, 1997. 9, 21

[6] Donald F Specht. **A general regression neural network**. *Neural Networks, IEEE Transactions on*, **2**(6):568–576, 1991. 10, 21

[7] Pivonka Petr and DOHNAL JIŘÍ. **On-line Identification Based on Neural Networks using of Levenberg-Marquardt Method and Back-propagation Algorithm**. In *Proceedings of the 5th WSEAS Int. Conf. on Neural Networks and Applications*, 2004. 11

[8] Jyh-Shing Roger Jang, Chuen-Tsai Sun, and Eiji Mizutani. **Neuro-fuzzy and soft computing; a computational approach to learning and machine intelligence**. 1997. 11

[9] Petr Hájek. *Metamathematics of fuzzy logic*, **4**. Springer, 1998. 11

[10] Robert Fullér. *Introduction to neuro-fuzzy systems*, **2**. Springer, 2000. 12

[11] G Nalbantov, Patrick JF Groenen, and Jan C Bioch. **Support Vector Regression Basics**. 12

[12] Alex J Smola and Bernhard Schölkopf. **A tutorial on support vector regression**. *Statistics and computing*, **14**(3):199–222, 2004. 13

[13] Taylan Altan, Gracious Ngaile, and Gangshu Shen. *Cold and hot forging: fundamentals and applications*, **1**. ASM international, 2005. 15

[14] Guy Snape, Sally Clift, and Alan Bramley. **Parametric sensitivity analyses for FEA of hot steel forging**. *Journal of materials processing technology*, **125**:353–360, 2002. 15

[15] BI Tomov, VI Gagov, and RH Radev. **Numerical simulations of hot die forging processes using finite element method**. *Journal of materials processing technology*, **153**:352–358, 2004. 15

[16] Hyunkee Kim, Tetsuji Yagi, and Masahito Yamanaka. **FE simulation as a must tool in cold/warm forging process and tool design**. *Journal of Materials Processing Technology*, **98**(2):143–149, 2000. 16

[17] William RD Wilson, Steven R Schmid, and Jiying Liu. **Advanced simulations for hot forging: heat transfer model for use with the finite element method**. *Journal of Materials Processing Technology*, **155**:1912–1917, 2004. 16

[18] Serope Kalpakjian. **Survey of the Feasibility of an Analytical Approach to Die Design in Closed-die Forging**. Technical report, DTIC Document, 1966. 16

REFERENCES

[19] Pfingsten JT. **Machine Learning for Mass Production and Industrial Engineering**. 2007. 21, 43

[20] M Jalali-Heravi, M Asadollahi-Baboli, and P Shahbazikhah. **QSAR study of heparanase inhibitors activity using artificial neural networks and Levenberg–Marquardt algorithm**. *European journal of medicinal chemistry*, **43**(3):548–556, 2008. 21

[21] T Dudley Wallace. **Weaker criteria and tests for linear restrictions in regression**. *Econometrica: Journal of the Econometric Society*, pages 689–698, 1972. 22

[22] Ridha Hambli. **Statistical damage analysis of extrusion processes using finite element method and neural networks simulation**. *Finite Elements in Analysis and Design*, **45**(10):640–649, 2009. 22

[23] Feng Yang. **Neural network metamodeling for cycle time-throughput profiles in manufacturing**. *European Journal of Operational Research*, **205**(1):172–185, 2010. 22

[24] Demetgul M Kucuk H Unal M, Onat M. **Fault diagnosis of rolling bearings using a genetic algorithm optimized neural network**. *Measurement*, 2014. 22

[25] Saravanan R Venkatesan D, Kannan K. **A genetic algorithm-based artificial neural network model for the optimization of machining processes**. *Neural Computing and Applications*, **18**(2):135–140, 2009. 22

[26] YL Yao and XD Fang. **Assessment of chip forming patterns with tool wear progression in machining via neural networks**. *International Journal of Machine Tools and Manufacture*, **33**(1):89–102, 1993. 23

[27] Gagliardi F Ambrogio G. **Design of an optimized procedure to predict opposite performances in porthole die extrusion**. *Neural Computing and Applications*, **23**(1):195–206, 2013. 23, 46

[28] Fiğlali A Özcan B. **Artificial neural networks for the cost estimation of stamping dies**. *Neural Computing and Applications*, pages 1–10. 23

[29] Yusuff RM Saberi S. **Neural network application in predicting advanced manufacturing technology implementation performance**. *Neural Computing and Applications*, **21**(6):1191–1204, 2012. 23

[30] Bouzerdoum A Arulampalam G. **A generalized feedforward neural network architecture for classification and regression**. *Neural Networks*, **16**(5):561–568, 2003. 23

[31] Jerez JM Gómez I, Franco L. **Neural network architecture selection: Can function complexity help?** *Neural Processing Letters*, **30**(2):71–87, 2009. 23

[32] Haussler D Baum EB. **What size net gives valid generalization?** *Neural computation*, **1**(1):151–160, 1989. 23

[33] Yoneyama T Camargo L. **Specification of training sets and the number of hidden neurons for multilayer perceptrons**. *Neural computation*, **13**(12):2673–2680, 2001. 24

[34] Qing M Jun-Hai Z Xi-Zhao W, Qing-Yan S. **Architecture selection for networks trained with extreme learning machine using localized generalization error model**. *Neurocomputing*, **102**:3–9, 2013. 24

[35] Lirov Y. **Computer aided neural network engineering**. *Neural networks*, **5**(4):711–719, 1992. 24

[36] Haykin S. *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994. 24

[37] Berger JO. *Statistical decision theory and Bayesian analysis*. Springer, 1985. 24

[38] Smith AFM Bernardo JM. *Bayesian theory*, **405**. John Wiley & Sons, 2009. 24

[39] Johnson JD Sexton RS, Dorsey RE. **Optimization of neural networks: A comparative analysis of the genetic algorithm and simulated annealing**. *European Journal of Operational Research*, **114**(3):589–601, 1999. 25, 28

[40] Araújo R Soares SA, Carlos H. **Comparison of a genetic algorithm and simulated annealing for automatic neural network ensemble development**. *Neurocomputing*, **121**:498–511, 2013. 25, 28

[41] Bogart C Whitley D, Starkweather T. **Genetic algorithms and neural networks: Optimizing connections and connectivity**. *Parallel computing*, **14**(3):347–361, 1990. 25, 28

[42] Kenneth A De Jong and William M Spears. **A formal analysis of the role of multi-point crossover in genetic algorithms**. *Annals of Mathematics and Artificial Intelligence*, **5**(1):1–26, 1992. 25, 30

[43] Ram Bhusan Agrawal, Kalyanmoy Deb, and Ram Bhushan Agrawal. **Simulated binary crossover for continuous search space**. 1994. 25, 30, 91

[44] Glover F. **Artificial intelligence, heuristic frameworks and tabu search**. *Managerial and Decision Economics*, **11**(5):365–375, 1990. 25, 31

[45] Li M Ruan X Ye J, Qiao. **A tabu based neural network learning algorithm**. *Neurocomputing*, **70**(4):875–882, 2007. 25, 31

[46] Lim LEN Khaw JFC, Lim BS. **Optimal design of neural networks using the Taguchi method**. *Neurocomputing*, **7**(3):225–245, 1995. 25, 33

[47] Rowlands H Packianather MS, Drake PR. **Optimizing the parameters of multilayered feedforward neural networks through Taguchi design of experiments**. *Quality and reliability engineering international*, **16**(6):461–473, 2000. 25, 33

[48] Tannock J Sukthomya W. **The optimisation of neural network parameters using Taguchi's design of experiments approach: an application in manufacturing process modelling**. *Neural Computing & Applications*, **14**(4):337–344, 2005. 25, 33

[49] Quinlan JR. **Learning decision tree classifiers**. *ACM Computing Surveys (CSUR)*, **28**(1):71–72, 1996. 25, 34

[50] Neville PG. **Decision trees for predictive modeling**. *SAS Institute Inc*, 1999. 25, 34

[51] Kim YS. **Comparison of the decision tree, artificial neural network, and linear regression methods based on the number and types of independent variables and sample size**. *Expert Systems with Applications*, **34**(2):1227–1234, 2008. 25, 34

[52] Kubat M Ivanova I. **Initialization of neural networks by means of decision trees**. *Knowledge-Based Systems*, **8**(6):333–344, 1995. 25, 34

[53] Maji P. **Efficient design of neural network tree using a new splitting criterion**. *Neurocomputing*, **71**(4):787–800, 2008. 25, 34

[54] Pinkus A Schocken S Leshno M, Lin VY. **Multilayer feedforward networks with a nonpolynomial activation function can approximate any function**. *Neural networks*, **6**(6):861–867, 1993. 26

[55] Cybenko G. **Approximation by superpositions of a sigmoidal function**. *Mathematics of control, signals and systems*, **2**(4):303–314, 1989. 26

[56] Orr GB Müller KR LeCun YA, Bottou L. **Efficient backprop**. In *Neural networks: Tricks of the trade*, pages 9–50. Springer, 1998. 27, 38

[57] Sandberg IW Park J. **Universal approximation using radial-basis-function networks**. *Neural computation*, **3**(2):246–257, 1991. 27

[58] Argyros AA Lourakis MIA. **Is Levenberg-Marquardt the most efficient optimization algorithm for implementing bundle adjustment?** In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, **2**, pages 1526–1531. IEEE, 2005. 27

[59] Stafylopatis A Likas A. **Training the random neural network using quasi-Newton methods**. *European Journal of Operational Research*, **126**(2):331–339, 2000. 27

[60] Braun H Riedmiller M. **A direct adaptive method for faster backpropagation learning: The RPROP algorithm**. In *Neural Networks, 1993., IEEE International Conference on*, pages 586–591. IEEE, 1993. 28

[61] Al Timemy AH Al Naima FM. **Resilient Back Propagation Algorithm for Breast Biopsy Classification Based on Artificial Neural Networks**. *Breast cancer*, **4**(5):6–7. 28

[62] Khoshayanda MR Soltani Bozchalooic I-Dadgara A Rafiee-Tehrania M Ghaffari A, Abdollahib H. **Performance comparison of neural network training algorithms in modeling of bimodal drug delivery**. *International journal of pharmaceutics*, **327**(1):126–138, 2006. 28

[63] Ben Kröse, Ben Krose, Patrick van der Smagt, and Patrick Smagt. *An introduction to neural networks*. Citeseer, 1996. 36

[64] Georg Thimm and Emile Fiesler. **Neural network initialization**. In *From Natural to Artificial Neural Computation*, pages 535–542. Springer, 1995. 38

REFERENCES

[65] D Lamy and P Borne. **Neural networks initialization**. In *Systems, Man and Cybernetics, 1993.'Systems Engineering in the Service of Humans', Conference Proceedings., International Conference on*, pages 491–495. IEEE, 1993. 38

[66] Derrick Nguyen and Bernard Widrow. **Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights**. In *Neural Networks, 1990., 1990 IJCNN International Joint Conference on*, pages 21–26. IEEE, 1990. 38

[67] Jim YF Yam and Tommy WS Chow. **A weight initialization method for improving training speed in feedforward neural network**. *Neurocomputing*, **30**(1):219–232, 2000. 38

[68] Jonathan Engel. **Teaching feed-forward neural networks by simulated annealing**. *Complex Systems*, **2**(6):641–648, 1988. 40

[69] Kenneth D Boese and Andrew B Kahng. **Simulated annealing of neural networks: thecooling'strategy reconsidered**. In *Circuits and Systems, 1993., ISCAS'93, 1993 IEEE International Symposium on*, pages 2572–2575. IEEE, 1993. 41

[70] Jeffrey Fluhrer. **DEFORMTM 3D Version 5.0 User's Manual [Z]**. *Columbus: Scientific Forming Technologles Corporation*, 2005. 44

[71] Schmid SR Kalpakjian S. *Manufacturing processes for engineering materials*, **5**. Pearson education, 2010. 45

[72] Gagliardi F Giardini C Ceretti E, Fratini L. **A new approach to study material bonding in extrusion porthole dies**. *CIRP Annals-Manufacturing Technology*, **58**(1):259–262, 2009. 48, 119

[73] Umbrello D Ambrogio G, Filice L. **Numerical analysis of the fracture surface in thick sheets blanking**. In *Proc. of the 7th ESAFORM Conference*, pages 757–760, 2004. 49

[74] Filice L Ambrogio G, Gagliardi F. **First experimental and numerical evidences concerning the Hydropiercing process**. *7th Esaform conference on material forming, Trondheim (Norway), 28-30 April 2004*, 2004. 49

[75] Peter ZG Qian, Huaiqing Wu, and CF Jeff Wu. **Gaussian process models for computer experiments with qualitative and quantitative factors**. *Technometrics*, **50**(3), 2008. 54

[76] P Jaeckel and R Rebonato. **The most general methodology for creating a valid correlation matrix for risk management and option pricing purposes**. *Journal of risk*, **2**(2):17–28, 1999. 54

[77] J Jeswiet, F Micari, G Hirt, A Bramley, Joost Duflou, and J Allwood. **Asymmetric single point incremental forming of sheet metal**. *CIRP Annals-Manufacturing Technology*, **54**(2):88–114, 2005. 55, 57

[78] Giuseppina Ambrogio, L De Napoli, Luigino Filice, and M Muzzupappa. **Experimental evidences concerning geometrical accuracy after unclamping and trimming incrementally formed components**. *Key Engineering Materials*, **344**:535–542, 2007. 56

[79] Alessandro Baldi Antognini and Maroussa Zagoraiou. **Exact optimal designs for computer experiments via Kriging metamodelling**. *Journal of Statistical Planning and Inference*, **140**(9):2607–2617, 2010. 58

[80] Gavin C Cawley and Nicola LC Talbot. **Efficient leave-one-out cross-validation of kernel Fisher discriminant classifiers**. *Pattern Recognition*, **36**(11):2585–2592, 2003. 59, 87

[81] Omer El Fakir, LiLiang Wang, Daniel Balint, John P Dear, and Jianguo Lin. **Predicting Effect of Temperature, Strain Rate and Strain Path Changes on Forming Limit of Lightweight Sheet Metal Alloys**. *Procedia Engineering*, **81**:736–741, 2014. 67

[82] D Young and J Jeswiet. **Wall thickness variations in single-point incremental forming**. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, **218**(11):1453–1459, 2004. 67

[83] LUCA GAMBIRASIO and EGIDIO RIZZI. **On the Constitutive Modeling of Strain Rate and Temperature Dependent Materials. Part I- Calibration Strategies of the Johnson-Cook Strength Model: Discussion and Applications to Experimental Data**. 2013. 68

[84] Jeffrey Fluhrer. **DEFORM-3D Version 5.0 users manual**. *Scientific Forming Technologies Corporation*, 2004. 68, 73, 74

[85] Eliane Giraud, Frederic Rossi, Guénaël Germain, and JC Outeiro. **Temperature variation during high speed incremental forming on different lightweight alloys**. *International Journal of Advanced Manufacturing Technology*, 2014. 69

[86] M Durante, A Formisano, A Langella, and F Memola Capece Minutolo. **The influence of tool rotation on an incremental forming process**. *Journal of Materials Processing Technology*, **209**(9):4621–4626, 2009. 73

[87] Meftah Hrairi and Salah BM Echrif. **PROCESS SIMULATION AND QUALITY EVALUATION IN INCREMENTAL SHEET FORMING**. *IIUM Engineering Journal*, **12**(3), 2011. 74

[88] L Filice, G Ambrogio, and F Micari. **On-line control of single point incremental forming operations through punch force monitoring**. *CIRP annals-Manufacturing technology*, **55**(1):245–248, 2006. 74

[89] ASM International. Handbook: Aluminum and Aluminum Alloys. *ASM handbook*, **13**. Asm International, 2005. 74

[90] Christoph Leyens and Manfred Peters. *Titanium and titanium alloys*. Wiley Online Library, 2003. 74

[91] Thorsten Wuest, Christopher Irgens, and Klaus-Dieter Thoben. **Analysis of manufacturing process sequences, using machine learning on intermediate product states (as process proxy data)**. In *Advances in Production Management Systems. Competitive Manufacturing for Innovative Products and Services*, pages 1–8. Springer, 2013. 81

[92] Stefan Bergman. *The kernel function and conformal mapping*, **5**. American Mathematical Soc., 1970. 87

[93] Debasish Basak, Srimanta Pal, and Dipak Chandra Patranabis. **Support vector regression**. *Neural Information Processing-Letters and Reviews*, **11**(10):203–224, 2007. 87

[94] Michael Collins. **Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms**. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. Association for Computational Linguistics, 2002. 87

[95] Purushottam Kar and Prateek Jain. **Similarity-based Learning via Data Driven Embeddings.** In *NIPS*, pages 1998–2006, 2011. 88

[96] Herman Chernoff. **Sequential design of experiments**. *The Annals of Mathematical Statistics*, pages 755–770, 1959. 89

[97] M Willjuice Iruthayarajan and S Baskar. **Evolutionary algorithms based design of multivariable PID controller**. *Expert systems with Applications*, **36**(5):9159–9167, 2009. 91

[98] Yang Dongxia, Li Xiaoyan, He Dingyong, Nie Zuoren, and Huang Hui. **Optimization of weld bead geometry in LASER WELDING with filler wire process using Taguchis approach**. *Optics and Laser Technology*, **44**(7):2020–2025, 2012. 91

[99] MG Forrest, WA Martilla, and TM Burke. **A comparative analysis of laser and resistance spot welds**. In *IBEC*, **94**, pages 45–51, 1994. 91

[100] P Sathiya, K Panneerselvam, and R Soundararajan. **Optimal design for laser beam butt welding process parameter using artificial neural networks and genetic algorithm for super austenitic stainless steel**. *Optics & Laser Technology*, **44**(6):1905–1914, 2012. 92

[101] KY Benyounis, AG Olabi, and MSJ Hashmi. **Effect of laser welding parameters on the heat input and weld-bead profile**. *Journal of Materials Processing Technology*, **164**:978–985, 2005. 92

[102] Mahmood Moradi and Majid Ghoreishi. **Influences of laser welding parameters on the geometric profile of NI-base superalloy Rene 80 weld-bead**. *The International Journal of Advanced Manufacturing Technology*, **55**(1-4):205–215, 2011. 92

[103] Ford. **Engineering Specification-Laser Beam Welding-Stitch Welds**. *Ford Internal Document*, 2010. 93

[104] T Sibillano, A Ancona, D Rizzi, S Saludes Rodil, J Rodríguez Nieto, AR Konuk, RGKM Aarts, and AJ Huis int Veld. **Study on the correlation between plasma electron temperature and penetration depth in laser welding processes**. *Physics Procedia*, **5**:429–436, 2010. 93

[105] Weichiat Chen, Paul Ackerson, and Pal Molian. **CO¡ sub¿ 2¡/sub¿ laser welding of galvanized steel sheets using vent holes**. *Materials & Design*, **30**(2):245–251, 2009. 93

[106] H Bley, L Weyand, and A Luft. **An alternative approach for the cost-efficient laser welding of zinc-coated sheet metal**. *CIRP Annals-Manufacturing Technology*, **56**(1):17–20, 2007. 93

[107] American Welding Society. **Standard Welding Terms and Definitions**. 1989. 93

REFERENCES

[108] ASM Handbook. **Welding, brazing and soldering**. *ASM Handbook*, **6**, 1993. 93

[109] W Sudnik, D Radaj, S Breitschwerdt, and W Erofeew. **Numerical simulation of weld pool geometry in laser beam welding**. *Journal of Physics D: Applied Physics*, **33**(6):662, 2000. 93

[110] KY Benyounis and AG Olabi. **Optimization of different welding processes using statistical and numerical approaches–A reference guide**. *Advances in Engineering Software*, **39**(6):483–496, 2008. 94

[111] MMA Khan, L Romoli, M Fiaschi, G Dini, and F Sarri. **Experimental design approach to the process parameter optimization for LASER WELDING of martensitic stainless steels in a constrained overlap configuration**. *Optics & Laser Technology*, **43**(1):158–172, 2011. 95

[112] G Casalino, F Curcio, and F Memola Capece Minutolo. **Investigation on Ti6Al4V LASER WELDING using statistical and Taguchi approaches**. *Journal of materials processing technology*, **167**(2):422–428, 2005. 95

[113] AG Olabi, G Casalino, KY Benyounis, and MSJ Hashmi. **An ANN and Taguchi algorithms integrated approach to the optimization of CO_2 LASER WELDING**. *Advances in Engineering Software*, **37**(10):643–648, 2006. 95

[114] U Reisgen, M Schleser, O Mokrov, and E Ahmed. **Statistical modeling of LASER WELDING of DP/TRIP steel sheets**. *Optics & Laser Technology*, **44**(1):92–101, 2012. 95

[115] Dong-Yoon Kim and Young-Whan Park. **Weldability evaluation and tensile strength estimation model for aluminum alloy lap joint welding using hybrid system with laser and scanner head**. *Transactions of Nonferrous Metals Society of China*, **22**:s596–s604, 2012. 95

[116] PK Palani, N Murugan, and B Karthikeyan. **Process parameter selection for optimising weld bead geometry in stainless steel cladding using Taguchi's approach**. *Materials SCIENCE AND TECHNOLOGY*, **22**(10):1193–1200, 2006. 95

[117] LM Galantucci, L Tricarico, and R Spina. **A quality evaluation method for LASER WELDING of Al alloys through neural networks**. *CIRP Annals-Manufacturing Technology*, **49**(1):131–134, 2000. 95

[118] Andres Ortiz. *Optimization of Remote Laser Welding Process Parameters using Design-of-Experiments and Adaptive Sampling*. PhD thesis, 2012. 96

[119] WMG. **TSB Project Reports**. Technical report, 2010. 97

[120] L Li, H Zhang, J Zhou, J Duszczyk, GY Li, and ZH Zhong. **Numerical and experimental study on the extrusion through a porthole die to produce a hollow magnesium profile with longitudinal weld seams**. *Materials & Design*, **29**(6):1190–1198, 2008. 118

[121] L Donati and L Tomesani. **The prediction of seam welds quality in aluminum extrusion**. *Journal of Materials Processing Technology*, **153**:366–373, 2004. 120

[122] Jeff Heaton. **Encog: Library of Interchangeable Machine Learning Models for Java and C#**. *arXiv preprint arXiv:1506.04776*, 2015. 122

Declaration

I herewith declare that I have produced this paper without the prohibited assistance of third parties and without making use of aids other than those specified; notions taken over directly or indirectly from other sources have been identified as such. This paper has not previously been presented in identical or similar form to any other Italian or foreign examination board.

The thesis work was conducted from November 2012 to October 2015 under the supervision of Prof. Roberto Musmanno at University of Calabria.

Arcavacata,