DISSERTATIONS IN
FORESTRY AND
NATURAL SCIENCES

MIKA LIUKKONEN

Intelligent Methods in the Electronics Industry

Quality Analysis of Automated Soldering

PUBLICATIONS OF THE UNIVERSITY OF EASTERN FINLAND
Dissertations in Forestry and Natural Sciences

UNIVERSITY OF
EASTERN FINLAND

MIKA LIUKKONEN

Intelligent Methods in the Electronics Industry

Quality Analysis of Automated Soldering

Publications of the University of Eastern Finland
Dissertations in Forestry and Natural Sciences
24

Academic Dissertation
To be presented by permission of the Faculty on Sciences and Forestry for public examination in the Auditorium MET in Mediteknia Building at the University of Eastern Finland, Kuopio, on December, 17, 2010, at 12 o'clock noon.

Department of Environmental Science

Kopijyvä
Kuopio, 2010
Editors: Prof. Pertti Pasanen
Lecturer Sinikka Parkkinen, Prof. Kai Peiponen
Distribution:
Eastern Finland University Library / Sales of publications
P.O.Box 107, FI-80101 Joensuu, Finland
tel. +358-50-3058396
http://www.uef.fi/kirjasto

ISBN: 978-952-61-0281-8
ISSN: 1798-5668
ISSNL 1798-5668
ISBN 978-952-61-0282-5 (PDF)
ISSN 1798-5676 (PDF)

Author's address: University of Eastern Finland
Department of Environmental Sciences
P.O.Box 1627
70211 KUOPIO
FINLAND
email: mika.liukkonen@uef.fi

Supervisors: Professor Yrjö Hiltunen, Ph.D
University of Eastern Finland
Department of Environmental Sciences
P.O.Box 1627
70211 KUOPIO
FINLAND
email: yrjo.hiltunen@uef.fi

Professor Juhani Ruuskanen, Ph.D
University of Eastern Finland
Department of Environmental Sciences
P.O.Box 1627
70211 KUOPIO
FINLAND
email: juhani.ruuskanen@uef.fi

Reviewers: Professor Sirkka-Liisa Jämsä-Jounela, D.Sc. (Tech.)
Aalto University School of Science and Technology
Laboratory of Process Control and Automation
P.O.Box 6100
02015 HUT
FINLAND
email: sirkka-l@tkk.fi

Professor Tommi Kärkkäinen, Ph.D
University of Jyväskylä
Department of Mathematical Information Technology
P.O.Box 35
40014 UNIVERSITY OF JYVÄSKYLÄ
FINLAND
email: tommi.karkkainen@jyu.fi

Opponent: Professor Kauko Leiviskä, D.Sc (Tech.)
University of Oulu
Control Engineering Laboratory
P.O.Box 4300
90014 UNIVERSITY OF OULU
FINLAND
email: kauko.leiviska@oulu.fi

ABSTRACT

Modern production of electronics is characterized by rapid changes in the production, products that are increasingly technical and complicated, the pressure to decrease production costs and, at the same time, higher demands for the quality of products. The environment of producing electronics is extremely challenging in terms of quality, because the products and their different versions are in a state of continuous change. All kind of changes, both sudden and long-term, in the production potentially cause problems, and problems often cause defective products. Therefore quality assurance is getting an increasingly important role in the electronics industry.

Recent development in the study called computational intelligence has produced new intelligent methods for automated extraction of useful information. Methods such as artificial neural networks, fuzzy sets, rough sets and evolutionary computation, which are generally associated with computational intelligence, are nowadays used widely in different industrial environments. These intelligent methods have several advantages over statistical methods that have been used traditionally by the electronics industry. An ability to learn from experience, self-organize, adapt in response to dynamically changing conditions and a considerable potential in solving real world problems, for example, are properties typically inherent in computationally intelligent systems. Despite this intelligent methods have not been utilized by the electronics industry on a large scale.

This thesis documents an application of intelligent data-based modeling methods to quality analysis of electronics production. The purpose is to benefit from the useful characteristics of computational intelligence, of which the key elements are listed above. Methods such as self-organizing maps and multilayer perceptrons are applied to the quality analysis of automated soldering, and a procedure for intelligent quality analysis of electronics manufacturing is created, starting from pre-processing of data and ending up to visualization and analysis of results.

The main conclusion of the thesis is that intelligent methods should be used in the electronics industry on a much larger scale than they are today. The results show that they provide an efficient way of analyzing quality in the electronics industry. Intelligent methods can reveal mutual interactions which are otherwise difficult to find, improve the goodness of models and decrease the number of variables needed for modeling or optimization. They can also offer a useful way of analyzing large data sets and provide a practical platform for representing them visually. Perhaps the most important thing is, however, that these methods are usable in generic data-based applications, which facilitates their implementation in the electronics industry.

Universal Decimal Classification: 004.8, 005.935.3, 006.015.5, 621.38, 621.791.3

INSPEC Thesaurus: artificial intelligence; neural nets; multilayer perceptrons; self-organising feature maps; electronics industry; soldering; quality assurance; quality control; data visualisation; data analysis; modelling

Yleinen suomalainen asiasanasto: tekoäly; neuroverkot; juotto; elektroniikkateollisuus; laadunvarmistus; laadunvalvonta; visualisointi; mallintaminen

Preface

The majority of the results presented in this thesis have been produced during two research projects organized in co-operation between the 3K Factory of Electronics and the University of Kuopio (currently the University of Eastern Finland) during the years 2006–2010.

ELMO (*Elektroniikan tuotantoprosessien mallinnus ja optimointi* - Electronics production processes: modelling and optimization) project was funded by the Finnish Funding Agency for Technology and Innovation (TEKES), European Regional Development Fund (ERDF), Enics Varkaus Ltd., Elcoteq Finland Ltd. and Savox Manufacturing Services Ltd. The project was organized during November 1, 2006 - December 31, 2007.

ETKO (*Elektroniikan tuotantoketjujen kokonaisvaltainen optimointi* - Comprehensive optimization of electronics production chains) project was funded by the Finnish Funding Agency for Technology and Innovation (TEKES), European Regional Development Fund (ERDF), Enics Finland Ltd., Kemppi Ltd. and Wisetime Ltd. The project was organized during January 1, 2008 - April 30, 2010.

I would like to thank all the partners that have been involved in funding the two research projects which have made writing this thesis possible.

I wish also to thank Professor Juhani Ruuskanen for supervising my thesis, and Professor Sirkka-Liisa Jämsä-Jounela and Professor Tommi Kärkkäinen for reviewing it.

Moreover, I am most grateful to all my co-authors, Elina Havia and Hannu Leinonen from the 3K Factory of Electronics, who offered their expert knowledge of the production of electronics, and Teri Hiltunen, who worked at the University of Kuopio at the time when the second article was published.

My special thanks go to Professor Yrjö Hiltunen for offering the opportunity to carry out this work. His guidance and encouragement have been invaluable over these years.

Finally, I wish to express my warm gratitude to my family, friends and colleagues for their endless support over the years and for giving me opportunities to relax.

Kuopio, December 2010
Mika Liukkonen

Glossary

Activation function: Mathematical function used in tuning the output of neurons in ANNs.

Artificial neural network: Group of machine learning methods consisting of simple processing units linked together which are used to create complex models and which exploit experimental knowledge in nonlinear problem solving.

Back-propagation: Group of neural network training algorithms which work by minimizing the error value between actual and expected outputs.

Clustering: Form of unsupervised learning in which data patterns are partitioned into subgroups with respect to their similarity.

Computational intelligence: Study of advanced computing methods that are adaptive and able to evolve, can learn from experience and solve problems in complex and changing environments.

Cross-validation: Method for evaluating the goodness of a model which works by dividing the data into subsets and using each subset at a time in the validation of the model.

Data mining: Step of knowledge discovery which involves the application of specific algorithms for extracting models from data.

Design of experiments: Method of experimental testing for acquiring information by doing experimentations in processes in which variation is present.

Imputing: Stage of data analysis in which the missing values of data are replaced artificially.

Index of agreement: Measure of correlation which describes the ratio between mean squared error and potential error.

Input layer: The part of neural network that distributes inputs to hidden layers.

Intelligent method: Method in which computational intelligence is utilized.

K-means: Iterative clustering method based on calculating squared errors.

Knowledge discovery: Process of identifying new and potentially useful patterns in data, in which theories, algorithms and methods from research fields such as machine learning, statistics, computational intelligence and visualization of data are combined.

Learning rate (parameter): Parameter of neural networks which determines how much the weights can be modified with respect to the direction and rate of change and which is used to tune the speed and stability of training.

Linear regression: Computing method that fits a linear equation to data points in order to estimate the unknown parameters of a model.

Machine learning: Study of computing algorithms designed for demanding tasks such as classification, pattern recognition, prediction and adaptive control.

Model: Artificial descriptions of real phenomena, which can be either quantitative or qualitative, and either mechanistic or data-driven.

Modeling: Development of models.

Momentum (constant): Constant parameter used in tuning ANNs by scaling the effect of the previous step on the current one, which aids the algorithm to overcome the problem of sticking to local minima.

Multilayer perceptron: Feed-forward neural network based on supervised learning which consists of computational units (neurons) and weighted connections and in which the input signals proceed forward layer by layer to produce a desired output.

Neuron: Single computational unit of a neural network.

Normalization: Group of data transformation techniques used in multivariate computation for equalizing the different ranges of variables to avoid domination due to a greater range.

Output layer: The part of neural network that outputs the result of computation.

Pre-processing: Stage of data analysis which includes replacement of missing values, scaling of variables and determining process lags, for example.

Printed circuit board: Popular technique in inter-component wiring and assembly of electronic equipment.

Quality: Combination of characteristics which define the ability of a product to meet the preset requirements.

Quality analysis: Analysis procedure which aims at quality improvement by studying the relationship between the realized quality and the potential reasons for it.

Quality management: Group of activities used by an organization for directing, controlling and coordinating quality.

Regression: Basic method for searching the relationship between a response variable and at least one explanatory variable by estimating the model parameters.

Self-organizing map: Neural network algorithm based on unsupervised learning, in which output neurons compete with each other to be activated, or a visualization structure in which features of data are presented in a map-like grid.

Sequential backward search: Method for selecting variables which works by eliminating the least promising variables one by one from the set including all variables.

Sequential forward search: Method for selecting variables in which variables are progressively included to larger and larger subsets so that the goodness of model is maximized.

Soldering: Stage of electronics assembly in which two metal surfaces are joined together by metallic bonds created when the molten solder placed between them is solidified.

Statistical process control: Tool of process control which utilizes statistical methods in measurement and analysis of variance in a process.

Supervised training: Form of computational training in which examples are used to learn an unknown function defined by the samples, by adjusting the weights of a neural network, which can then be used to produce estimates for the output.

Unsupervised learning: Form of computational learning in which neurons adapt to specific patterns in input data by auto-association and by competing with each other.

Variable selection: Stage of data processing in which the dimensionality of data (i.e. the number of variables) is reduced.

Variance scaling: Technique for transforming data, which is based on scaling the variance in original data.

Validation: Step of modeling in which the model performance is validated.

Wave soldering: Automated process to solder electronic components to printed circuit boards.

Abbreviations

ANN	Artificial neural network
ANOVA	Analysis of variance
AOI	Automated optical inspection
AXI	Automated X-ray inspection
CI	Computational intelligence
CS	A components-before-solder process; the components are placed onto the PCB before the solder is applied
DoE	Design of experiments
DT	Decision tree
GA	Genetic algorithm
FST	Fuzzy set theory
ICT	In-circuit testing
KDD	Knowledge discovery in databases
LCL	Lower control level
LVQ	Learning vector quantization
MANOVA	Multivariate analysis of variance
MLP	Multilayer perceptron
PCB	Printed circuit board
PNN	Probabilistic neural network
RBFN	Radial basis function network
RST	Rough set theory
SC	A solder-before-components process; the solder is deposited before the components are placed onto the PCB
SFS	Sequential forward selection of variables
SMC	Surface mount component
SOM	Self-organizing map
SPC	Statistical process control
SVM	Support vector machine
THC	Through-hole component
TQM	Total quality management
UCL	Upper control level

Symbols

General symbols

C	General symbol for correlation
D	General symbol for distance
e	General symbol for error
M	The number of neurons
m	General symbol for a neuron
N	The number of input vectors (data rows)
P	The number of variables
x	General symbol for a data row (input vector)
y	General symbol for output

Self-organizing maps

e_q	Quantization error
e_t	Topographic error
e_d	Distortion measure
h	Neighborhood function
k	Iteration round (discrete time coordinate)
R	Set of reference vectors
r	Reference vector
v	Location vector of a neuron
α	Learning rate factor
β	Index of the best matching unit
σ	Width of neighborhood

K-means

c	Cluster center
D	Distance between clusters
D_{DB}	Davies-Bouldin –index
k	Number of clusters
n	Number of data rows assigned to cluster
S	Within-cluster distance

Linear regression

w	Model parameters that are fitted
y	Model output
ρ	Residual; difference between the expected and the estimated value
Θ	Sum of squared residuals

Multilayer perceptrons

a	Symbol for a constant term
b	Bias term
d	Expected value of the response vector
e	Error signal for a neuron
K	Total number of iteration rounds
k	Iteration round (discrete time coordinate)
o	Output neuron
u	Output of linear combiner
w	Synaptic weight of a neuron
y	Output signal of a neuron
α	Learning rate factor
δ	Local gradient
Λ	Output layer
λ	Layer of the network
μ	Momentum constant
υ	Activation potential (induced local field) of a neuron
ϕ	Activation function

Pre- and postprocessing

C_{IA}	Index of agreement
D_{euc}	Euclidean distance
k	Number of subsets in cross-validation
O	Observation
Π	Selected subset of variables
σ	Standard deviation

LIST OF ORIGINAL ARTICLES

This thesis is based on data presented in the following articles, referred to by the Roman numerals I-IV. The articles are reproduced with the kind permission of their publishers.

I Liukkonen M., Havia E., Leinonen H., Hiltunen Y. Application of Self-Organizing Maps in Analysis of Wave Soldering Process. *Expert Systems with Applications, 36(3):4604–4609, 2009*.

II Liukkonen M., Hiltunen, T., Havia E., Leinonen H., Hiltunen Y. Modeling of Soldering Quality by Using Artificial Neural Networks. *IEEE Transactions on Electronics Packaging Manufacturing, 32(2):89–96, 2009.*

III Liukkonen M., Havia E., Leinonen H., Hiltunen Y. Quality-oriented Optimization of Wave Soldering Process by Using Self-Organizing Maps. *Applied Soft Computing, 11(1):214–220, 2011.*

IV Liukkonen M., Havia E., Leinonen H., Hiltunen Y. Expert System for Analysis of Quality in Production of Electronics. *Expert Systems with Applications, 2010*. Submitted for publication.

AUTHOR'S CONTRIBUTION

The four research articles included to this thesis have been written during the years 2006–2010. All the articles were produced by the author for the most part, including the processing and analyses of data, literature reviews and the preparation of manuscripts.

The roles of Elina Havia and Hannu Leinonen from the 3K-Factory of Electronics were to offer their expert knowledge of the production of electronics, especially the wave soldering process. Their knowledge was especially useful in calculating the unit costs for different defect types in article III. They have also kindly revised the description of the wave soldering process written by the author and evaluated the analysis results of all the articles from the process experts' point of view.

The role of Teri Hiltunen in the article II was to aid with the practical implementation of selecting variables based on multilayer perceptrons.

Professor Yrjö Hiltunen's role was supervisory, including his kind help in the structural organization and advice on writing the articles. He also offered generously his abundant knowledge of the analysis methods used and helped in solving the numerous practical problems encountered during the research.

Contents

1. Introduction

Electronics industry is confronting many challenges nowadays. Modern manufacturing of electronics is characterized by rapid changes in production, technically more complicated products, the pressure to decrease the production costs, and at the same time higher demands for the product quality. The challenging situation has created a need for new methods that could be used in process diagnosis. In this respect, the so called intelligent methods that utilize process history and have been uncommon in the electronics industry until today offer a promising platform for developing new procedures for data analysis. Intelligent data-based quality analysis can potentially provide a useful path to process improvement.

1.1 PROCESS INFORMATICS

Increasing amounts of numerical and other data containing information on the production are produced in modern manufacturing environments (Choudhary et al., 2009; Wang, 2007). These data may be related to design, products, machines, processes, materials, maintenance, control, assembly, quality, process performance and so forth. Modern manufacturing lines contain many sensors and computer-controlled devices which offer information that can be utilized for instance in process control and optimization (Fountain et al., 2000). Because of the increasing number of components in electronic products, their electronic testing produces massive amounts of data, for example, which may be collected and stored in databases without deeper analyses.

However, it seems that the use of process data in process improvement, for instance, has been only partial. One possible reason for this is the difficulty of data analysis caused by the characteristics of the industrial process data, e.g. the large volume of data and missing values due to failures in measurement devices. For this reason, engineers and managers are having difficulties in handling and understanding the process data (Wang, 2007). Furthermore, it seems that the simple statistical methods used conventionally in data analysis are not adequate for handling the increasing amount of data (Wang, 2007).

Despite these difficulties historical process data has potential to be used for process diagnosis and improvement. The target of *process informatics* is to develop methods for analyzing and refining large amounts of data from a variety of industrial processes. As new methodologies for data processing are being developed, the data that may seem useless can potentially be utilized in process improvement. Process informatics uses not only the traditional methods of data analysis but also the most recent methods of information technology to achieve this.

The Process Informatics research group of the University of Eastern Finland uses many data processing methods and algorithms from simple plotting to more sophisticated modeling methods, such as artificial neural networks. The main target of process informatics is to produce intelligent and adaptive systems and software that can be used in process improvement, monitoring and control based on real process data. Such applications are presented by Hiltunen *et al.* (2006), Heikkinen *et al.* (2008, 2009a–b, 2010), Juntunen *et al*. (2010a–b), and Liukkonen *et al.* (2007, 2008, 2009a–e, 2010a–g), for example. In addition, the group has gained expertise on the development of new measurements and measurement systems in challenging industrial environments.

1.2 ELECTRONICS INDUSTRY: A CHALLENGING ENVIRONMENT

Production of electronics involves many manufacturing processes from automated assembly lines to testing and final manual assembly (Khandpur, 2005). Manufacturing of electronics produces large amounts of information that could be used for process improvement through data analysis and modeling, for instance. The main problems in the electronics industry seem to be optimization problems such as how to improve production rates without affecting the quality. Thus in an ideal situation the production facility would be working at a 100% quality level (no defects), a 100% level of production load (no machines and other resources without work) and at the highest possible production rate with current resources.

Despite the fact that the process of automated manufacturing of electronics is several decades old, it is still under constant development. This is because of the trends prevailing in electronics manufacturing including time-based competition, increasing product variety and novel technologies (Helo, 2004). Especially certain environmental regulations (e.g. Directive 2002/95/EC: Restriction of the use of Hazardous Substances in Electrical and Electronic Equipment, ROHS), and the reduction of size and the increasing complexity of electronic products have created needs for further development in the 21st century (Barbini and Wang, 2005; Havia et al., 2005). Analysis and optimization of electronics production have remained extremely important because of these new challenges.

A factory used for manufacturing electronics is a challenging environment for process improvement in many ways. The environment evolves rapidly, and the electronic products tend to have short life cycles (Gebus & Leiviskä, 2009). Rapid exchange of different product types is essential in assembly lines, which necessitates a fast exchange of process parameters. In addition, introducing new products requires fast responding in the production. Automated optimization of production lines would be beneficial, but involves many problems such as the increasing frequency of developing new versions of products. Sometimes different product types are simply put through the process with the same process parameters, which increases the rework costs due to faulty products.

The models used in process improvement in the electronics production vary greatly from simple regression models to complex advanced models, such as artificial neural networks. It is important to note, however, that simple statistical methods are generally used for process analysis, because they are fast, simple to understand, and relatively easy to implement. The question is whether more advanced methods could be useful in the analysis of electronics production and whether they should be used more widely.

Quality assurance is getting an increasingly important role in electronics manufacturing (Khandpur, 2005). Improving the quality of final products is significant because the rework of faulty products may be expensive and binds resources that could be directed to some more productive work. On the other hand, achieving good quality also costs money, and sometimes it is difficult to determine the operational window of process parameters in which the optimum situation with respect to quality can be achieved. This is the part of data analysis in which advanced, multivariate data-based methods are needed and in which their benefits can be potentially exploited well.

1.3 ROLE OF COMPUTATIONAL INTELLIGENCE

The development of computational intelligence has created new intelligent methods for automated extraction of useful information (Wang, 2007), and recent years have involved an expansion of computationally intelligent applications to a large variety of industrial processes (Choudhary et al., 2009; Harding et al., 2006; Kadlec et al., 2009; Kohonen, 2001; Meireles et al., 2003). However, electronics manufacturers have adhered to traditional statistical or analytical analyses and models, and it seems that the process information available is not used as extensively and thoroughly as one could. It is quite common that modern production devices include good functions for data acquisition, but they are either not used at all or the data collected by the system are transferred to data bases without exploiting them thoroughly.

One of the advantages of computationally intelligent methods is their ability to learn and therefore generalize (Haykin, 2009), which means that they can be used to extract knowledge from large amounts of data. These methods have not, however, been adopted by the electronics industry in a larger scale. Thus it is necessary to determine whether the data from electronics production could be exploited more extensively to make the processes more effective by using intelligent methods such as artificial neural networks in data analysis. The question is whether these methods could provide new information when used in the quality analysis of electronics manufacturing.

Nonetheless, the possible applicability and usefulness of the methods in the electronics manufacturing is not adequate. Paying attention to usability, swiftness and robustness of the analysis methods is equally important from the manufacturer's point of view. Process engineers and other process experts often lack the time for tuning and learning the principles of the complicated computing algorithms, so the methods should preferably be applicable by pressing just a few buttons. Therefore the usability is a much more important issue in an industrial environment than for example reaching a slight improvement of model goodness by tuning its parameters. Good usability opens new opportunities for developing decision support systems and other data-based applications, for instance.

Thus, if the first question is whether intelligent methods could be used for process improvement in the electronics industry and the answer is yes, then the next question would logically be how it should be done. It would be most beneficial to create a procedure for modeling and optimization of electronics manufacturing processes, because that way the methods would become more easily applicable to a real industrial environment, for example in the form of different software applications.

1.4 SCOPE AND LIMITATIONS

The purpose of the study is to advance the use of intelligent data-based models in the production of electronics by exploring the current state of research and by applying intelligent methods to a real automated process of manufacturing electronics. The ultimate goal is to develop a methodology for modeling an electronics manufacturing process using intelligent methods.

The scope of this study is the quality analysis of soldering. Thus, other unit processes related to production of electronics are excluded from the literature review, for instance. Moreover, the concentration is on the applications of intelligent methods, and not in the theoretical issues of the methods and algorithms. On the other hand, such methods are presented and used in the analysis that are potentially applicable to an industrial environment and that are usable when large amounts of data are available.

The thesis consists of eight chapters. The typical production stages in electronics manufacturing are presented in Chapter 2, with a short presentation on quality management and conventional methods used in the analysis of quality in production of electronics. Intelligent methods for analyzing quality are described in Chapter 3. The aims of the study are presented in Chapter 4, whereas the wave soldering process and the data used in this study are presented in Chapter 5. The intelligent quality analysis of wave soldering is described in the subsequent Chapter 6, including the justification for selecting the computational methods and the main findings of the research. Moreover, the results are discussed in Chapter 7. These are followed by conclusions and some ideas presented for future work in Chapter 8.

2. Production of electronics

2.1 PROCESSES FOR PRODUCING ELECTRONICS

According to Khandpur (2005), "*Electronic equipment is a combination of electrical and electronic components connected to produce a certain designed function*". As the definition states, an essential part of an electronic device is the connection between components. The technique used mostly today in the inter-component wiring and assembly of electronic equipment is the *printed circuit board*, or *PCB*, which is also known as the *printed wiring board*. There are a large number of techniques for producing printed circuit board assemblies. The technique to be used is usually selected on the basis of the layout of the product, i.e. what sort of components it involves.

The types of electronic components can be divided into two basic categories, i.e. *through-hole components* (THC) and *surface mount components* (SMC), as presented in Figure 1. Generally different methods are used to mount these two component types. Through-hole assembly is typically a components-before-solder (CS) process (Judd & Brindley, 1999), whereas the surface mount assembly is most likely a solder-before component (SC) process. This classification is based on the order in which the placement of components and the application of solder are performed. In a CS process the components are positioned before the solder is applied, and in a SC process the solder is applied first.

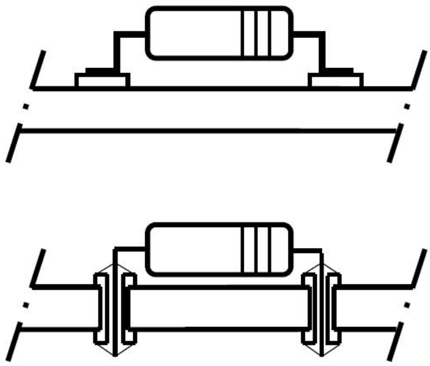

Figure 1: *The two main printed circuit board technologies in use. A surface mount component above, and a through-hole component below.*

Some typical assembly methods for different printed circuit board layouts according to Judd & Brindley (1999) and modified by the author are presented in Figure 2. Through-hole components are typically inserted before the soldering (Figure 2a), whereas in case of mounting SMCs the solder is generally applied first (Figure 2b). It is quite usual to use both THCs and SMCs in combination, however, which requires a combination of assembly methods. There are several possible ways to combine these technologies, of which an example is given in Figure 2c, in which SMCs are handled first via SC soldering, and THCs are attached afterwards by the CS soldering method.

It is also possible to attach SMCs with CS soldering. This necessitates an application of adhesive, however, to keep the component in place as it is transported to a soldering station. This method is beneficial if there are both THCs and SMCs on one side of the printed circuit board, because then all the components can be soldered at one operational stage.

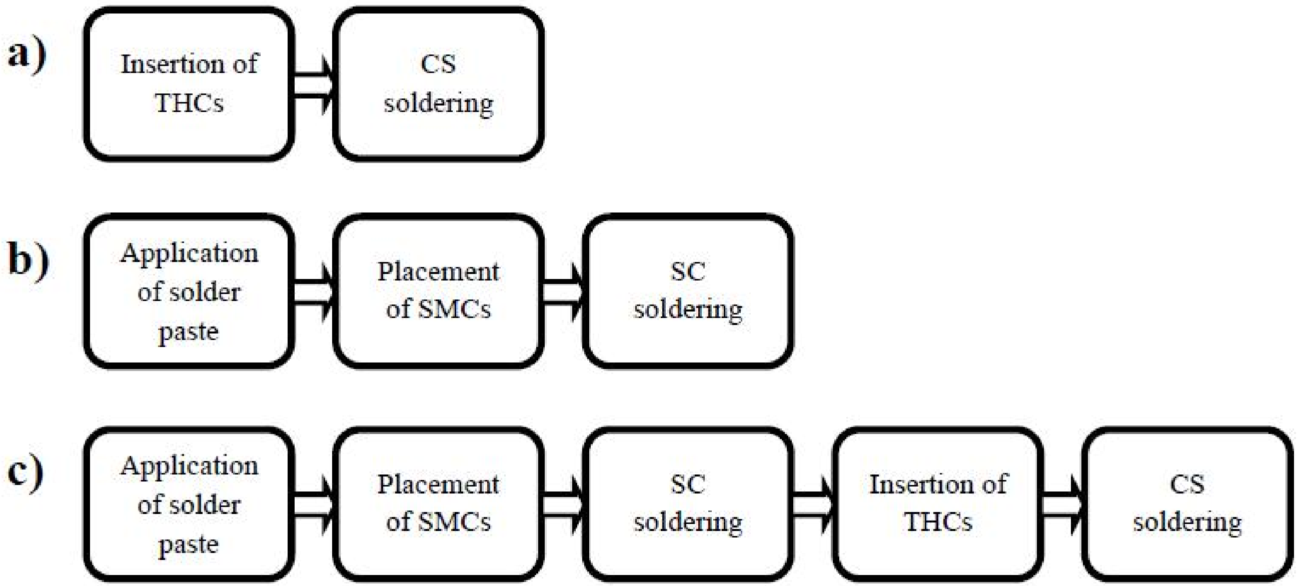

Figure 2: *Methods for automated production of electronic assemblies. a) Typical assembly method for through-hole components (THC), b) Typical assembly method for surface mount components (SMC), c) an example on the combination of THC and SMC assembly. SC denotes solder-before-components process, and CS components-before solder.*

2.1.1 Processes prior to soldering

Insertion of through-hole components: There are two methods for inserting THCs to printed circuit boards: manual and automated insertion. In manual placement a worker inserts the components following a specified placement procedure. Manual insertion is reasonable when there are a relatively low number of components to be inserted or especially when several small batches of different products are produced. This is because a change of product reduces the overall mounting rate of automated insertion machines, as the insertion program and reel packs have to be changed. Manual insertion is also obligatory in special cases, since some component types cannot be inserted automatically.

In automated placement the components are inserted by a machine. An automated through-hole insertion machine consists of a placement head, tools for picking up components and sensors and vision inspection cameras for verifying the correct placement of components. The machine picks up the components from the reel packs, inserts them accurately by following an insertion program and finally bends or cuts the leads of the components to a suitable length.

Application of solder paste or adhesive: Automated deposition of solder paste or adhesive can be performed using *stencil printing*, *dispensing* or *pin-transfer*, of which stencil printing is the most commonly used technology for the deposition of solder paste (Lee, 2002). Also other methods for the deposition of paste or adhesive exist, but their use is marginal compared to the three methods mentioned.

In *stencil printing* (see Figure 3a) a metal foil (= stencil) with a pattern of apertures matching the connection pads of the PCB to be soldered is placed precisely on top of the board. Next, the solder paste is applied onto one side of the stencil. The paste is then wiped across the stencil by using a squeegee, and as the PCB is detached from the stencil, the solder paste remains on top of the corresponding pads. (Lee, 2002).

The benefits of stencil printing include a high speed, high throughput and better control of the volume of solder paste, for example (Lee, 2002). Sometimes stencil printing is not possible or desirable, however. A requirement for flat surface, for example, limits the use of stencil printing in rework or in attaching components onto non-flat surfaces (Lee, 2002). In such cases other methods have to be considered.

In a *dispensing* process (see Figure 3b) solder paste (or adhesive) is forced through a needle to the connection points of a PCB. Dispensing can be cost-effective when the product batches are small, for instance (Judd & Brindley, 1999). In *pin-transfer* (see Figure 3c) a matrix of pins is mounted on a holder with a pattern that matches the connection pads of the PCB to be soldered (Lee, 2002). Pin-transfer is a quite rarely used method for depositing solder paste or adhesive, and it is used mostly in special cases.

***Placement of surface mount components*:** Placement of SMCs is usually performed using automated pick-and-place machines, or *placement machines*. The placement machines can be categorized broadly into two groups according to their placement timing (Khandpur, 2005): sequential and simultaneous placement.

In sequential placement components are placed one after another in a specified order, in which the sequence is determined by a placement program. Simultaneous configuration is designed for placing all components onto a PCB in a single operation. Simultaneous placement systems can have a placement rate of as high as 200 000 components per hour and are suitable for companies with very high throughput requirements (Khandpur, 2005). The sequential placement is more beneficial for small and medium batch sizes, however.

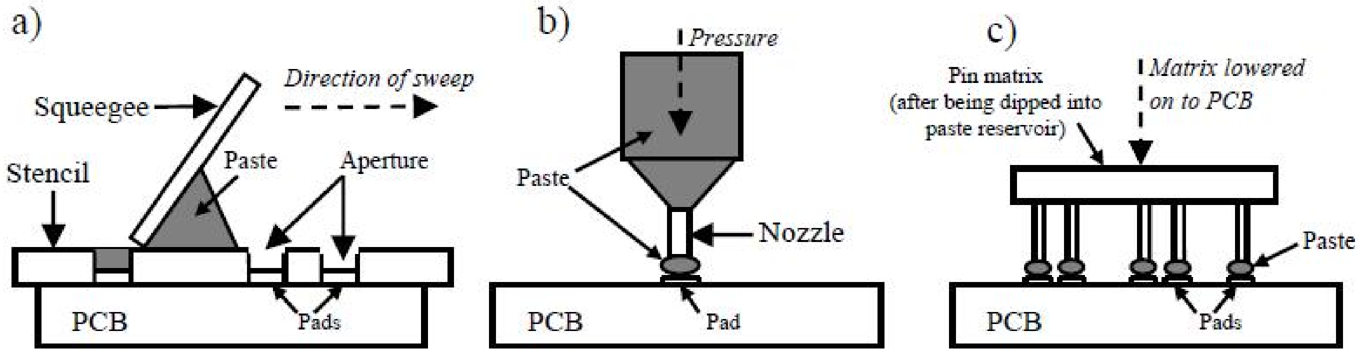

Figure 3: *Methods for deposition of solder paste or adhesive. a) Stencil printing, b) dispensing, and c) pin transfer.*

Chip shooters can be used in PCB assembly of special applications. Chip shooters are placement machines which operate at high speed and are designed particularly for chip components. Generally chip shooters are used to place small, passive components such as resistors and capacitors, while pick-and-place machines are used to place packages of a larger size (Khandpur, 2005).

2.1.2 Soldering methods

Electronics assembly generally involves soldering of components onto a printed circuit board. Two metal surfaces are joined together in soldering by metallic bonds, which are created as the molten solder between the PCBs connecting pad and the termination of the component solidifies (Judd & Brindley, 1999).

Hand soldering**:** *Hand soldering* is a process in which the components are soldered individually and manually with a soldering iron (Judd & Brindley, 1999). Because it is time-consuming, the use of hand soldering is generally restricted to special cases such as soldering individual components that are difficult or problematic to solder in mass, or reworking and repairing. In addition, soldering of small SMCs manually is difficult, which has decreased the popularity of hand soldering in mass production of electronics.

Mass soldering**:** In *mass soldering,* also called machine soldering or automatic soldering, several components are soldered onto a board simultaneously without manual application of solder. Machine soldering methods can be further divided into four basic types: *dip soldering*, *drag soldering*, *wave soldering* and *reflow soldering* (Khandpur, 2005). Another way of categorizing the processes is based on the order in which the components and the solder are added onto the board (Judd & Brindley, 1999). In wave soldering, for example, components are inserted before the solder is applied (CS process). In contrast, reflow soldering involves an application of solder prior to placement of components (SC process). Dip, drag and wave soldering (see Figure 4) are generally used as CS processes, whereas reflow soldering is usually considered a SC process.

In dip soldering an assembled PCB is lowered into a bath of molten solder, as presented in Figure 4a. The board is kept in the bath for a suitable time (typically 2–3 seconds according to Khandpur, 2005), and then lifted off. Figure 4b presents the drag soldering process, in which the PCB is dragged over the surface of molten solder. (Judd & Brindley, 1999)

Dip and drag soldering are nowadays rarely used for mass soldering applications. Wave soldering (see Figure 4c) is the standard method for mass soldering of leaded through-hole components (Khandpur, 2005). In wave soldering a solder pump creates a wave of molten solder, over which the PCB assembly is transported. The typical contact times vary from 1 to 4 seconds according to Khandpur (2005). Wave soldering is discussed more deeply in Chapter 5.

Surface mount assembly is a process of *reflow soldering*, above all. Commonly used mass reflow methods include infrared, convection, vapor phase, and conduction-based reflow systems, although there are also other, not so common methods used especially in low volume production. Combinations of different heating systems are also commonly used. (Lee, 2002)

Characteristic for all reflow soldering methods is that the PCB assembly, with the solder paste applied and components placed onto it, is heated in a reflow oven, whereby the paste melts and the solder forms the connection between components and connection pads. Reflow soldering has become the primary technology for mass soldering of surface mount components because of its flexibility and high throughput rate.

2.1.3 Other processes

Production of electronics often comprises a complex and varying set of unit processes. It is noteworthy that the production of an individual electronic product can involve dozens of unit processes varying greatly with respect to their complexity and duration. This poses many challenges to process management of a company manufacturing electronics.

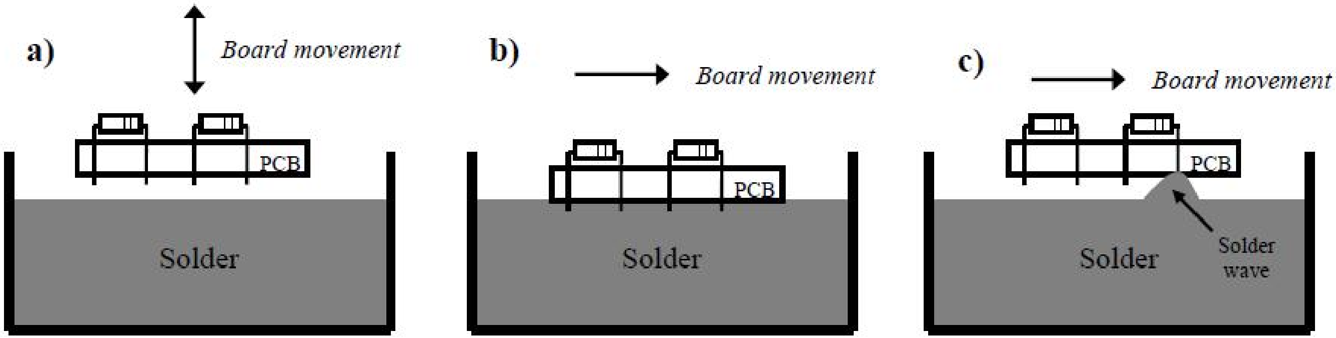

Figure 4: *CS soldering methods. a) dip soldering, b) drag soldering, and c) wave soldering.*

2.2 QUALITY MANAGEMENT IN THE ELECTRONICS INDUSTRY

2.2.1 Definitions

The word *quality* often evokes an illusion of a marvelous product or service that surpasses, or at least fulfills, our expectations. However, quality is also defined by the price paid for the product, which complicates its exact definition. According to DIN 55350 standard (Quality assurance and statistics terms) part 11, quality is the combination of characteristics of a device with regard to its eligibility for satisfying the specified and assumed requirements. In the industry these requirements often come directly from the customer, but they can also be specifications or goals set by the manufacturer himself.

A quite similar definition for quality is presented in the ISO 9000 (Quality management systems - Fundamentals and vocabulary) standard. The standard states that the quality can be determined by comparing a set of inherent characteristics with a set of requirements. If the characteristics meet the requirements, high quality is achieved. In contrast, if the characteristics do not fit the requirements, the quality is low. In other words, quality is a question of degree that can be measured with a pre-defined scale.

According to ISO 9000 standard, *quality management* includes all the activities that organizations use to direct, control and coordinate quality. Quality management can be considered to consist of five elements having different goals, as presented in Figure 5: quality policy, quality planning, quality control, quality assurance and quality improvement, between which the limits sometimes overlap each other. Especially the terms *quality control*, *quality assurance* and *quality improvement* are often mixed with each other in the literature.

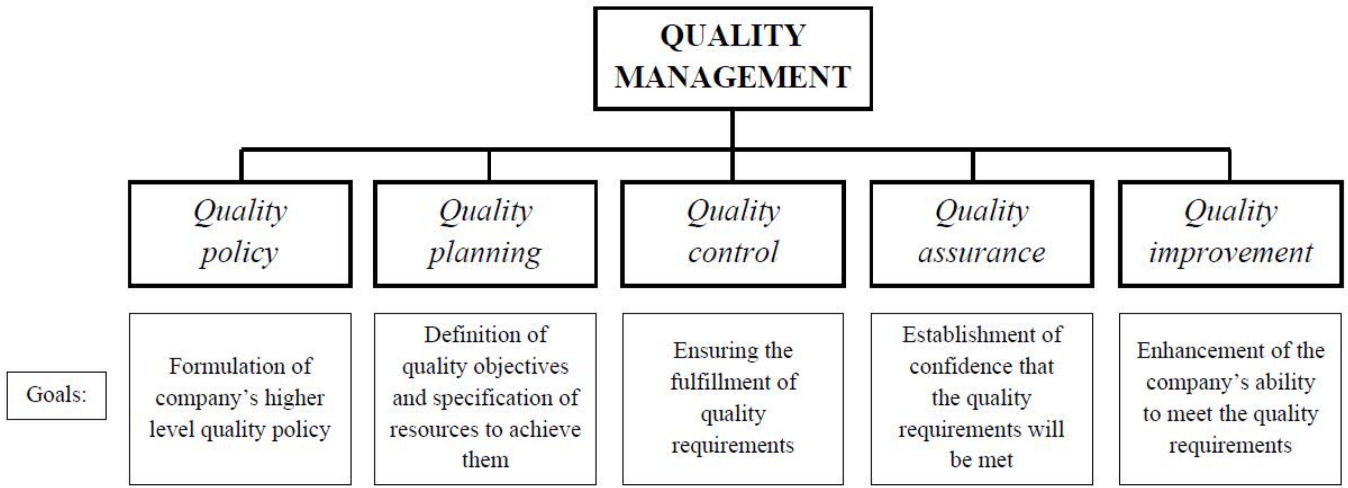

***Figure 5**: The quality management activities of a company and their primary goals according to ISO 9000 standard.*

Nonetheless, certain distinctive features can be extracted from these concepts. Quality control can be seen as a continuous effort to maintain the reliability of single products including routine technical activities for measuring and controlling quality, whereas quality assurance seeks to guarantee a standard level of quality in the whole production from raw materials to delivery. Testing of products can be considered a part of quality control, for example, and a quality analysis of a set of manufactured products can be considered a part of quality assurance. From this point of view quality control can be seen as one part of quality assurance. Quality improvement, on the other hand, can be seen as a one-time purposeful change of a process to *improve* quality, i.e. the suitability of a product for the purpose it is intended to. Thus, it can include process optimization, for instance.

2.2.2 Quality in the electronics industry

The global-scale competition has made it crucial to manufacture products with a low cost and a high quality (Gebus & Leiviskä, 2009). In terms of quality, the environment for producing electronics is challenging, because products and their different versions are changing continuously. In addition, the gamut of different products can be wide, which increases the number of different line specifications needed. Furthermore, new component types and assembly methods set requirements for production, which has to be taken into account in process design.

All kinds of changes in the production, both sudden and long-term, potentially cause problems, and the problems often cause defective products. Detecting these defects and identifying the reasons for their formation is important because a large part of the safety and reliability of an electronic device depends on the proper function of its electric components and their solder joints.

A *defect* can be defined as any aspect or parameter of the PCB that does not fit the specified requirements (Khandpur, 2005). Defect types can be classified according to the degree of their seriousness. The most serious defects, or *critical defects*, are likely to cause hazardous conditions for the user of the product. *Major defects* of PCBs are likely to produce a failure in the final product. In addition, such defects may appear that are not necessary to be repaired. These can be merely cosmetic flaws such as flux residues or small solder spatters, for instance. The defects not reducing the usability of the product are called *minor defects*.

2.2.3 Quality assurance of electronics production

Quality assurance is an essential and increasingly important concept in the production of electronics (Khandpur, 2005). In a wider context quality assurance involves the whole production chain: product design, inspection of incoming materials, methods for preventive quality assurance, techniques for quality inspection and testing of the semi-products, methods for online quality control, final testing and methods for analyzing the quality after production. The ideal situation would be that the quality-related feedback from the downstream production stages would be available in the upstream stages, the most preferably as far as at the designing stage. This would offer a path to predictive quality assurance.

Thus, the main function of quality assurance should be continuous monitoring of all factors and parameters contributing to reliability and other quality-related properties of the manufactured products. Quality assurance should cover the whole spectrum of production stages, because design, fabrication, assembly, soldering, quality inspection, packing etc. are all stages susceptible to problems and defects. It is also vital that the production data used in quality assurance are adequate, and that the methods used in data analyses are able to extract reliable and useful knowledge.

2.2.4 Techniques for quality inspection and testing

Production of PCBs involves a large number of process steps, so it is important to perform proper inspection and testing in different stages to ensure the quality of products (Khandpur, 2005). Especially final testing is vital to assure the quality of final products. Different quality tests have been developed to aid in determining how well a manufactured item satisfies the requirements for quality (Sauer et al., 2006). Plenty of inspection and testing methods have been introduced, so only the most common of them are presented here shortly.

Visual inspection: *Visual inspection* means simply the visual quality check of a semi-product or product performed by personnel specially trained for it. Visual inspection is an efficient method for detecting visible flaws, e.g. missing components or solder bridges. Usually certain locations on the PCB, i.e. solder joints, susceptible to faults are inspected with special care, but the method also makes it possible to get a general overview of the quality of the product. Visual inspection is still one of the most popular methods for inspecting the quality of electronic products, and it has many benefits such as flexibility and speed (Manko, 2001).

Despite its strong advantages visual inspection has become less effective for double-sided and multilayered boards, because component densities on PCBs have increased (Khandpur, 2005). In addition, the increasing use of small components, e.g. ball grid arrays, has reduced the popularity of visual inspection (Judd & Brindley, 1999). This has necessitated the development of penetrating methods such as X-ray scanning for quality inspection.

Automated optical inspection (AOI): *Automated optical inspection* is an automated optical method based on machine vision for detecting component misalignments or solder joints on PCBs. The system utilizes an inspection algorithm which compares an ideal reference picture to scanned images of boards in the inspection, taken by one or more cameras (Moganti et al., 1996). The other option is to use dimensional verification exploiting CAD files of the boards (Gebus, 2006).

Automated X-ray inspection (AXI): In *automated X-ray inspection* the solder joints are scanned by using X-rays, which produces an image in which soldering defects can be tracked by detecting differences in material density or thickness (Neubauer, 1997). X-ray inspection has been developed, because it is almost impossible to detect certain defects in the inner layers of multilayered boards by visual or electrical continuity check (Khandpur, 2005). AXI is a relatively slow method, so it is generally used in inspecting individual samples from production.

In-circuit testing (ICT): The development of surface mount technology has lead to increasing component densities on PCBs. For this reason, different automated electrical testing methods have been developed for testing high density boards. *In-circuit testing* is used to locate defects and isolate misaligned or missing components on PCB assemblies (Khandpur, 2005). ICT is an electrical method for analyzing the internal function of an electronic device. IC-tester consists of a matrix of probes that is connected to the circuit nodes on the board, after which a signal is sent to stimulate each node and the measured response is then compared to predefined limits (Khandpur, 2005). There are different techniques used to connect the testing equipment with the board, the most popular of which are a bed of nails, probing and boundary scan.

Bed of nails tester equipment consists of spring loaded pins lowered on certain test points on the board (Khandpur, 2005). Under the control of a test program, signals are sent via the pins to check the electrical continuity of a PCB. The responses are then used to detect possible wrong components, short-circuits and other soldering defects.

Probe testing is a testing set-up in which two or more probes mounted onto small heads move in an X-Y plane to test predefined points on a PCB (Khandpur, 2005). Although probe testing is relatively affordable, it is rather limited by its capacity: probe testers can test only few points at one time, whereas a bed of nails tester is able to test thousands of test points at a time, (Khandpur, 2005).

Functional testing (FT): *Functional testing* aims at testing the operations and functioning of a PCB. In FT, data corresponding to its real life working conditions are input to the device, and the response is used to analyze its functionality (Khandpur, 2005). It is thus an effective method for locating faulty components (Khandpur, 2005). The drawbacks of functional testing include complexity, difficult fault localization and its incapability of performing diagnostic at the component level (Gebus, 2006), for which it is often used in combination with IC-testing. This enables detecting both functional and productive flaws.

Environmental testing: An electronic device should be able to endure the conditions of the environment it is intended to. *Environmental testing* can be used to discover the reliability of the device and its resistance to different climatic, mechanical and chemical conditions. This is carried out by exposing the device to environmental strain induced in a predefined testing program. Strain tests of this kind are thermal stress, thermal shock and moisture resistance tests, for example (Khandpur, 2005).

2.2.5 Statistical process control (SPC)

The purpose of process control is to monitor individual manufacturing processes to ensure that the required quality is reached (Sauer et al., 2006). *Statistical process control* (SPC) is the application of statistical methods to the measurement and analysis of variance in a process (Khandpur, 2005), and is thus a general tool for process control. SPC is employed commonly for managing quality in the electronics industry (Smith & Whitehall, 1997). Moreover, SPC methods belong to the group of standard techniques of quality assurance in mechanical engineering (Sauer et al., 2006). Sampling is an integral part of statistical process control. According to Sauer *et al.* (2006), for example, the SPC methods "*permit the conclusion drawn from a few observation values of a sample to be applied to a total set of products produced under the same conditions*".

Product attributes are generally mapped as distributions in statistical control. A process that is in control produces only random variation within acceptable limits, whereas in a process that is out of control assignable causes of variations occur, producing unpredictable results (Judd & Brindley, 1999). These variations have to be controlled, which is performed by varying the parameters of the process. SPC can include a variety of statistical methods for analyzing data, ranging from simple methods such as design of experiments and control charts to more advanced statistical methods such as variance analysis and regression. The main stages of SPC are:

1) Data gathering
2) Data analysis
3) Data usage in process control

Because real production data is used by the SPC, the data has to be collected first. For this reason, many production devices have nowadays versatile features with regard to gathering of data. After data are analyzed, the extracted information is used in process control to ensure adequate quality of the products.

2.2.6 Programs of quality management

Over the years two strategies have been exploited on a larger scale by the electronics manufacturers to improve the total quality of their organizations. Both of them use statistical methods to improve the quality of production.

Total quality management (TQM): *Total quality management* (TQM), which was introduced in mid-1980s, has been widely applied to improve competitiveness around the world (Samson and Terziovski, 1999). TQM is a management strategy used to enhance quality in all organizational levels, the main goal of which is in long-term improvement of quality through customer satisfaction (Gebus, 2006). Statistical methodologies such as SPC and experimentation are used in TQM as tools for improving quality (Smith & Whitehall, 1997).

Sigma programs: *Six Sigma* has become a popular methodology in quality management of manufacturing. Its goal is to eliminate defects in a process by aiming at six standard deviations between the mean and the nearest specification limit (Sleeper, 2005). Six Sigma is a measurement-based strategy for business management, and its fundamental objective is to identify and remove the causes of defects and variation in manufacturing (Gebus, 2006). Six Sigma uses different data-driven statistical methods to achieve this goal.

Six Sigma is loosely founded on *3σ*, which is an earlier specification of process capability. 3σ program involves three general rules (Smith & Whitehall, 1996):

1) Process output is normally distributed.
2) Process mean is used as the target value.
3) The acceptable range of outputs ranges from -3σ (the mean minus 3 standard deviations) to +3σ (the mean plus 3 standard deviations), as presented in Figure 6.

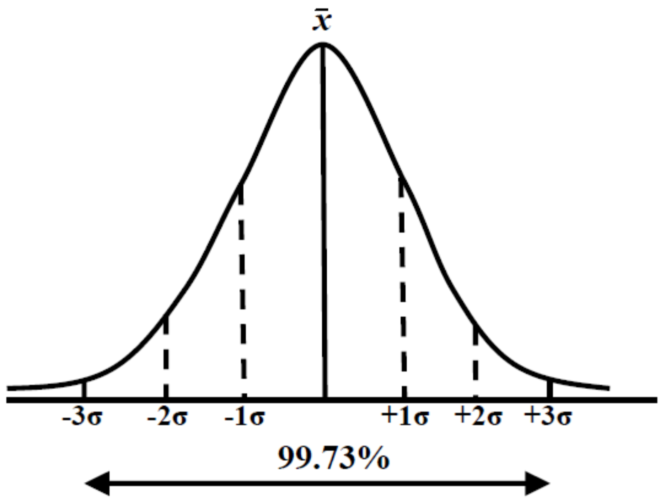

Figure 6*: The range of 3σ quality (Modified from Smith & Whitehall, 1996).*

The Six Sigma program, which was derived from the 3σ philosophy in the mid 1980s by Motorola, aims at improving the process capability so that it operates in the range of six standard deviations (Sleeper, 2005). In practice, a process with a Six Sigma performance for quality would generate only 3.4 defects per million opportunities. This provides the assumption that the mean value is not constant, however (Smith & Whitehall, 1996).

Six Sigma can also be considered a quality improvement strategy in a wider context. According to Pande & Holpp (2002) the main targets of the method are:

- Improvement of customer satisfaction
- Reduction of cycle time
- Reduction of defects

Kwak and Anbari (2006) have reported that Six Sigma has been successfully applied by several organizations that manufacture electronics. Despite the popularity of this strategy in the electronics industry, it has also encountered criticism. Smith & Whitehall (1996) stated, for example, that one of the failings of Six Sigma is its inability to recognize differences in production challenges. Furthermore, the fact that the methodology does not take the financial aspects into account is a major drawback according to them. For a more detailed contemplation of pros and cons of Six Sigma, the reader may refer to Antony (2004).

2.3 STATISTICAL METHODS FOR ANALYZING QUALITY

Methods that have been used traditionally in the data analysis of soldering quality include troubleshooting charts, control charts and design of experiments. Usually these methodologies exploit conventional statistical calculus such as regression or variance analysis. Taguchi is a more recent statistical method that is popular in quality analysis nowadays.

2.3.1 Troubleshooting charts

Troubleshooting chart is a conventional and user-friendly method for diagnosing the quality of a process. The problem-solving knowledge of these diagrams is founded on process expertise and former experience. The user is guided through the troubleshooting by following a specified procedure of questions. Ultimately the guide gives the probable causes of the problem concerned. Bernard (1977), Borneman (1981) and Pascoe (1982), for example, developed early troubleshooting guides for wave soldering. Expert systems relying on prior expert knowledge restored in a database, e.g. as presented by Randhawa *et al.*

(1986) and Fidan & Kraft (2000), can be considered more advanced methods for troubleshooting.

2.3.2 Control charts

A control chart is a visual method for evaluating the amount of variation in a process and for identifying the situations when the process goes out of control (Smith & Whitehall, 1997). The basic purpose of a control chart is to ensure that the characteristic values of a product or process-related data remain within specified limits (Sauer et al, 2006), so it is thus a tool for process control. An example of a control chart is given in Figure 7.

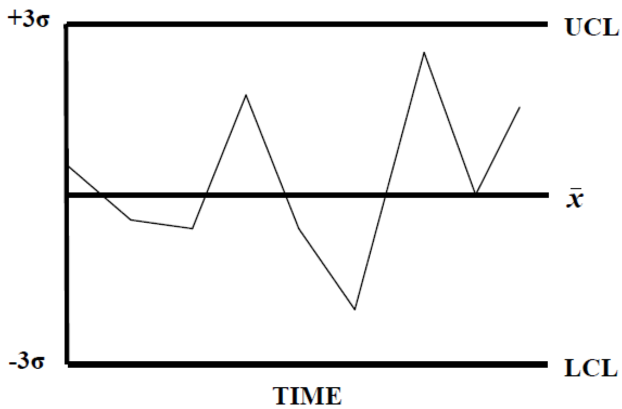

Figure 7*: A control chart of a process with a normal situation (no control needed). UCL = upper control level, LCL = lower control level.*

In principle, working with control charts means that the numerical values of a variable are monitored, and if they differ more than a preset value, say e.g. three standard deviations, from the mean value of the variable that is monitored, the process will be controlled.

Applications: The control chart, also known as the Shewhart chart, was originally developed by Shewhart in 1920s. Control charts can be found nowadays in most electronics assembly plants (Smith & Whitehall, 1997). For example Prasad and Fitzsimmons (1984), Brinkley (1993) and Santos *et al.* (1997) have reported on the use of control charts in the evaluation of a wave soldering process.

There are also several variants of control charts in use. Ho *et al.* (2003) suggested, for example, that the so called demerit control charts developed by Dodge and Torrey (1977) can outperform the standard control charts when dealing with various defect types with different degrees of severity in reflow and wave soldering. As some defect types can have serious consequences on the quality of the product and others only minor, the demerit chart takes these different degrees of seriousness into account when defining the control limits for a process. Stuart *et al.* (1996) have presented a compact review on different control charts.

2.3.3 Design of experiments (DoE)

Design of experiments (Fisher, 1935), which is also called *experimental design*, is one of the most important methods for statistical data analysis. It is a discipline of statistical test planning used to gather information by doing experimentations in processes in which variation is present. DoE gives an opportunity to analyze processes for the minimization, maximization, or specific setting of response variables (Sauer et al., 2006).

Careful implementation of DoE is vital when dealing with data-driven applications. An empirical model based on data of insufficient quality will not be able to predict the output reliably. Problems can also occur if data samples do not reflect the entire range of practicable machine operation or do not even include the optimal process settings (Coit et al., 1998). Successful factorial experimentation depends largely on the selection of variables included to the experiments (Smith & Whitehall, 1997).

Applications: There are several examples of using experimental design in automated wave soldering. Lim (1989 & 1990), Briggs and Yang (1990), Sterritt (1991), Brinkley (1993) and Mesenbrink *et al.* (1994) exploited the DoE methodology in the wave soldering process in the early 1990s, for instance. In addition, Hoe *et al.* (1998) used a combination of a physical model and a DoE-based verification to model the thermal profile of the preheat section in wave soldering. Moreover, Snitko (1998) described the key steps of experimental design with an actual DoE performed on the process, and Chowdhury *et al.* (2000b) used the method to improve a wave solder process with a high defect level, achieving a remarkable reduction of defects.

Some more recent applications indicate that DoE is still an important method for improving the wave soldering process. Arra *et al.* (2002), for example, developed a DoE for lead-free wave soldering to study the effects of different factors on soldering performance. Tsenev and Marinov (2004) described a successful application of DoE to reduce the defects in the process, and Dror and Steinberg (2006) proposed a robust technique of experimental design for multivariate generalized linear models and tested it in a wave soldering process. Furthermore, Boulos *et al.* (2009) used experimental design to define the optimal parameters for selective wave soldering.

2.3.4 Regression analysis

Regression analysis comprises techniques for modeling several variables. Its goal is in finding the relationship between a dependent (*response*) variable and at least one independent, or *explanatory*, variable (Rawlings et al., 1998). The goal of regression analysis can be curve fitting or parameter estimation, for instance.

Linear regression: *Linear regression* is the most popular and simplest regression method. In linear regression the response variable is expected to be a linear combination of certain explanatory variables.

Nonlinear regression: In *nonlinear regression* the data are modeled using a function which is a nonlinear combination of the model parameters (Rawlings et al., 1998). As in linear regression, the goal is to discover such parameters for the model that minimize the defined error function. The response variable can depend on one or more explanatory variables. Several iterative parameter estimation methods have been used for nonlinear regression, of which the feed-forward neural network is one of the most popular (see Chapter 3).

Applications: There are a few examples of using the regression method in the analysis of wave soldering. Scheuhing & Cascini (1990) and Dror & Steinberg (2006), for instance, used multivariate linear regression to model the process. Moreover, Mesenbrink *et al* (1994) used weighted polynomial regression to model the performance of different soldered leads as a part of a procedure for quality improvement of a wave soldering process. Coit *et al.* (2002) used linear regression analysis to predict mean temperature and temperature gradient at the wave, and the mean temperatures at pre-heaters 1 and 2 in a wave soldering process, managing to get successful results only for the last case. In contrast, their application of neural networks to the same problems gave a better performance. Furthermore, Sauer *et al.* (2006) presented an example of the use of regression analysis in proving a relationship between the duration of a compressed air blast and the applied volume of a paste medium in an automated dispenser machine.

2.3.5 Variance analysis

Comparison of variances is a simple and useful method for comparing the quality of machines and processes. If the variance σ^2_{P1} of the process $P1$ is smaller than the variance σ^2_{P2} of the process $P2$, for example, the conclusion is that the process $P1$ can provide better quality (Sauer et al., 2006). If the variances σ^2_{P1} and σ^2_{P2} are not equal, the variance behavior is different and the capability of the processes to produce quality is also different.

It is often necessary to compare more than two machines or processes, however, which can make the above comparison difficult and expensive. Analysis of variance (ANOVA) can be used to perform these comparisons independently of the number of objects (Sauer et al., 2006). In ANOVA the variation in a response variable is partitioned into components that correspond to different sources of variation, i.e. to different explanatory variables (Yang & Trewn, 2004). The aim is to divide total variation into a portion caused by random error and portions caused by changes in the explanatory variables.

In its simplest form ANOVA is a parametric model that can be used to test the hypothesis that the means among two or more groups are equal, assuming that the samples are normally distributed (Yang & Trewn, 2004). Thus it expands the Student's t-test by involving more than two groups. Multivariate analysis of variance (MANOVA) involves several response (dependent) variables.

Applications: Mesenbrink *et al.* (1994) and Tsenev & Marinov (2004) used the analysis of variance in the analysis of wave soldering. Chowdhury and Mitra (2000), and Chowdhury *et al.* (2000) also exploited the analysis of variance in the quality improvement of a wave soldering process struggling with a high defect level. In addition, Lin *et al.* (2007) conducted ANOVA to identify the factors that potentially had significant impacts to a reflow soldering process. This was performed as one part of their modeling and optimization scheme, which also involved the use of a neural network.

2.3.6 Taguchi methods

Genichi Taguchis original way of pursuing robustness and reduction of variation by experimental design had a major effect on the statistical process control and improvement in the 1990s (Stuart et al., 1996). Taguchi is a statistical methodology that can be used in determining the parameters affecting quality and identifying the factors causing variance in a process through analysis of variance, interaction charts and optimization (Yang & El-Haik, 2009). The aim is to minimize variance by finding the optimal values for controllable variables.

Taguchi is a combination of engineering design principles (robust parameter design) and a special version of experimental design, which is known as the *orthogonal array experiment* (Yang & El-Haik, 2009). The experiment outputs an interaction table and linear graphs associated with it, which show the interactions between variables. Only the main effects and the interactions between two factors are considered in a Taguchi experiment, and interactions of higher order are assumed to be nonexistent (Yang & El-Haik, 2009). This is a difference to the regular experimental design.

Applications: Shina (1989) and Lulu & Rao (1990) presented one of the earliest applications of Taguchi to the optimization of wave soldering. No sooner than in the late 1990s and in the 21st century, however, the Taguchi method has become more popular than ever in the electronics industry. Williams and Raaijmakers (1999), for example, used Taguchi in the analysis of wave soldering, and Diepstraten (2001) analyzed lead-free soldering defects in wave soldering and defined the parameters for the process using the method.

Barbini and Wang (2005) exploited Taguchi to address certain issues and knowledge gaps in lead-free wave soldering. Moreover, Zhou *et al.* (2008) used the method as a pre-processing element for an artificial neural network they used for modeling fatigue reliability of solder joints in plastic ball grid array packages. Also de Tino (2008) used the orthogonal array experiment in the robust design of a lead-free wave soldering process.

3. *Intelligent methods for analyzing quality*

3.1 COMPUTATIONAL INTELLIGENCE

The volume of data grows fast in modern manufacturing due to new digital production environments, which exploit technologies such as barcodes, sensors and vision systems in data acquisition (Choudhary et al., 2009). On the other hand, recent advancements in information technology, data acquisition systems and storage technology have promoted the use of manufacturing-related data in process improvement (Harding et al., 2006).

Moreover, recent development in the study called *computational intelligence* has produced new intelligent methods for automated extraction of useful information (Wang, 2007). Methods such as artificial neural networks, fuzzy sets, rough sets and evolutionary computation, which are generally associated with computational intelligence (Pal & Pal, 2002), are nowadays widely used in different industrial environments (Choudhary et al., 2009; Harding et al., 2006; Kadlec et al., 2009; Kohonen, 2001; Meireles et al., 2003).

Which features characterize the term *computational intelligence*? Different definitions proposed in the literature over the years show that *computational intelligence* (CI) is not an unambiguous concept. Poole *et al.* (1998), for example, presented the following definition for CI:

Computational intelligence is the study of the design of intelligent agents. An agent is something that acts in an environment—it does something. -- An intelligent agent is a system that acts intelligently: What it does is appropriate for its circumstances and its goal, it is flexible to changing environments and changing goals, it learns from *experience, and it makes appropriate choices given perceptual limitations and finite computation.*

According to the definition, computational intelligence is a rather wide concept. Certain elements can be distinguished in it, however: flexibility, ability to learn from experience and decision-making. Another definition was proposed by Engelbrecht (2007):

Computational intelligence is the study of adaptive mechanisms to enable or facilitate intelligent behavior in complex and changing environments.

This definition emphasizes the target, which is the complex or changing environment, at which the efforts are aimed. This is useful because the complexity and unpredictability of the system to be modeled are often the reason for which computational intelligence is relied on.

The definition suggested by Pal and Pal (2002) states that a computationally intelligent system should possess the following four properties:

- Considerable potential in solving real world problems.
- Ability to learn from experience.
- Ability to self-organize.
- Ability to adapt in response to dynamically changing conditions.

The characteristics listed above include elements of intelligent behavior such as observed in human beings (Pal & Pal, 2002). The definition presents also the purpose of CI, which is discovering solutions to problems. Computational methods in which the characteristics listed above are inherent are referred to as *intelligent methods* in this thesis.

CI can utilize a variety of methods to achieve its ultimate goal, which is solving problems. The most popular methods of computational intelligence are artificial neural networks, fuzzy sets, rough sets and evolutionary computation (Pal & Pal, 2002).

3.1.1 Hierarchy of concepts

Because humans are the end users of computational intelligence, it is useful to consider the objects of intelligence from the human's point of view. Ackoff (1989) proposed that the content of human mind can be grouped into five hierarchical categories:

1) *Data*: Symbols
2) *Information*: Data processed to a useful form
3) *Knowledge*: Application of data and information
4) *Understanding*: Comprehension of reasons
5) *Wisdom*: Evaluated understanding

The following example illustrates these terms more deeply. Numerical process data can be refined further to extract information by adding relational connections between data elements. When the information is used for process improvement, it turns into knowledge. Understanding a phenomenon within a process requires an analytical process based on previous knowledge, or at least information.

Wisdom is evaluated understanding with an aspect into future. It asks questions that have no answers yet, which offers an opportunity to improve the process.

3.2 KNOWLEDGE DISCOVERY AND DATA MINING

Data from almost all the processes such as design, material planning and control, assembly, scheduling, and maintenance, just to name a few, are recorded in modern manufacturing (Choudhary et al., 2009). These data have enormous potential of serving as a new source of information and knowledge, which can be used in modeling, classifying and making predictions, for instance (Harding et al., 2006). Thus, exploitation of collected data is becoming increasingly important in modern manufacturing. Extracting useful knowledge from an increasing amount of data has become extremely challenging, however, which has created a growing need for intelligent and automated methods for data analysis.

There are two important terms with respect to knowledge extraction: knowledge discovery in databases and data mining. The following definition for *knowledge discovery in databases* (KDD) was presented by Fayyad *et al.* (1996):

KDD is the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data.

KDD combines theories, algorithms and methods from several research fields such as database technology, machine learning, statistics, artificial intelligence, knowledge-based systems and visualization of data. *Data mining* is a special step in the KDD process, which involves applying computer algorithms to extract models from data. Besides data mining KDD includes data preparation, data selection and cleaning, integration of prior knowledge, and proper interpretation of results to ensure that useful knowledge is derived from data. (Choudhary et al., 2008). According to another definition (Hand et al., 2001):

Data-mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful for the data owner.

This definition highlights that a data mining method should be able to discover relationships that are not easily detectable and present the results in an understandable way. Data mining can involve the use of any technique for data analysis such as simple statistics as well as artificial neural networks.

3.3 MACHINE LEARNING

Machine learning is an important concept with respect to computational intelligence. Mitchell (1997) presented the following definition for machine learning:

Machine learning is the study of computer algorithms that improve automatically through experience.

The definition emphasizes the use of computer algorithms in learning and especially the adaptivity, or the ability to improve, of these algorithms. The main goal of machine learning is thus to create algorithms that utilize past experience, or example data, in solving problems.

Machine learning techniques can be used in computationally demanding tasks such as knowledge discovery in databases, language processing, function approximation, adaptive control and pattern recognition (Dietterich, 1997; Haykin, 2009). The methods and algorithms of machine learning include decision tree learning, artificial neural networks, genetic algorithms, and rule set learning, among many others.

3.4 ARTIFICIAL NEURAL NETWORKS (ANN)

Artificial neural networks form a large group among the methods of machine learning. Reed and Marks II (1999) proposed a definition which states that ANNs are nonlinear mapping systems whose arrangement is slackly based on the principles of nervous systems observed in humans and animals. The ANNs operate by connecting simple computational units together in suitable ways to create complex and interesting behaviors (Reed & Marks II, 1999). Haykin (2009) presented an alternate definition for a (artificial) neural network, modified from the one proposed by Aleksander and Morton (1990):

A neural network is a massively parallel distributed processor made up of simple processing units that has a natural propensity for storing experimental knowledge and making it available for use.

Meireles *et al.* (2003) suggested that:

Artificial neural networks (ANNs) implement algorithms that attempt to achieve a neurological related performance, such as learning from experience and *making generalizations from similar situations.*

A more technical definition is that neural networks are a collection of simple computational units linked to each other by a system of connections (Cheng & Titterington, 1994). The number of computational units can be large and the interlinking connections complex.

There are several ways of grouping the artificial neural networks. Sometimes they are categorized by the way they process data through the network (Meireles et al., 2003). Feed forward, or nonrecurrent, neural networks always process their outputs forward in the network: the neurons in an input layer propagate their outputs to a hidden layer, and the neurons in the hidden layer forward their outputs to an output layer. In contrast an ANN, in which the outputs can proceed both forward and backward, is called recurrent. This makes it possible to benefit from feedback information.

ANN methods can be classified also by the learning method. This kind of grouping separates the methods based on supervised and unsupervised (self-organizing) learning (Meireles et al., 2003). In *supervised learning*, the weights of a network are adjusted so that it produces a desired mapping of input to output activations (Riedmiller, 1994). The mapping is represented by a set of patterns, which includes examples of the desired function.

In supervised learning correct results, or desired outputs, are known, whereas in *unsupervised learning* training is completely data-driven. Unsupervised learning aims at auto-associating information from the network inputs with a fundamental reduction of data dimension, in the same way as extracting principal components in linear systems (Meireles et al., 2003). In unsupervised learning the neighboring units compete in their activities by means of mutual lateral information and adapt to specific patterns in the input signals (Kohonen, 1990). For this reason, this form of learning is also called *competitive learning*.

Some of the most popular artificial neural networks are multilayer perceptron (MLP) networks, radial basis function (RBF) networks, learning vector quantization (LVQ), support vector machines (SVM), self-organizing maps (SOM) and Hopfield networks. SOM and MLP are discussed in more detail in Chapter 6.

Self-organizing maps: *Self-organizing maps* (Kohonen, 2001) are a special type of unsupervised artificial neural networks based on *competitive learning*: the output neurons compete with each other in training to be activated (Haykin, 2009). In a self-organizing map the neurons are located at the nodes of a lattice including usually two dimensions. The self-organizing network is trained with a "winner-takes-all" rule, which is based on defining the best matching unit (BMU) for each input vector. As a result, the nonlinear relationships between the elements of high-dimensional data are transformed into simple geometric relationships of their image points on a low-dimensional display (Kohonen, 2001), which makes it an effective method for visualizing multivariate data.

Learning vector quantization: *Learning vector quantization* (LVQ; Kohonen, 1986) is a supervised learning algorithm, in which each class of input examples is represented by its own set of reference vectors. Although the method does not involve unsupervised learning, it is closely related to SOM. The purpose of LVQ is to describe borders between classes by using the nearest neighbor rule (Kohonen, 2001). New incoming data vectors are separated on the basis of the so called quantization regions, or *Voronoi sets*, defined by hyperplanes between the neighboring reference vectors. Thus, LVQ can be used in pattern recognition or classification, for example. The applications of LVQ have included image analysis, speech analysis and recognition, signal processing, and different industrial measurements, for instance (Kohonen, 2001).

Multilayer perceptron: Multilayer perceptrons (MLP) are widely-used feed-forward neural networks (Haykin, 2009; Kadlec et al., 2009; Meireles et al., 2003), which consist of processing elements and connections. The processing elements include an input layer, one or more hidden layers, and an output layer. In MLP networks, input signals are forwarded through successive layers of neurons on a layer-by-layer basis (Haykin, 2009). First the input layer distributes the inputs to the first hidden layer. Next, the neurons in the hidden layer summarize the inputs based on predefined weights, which either strengthen or weaken the impact of each input. The weights are defined by learning from examples (supervised learning). The inputs are next processed by a transfer function and the neurons transfer the result as a linear combination to the next layer, which is usually the output layer.

Radial basis function networks: *Radial basis function network* (RBFN) was introduced by Broomhead and Lowe (1988). The main difference between RBFN and MLP is that the links connecting the neurons of the input layer to the neurons of the hidden layer are direct connections with no weights (Haykin, 2009). Thus, the size of the hidden layer, which consists of nonlinear radial basis functions, equals the number of inputs. The second layer of the network is weighted and the output neurons are simple summing junctions (Meireles et al., 2003). Because of its structure the main limitation of RBFNs is the high demand of computational resources, especially when dealing with a large number of training samples.

Support vector machine: *Support vector machine* (SVM) is a category of feed-forward neural networks pioneered by Vapnik (1998; original description of the method by Boser et al., 1992). This supervised learning method can be used for classification and regression. The SVM is based on using hyperplanes as decision surfaces in a multidimensional space, in which the optimal separation is reached with the largest distance to the nearest neighboring data points used in training. A data point is considered an n-dimensional vector and the goal is to separate the data points with a linear classifier, an n-1 dimensional hyperplane. The major drawback of SVMs is the fast increase of computing and storage requirements with the number of training samples (Haykin, 2009), which limits their use in practical applications.

Recurrent neural networks: A *recurrent neural network* architecture is different from a feed-forward neural network in that it has at least one feedback loop (Haykin, 2009). In consequence, a neuron receives inputs both externally from network inputs and internally from feedback loops. Perhaps the most popular of recurrent networks is the *Hopfield network*, which generally consists of a single layer of neurons. The Hopfield network is totally interconnected, which means that all the neurons are connected to each other (Meireles et al., 2003). Thus it forms a multiple-loop feedback system (Haykin, 2009). The Hopfield network can be used as associative memory and in optimization problems, for instance (Meireles et al., 2003).

Probabilistic neural networks: *Probabilistic neural networks* (PNN) (Specht, 1988) are neural networks that utilize kernel-based approximation in forming an estimate of the probability density functions of classes. The method is used especially in problems of classification and pattern recognition. PNNs are almost similar in structure to MLPs (Meireles et al., 2003). The main differences between the methods are in the activation and in the connection patterns between neurons. An advantage over MLP is that PNN works entirely in parallel and the input signals proceed in one direction, without a need for feedback from the neurons to the inputs (Meireles et al., 2003).

3.5 CLUSTERING

Clustering, or cluster analysis, means partitioning data samples into subgroups according to their similarity. Cluster analysis is an important part of exploratory data analysis, which is typically used in exploring the internal structure of a complex data set, which cannot be described only through classical statistics (Äyrämö & Kärkkäinen, 2006).

According to a short definition presented by Jain *et al.* (1999), clustering is the unsupervised classification of patterns into groups. By patterns the authors mean data items, e.g. observations. A more detailed definition for clustering was presented by Haykin (2009):

> *Clustering is a form of unsupervised learning whereby a set of observations … is partitioned into natural groupings or clusters of patterns in such a way that the measure of similarity between any pair of observations assigned to each cluster minimizes a specified cost function.*

Similarity of data vectors consisting of several variables is of course difficult to define. The specification of proximity and how to measure it are the crucial problems in identifying clusters (Jain & Dubes, 1988), because the definition of proximity is problem dependent. Numerous clustering algorithms have therefore been developed.

3.6 OTHER INTELLIGENT METHODS

Decision tree learning: *Decision tree* (DT) *learning* is a method of data mining that can be used to partition data using the input variables and a class purity measure (Hand et al., 2001). In DT learning, a decision tree is used as a predictive model to estimate an output variable based on several input variables. The goal of DT learning is to form a tree-like structure in which most of the data points included in one node belong to the same class. Thus, different levels of the resulting tree structure represent hierarchical information on the clustering behavior of the data. The most famous DT algorithms include Classification and Regression Decision Trees (CART; Breiman et al., 1984), Iterative Dichotomiser 3 (ID3; Quinlan, 1986) and C4.5 and C5.0, which are extensions of the ID3.

Fuzzy sets and fuzzy logic: *Fuzzy set theory* (FST) is an exact mathematical framework in which vague conceptual phenomena can be examined rigorously (Tripathy, 2009). According to Zadeh (1965) who first introduced the fuzzy sets, a fuzzy set is *a class of objects with a continuum of grades of membership*. The fundament of FST is to enable graded membership of data elements instead of two-valued (true/false) membership logic (Tripathy, 2009). *Fuzzy logic* is a multi-valued logic derived from the fuzzy set theory, which makes it possible to apply approximate capabilities of human reasoning to knowledge-based systems., Fuzzy logic has emerged in a large variety of applications in recent years (Alavala, 2008).

Rough set theory: *Rough set theory* (RST) is a mathematical theory developed by Pawlak (1982) for classificatory analysis of data tables. It is based on creating rough sets, which are estimations of precise sets, approximated by a pair of sets, which are called a lower and upper approximation of the original set (Tripathy, 2009). Thus, RST can handle imperfect and vague datasets. Like the fuzzy set theory, the original purpose of RST is to understand and manipulate imperfect knowledge (Tripathy, 2009). The main difference between the theories is that in RST a membership function is not needed like in fuzzy sets, and the method can therefore avoid pre-assumptions. RST provides a method for data mining that can be used for finding relevant features from data or decision rules for classification and for reducing the size of data, just to name a few examples.

Evolutionary computation: Evolutionary computation techniques adapt the evolutionary principles of the nature into algorithms that can be used to solve problems. *Genetic algorithms* (GA; Holland, 1975), the most famous form of evolutionary computation, are search algorithms loosely based on the mechanics of natural selection and genetics. GAs start with a set of random solutions called a population, which is a difference to conventional search strategies. The population consists of individuals, each of which represent a solution to the problem. The solutions are evaluated during successive iterations using fitness measures and improve with every generation, which constantly leads to a new generation of evolved solutions. Ultimately, the algorithm converges to the best solution. (Gen & Cheng, 1997)

Hybrid intelligence: *Hybrid intelligence* is nowadays a widely-used approach of computational intelligence. Particularly combining neural networks and fuzzy systems in a united framework has become popular in recent years. The purpose of these so called *hybrid systems* is to benefit from the advantageous special properties of different methods and to reach a better performance in problem-solving than by using only the standard methods as such. Alavala (2008) grouped the hybrid systems into three different categories:

- *Sequential* hybrid systems: the output of one method becomes the input for another method, so the methods do not work in integrated combination.
- *Auxiliary* hybrid systems: one method uses the other as a subroutine to process data by it (master-slave system).
- *Embedded* hybrid systems: the methods are integrated into a complete fusion, so that both methods need the other to solve the problem.

3.7 INTELLIGENT METHODS IN MASS SOLDERING OF ELECTRONICS

As was discussed in Chapter 2, the processes of mass soldering of electronics have been conventionally analyzed using simple statistical methods. There are certain operations such as inspecting the soldering quality in which computationally intelligent methods have been used more widely, however. The earliest intelligent applications to mass soldering originate from the early 1990s (see Figure 8).

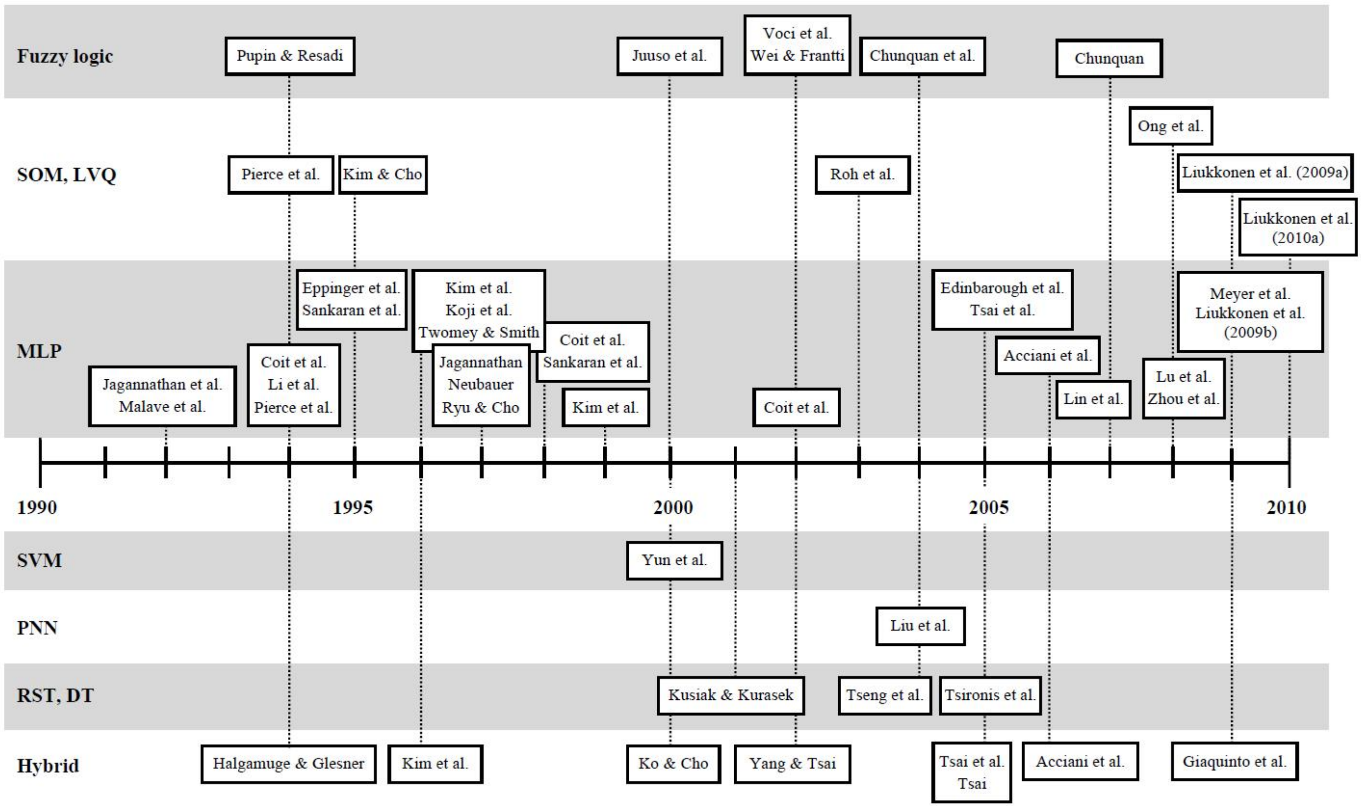

Figure 8: *Time chart showing the history of intelligent methods in the mass soldering of electronics. SOM = self-organizing map, LVQ = learning vector quantization, MLP = multilayer perceptron, SVM = support vector machine, PNN = probabilistic neural network, RST = rough sets, and DT = decision tree.*

The history of using intelligent methods in mass soldering of electronics is presented in Figure 8. Quality management and control have been the main application fields when using intelligent methods in the automated soldering. As can be seen, MLP has been used in most cases, covering approximately 50 % of the applications.

3.7.1 Applications to quality management

Quality management has been the main application field of intelligent methods with regard to mass soldering of electronics. One of the earliest applications was presented by Li *et al.* (1994) who studied the effects of design factors on PCB assembly yield using both regression and neural network models trained with back-propagation. The authors used analysis of variance to reduce the number of parameters in the regression models. The ANN models (average R^2 = 89.5) performed better in their experiments than the regression models (average R^2 = 81.3). Juuso *et al.* (2000) developed a tool for quality forecasting that can be used to model product quality on the basis of certain product-specific characteristics. The software relies mainly on simple statistics, but it includes additional features for quality forecasting using fuzzy logic and linguistic equations.

Kusiak and Kurasek (2001) used an approach based on the rough set theory to data mining of printed circuit board defects. They managed to identify the causes of quality faults in a SMT assembly line with a rule-based method for extracting knowledge. In addition, the method was able to recognize PCBs with no faults accurately. Nonetheless, the classification accuracy for faulty objects was not so good.

Yang and Tsai (2002) implemented a neurofuzzy system for defect prediction and control of a surface mount assembly line. The system was able to identify all defects, but produced, however, a few false alarms. The authors reported that the downtime inflicted by defects was reduced remarkably by 47% after implementing the system. A little later, Tseng *et al.* (2004) used an extended approach based on the rough set theory to a problem of quality control in PCB assembly. They managed to predict 97% of the solder balls, so the method overcame clearly the statistical methods used in their comparative studies.

Moreover, Tsironis *et al.* (2005) studied the use of machine learning in quality management of a process for manufacturing ISDN modems. The purpose was to demonstrate the applicability of two machine learning methods, i.e. decision tree induction and association rules mining, by extracting rules for quality management. The extracted rules were then used in the modeling of cause-effect relationships associated with faulty products. The authors found that both approaches exhibited a good accuracy of results (average error ca. 9%). Rule mining outperformed the decision trees in the correctness of learned rules, however.

More recently, Lin *et al.* (2007) created a predictive model for the sheer force of reflow-soldered joints of ball grid array packages. Their system was based on integrated use of a multilayer feed-forward neural network for modeling and sequential quadratic programming for optimization. The ANN model was able to achieve a correlation of 0.933 in terms of R^2 between the real and predicted shear force. Zhou *et al.* (2008) combined a back-propagating ANN with particle swarm optimization to optimize thermo-mechanical fatigue reliability of solder joints in plastic ball grid arrays. They stated that their procedure for intelligent optimization was a great improvement to the traditional (Taguchi) optimization method.

Furthermore, Meyer *et al.* (2009) used neural networks with back-propagation to predict void generation in a PCB assembly process, achieving over 90% accuracies for different component types. They also studied the effects of different parameters on model accuracy and concluded that the influencing parameters change from model to model.

3.7.2 Inspection of solder joints (quality control)

Detection of faulty solder joints is an important part of electronics production, because flaws can cause short circuits and missing contacts impeding the correct functioning of products. For this reason, a variety of applications have been developed to recognize faulty solder joints. Cho and Park (2002), for example, presented a variety of optical inspection systems based on neural networks for this purpose. As a matter of fact, inspection of solder joints can be considered a part of quality management, but it is separated here due to the diversity of applications and because it forms a clear and separate entity within quality management.

Intelligent methods have been used relatively widely in the inspection of solder joints. Particularly multilayer perceptrons with back-propagation have been a popular method for inspecting solder joints (see Jagannathan *et al.*, 1992; Pierce *et al.*, 1994; Eppinger *et al.*, 1995; Sankaran *et al.*, 1995 & 1998; Kim *et al.*, 1996a; Koji *et al.*, 1996; Ryu & Cho, 1997; Neubauer, 1997; Jagannathan, 1997; Kim *et al*, 1999; Edinbarough et al., 2005; Acciani *et al.*, 2006a; Lu *et al.*, 2008). Nevertheless, there are also applications based on self-organizing maps and learning vector quantization (Pierce *et al.*, 1994; Kim & Cho, 1995; Roh *et al.*, 2003; Ong *et al.*, 2008), support vector machines (Yun *et al.*, 2000), probabilistic neural networks (Liu *et al.*, 2004), fuzzy logic (Pupin & Resadi, 1994; Voci *et al.*, 2002; Wei & Frantti, 2002; Chunquan *et al.*, 2004; Chunquan, 2007) and hybrid intelligence (Halgamuge & Glesner, 1994; Kim *et al.*, 1996b; Ko & Cho, 2000; Acciani *et al.*, 2006b; Giaquinto *et al.*, 2009). These applications are discussed more deeply in the following paragraphs.

MLP and back-propagation: The earliest intelligent systems for solder joint evaluation originate from the early 1990s. Jagannathan *et al.* (1992) proposed a system using intelligent machine vision to inspect wave soldered joints. They reached a ratio of 98.75% successful classifications in classifying the joints into defective and non-defective ones. Pierce *et al.* (1994) developed an automated inspection system that identified solder joints from an X-ray image and classified them as good or corrupted using both back-propagation and a Kohonen network. The back-propagation routine was reported to classify 86% of the solder joints correctly, whereas the Kohonen network managed to classify at best 77.5% of the joints right. Eppinger *et al.* (1995) used a data set from an automated solder joint inspection system to demonstrate the benefits of neural networks over statistical methods in both feature selection and classification. They discovered that the applied multilayer neural network with back-propagation produced a significant improvement in performance when compared to traditional classification methods. The improvement was obtained at the expense of significant computational resources, however (Eppinger *et al.*, 1995). After all, the errors of classification were also relatively high (9% at their lowest) on the whole.

Sankaran *et al.* (1995 & 1998) reported a performance as high as 92% in identifying solder joint defects. They used visible light images as source data and analyzed the data with a back-propagating neural network, using additionally several other methods for data compression and feature extraction. Kim *et al.* (1996a) used a neural network based on back-propagation to classify solder joints of commercially manufactured PCB assemblies, attaining high rates (97–100%) for right classifications. Koji *et al.* (1996) used a neural network with one hidden layer to inspect soldering quality utilizing optical images taken from soldered leads of semiconductor packages. They managed to reach a 100% detection rate for defective samples and 95.7% detection rate for normal solder joints.

Ryu and Cho (1997) used a neural network to classify accurately (ca. 98%) two kinds of solder joints with respect to soldering quality, using 10 data features from automatic visual inspection. Neubauer (1997) used a three-layered MLP successfully in detecting voids in solder joints imaged by automated X-ray inspection. Moreover, Jagannathan (1997) used back-propagation in a two-stage classifier for wave soldered joints, which classified the samples into three different categories (good, excess, no solder) accurately (100%).

Kim *et al.* (1999) reported on classification of four types of solder joints (good, none, insufficient, excess solder) using a multilayer perceptron network, which produced 98–100% accuracies within the classes. In addition they used a Bayesian classifier in uncertain cases. Edinbarough *et al.* (2005) developed a visual inspection system that utilizes a single layer neural network with multiple neurons in identifying common defects in electronics manufacturing and managed to reach a 100% performance using the system. Acciani *et al.* (2006a) experimented with both multilayer perceptrons and learning vector quantization in classifying solder joints into five different categories from poor to excess solder. By combining the geometric and wavelet features extracted from the images of joints they were able to achieve the performance of 98.8% for the MLP and 97.1% for the LVQ method. Lu *et al.* (2008) improved an automated optical inspection system by developing an intelligent application based on BP neural networks to the classification of solder joints. They reported on achieving a high accuracy for classification.

SOM and LVQ: Pierce *et al.* (1994) used X-ray images and analyzed them with both back-propagation and Kohonen network to inspect the quality of through-hole solder joints. The back-propagation routine was able to classify 86% of the solder joints correctly, whereas the performance of the Kohonen network was significantly lower (77.5%). Kim and Cho (1995) used LVQ to classify solder joints into five classes, ranging from insufficient to excess solder. They compared the performance of the method to that of methods based on back-propagation and Kohonen self-organizing networks and managed to attain a fairly good accuracy for both the BP and LVQ methods (94% and 93%, respectively). LVQ was the method chosen by the authors, however, because it was faster and simpler to implement.

Furthermore, Roh *et al.* (2003) used a self-organizing map in enhancing the image quality of a 3D X-ray imaging system used in the inspection of solder joints. Ong *et al.* (2008) introduced a technique that utilized a camera with an orthogonal view in combination with one having an oblique view for inspecting the quality of solder joints. They succeeded to categorize the joints into three different quality classes with no false classifications using learning vector quantization.

SVM: Yun *et al.* (2000) compared k-means, back-propagation and support vector machines in the classification of solder joints in surface mounted devices, which were inspected by a circular illumination technique, with respect to the amount of solder present in the joints. The support vector machine performed slightly better than the other two methods, reaching a rate of 96–100% correct classifications within the classes.

PNN: Liu *et al.* (2004) developed a system for inspecting flip-chip solder joints, which was based on analyzing ultrasound waveforms and which utilized probabilistic neural networks in the automated pattern recognition of soldering defects. Their experimentations with the system produced a rate of 95% correct classifications for 20 samples.

Fuzzy logic: Pupin and Resadi (1994) reported on a machine that could be used in inspecting the quality of solder joints and presented new approaches to analyze the images of joints and a modern approach based on fuzzy theory to judge soldering defects. Voci *et al.* (2002) developed a system based on fuzzy rules for detecting short circuits from X-ray images of printed circuit boards. They managed to enhance the images by fuzzy filtering so that the detection of short circuits was facilitated. Wei and Frantti (2002) presented online embedded software based on fuzzy logic for inspecting soldering defects in products for signal transmission. Their system was able to improve the results of X-ray inspection by reducing false alarms by 44%. Furthermore, Chunquan *et al.* (2004) and Chunquan (2007) proposed an application based on fuzzy rules to the diagnosis of surface mounted solder joints. The research was based on the theory on the 3D geometrical shape of solder joints.

Hybrid intelligence: Intelligent hybrid systems have become popular methods for inspecting solder joints. These methods utilize at least two methodologies in solving problems. Halgamuge and Glesner (1994) used a multilayer perceptron architecture known as FuNe I to generate fuzzy systems for different real world applications. They used 3D surface information and 2D gray-level information from solder joint images, for instance, in distinguishing good solder joints from the bad ones automatically, reporting on a 99% accuracy of classification.

Kim *et al.* (1996b) used a back-propagation algorithm in classifying solder joints into four different types (good, none, insufficient, excess solder) and used an additional Bayesian classifier in unclear cases. They reported on a performance of 98–100% correct classifications within classes. Ko and Cho (2000) presented a classification method consisting of two modules, of which one was based on an unsupervised LVQ classifier and the other on fuzzy set theory. The purpose of the latter module was to correct possible misclassifications produced by the LVQ module. The joints were classified into five categories, and the method reached an accuracy of 96% for test samples.

More recently, Acciani *et al.* (2006b) introduced a diagnostic system based on using multilayer perceptrons and learning vector quantization in combination, which was used to analyze images of solder joints in integrated circuits. They classified the joints into five classes (from insufficient to excess amount of solder) and managed to reach a recognition rate of as high as 99.5%. Giaquinto *et al.* (2009) presented a neurofuzzy method for analyzing soldering quality by evaluating each soldering on a five-degree scale, ranging from poor to excess solder. Their methodology comprised three supervised MLP networks and two modules based on fuzzy rules and attained an overall recognition rate of 97.8%.

3.7.3 Other applications

There are also applications in which automated soldering is modeled for some other purpose than improving the quality of its outcome. Modeling of the thermal profile of soldering is one existing application field. Malave *et al.* (1992), for example, used a neural network to model the pre-heat temperature and line speed of a wave solder machine using PCB design characteristics (15 variables) as model inputs. An MLP with two hidden layers and back-propagation training was selected for the modeling method. The created network did not produce successful results, however. The authors concluded that a better selection of design parameters and a larger amount of data would possibly improve the performance of their model.

Coit *et al.* (1994, 1998 & 2002) used a back-propagating neural network to model the temperature profile of a wave soldering process with two preheating stages. The results presented by Coit *et al.* (1994) showed that neural networks can overcome statistical methods such as linear and polynomial regression in the analysis of wave soldering. The authors used both experimental and production data to predict the mean temperature, standard deviation and the rate of change at the wave. Nonetheless, their data set was quite small, comprising 100 samples. As a result, the precision for predicting mean temperature was relatively good, whereas the predictions for standard deviation and gradient were less precise.

The research was continued by the authors by predicting the quality in wave soldering (Coit *et al.*, 2002). A back-propagation algorithm and a network consisting of 13 input variables and two hidden layers were used to produce three binary outputs: excellent, good and fair solder quality. The results of the study were moderate, i.e. about 80% of classifications were correct, but it was beneficial that in erroneous cases the network predicted a higher defect level rather than a lower one.

Furthermore, Twomey and Smith (1996) extended the research of Coit *et al.* (1994) by presenting an ANN-based approach for controlling the wave soldering process. They used a neural network with back-propagation and divided the data differently to train and test sets in their experiments. The data included four parameters to be used as inputs and the mean temperature of the wave to be used as model output. The authors reported that a committee network, in which several networks are trained and the output is a combination of the outputs of all the networks, performed generally better than the other two networks tested, which were created using conventional train-and-test and re-substitution methods. Unfortunately the computational time required by the methods was not reported.

Tsai *et al.* (2005a) presented a two-layered back-propagating neural network for modeling the thermal profile of a reflow soldering process. The authors predicted seven variables with regard to thermal profile using eight inputs, which were parameters of the soldering machine. High correlations ($R > 0.99$) between training data and model outputs were achieved. Moreover, Tsai *et al.* (2005b) presented a neurofuzzy system for controlling the thermal profile of reflow convection ovens, which utilized training vectors consisting of eight input elements in extracting rules which were then used to produce six responses with regard to thermal profile. In addition, Tsai (2005c) introduced an industrial application of intelligent knowledge-based system for controlling reflow soldering, which is based on knowledge extraction through neurofuzzy training and which can be used to determine process parameters and assess the outputs of a surface mount assembly line.

4. Aims of the study

The purpose of this work as a whole is to provide understanding on the application of intelligent data-based modeling methods to production of electronics. This is performed by surveying the current state of research in this field and using intelligent methods in the quality analysis of a real process, i.e. wave soldering.

The ultimate target of the thesis is to create a procedure for analyzing the quality of electronic products using process data and intelligent methods. The purpose is that the system would benefit from the most useful characteristics of computational intelligence, which include an ability to solve real world problems, to learn from experience, to self-organize and to adapt in response to dynamically changing conditions. Methods that fill these requirements are used in the quality analysis of automated soldering in this thesis.

In summary, the aims of the work are:

1) To determine whether intelligent data-based methods could provide additional value to production of electronics by using them in the quality analysis of a real manufacturing process.
2) To create a procedure for analyzing quality in electronics manufacturing including all the necessary stages from pre-processing of data to visualization and analysis of results.
3) To increase the awareness of electronics manufacturers of intelligent data-based methods and their potential for being used in process improvement.

The purpose of the first aim is to answer the question: Can intelligent methods be utilized in analyzing quality in the electronics industry? Consequently, if the answer for the first question is yes, the second aim is for answering the question: How should it be done?

The intelligent methods were selected for quality analysis by evaluating their usability, swiftness and robustness from the manufacturer's point of view, because good usability is the key element of any industrial real-world applications such as decision support systems, which concerns especially the rapidly changing environment of electronics production. Therefore the most sophisticated intelligent methods are excluded from the scope of the thesis.

5. Wave soldering process and data

5.1 WAVE SOLDERING PROCESS

Wave soldering is a classic technique for automated soldering of electronic components on PCBs. The technology originates from the mid 1950s when Fry's Metals of Mitcham introduced the first wave soldering machine (Noble, 1989). Typically PCBs with through hole components (THC) are first fluxed in the process by a foam fluxer and then conveyed over a laminar solder wave to solder the THCs to the leads of the board (Lee, 2002).

Wave soldering is originally designed for soldering through-hole components, but also surface mount components can be attached with wave soldering if they are first glued onto the bottom side of the board (Manko, 2001). This requires an additional working stage, i.e. dispensing of an adhesive, however. Solder waves can be used also in other applications such as tape tinning, attaching leads to hybrids and soldering armatures for automobiles (Manko, 2001).

The main stages of wave soldering, as illustrated in Figure 9, are fluxing, preheating and soldering. *Fluxing* is accomplished using foam, spray or wave fluxing. In foam fluxing, air is forced through an aerator into the flux, which generates flux foam that is applied to the surface of a board. Spray fluxing, which was used in the case study, involves spraying of flux through small nozzles and coating the underside of a PCB passing through the fluxing station. In wave fluxing, a liquid wave applicator creates a wave of flux in a similar manner that the solder is applied. (Judd & Brindley, 1999).

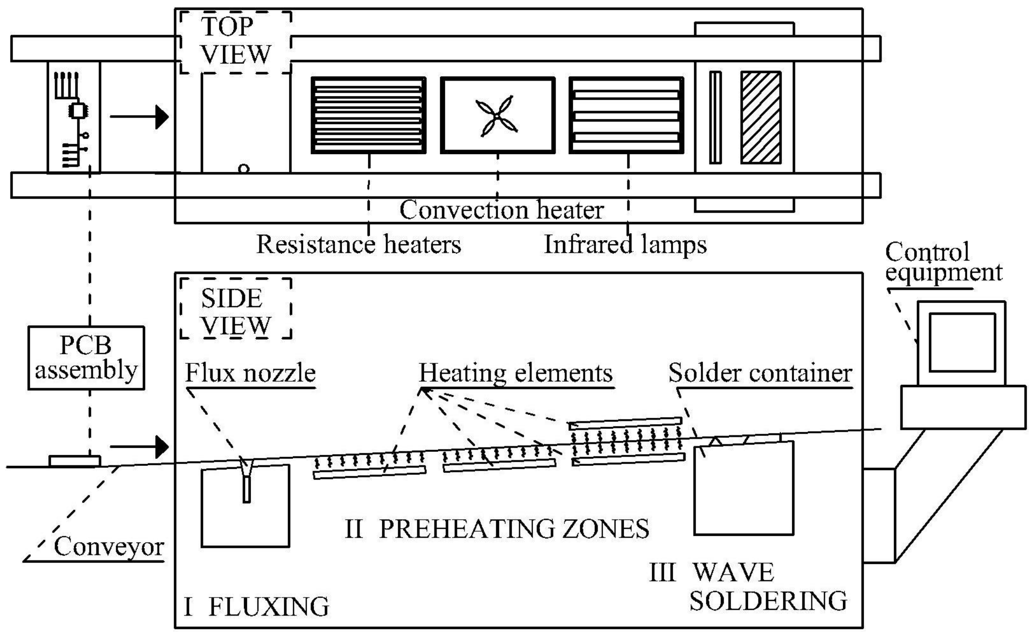

Figure 9: *The stages of the wave soldering process and the main parts of the wave soldering machine (Liukkonen et al., 2010b).*

The purpose of preheating is to prepare the fluxed PCB assembly for soldering. The main function of preheating is to speed up the soldering process by enabling the use of shorter contact times in soldering phase. The additional functions of preheating are drying and activating the flux and reducing thermal shock, which can cause damage to components or warpage of the board. (Judd & Brindley, 1999).

The assemblies are next simply moved over a wave of molten solder in the soldering phase, in which the solder raises the surface temperature of the board sufficiently, completes the activation of flux and causes the component leads to be wetted by the solder (Judd & Brindley, 1999). The final connection is formed as the solder solidifies by cooling.

Wave soldering has long been the prevailing method for automated soldering in the era of through-hole technology (Lee, 2002). There has been a constant evolution in the electronics industry towards higher functional density, further miniaturization and higher yield, however (Lee, 2002). This is especially the case in the manufacturing of consumer electronics. For these reasons, surface mount technology has prevailed over through-hole technology in many products, which has reduced the popularity of wave soldering.

There are applications in which wave soldering remains the prevailing soldering method, however. The method is still an ideal technique for soldering boards with conventional leaded components and for surface mount boards with larger components. This is usual in the manufacturing of industrial electronics, e.g. large power devices. Wave soldering has also the significant benefit that it can be used easily in soldering assemblies containing mixed technology, i.e. PCBs that contain both through-hole and surface mounted components (Khandpur, 2005). Furthermore, wave soldering is also used in special applications such as continuous-wire or tape tinning, soldering armatures for automobiles, attaching leads to hybrids and pre-tinning components and flat packs (Manko, 2001).

5.2 WAVE SOLDERING DEFECTS

Wave soldering defects can result from any of the stages of the process (Sterritt, 1991). The most serious faults occurring in wave soldering are those that cause malfunctions in the final product or even pose a threat to its user. These kinds of defects are unsoldered components and solder bridges that cause short circuits, for example. Also solder balls detaching afterwards can prove problematic, because they can cause short circuits while moving inside the final product. In addition, minor faults such as flux residues and small solder spatters may appear. These flaws are sometimes merely cosmetic and do not necessarily need any reworking.

Unsoldered joints: *Unsoldered joints* of surface mount components can occur in wave soldering due to a too low chip wave or the so called shadowing effect of another component, which hinders streaming of solder to the joint area. The layout design of the board is the most important means to ensure formation of proper SMC joints.

Bridging: Forming of solder bridges between neighboring solder joints is called *bridging* (Lee, 2002). Bridging occurs due to locally excessive solder and may extend over more than two joints (Lee, 2002). Solder bridges are critical or at least major defects, because they cause malfunction of products almost beyond exception. A short circuit is a typical consequence of bridging. In wave soldering, correct flux dosage, nitrogen atmosphere and the layout design of the board are the typical means to reduce bridging.

Poor wetting: Successful soldering depends essentially on wetting, in which the contact between the solder and the metal in component joints is formed (Manko, 2001). In wetting, the liquid solder leaves a consistent and permanent coating onto the surface of soldered metal.

Poor through-hole wetting can be problematic in soldering through-hole components. The defect has been reported to be a fairly typical problem in lead-free wave soldering (Havia et al., 2005). The problem can be solved by increasing the contact time or temperature of solder, which may cause other problems, however (Havia et al., 2005). Poor wetting is typically considered a major defect.

Balled solders: A *balled solder* is a large, individual ball of solidified solder, which is formed in the tip of a leg of a through-hole component, for example, or to one side of a surface mount component. Balled solders may inflict malfunction in electronic products, so it has to be considered a major defect.

Solder flags: A *solder flag* is a spiked formation of excess solder located in the soldered connections of a PCB. Usually solder flags are cosmetic flaws, but sometimes they increase the risk of short circuits or bridging. Possible reasons for flagging include insufficient preheating, uneven spreading of flux and contaminated solder.

Solder balls: *Solder balls* and *spatters* are small, ball-shaped solder residues on the surface of a PCB. Possible reasons for the formation of solder balls include insufficient preheating, wrong solder temperature and an unevenly distributed wave of solder. Generally solder balls are harmless minor defects, but sometimes they can cause short circuits if they come loose and end up in critical locations.

Flux residue: *Flux residue* is leftover flux on the surface of a soldered PCB. Usually flux residue is merely a cosmetic flaw, but when a flux with a high solid content is used, it can cause also functional problems.

5.2.1 Prioritization of defect types

Soldering defects are often prioritized with respect to their seriousness. Two aspects are important when increasing the total quality of an electronic product:

1) Removal of critical defects
2) Minimization of quality cost

In the first point it is assumed that there are defects that cause malfunction in products. These fatal flaws either have to be repaired or the product is useless and will be discarded.

In the second place comes the minimization of a cost factor that consists of a monetary loss due to occurred defects. This aspect involves the deliberation on the profitableness of fixing defects. In other words, less weight is given to cosmetic flaws and more weight to critical or major defects that affect the correct functioning of the product. Thus, total cost of bad quality can be assessed by calculating the sum of weighted defect-specific cost estimates, as presented by Liukkonen *et al.* (2010a).

5.3 CHALLENGES IN MODERN WAVE SOLDERING

Despite the fact that the process is over 50 years old, wave soldering is under constant development. Especially certain legislative acts and the general trend of reducing the size of electronic products have created needs for further development in recent years.

Legislation: Directive 2002/95/EC (Restriction of the use of Hazardous Substances in Electrical and Electronic Equipment, ROHS) of the European council, which came into force in 2006, restricted the use of hazardous substances such as lead in the electronics industry. Adoption of lead-free solders to replace the conventional tin-lead solders has brought completely new challenges to the implementation of wave soldering in the 21st century (Barbini & Wang, 2005; Havia et al., 2005; Mendez et al., 2008; Morris & Szymanovski, 2009). Process temperatures must be kept higher, for example, because the melting point of lead-free solder is higher than the melting point of tin-lead solder (Barbini & Wang, 2005). Longer contact times and higher soldering temperatures affect also the behavior of flux (Barbini & Wang, 2005).

Furthermore, other environmental regulations such as Directives 1999/13/EC and 2004/42/EC of the European Council, which aim at the reduction of volatile organic compounds (VOC), have also influenced wave soldering in the 21st century. Therefore the trend has been to replace alcohol-based fluxes with the ones containing a low amount of VOC. Changes in the composition of flux have led to modifications in the process, since VOC-free fluxes have effects on fluxing technology as well as preheating technologies and temperatures (Barbini & Wang, 2005).

Size reduction: Decreasing the component pitch in PCBs has also been a challenge for wave soldering because the wetting behavior of lead-free alloys is different when compared to that of tin-lead alloys (Barbini & Wang, 2005). Because of higher temperatures the process is near to the breaking point of materials designed primarily for tin-lead processes, which means that the so called process window is narrower than before (Havia et al., 2005).

Analyzing and optimizing the wave soldering process have remained important because of these new challenges, although the process is old and otherwise well-known. When added to the urge to improve overall process efficiency in the electronics industry, the pressure to modify the process due to environmental regulations and the general trend for reducing the size of electronic products highlights the importance of finding new methods for process improvement.

5.4 WAVE SOLDERING DATA

The data from a lead-free wave soldering process include 1 073 rows with 40 variables in columns. The variables are presented in Table 1.

***Table 1**: The variables of the wave soldering data (modified from Liukkonen et al., 2009a).*

VARIABLE	Data inputs
PCB PROPERTIES	
PCB surface finish: 5 materials	5
PCB material: FR-4 or CEM	1
Number of layers	1
FLUXING	
Liquid base of flux: water or low VOC	2
Acid number of flux [mg KOH/g]	1
Solid content of flux [%]	1
Flux: 10 fluxes	10
Flux pump frequency [Hz]	1
COMMON FEATURES OF PROCESS	
Nitrogen on/off	1
Track speed [cm/min]	1
PRE-HEATING	
Zone 1 Resistance heater temperature [°C]	1
Zone 2 Convection heater temperature [°C]	1
Zone 3 IR Lamps [%]	1
Top Zone 3 IR Lamps [%]	1
SOLDERING	
Chipwave pump [rpm]	1
Chipwave on/off	1
Solderwave pump [rpm]	1
Solder temperature[°C]	1
Back plate [mm]	1
Cooling on/off	1
DEFECTS PER BOARD	
Poor through-hole wetting	1
Solder bridges	1
Balled solders	1
Solder balls	1
Solder flags	1
Degree of flux residue amount [0, 2]	1

6. Intelligent quality analysis of wave soldering

6.1 COMPUTATIONAL METHODS

The general aims of the study have served as the basis when selecting the computational methods for the intelligent analysis of wave soldering. Therefore, generic methods which could help in answering the question of the usability of the intelligent methods in the electronics industry and, on the other hand, in answering the question of how it should be done, have been included to the analysis. As a rule of thumb, generic and robust but still nonlinear intelligent methods have been selected for the analysis. Another criterion for choosing the computational methods has been that the time required for computation remains within reasonable limits.

When it comes to selecting the algorithms and parameters used by each method, the general principle has been to use standard methods and not performing any large-scale optimization of the parameters. Nevertheless, if a clear improvement over the standard method has been reported in the literature, those algorithms have been used. This concerns especially the training algorithms of self-organizing maps, of which the batch version is faster than the basic SOM when Matlab is used (Kohonen, 1999; Vesanto et al., 1999b), and it is therefore more applicable to industrial processes involved with large data sets.

Self-organizing maps (SOM) provide a projection upon which different properties of the data can be shown, and therefore provides a versatile basis on which to build an analysis process (Vesanto, 2002). This makes it an ideal method to be used in the early stages of data analysis. The SOM is especially efficient when used in visualizing dependencies and interrelations between variables of high-dimensional data (Alhoniemi, 2002; Laine, 2003). For these reasons, SOM has been included to the group of methods used in the intelligent quality analysis.

Clustering of SOM is another important method which has been used in this thesis. There are several advantages favoring this. Vesanto and Alhoniemi (2000) and Vesanto (2002), for example, have observed that the use of SOM as the first abstraction level in clustering has significant benefits. First, the original data set is represented using a smaller set of reference vectors, which enables efficient use of clustering algorithms, which means that clustering with the SOM is considerably faster than clustering the original data directly. Second, the low-dimensional arrangement of neurons allows easier visual presentation and interpretation of clusters. K-means was selected for the clustering method, because it is widely-used, simple and relatively easy to implement. In addition, k-means is advantageous in applications involved with large data sets for which the construction of a dendrogram is computationally heavy.

In addition to SOM, which is an unsupervised method, also an intelligent method with supervised capabilities was needed for the quality analysis. Multilayer perceptrons (MLP) can be nowadays considered a standard neural network, because it is widely used in different applications including diverse industrial use (Kadlec et al., 2009; Meireles et al., 2003). When combined with back-propagation, MLP offers a method that is computationally highly effective (Haykin, 2009). In addition, it offers a nonlinear modeling method for quality analysis and provides a platform on which adaptivity can be built. Therefore MLP has been an obvious selection for the intelligent quality analysis.

6.1.1 Self-organizing maps

Background: The self-organizing map (SOM) is a neural network algorithm developed by Teuvo Kohonen (Kohonen, 2001) in the early 1980s. A large variety of SOM-based applications have been developed since then. The conventional application areas of SOM have been machine vision and image analysis, signal processing, telecommunications, industrial measurements, exploratory data analysis, pattern recognition, speech analysis, industrial and medical diagnostics, robotics and instrumentation, and even process control (Kohonen, 2001), just to name a few.

Several literature reviews and surveys on the SOM and its applications have been presented. Kohonen *et al.* (1996), for example, concentrated on different engineering applications of SOM. Oja *et al.* (2002) gathered an extensive listing of books and research papers related to SOM. Moreover, by the date of publishing this thesis the Neural Network Research Centre from the Helsinki University of Technology had listed over 7 500 SOM-related references, of which the most recent, however, were from 2005.

In addition, several review articles specializing in narrower application fields exist. The early applications of SOM to robotics were discussed more deeply by Ritter *et al.* (1992). Tokutaka (1997) listed some SOM-based research applications in Japan. Seiffert & Jain (2002) presented advances of SOM in image analysis, speech processing and financial forecasting among others. More recently, Kalteh *et al.* (2008) reviewed the use of SOM in the analysis and modeling of water resources, and Barreto (2008) presented a review on time-series prediction with the self-organizing map.

The SOM has also served as the basis of intelligent applications to process improvement and monitoring in numerous industrial processes. Abonyi *et al.* (2003), Alhoniemi *et al.* (1999), Heikkinen *et al.* (2009a–b, 2010), Hiltunen *et al.* (2006), Jämsä-Jounela *et al.* (2003), Liukkonen *et al.* (2007, 2009a, 2009c–e, 2010a–b, 2010e–g) and Vermasvuori *et al.* (2002), for example, provided examples of these kinds of systems.

Basics of SOM: Training of SOM results in a topological arrangement of output neurons, each of which has a special property vector describing its *hits*, or input vectors. Each neuron of the SOM is defined on one hand by its location on the map grid and, on the other hand, by this property vector, which has the same dimensionality as input vectors. The property vector is called in this context as a *reference vector*, although it has been also called as a *codebook*, *prototype*, or *weight* vector in the literature. The reference vector can be defined as follows:

$$r_m = (r_{m1}, r_{m2}, \dots, r_{mP}), (m = 1, \dots, M), \tag{1}$$

where P is the number of variables, and M refers to the number of map neurons.

At the beginning of training the SOM is initialized. In *random initialization* the map is initialized using arbitrary values for reference vectors. In *linear initialization* the SOM is initialized linearly along the dimensions of the map, with respect to the greatest eigenvectors of training data. Linear initialization results in an ordered initial state for reference vectors instead of arbitrary values obtained by random initialization (Kohonen, 2001). Linear initialization is also faster and computationally less arduous than the classic random initialization (Kohonen, 2001), which makes it a good option for initializing maps for large data sets.

In the original incremental SOM input vectors are presented to the algorithm one at a time in a random order. The *best matching unit* (BMU) is the neuron with the smallest Euclidean distance to the input vector:

$$\beta(x_i, R) = arg\ \min_j \|x_i - r_j\|, \tag{2}$$

where β is the index of BMU, x_i signifies an input vector and R includes all reference vectors.

The BMU and a group of its neighboring neurons are trained according to the following update rule (Kohonen, 2001):

$$r_m(k+1) = r_m(k) + h_{\beta m}(k)[x_i - r_m(k)], \tag{3}$$

where k is the iteration round and m signifies the index of the neuron that is updated. A widely used neighborhood function is the Gaussian function (Kohonen, 2001):

$$h_{\beta m}(k) = \alpha(k)\exp\left(-\frac{\|v_\beta - v_m\|^2}{2\sigma^2(k)}\right), \tag{4}$$

where v_β and v_m symbolize the location vectors of two neurons, α refers to the factor of learning rate and σ is the parameter which defines the width of the kernel, i.e. the neighborhood of a single neuron.

It is noteworthy that practical applications of up to hundreds of neurons are not sensitive to factors α and σ, so usually a simpler function can be used to define the neighborhood (Kohonen, 2001):

$$h_{\beta m}(k) = h(\|v_\beta - v_m\|, k). \tag{5}$$

It is recommended that the training of SOM is performed in two phases (Kohonen, 2001). In the first phase the learning rate factor and neighborhood radius are decreased. Then the second phase, *fine tuning*, is started using small values for the learning rate and neighborhood radius. Generally the first ordering phase should include 1 000 steps and the fine tuning phase should have a number steps as large as 500 times the number of map units (Kohonen, 2001).

In summary, the training of SOM includes the following stages:

1) Initialize the map.
2) Find the BMU of the input vector using Euclidean distance (equation 2).
3) Move the reference vector of the BMU towards the input vector (equation 3).

4) Move the reference vectors of the neighboring neurons towards the input vector (equation 3).
5) Repeat steps 2–4 for all input vectors successively.
6) Repeat steps 2–5 using a smaller learning rate factor (fine tuning).
7) Find the final BMUs for input vectors (equation 2).

Training algorithms of SOM: The basic SOM algorithm which utilizes the update rule presented in Equation 3 is also called the *sequential training* algorithm in the literature (Kohonen, 1999; Vesanto, 1999b). Another option for training is the *batch training* algorithm (Kohonen, 1999), which is also iterative. In batch training the whole data set is brought to a map before any adjustments (Vesanto, 1999b). In training each data vector is mapped to the closest neuron according to the so called Voronoi regions of reference vectors. The update rule for reference vectors in batch training is (Kohonen, 1999):

$$r_m(k+1) = \frac{\sum_{i=1}^{N} h_{\beta m}(k) * x_i}{\sum_{i=1}^{N} h_{\beta m}(k)}, \quad (6)$$

where N is the number of original input vectors. As the formula suggests, the new reference vector is a weighted average of the original data samples assimilated to it.

The batch computation version of SOM is significantly faster than the basic SOM when Matlab is used (Kohonen, 1999; Vesanto et al., 1999b), which makes it more applicable to industrial processes involved with large data sets. On the other hand, expenditure of memory is a deficiency of the batch algorithm (Vesanto et al., 1999b).

Goodness of SOM: Many assumptions have to be made with respect to learning parameters when training SOM. For this reason it is important to test these parameters experimentally before their final selection. Determining the size of the map is the most common problem when using SOM. Usually different map sizes are therefore tested and the optimum size is chosen based on minimum errors. There are several measures to evaluate the goodness of a map.

Quantization error (e_q) is a widely used error measure of SOM. It can be presented as follows (Kohonen, 2001):

$$e_q = \frac{1}{N}\sum\|x_i - r_\beta\|, \tag{7}$$

where N refers to the number of original data vectors and r_β is the BMU of the data vector x_i. As can be seen, the quantization error is a measure of the average distance between data vectors and their BMUs, so it evaluates the overall fitting of SOM to the data. Thus, the smaller the value of e_q is, the closer the original data vectors are to their reference vectors. Nonetheless, it is important to note that the quantization error can be reduced simply by increasing the number of map units, because the data samples are then distributed more sparsely on the map.

Another important goodness measure of SOM is *topographic error* (e_t), which measures the continuity of mapping. This measure of error utilizes input vectors to define the continuity of mapping from the input space to map grid. There are various ways of calculating the topographic error, one of the mostly used of which is presented by (Kiviluoto, 1996):

$$e_t = \frac{1}{N}\sum_{i=1}^{N} u(x_i), \tag{8}$$

where $u(x_i)$ gets the value of 1 if the best and the second-best-matching units of an input vector are non-adjacent, and 0 otherwise. In other words, the value of e_t describes the proportion of those input vectors for which the first and second-best-matching units are not adjacent vectors.

The lower the topology error is, the better the SOM preserves its topology. It must be noted, however, that the topology error generally increases with the size of the map due to growing complexity of arranging the neurons, because the number of reference vectors also increases (Uriarte et al., 2006).

Many algorithms include a cost function for defining the optimal situation in training. Nonetheless, Erwin et al. (1992) have shown that the basic SOM algorithm is not the gradient of any cost function in a general case. If the data set is discrete and the neighborhood radius constant, *distortion measure* (e_d) can be considered a local cost function of a SOM (Kohonen, 1991; Vesanto, 2002). The distortion of a SOM is defined as:

$$e_d = \sum_{i=1}^{N} \sum_{j=1}^{M} h_{\beta_{ij}} \left\| x_i - r_j \right\|^2. \qquad (9)$$

By remembering the limitations mentioned above, the distortion measure can be used in selecting the best fitting SOM from the group of maps trained with the same data.

Visualization of SOM: One of the main advantages of SOM is the large variety of visualization methods that can be used. Perhaps the mostly used method for visualization is the 2-dimensional mapping of neurons and color coding, which can be used for visualizing features on the map. A component plane of a SOM is illustrated this way in Figure 10a. Each variable is presented in a separate component plane in this approach.

Another illustrative method is to use 3-dimensional visualization of component planes, as presented in Figure 10b. In this presentation the arrangement of neurons forms the first two dimensions while the third dimension represents the desired output feature, or vector component.

The component planes of SOM can also be represented in a 2-dimensional organization, as presented by Liukkonen *et al.* (2009a), for example. In this approach the values for neurons are obtained from their reference vectors, which offers an illustrative way to explore dependencies between two variables. A third variable can be additionally included in the presentation by using color coding.

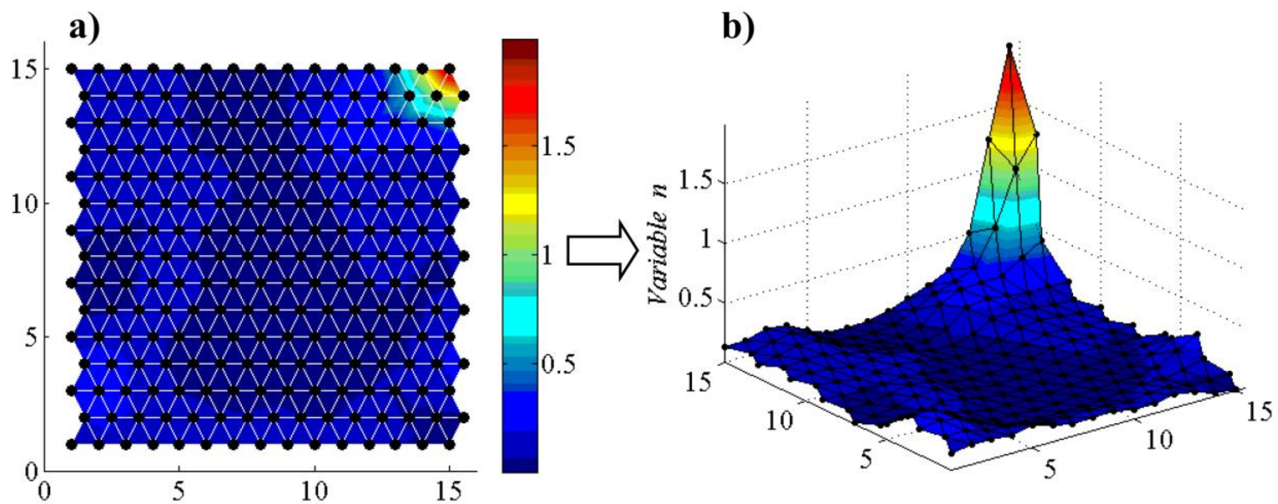

***Figure 10**: Visualization of the component planes of a 15×15 SOM (Liukkonen et al., 2010a). a) 2-dimensional representation in which the two axes indicate the coordinates of neurons and the color scale the values of a variable in each reference vector, and (b) 3-dimensional representation in which the vertical axis indicates the values of the variable in each reference vector.*

Alternatively reference vectors can be presented as bars, as presented by Liukkonen *et al.* (2009d, 2010a), for example. This is a useful way of studying differences between two neurons. If the SOM includes a neuron associated with a low number of soldering defects, for example, and one associated with a high number of those, the bar presentation can be used for identifying reasons for high defect levels. Alternatively the two reference vectors can be subtracted from each other to produce a vector which illustrates the main differences between them directly (see Liukkonen et al., 2009d, for example).

The U-matrix representation developed by Ultsch and Siemon (1989) illustrates the relative average distances between neighboring reference vectors by shades in a gray scale or by different colors in a color scale, so it can be used for indicating the clustering behavior of reference vectors. The U-matrix is computed by determining the average of distances between the reference vectors of neighboring neurons of the reference vector in target. The resulting value can be associated with each single neuron and used as a basis of color coding, for example.

6.1.2 K-means clustering

K-means (MacQueen, 1967), a partitional clustering algorithm based on the calculation of squared errors, is one of the most popular methods used for clustering, because it is fast, simple, relatively easy to implement and yet effective in performance and computationally efficient (Haykin, 2009; Äyrämö & Kärkkäinen, 2006). Partitional clustering is especially advantageous in applications involved with large data sets for which the construction of a dendrogram would be computationally demanding or even prohibitive (Jain et al., 1999). A drawback is that the method is sensitive to the selecting of initial partition (Jain *et al.*, 1999), and the result may thus converge to a local minimum. The stages of the algorithm are as follows:

1) Define k cluster centers and assign k random data vectors to them.

2) Address each data vector to the cluster with the most similar cluster center. For example Euclidean distance can be used as a measure of similarity.
3) Update the cluster centers to new memberships.
4) If the criterion for convergence is not met, return to step 2. A minimal decrease in the squared error, for example, can be used to stop iteration.

The squared error of a clustering with k clusters can be expressed as:

$$e^2 = \sum_{j=1}^{k} \sum_{i=1}^{n_j} \left\| x_i^{(j)} - c_j \right\|^2, \quad (10)$$

where $x_i^{(j)}$ is the ith input vector belonging to the cluster j, n_j equals the number of data rows assigned to cluster j, and c_j denotes the center of cluster j.

Goodness of clustering: The optimal number of clusters can be determined using Davies-Bouldin index (Davies & Bouldin, 1979):

$$D_{DB} = \frac{1}{k} \sum_{i=1}^{k} \max_{j, j \neq i} \frac{s_i + s_j}{D_{ij}}, \quad (11)$$

where s_i is the average distance of the input vectors associated with cluster i to the center of that cluster, s_j is that of the input vectors associated with cluster j, and D_{ij} denotes the distance between clusters i and j. As the equation (11) shows, small values of DB-index correspond to clusters the centers of which are far from each other, so the optimal number of clusters is indicated by the minimum value of the index. This eliminates the need for knowing the clusters *a priori*.

6.1.3 Linear regression

The regression equation for the value y of a dependent variable (model output) can be written as follows:

$$y = f(p, w), \quad (12)$$

where p is the independent variable and w is a set of unknown parameters to be fitted.

The values for parameters w can be solved by minimizing the distance between measured and predicted values of the dependent variable.

Simple linear regression can be expressed as follows:

$$y_i = w_0 + w_1 p_i + \varepsilon_i,\ i = 1, \dots, N \quad (13)$$

where N is the total number of data points and ε refers to an error term which includes the uncontrolled factors and experimental errors of the model.

Multiple linear regression: In *multiple linear regression* involved with at least two explanatory variables the equation turns to:

$$y_i = w_0 + w_1 p_{i1} + \dots + w_P p_{iP} + \varepsilon_i,\ \ i = 1, \dots, N \quad (14)$$

where P is the number of independent variables included to the model.

The function used mostly in estimating parameters in linear regression is the method of least squares. The function works by minimizing the sum of squared residuals (Θ):

$$\Theta = \sum_{i=1}^{N} (\rho_i)^2, \quad (15)$$

where ρ_i is the residual, i.e. the difference between the value of a dependent variable (y_i) and the predicted value from the estimated model ($\hat{y}_i$):

$$\rho_i = y_i - \hat{y}_i. \quad (16)$$

6.1.4 Multilayer perceptron

A *multilayer perceptron* (MLP) network consists of processing elements (neurons) and weighted connections (Haykin, 2009), as presented in a simple form in Figure 11. The processing elements of MLP include an input layer, one or more hidden layers and an output layer. The network structure is organized so that each neuron output is connected to all neurons of the subsequent layer, whereas there are no connections between neurons in the same layer (Meireles *et al.*, 2003).

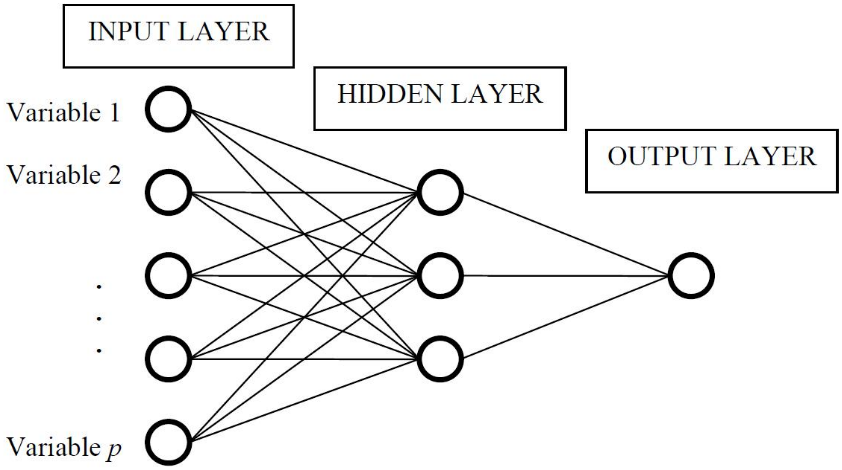

Figure 11*: The structure of a MLP network. The input layer has as many neurons as there are input variables.*

The purpose of the input layer is to distribute inputs to the first hidden layer. Computing is performed in neurons of the hidden layer, which summarize the inputs based on predefined weights, process them by a transfer function and transfer the result to the next layer, which is usually the output layer, as a linear combination. Finally, the network outputs are calculated by a transfer function, which can be hyperbolic or sigmoid, for instance (Haykin, 2009).

Neuron model: A nonlinear model of a single neuron originally presented by Haykin (2009) is illustrated in Figure 12, in which the three basic elements of a neuron can be seen.

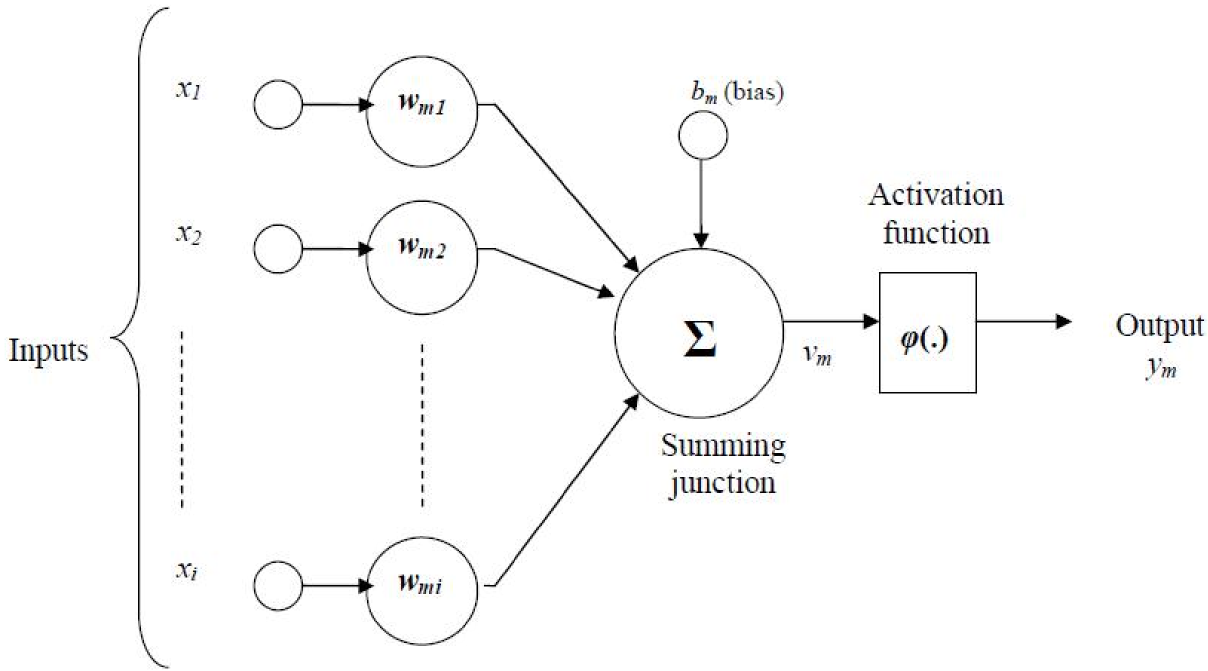

Figure 12: *Nonlinear model of a single neuron m (modified from Haykin, 2009). w denotes the synaptic weights in connections and* v_m *is the activation potential.*

First, the connections, or *synapses*, characterized by a weight by which the proceeding signals are multiplied, are used for transfering the inputs through the network. Second, the function of a summing junction, or an *adder*, is to sum the weighted signals to form a linear combination. Third, an *activation function* is needed to limit the amplitude of the neuron output. In addition, the neural model includes a *bias* term, which is used to tune the input of the activation function into an effective interval.

The output signal of neuron *m* can be presented as follows (Haykin, 2009):

$$y_m = \varphi(u_m + b_m), \tag{17}$$

where ϕ denotes the activation function and b_m is the bias. Symbol u_m signifies the output of a linear combiner, which can be expressed as follows:

$$u_m = \sum_{j=1}^{N} w_{mj} x_j, \tag{18}$$

where x_1 to x_N are the input signals, w_{m1} to w_{mN} are the respective synaptic weights and *N* is the total number of inputs. The term $u_m + b_m$ in Equation 17 is the activation potential (v_m) of neuron *m*. In other words the output signal y_m can also be presented as follows:

$$y_m = \varphi(v_m). \tag{19}$$

Back-propagation training algorithm: The supervised MLP networks have to be trained to a problem. Shortly, the purpose of training is to minimize the error value between actual and expected outputs for all input patterns. The most commonly used supervised training algorithm for MLP networks is the *back-propagation* (Werbos, 1974 & 1994) algorithm.

Basic back-propagation works by the following manner, as presented by Haykin (2009). The implementation of the algorithm starts by initializing a network by picking up the synaptic weights and thresholds from a uniform distribution. Next, an epoch of training samples is input to the network. The activation potentials, or *induced local fields*, and output signals are then computed by proceeding forward layer by layer through the network. The induced local field $v_{m,\lambda}(k)$ for neuron m in layer λ at current iteration k can be presented as follows:

$$v_{m,\lambda}(k) = \sum_l w_{m,\lambda}(k) y_{l,\lambda-1}(k), \quad k = 1, \dots, K \tag{20}$$

where $w_{m,\lambda}(k)$ is the synaptic weight of neuron m in layer λ that is fed from neuron l in the previous layer $\lambda–1$, $y_{l,\lambda-1}(k)$ is the output signal of neuron l in layer $\lambda–1$ at iteration k, and K is the total number of iteration rounds.

If a sigmoid activation function is used, the output signal of neuron m in layer λ is:

$$y_{m,\lambda} = \varphi_m(v_m(k)), \tag{21}$$

where ϕ_m is the activation function. Next, an error signal can be computed:

$$e_m(k) = d_m(k) - y_{m,\Lambda}(k), \tag{22}$$

where $d_m(k)$ is the mth element of the desired response vector and $y_{m,\Lambda}(k)$ denotes the output signal in the output layer Λ. Then local gradients (δ) for neuron m can be computed backwards:

$$\delta_{m,\lambda}(k) = \begin{cases} e_{m,\Lambda}(k)\varphi'_m\left(v_{m,\Lambda}(k)\right) \text{in output layer } \Lambda \\ \varphi'_m\left(v_{m,\lambda}(k)\right) \sum_o \delta_{o,\lambda+1}(k) w_{o,\lambda+1}(k) \\ \qquad \text{in hidden layer } \lambda \end{cases} \tag{23}$$

where the prime in the context of activation function (ϕ'_m) denotes differentiation and o refers to the output neuron. Subsequently, the network weights are adjusted iteratively using generalized delta rule:

$$w_{m,\lambda}(k+1) = w_{m,\lambda}(k) + \mu\left[w_{m,\lambda}(k-1)\right] + \alpha\delta_{m,\lambda}(k)y_{l,\lambda-1}(k) \quad (24)$$

where α is the parameter for learning rate and μ denotes the momentum constant. The momentum parameter scales the effect of the previous step on the current one, which helps the algorithm to overcome the problem of getting stuck in local minima.

At the final stage, forward and backward computations are iterated by introducing new epochs of training examples to the network until the stopping criterion is met. Fundamentally, the learning described above can be defined as the minimization of an error signal by *gradient descent* through an error surface in weight space.

To summarize the above, back-propagation training works in two phases (Haykin, 2009; Bishop, 1995):

1) *Forward phase*. Network weights are fixed and the input is forwarded through the network until it reaches the output (equations 20 & 21).
2) *Backward phase*. The output of the network is compared with the desired response to get an error signal (eq. 22), which is then propagated backward in the network (eq. 23). In the meantime, the network weights are adjusted successively to minimize the error (eq. 24).

Other algorithms for training: Major problems associated with the basic back-propagation algorithm described above are its slowness in learning and poor generalization (Tsaptsinos, 1995). Furthermore, basic back-propagation is at risk of being trapped on a local minimum in which even a small variation in weights increases the cost function (Haykin, 2009).

Over the years, many algorithms have been proposed to overcome these problems inherent in the standard gradient descent algorithm. These techniques include the so called *adaptive techniques*, which seek to avoid the local minima by using an adaptive learning rate (Riedmiller, 1994). *Delta-bar-delta rule* (Jacobs, 1988), *super self-adapting back-propagation* (Tollenaere, 1990) and *resilient back-propagation* (Riedmiller & Braun, 1993) are examples of these methods.

In addition, back-propagation techniques based on numerical optimization have been developed (Haykin, 2009). These include the so called *quasi-Newton methods* based on the famous Newton's method for optimization, *Levenberg-Marquardt algorithm* (Hagan & Menhaj, 1994), and the *conjugate gradient* (Charalambous, 1992) and *scaled conjugate gradient* algorithm (Moller, 1993).

Activation functions: The output of a neuron is defined by the activation function ϕ. This function must be continuous, because the computation of local gradients (δs) requires the existence of a derivative of the activation function (See eq. 23). Therefore, differentiability is the only requirement for the activation function (Haykin, 2009). Two basic types of activation functions can be identified (Haykin, 2009):

1) Threshold function
2) Sigmoid function

These types are presented in Figure 13. As can be seen, the sigmoid function presented in Figure 13b approaches the value of one in case of large positive numbers and zero in case of large negative numbers, which permits a smooth transition between a high and low output of a neuron. Another common sigmoid activation function in use is the hyperbolic tangent function:

$$\varphi(v) = a \tanh(v). \tag{25}$$

where *a* refers to a positive constant which defines the steepness of the slope. This function can assume also negative values, which possibly yields practical benefits over the logistic sigmoid function (Haykin, 2009).

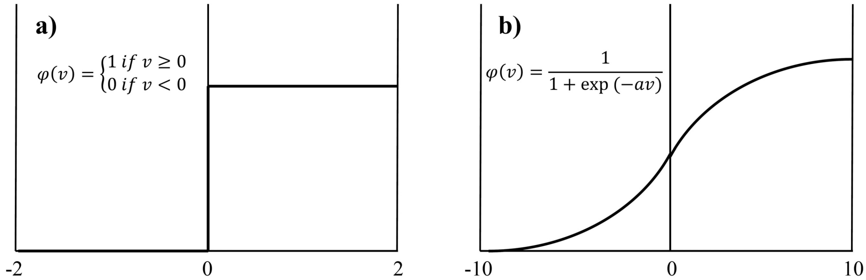

Figure 13: *Examples of the two basic types of activation functions (modified from Haykin, 2009). a) Threshold function, b) sigmoid (logistic) function. a denotes the slope parameter used for changing the steepness of the slope.*

Nevertheless, it must be noted that, although sigmoid functions are used most commonly as transfer functions, it is not self-evident that they always provide optimal decision borders (Duch & Jankowski, 1999). For this reason, many alternatives for activation functions have been proposed in the literature. Duch and Jankowski (1999) may be referred to for further information on these.

Industrial applications of MLP: Multilayer perceptrons have been used widely in a variety of industrial applications, especially in those related to modeling and identification, classification and process control (Meireles *et al.*, 2003). These applications cover a large spectrum of industrial processes and machines, e.g. induction motors, nuclear power plants, robotic systems, water supply systems, generators, welding, chemical processes, powder metallurgy, gas industry, paper making and plate rolling (Meireles *et al.*, 2003). Data-driven soft sensors in the process industry form a newer application field in which the use of MLP is growing rapidly (Kadlec et al., 2009).

Furthermore, MLP has served as the basis of intelligent applications to process improvement and monitoring in different processes such as fluidized bed combustion, production of polystyrene, water treatment and soldering of electronics. Heikkinen *et al.* (2008), Juntunen *et al.* (2010a–b) and Liukkonen *et al.* (2008, 2009b–c, 2010b, 2010e, 2010g) may be referred to if deeper information on these applications is desired.

6.2 STAGES OF INTELLIGENT QUALITY ANALYSIS

An intelligent data-based quality analysis includes several important stages. A description of the procedure is illustrated in Figure 14. The main stages of the analysis are preprocessing, selecting variables, modeling and post-processing.

6.2.1 Preprocessing

Proper preprocessing is an essential step of data analysis. Erroneous or missing data, for instance, can complicate modeling, because most analysis methods require complete data (Bishop, 1995). Compensating missing data, scaling and analyzing process lags are all important stages of preprocessing.

Missing data: Many computational methods provide that data samples do not contain any missing values. *Case deletion*, which means discarding all incomplete data rows, is one way of handling missing values. Unfortunately some information will be lost at the same time. Filling or compensating the missing values by some technique is called *data imputing*. Imputing is necessary when analyzing continuous time-series data, especially, because removing data rows is not preferable in those cases due to the cyclic characteristics of data.

The methods for solving the problem of incomplete data have been studied quite thoroughly in the past (see e.g. Junninen *et al.*, 2004; Äyrämö, 2006) and many computational techniques for compensating the missing values have been developed (Little & Rubin, 1987; Schafer, 1997). Junninen *et al.* (2004) have stated, however, that multivariate imputing techniques perform generally better than the simple methods such as imputation with mean, median or random values.

One possible way to deal with missing values is the self-organizing map (SOM) algorithm (Kohonen, 2001), which can easily use also partial training data (Samad et al., 1992). Missing values can be either ignored or included during the adaptation of SOM, and after training the estimators for missing values can be taken from the nearest reference vectors determined by the smallest Euclidean distance, for instance (Junninen et al., 2004).

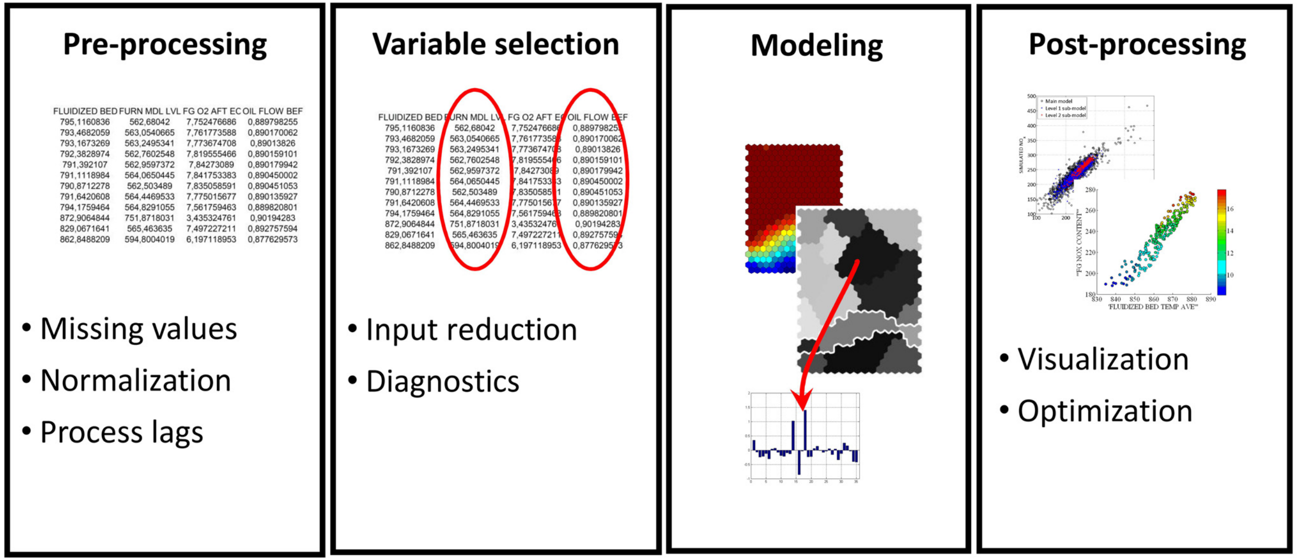

***Figure 14**: Diagram of the stages of data-based quality analysis.*

Transformation of data: Different variables usually have different ranges. Some modeling methods are sensitive to these ranges, which may lead to a situation in which the variables with a greater range overpower the influence of other variables. The variables with wider ranges will dominate in the process especially in algorithms which utilize distances between data vectors. For this reason, the numerical values of variables should be *normalized* before modeling.

In *variance scaling* the data vectors are linearly scaled to have a variance equal to one:

$$x_i' = \frac{x_i - \bar{x}}{\sigma_x}, \sigma_x \neq 0 \qquad (26)$$

where $\bar{x}$ is the average of values in vector x and σ_x denotes the standard deviation of those values. Thus, variance scaling not only equalizes the effect of those variables having a different range, but also reduces the effect of possible outliers in the data.

Process lags: Process lags can be considerable in the process industry, for example, and should therefore be taken into account in the modeling of processes in which they exist. When dealing with relatively slow fluid flows, for instance, data associated with each time stamp may not be comparable as such. Process lags can be determined using a cross-correlation method, in which the correlations between variables are calculated in a time window. (Heikkinen et al., 2009b). When it comes to wave soldering, which is a batch process, it is assumed that no lags exist between process variables, because the different parameters are measured for each product separately.

6.2.2 Selecting variables

The enormously increased flow of information has caused that selecting variables has become a relevant part of data analysis (Blum & Langley, 1997; Guyon & Elisseeff, 2003; Jain et al., 2000; Liu & Motoda, 2008). The purpose of variable selection can be improving the prediction performance of a model, providing faster processing of data or providing a better understanding of a process, for example (Guyon & Elisseeff, 2003). When using artificial neural networks in computation, for instance, reducing the number of model inputs may also reduce the computing time considerably. With respect to quality improvement, for example, it is also useful to discover the main factors affecting different physical phenomena, for which the selecting of variables can offer a good method.

According to Guyon and Elisseeff (2003), selecting variables brings many potential benefits to data analysis, e.g. facilitating visualization and understanding of data, reducing requirements for measurements and storage, reducing the time consumed by training and improving the performance of prediction. On the other hand, sometimes it is beneficial to use variable selection to discover the factors affecting a phenomenon that is yet not known well.

When reducing the dimensionality of data, two concepts are often separated, as presented in Figure 15. *Feature selection* is used to choose a subset of features, or variables, from the set of *P*, while *feature extraction* aims at creating a smaller set of combined, *derivative* variables by means of, for example, principal component analysis. Feature extraction is not further discussed in this context, however.

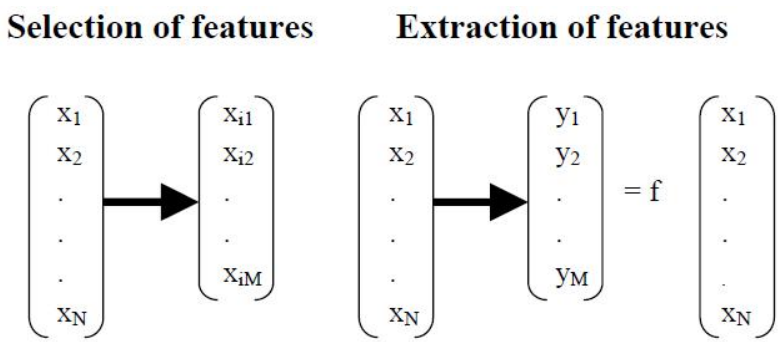

***Figure 15**: The difference between selecting features and extracting features from data.*

Variable selection can be performed either linearly or nonlinearly. Eppinger *et al.* (1995) proposed that in some cases a nonlinear neural network can reach a reasonable accuracy while the linear methods fail. It is also widely recognized, however, that the better performance of nonlinear methods is often achieved at the expense of vast computational resources. For this reason, choosing the method for selecting variables is always case-dependent.

In practice, the aim of selecting variables is at selecting a subset *Π* from the set of *P* variables without appreciably degrading the performance of the model and possibly improving it. Although exhaustive subset selection methods involve the evaluation of a very large number of subsets, the number to be evaluated can be reduced significantly by using suboptimal search procedures (Whitney, 1971). Therefore they are ideal for large data sets.

Sequential forward selection (SFS): *Sequential forward selection* is a simple suboptimal method for selecting variables in which the variables are included in progressively larger subsets so that the prediction performance of the model is maximized. To select *Π* variables from the set *P*:

1) Determine the variable that gives the best value for selected criterion.
2) Search for the variable that gives the best value *with* the variable(s) selected in stage 1.
3) Repeat stage 2 until *Π* variables have been selected or a stopping criterion has been met.

Variables can be selected also backwards. In *sequential backward selection* one starts with the set *P* and eliminates progressively the least promising variables (Guyon & Elisseeff, 2003). The method requires generally more computation than the forward selection, because all variables are included in the model at the beginning.

6.2.3 Modeling

The modeling phase is an essential part of quality analysis. There is an enormous variety of methods available for modeling, some of which rely on physical phenomena and some others of which exploit process history in the form of numerical data.

Venkatasubramanian *et al.* (2003) have presented a classification for methods of process diagnostic (see Figure 16). The authors suggest that the methods can be divided to model-based and to those based on process history, which can be further divided to quantitative and qualitative approaches. This work is focused on quantitative process history based (data-based) modeling methods.

Multi-dimensional distance measures: When dealing with models based on multivariate data, a computational method used has to be capable of handling multidimensional distances. Comparison of two data vectors requires the use of a distance measure, or *a metric*. A metric is an essential part of a learning algorithm especially in classification applications, because it determines the relative distance between multidimensional vectors.

Euclidean distance is one of the mostly used distance metrics. For two vectors x_i and x_j, it can be defined as follows:

$$D_{euc}(x_i, x_j) = \sqrt{(x_i - x_j)^T (x_i - x_j)} = \sqrt{\sum_{k=1}^{P} (x_{ik} - x_{jk})^2}, \quad (27)$$

where P is the number of vector elements (variables) and T denotes transpose. It must be noted that data transformation is a necessary pre-processing step before the use of Euclidean distance, because the metric assumes compatibility between variables. Otherwise the largest-scaled feature tends to dominate the others (Jain *et al.*, 1999).

Mahalanobis distance is another wide-spread distance metric (Jain *et al.*, 1999). Nonetheless, a drawback of the method is that the computation can be arduous with patterns of high dimensionality (Kohonen, 2001). This restricts essentially its use in real-world industrial applications in which large amounts of data are analyzed.

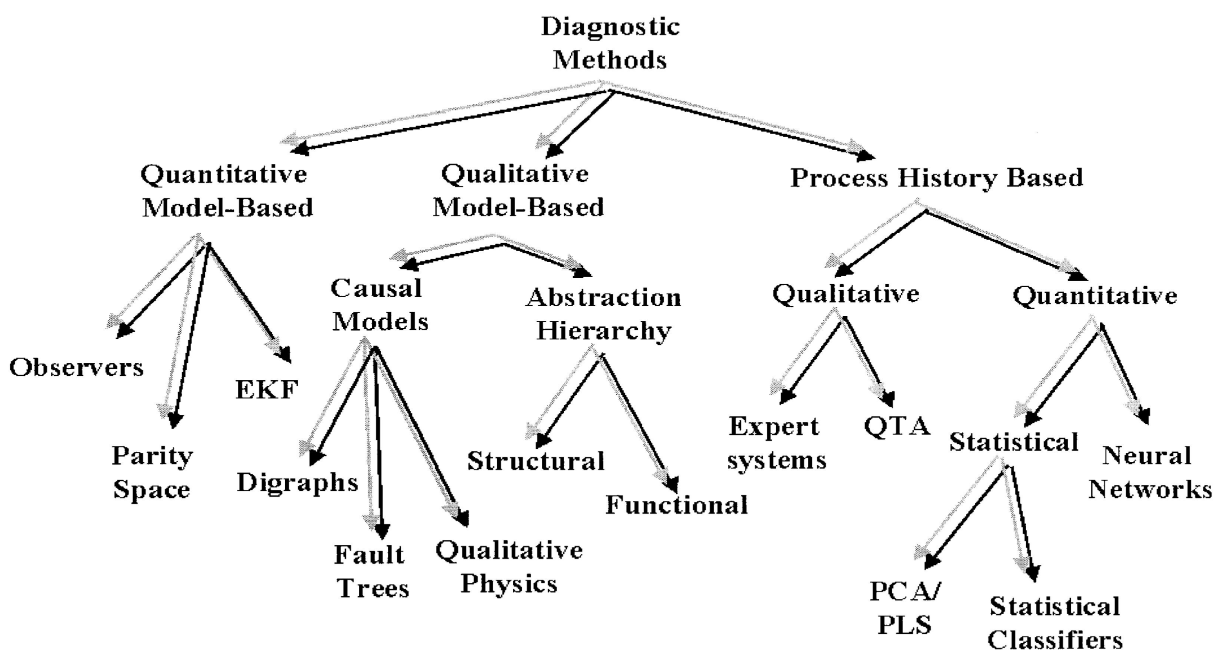

Figure 16: *Classification of diagnostic methods according to Venkatasubramanian et al. (2003).*

6.2.4 Model evaluation

Performance indicators: The goodness, or *performance*, of a model can be presented by indicators of performance. These measures use real values and the ones calculated by a model to compute a value that describes the similarity of these two vectors. The methods conventionally used for indicating performance include *Pearson's product moment correlation coefficient* and *coefficient of determination*.

Willmott (1982) has argued that the correlation measures used commonly are often inappropriate or misleading when used in comparisons between model-predicted and observed values. *Index of agreement* (Willmott, 1981) is a measure that can also be used to describe the goodness of a model:

$$C_{IA} = 1 - \frac{\sum_{i=1}^{N}(y_i - O_i)^2}{\sum_{i=1}^{N}(|y_i - \bar{O}| + |O_i - \bar{O}|)^2}$$

$$= 1 - N\frac{MSE}{PE}, \quad (28)$$

where y_i denotes a predicted element, O_i equals an observed element and $\bar{O}$ is the symbol for the average of observations. As can be seen, C_{IA} is a relative and dimensionless measure ranging from 0 to 1, the values of which approaching 1 indicate a good model. The index of agreement delineates the ratio between mean squared error (*MSE*) and potential error (*PE*), which is multiplied by *N* and then subtracted from unity. *PE* represents the largest value that the squared difference of each pair of an observation and an estimate can potentially reach. As a relative average error measure, C_{IA} presents an improvement over correlation-based measures such as the coefficient of determination, because it is more sensitive to additive and proportional differences between observations and estimates.

Validation methods: *Hold-out method* is the simplest way to validate the goodness of a model. In this approach the data set is divided into two sets, i.e. the training set and the validation set (hold-out set), of which only the training set is used for training the model. The goodness of the model is defined by predicting the values of the output in the validation data using the created model and comparing the predicted values to the real ones. The hold-out method is a fast way of validating models, but its major drawback is that the evaluation depends strongly on how data samples are divided into the training and validation sets.

Cross-validation is also a common method for evaluating the goodness of a model. The method can be defined as the test of the effectiveness of weights derived from a sample on an independently selected second sample (Mosier, 1951). Cross-validation is performed by evaluating the quality predictions with the rules extracted from a data set. There are several variants of cross-validation (see e.g. Cooil *et al.*, 1987), of which the so called *k-fold cross-validation* (Stone, 1974) is perhaps the most famous.

In k-fold cross-validation the training data set is divided into *k* subsets, or folds, of which one subset is separated to be used in independent validation. The rules for prediction are obtained from the remaining *k*–1 subsets. Then the values for the objects in the validation subset are predicted based on the rules attained from the training data. The process is repeated until every subset *k* has served as the validation data set.

L*eave-one-out cross-validation* is a special form of k-fold cross-validation, in which *k* equals the number of original data rows. In other words, the method uses a single original data sample as the validation data, while the remaining samples serve as the training data. The process is repeated until each sample has served once as the validation sample. A disadvantage of the method is that the large number of training rounds makes it computationally arduous and time-consuming, which limits its use in applications involved with large data sets. Creating a model from 1 000 original data rows, for example, involves 1 000 training rounds using the leave-one-out method, whereas a 5-fold cross-validation requires only five rounds.

6.2.5 Post-processing

Post-processing is the processing of results obtained by modeling to a useful form with regard to the application. This is an important part of quality analysis, because it assures the proper interpretation of analysis results and enables the refinement of information into knowledge that can be potentially used for process improvement.

Visualization is one of the most important parts of post-processing, especially when performing diagnosis in an industrial environment, because good visualization makes it possible for the end-user to view the results at one glance. Visualization of models can be performed in several ways depending on the application and on the methods used.

Another possible option to post-process the created model is *parameter estimation*. The model can be used for searching optimal parameter combinations to maximize the quality of products or to minimize the total cost of quality, for example, as presented by Liukkonen *et al*. (2010a).

6.3 RESULTS OF QUALITY ANALYSIS

The methodology proposed for the intelligent quality analysis of wave soldering and the results of the analysis are presented in four scientific papers, of which the main findings are described shortly here.

An application based on self-organizing maps for the analysis of wave soldering process is suggested in paper I (Liukkonen et al., 2009a). The procedure for data analysis is such that first process data (see Table 1) are coded as inputs for a SOM. Next, the reference vectors of the SOM neurons are clustered by k-means. At the final stage, the clusters are treated as sub-models to indicate variable dependencies within the clusters. Some of the sub-models discovered are presented in Figure 17.

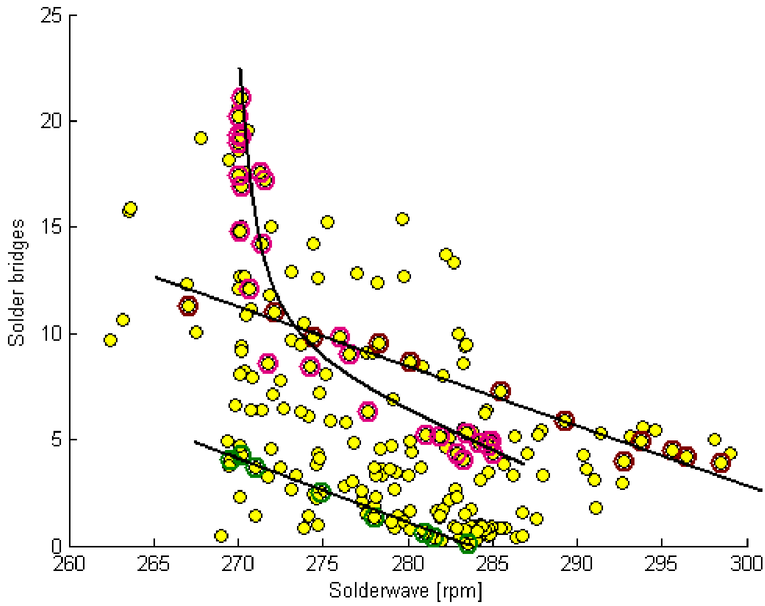

Figure 17*: Sub-models for solder bridges in wave soldering. Yellow balls represent the reference vector values from the neurons of SOM. The colored circles with fitted lines represent different clusters after clustering the reference vectors using k-means.*

Modeling of soldering quality using multilayer perceptrons is studied and discussed in Paper II (Liukkonen et al., 2009b). The paper concentrates on selecting the most important variables with respect to different soldering defects using both a linear method (multiple linear regression) and a nonlinear one (multilayer perceptron). The comparison of the performance of the methods in the analysis of balled solders can be seen in Figure 18.

Quality-oriented optimization of wave soldering using self-organizing maps is presented in Paper III (Liukkonen et al., 2010a). The main finding of the paper is that the SOM offers a visual and relatively easy alternative to the nonlinear modeling and optimization of a soldering process. SOM component planes with regard to the estimated costs of different defect types in wave soldering, for example, give a good overview of the economical significance of single defect types. In addition, the 3-dimensional visualization of SOM (see **Figure 19**) offers a practical platform for indicating and visualizing the variation of the total quality cost, which is especially useful when analyzing large data sets. The method can also be used in estimating optimal parameters on the basis of process history.

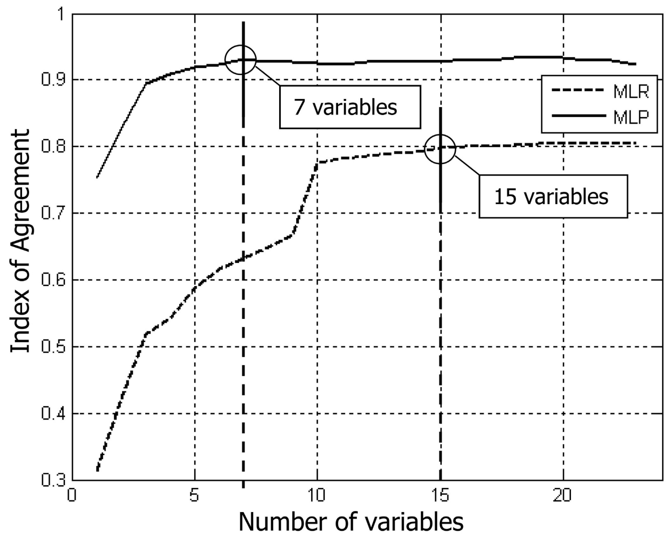

***Figure 18**: The performance of variable selection for balled solders in wave soldering using multiple linear regression (MLR) and multilayer perceptrons (MLP).*

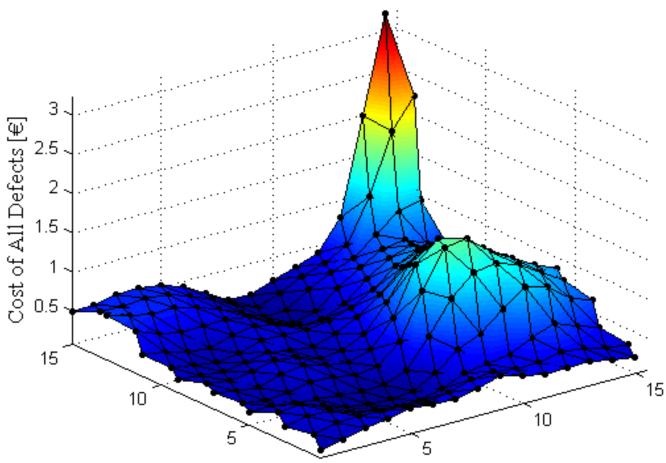

***Figure 19**: 3D self-organizing map with 15 x 15 neurons indicating the variation of the total cost of quality in the wave soldering research case.*

Paper IV (Liukkonen et al., 2010b) is focused on creating a generic intelligent optimization and modeling system for electronics production (see Figure 20). The application can be used in diagnostics and proactive quality improvement of electronics production and is intended primarily for process experts, who have the skills and knowledge to validate the results before they are introduced to the operational level. The system utilizes real production data and can be used for diagnosing and optimizing the processes for manufacturing electronics. It contains three modules which consist of computer algorithms specifically tailored to each task, i.e. preprocessing, selecting variables and optimizing.

The module for selecting variables can be used either for finding the factors affecting quality or for obtaining a model for predicting quality. Either linear regression or MLP can be selected for the modeling method. The optimization module outputs the optimal parameters and a visual SOM model, which describes the behavior of quality cost.

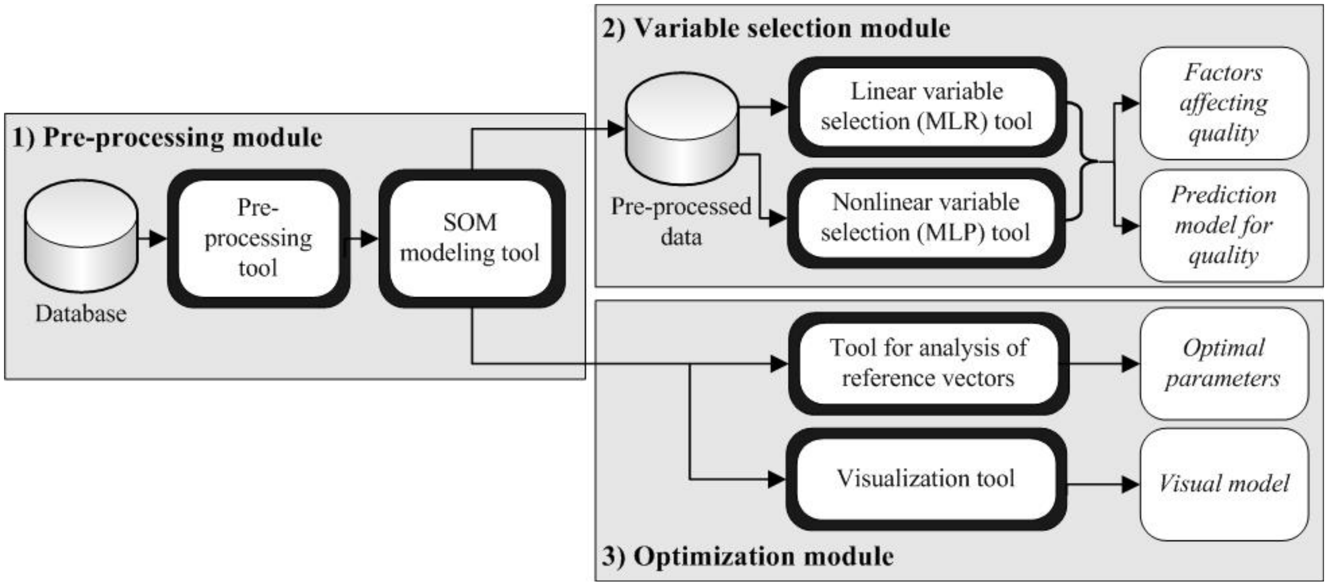

Figure 20: *The system developed for intelligent quality analysis of electronics production, which utilizes archived process history.*

7. Discussion

This work documents the application of intelligent data-based modeling methods in the quality analysis of electronics production. The purpose was to benefit from the most useful characteristics of computational intelligence, which include an ability to solve real world problems, to learn from experience, to self-organize and to adapt in response to dynamically changing conditions. Methods that fill these requirements were applied to the quality analysis of automated soldering in this thesis.

As the literature review in Chapter 3 indicates, the only production stage of electronics in which intelligent methods have been used in a larger scale has been the inspection of solder joints after soldering. Further quality analysis and process improvement are still often performed using traditional statistical methods. The research presented in this thesis involves the application of intelligent data-based analysis methods, which are new to this field, to modeling and optimization of an automated soldering process. Both advanced data-driven methods as well as the traditional statistical methods were used in the analyses.

Generally speaking, most of the advantages of computational intelligence were utilized in the study. Real problems with respect to product quality were solved using intelligent methods and process history. The ability of the methods to self-organize was illustrated clearly especially when the data were analyzed using the SOM method. Only the ability of the methods to adapt to changing conditions by learning could not be exploited thoroughly, because of the nature of the experimental data used in the analysis. In this case the data set, although it was the best available one, was quite small, for which the ability of the methods to adapt could not be tested properly. It is usual in the electronics industry, however, that the data sets are much larger than the one used here, and therefore the adaptivity is an important issue that has to be taken into account in the future.

The aims set for the research presented in the thesis were:

- To determine whether intelligent data-based methods could offer additional value to production of electronics.
- To create a procedure of quality analysis for electronics manufacturing, starting from pre-processing of data and ending up to visualization and analysis of results.
- To promote the awareness of electronics producers of the intelligent data-based methods and their exploitation possibilities in process improvement.

When it comes to reaching the first aim, many interesting and previously unseen results were obtained from the wave soldering process by means of the applied intelligent methods. The results indicate that computationally intelligent methods can be applied successfully to analysis and optimization of electronics production. The results offer many opportunities for further consideration in the future.

The SOM-based methodology suggested for data analysis can reveal dependencies between data variables relatively fast and easily, which would be much more laborious by using the methods for data processing that are traditional in the electronics industry. A surprising result is, for example, that the use of different flux types results in tremendously different effects of certain process parameters on the defect formation in wave soldering. This result would not have been reached using the conventional computational methods, which suggests that the intelligent methods are useful in this field and therefore supports the achievement of the first aim.

Several facts make the use of the SOM-based method in the analysis of wave soldering reasonable. Firstly, if a large number of the values of the data are missing, the conventional statistical methods would be time-consuming and difficult to use. By using the SOM method with the batch algorithm instead, this problem does not arise, because the possible missing values in the data are simply ignored as the reference vector values are calculated and BMUs for the input vectors are selected. On the other hand, if the data set is large, separating the desired subsets from it before the analysis stage would be a time-consuming operation and demand a lot of resources, whereas the proposed methodology makes this process easier.

Perhaps most important, however, is that the clustering behavior of the data is usually not known before the quality analysis. In this case the cluster borders seem to follow the different flux types, but in other cases some other factors may be dominant in the formation of clusters. The methodology suggested for quality analysis does not necessitate *a priori* knowledge of the clusters, which makes it a useful way of analyzing the structure of data.

The SOM-based method also provides a descriptive and relatively simple way of visualizing a large amount of manufacturing data, which is typical in the electronics industry. The results presented demonstrate that the self-organizing map provides an efficient and useful algorithm for revealing the most characteristic features of input data, which makes it a powerful method for discovering generic phenomena and visualizing the behavior of an automated soldering process. Especially the 3D representation of SOM component planes provides a visual way of presenting a large amount of quality-related data. The results suggest that the method facilitates data analysis and can be used to diagnose the performance of a process in a convenient and user-friendly manner. It is presumable that these benefits of the method are emphasized even more when larger data sets than the one used here are analyzed.

Searching for the most important factors for defect formation and predicting the product quality form an interesting part of quality analysis and can be considered a tempting possibility. When having a tentative prediction for defect numbers, it is easier to direct resources to repairing operations. It is also easier to start process improvement and optimization if the most important factors for each defect type are known. The presented method offers a fruitful way of performing this kind of process analysis. The results suggest that using the nonlinear method based on multilayer perceptrons improves the goodness of models. The other benefit of the nonlinear method is that fewer variables are generally needed to obtain an optimal model. Generally speaking, the results show that the MLP-based method provides plenty of extra value to quality analysis and is thus one part of achieving the first aim.

The prediction accuracy of the MLP model for different soldering defect types is generally good based on the results. The relatively large number of samples may not be adequate enough with respect to some of the defect types, however, which possibly reduces the goodness of some of the models. On the other hand, the defect numbers were visually detected and manually recorded, so it is possible that there are flaws in the raw data that weaken the model performance. In addition, the cross-validation used in model validation ensures that the selection of training data does not affect the modeling results, because the entire data set is exploited both in training and in independent validation. Data sets available in mass production of electronics are generally larger, however, which ensures an adequate number of samples and thus enables reliable automated applications. In addition, the detection of defects by automated optical inspection, as often is the case in the modern mass production of electronics, would eliminate any human-inflicted errors in data.

On a more general level, the results show that the advantages of using the MLP-based approach in the analysis of the soldering process are considerable, because the method benefits greatly from the general characteristics of computational intelligence. The method has a high computing power and is able to find nonlinear connections in the data. Because the model is obtained by training with real process data, it is also able to adapt to exceptional situations in the production, unlike the purely physical models based on predetermined functions, and therefore provides an efficient way of solving problems.

Thus, the method is especially suitable for cases in which the physical processes are not well known or are highly complex.

The results of the quality-oriented optimization of wave soldering can be interpreted in two ways. First of all, if the numbers of single defect types are desired to be maintained at low levels, certain parameters should be used in the production. On the other hand, the optimization routine in which all the defect types are considered supports the use of other process parameters. It is evident that the most reasonable goal would be to minimize the total cost of repairs, because then the total cost of production would most likely also decrease. Sometimes it may be beneficial to minimize only certain defect types, however. For instance, if it is difficult to determine the repairing costs of defects accurately enough it could be reasonable to aim at reducing the number of those defect types that are considered most problematic in the production.

The method presented for quality-oriented optimization using the SOM provides an easy way of estimating optimal process parameters with respect to product quality. The parameters can be estimated accurately using the presented method if such data exist that cover the search space. The apparent benefits of the method are flexibility, nonlinearity and a strong computing power. The method used is also very illustrative and relatively simple to use, and therefore provides a useful and efficient way to define the optimal parameters of a manufacturing process, which means that the first aim of the thesis can be considered achieved. On the whole, the method also presents a good example of the utilization of the key elements of computational intelligence listed at the beginning of this chapter.

The optimization of the cost of quality can be considered a one step forward in quality assurance. It is an important result, because it presents the bad quality as an expense in a comprehensive manner. It enables the reworking of defects in a two-step approach:

1) The critical defects have to be removed.
2) The quality cost has to be minimized.

It is often reasonable to minimize the formation of the defects of type 1 already in the production. The latter aspect offers a possibility to decide whether it is reasonable to speed up the production at the expense of quality and rework the products later, for example. This may be worth considering if the defects are easily repairable and their rework cost is low. This kind of philosophy can make the production more flexible by making the use of resources more efficient.

The intelligent optimization and modeling system developed for electronics production shows that intelligent methods are applicable to generic analysis applications in this field. This is useful because modern electronics process equipment typically store large amounts of data that may be used in statistical process control, but are not generally exploited in the proactive quality improvement and diagnosis of processes. The system is especially useful in processing large data sets, because the SOM enables condensing large amounts of numerical information.

It is important to emphasize, however, that creating nonlinear models demands more expertise from the user than creating the linear ones used traditionally by the electronics industry. Therefore the system developed is intended primarily for process experts, who have the skills and knowledge to validate the results before they are introduced in the operational level. The system makes it possible to automate arduous data processing, however, which facilitates quality management and so enables achieving better quality in the electronics production.

Both linear and nonlinear methods are included to the system, although the case problem (defect formation) seems to be nonlinear. This is because the linear routine for selecting variables is computationally much faster than the nonlinear one. Thus, the linear routine can be used as a fast first-stage method when analyzing large data sets. Moreover, because the data used in the case study consists of separate sets of test arrangements, the MLP model, although performing generally better, is also more sensitive to the selection of training and validation data sets, which decreases the generalization ability of the model. This supports strongly the use of cross-validation in the selection of variables, especially if the data set is small.

The intelligent quality analyzer answers directly the question on how the application of intelligent methods can be done in the electronics industry, and therefore supports the achievement of the second aim. The system consists of modules comprising the whole procedure of quality analysis including the pre-processing of data, selecting variables, optimization and so on (see Fig. 20).

The third aim of the thesis was to promote the awareness of electronics producers of the intelligent data-based methods and their exploitation possibilities in process improvement. The results have been published in three separate international journals, which has promoted the distribution of the gained knowledge. In addition, co-operation has been organized with seven different companies from the field of electronics production in three separate research projects during the years 2006–2010, and the projects seem to continue even further. In this sense, the third aim can be considered achieved.

On the whole, the aims of the thesis were achieved. In summary, the results show that data-based intelligent methods can improve quality analyses in the electronics industry in many ways, e.g. by:

- Speeding up the processing of large amounts of data.
- Facilitating the processing of missing data.
- Enabling the detection of multivariate dependencies.
- Enabling the detection of nonlinear dependencies.
- Improving the goodness of prediction models.
- Offering new illustrative ways of visualizing multivariate interactions.
- Providing an efficient way of solving problems.
- Enabling a more efficient diagnosis of quality and quality cost.

- Providing an adaptive platform for intelligent applications.

Despite the promising results, it is important to bear in mind that every time a defect is detected it is already too late to make any changes to materials, design or process. The only options left are to rework the product or to discard the product, and both ways materials and other resources are wasted. Therefore preventive actions are preferable in quality assurance. In practice, such links between process diagnosis and product and process design should exist which would ensure the continuous flow of feedback from the intelligent quality analysis to designers. The intelligent system offers an alternative for quality-oriented analysis of processes and provides a means to extract knowledge that can be used in preventive quality assurance.

The possibilities for utilizing the intelligent procedure for quality analysis more widely include a large spectrum of applications. One of these is process diagnostics and the obtaining of reasons for bad quality. Multivariate interactions between process variables, for example, can be seen at a glance on the SOM, and the selecting of variables can provide deeper information on the process and its quality issues. Optimization of process parameters with respect to quality cost is another potential application field. The approach could be utilized in high-level control systems, which would suggest the use of certain parameters based on previous knowledge, for example. The procedure could be also useful in predictive systems, which would estimate the produced quality or the cost of quality in the future. Moreover, a comprehensive quality model which would make it possible to minimize the cost of bad quality would enable improving the design of electronic products and their manufacturing.

8. Conclusions and future

8.1 CONCLUSIONS

Data-based diagnosis of processes has become an essential part of quality improvement in recent years, because archived process data have the potential for being used in optimization and improvement of productivity. It seems that there is a constant need for new data-driven systems for process diagnosis, which can process even larger amounts of data and which can be used in process monitoring and analysis to improve the process and the quality of final products. Computationally intelligent methods, which have not been utilized by the electronics industry on a large scale, seem to provide a respectable option for analyzing quality in the production of electronics.

The purpose of this study was to advance the use of intelligent data-based methods in the production of electronics by exploring the current state of research on their use in this field and by applying them to a real automated process of manufacturing electronics. The ultimate goal was to develop a methodology for the quality analysis of electronics production using intelligent methods, which was achieved in practice by using the wave soldering process as an example.

The main conclusion of the thesis is that intelligent methods should be used in the electronics industry on a much larger scale than they are today. As the results suggest, they provide an efficient way of analyzing quality in the electronics industry. They can reveal mutual interactions which are otherwise difficult to find, improve the goodness of models and decrease the number of variables needed for modeling or testing. Intelligent methods can also offer a useful way of analyzing large data sets and provide a practical platform for representing them visually. Perhaps the most important thing is, however, that they are applicable to generic data-based applications, which facilitates their implementation in the electronics industry.

8.2 IDEAS FOR FUTURE WORK

Quality and its monitoring form an important part of manufacturing certain special products containing electronics. Typically these special products are electronic devices having a long lifecycle and intended to professional use, in which reliability and durability are considered highly important unlike in the large-volume mass products of these days. It is important in manufacturing these products that the production-related information on the materials and the process parameters used, for example, would be available also at the later stages of the lifecycle of the products, for instance if they are delivered back to be repaired under warranty at some stage. Assuring the traceability of product-related data would be significant also because it would enable quality analyses like presented in this thesis but on a much larger scale, which would make it possible to improve the quality of products comprehensively and even during their entire lifecycle.

This is somewhat problematic in practice, however. Special products containing complicated electronics often consist of many separate electronic parts such as PCBs and other electronic components. It would be necessary to individualize those semi-finished products that are to be integrated to the final product, because that would make it possible to assign information from all the production stages to the final product. In an ideal situation this would prepare the way for finding causal connections between the production stages of semi-finished products and the functionality of assembled products in the final testing, for example. This necessitates integration of traceability to the production using product-specific identifiers, however.

In theory it is possible to implement product-specific monitoring of electronic products during their lifecycle using new technologies such as *radio frequency identification* (RFID). Particularly when it comes to products with a long lifecycle, the assurance of traceability would be significant, because it would enable comprehensive improvement of quality, and therefore cost savings and even prolongation of the lifecycle of products. In addition, traceability would be obviously an asset for a product, because its standard is raised by the new attribute.

Despite the optimistic prospects of new technologies for assuring traceability, their integration to production of electronics and exploitation during the lifecycle of products require a great deal of experimentation and research. It seems that there is a chance for arranging wireless traceability of items, however, which would enable even more efficient intelligent quality analyses in the future.

References

Abonyi J., Nemeth S., Vincze C., and Arva P., 2003. Process Analysis and Product Quality Estimation by Self-Organizing Maps with an Application to Polyethylene Production. *Computers in Industry*, vol. 52, pp. 221–234.

Acciani G., Brunetti G., and Fornarelli G., 2006a. Application of Neural Networks in Optical Inspection and Classification of Solder Joints in Surface Mount Technology. *IEEE Transactions on Industrial Informatics*, vol. 2, no. 3, pp. 200–209.

Acciani G., Brunetti G., and Fornarelli G., 2006b. A Multiple Neural Network System to Classify Solder Joints on Integrated Circuits. *International Journal of Computational Intelligence Research*, vol. 2, no. 4, pp. 337–348.

Ackoff R.L., 1989. From Data to Wisdom. *Journal of Applied Systems Analysis*, vol. 16, pp. 3–9.

Alavala C.R., 2008. *Fuzzy Logic and Neural Networks: Basic Concepts and Applications*. New Age International Publishers, New Delhi.

Aleksander I. and Morton H., 1990. *An Introduction to Neural Computing*. Chapman and Hall, London.

Alhoniemi E., Hollmén J., Simula O., and Vesanto J., 1999. Process Monitoring and Modeling Using the Self-Organizing Map. *Integrated Computer Aided Engineering*, vol. 6, no.1, pp. 3–14.

Alhoniemi E., 2002. *Unsupervised Pattern Recognition Methods for Exploratory Analysis of Industrial Process Data*. Doctoral thesis, Helsinki University of Technology, Finland.

Allen W.J.J., Curran E.P., and Stewart J.J.T., 1995. A Design for Manufacture Knowledge-Based System in Printed Board Assembly Production for Northern Telecom (Northern Ireland Ltd.). *Journal of Electronics Manufacturing*, vol. 5, no. 1, pp. 57–63.

Antony J., 2004. Some Pros and Cons of Six Sigma: An Academic Perspective. *The TQM Magazine*, vol. 16, no. 4, pp. 303–306.

Arra M., Shangguan D., Yi S., Thalhammer R., and Fockenberger H., 2002. Development of Lead-Free Wave Soldering Process. *IEEE Trans. Electronics Packaging Manufacturing*, vol. 25, no. 4, pp. 289–299.

Äyrämö S., 2006. *Knowledge Mining Using Robust Clustering*. Doctoral thesis, University of Jyväskylä, Finland.
Available via: http://urn.fi/URN:ISBN:951-39-2655-9

Äyrämö S. and Kärkkäinen T., 2006. Introduction to Partitioning-based Clustering Methods with a Robust Example. In: Reports of the Department of Mathematical Information Technology, Series C (Software and Computational Engineering), No. C. 1/2006. University of Jyväskylä.

Barbini D. and Wang P., 2005. Implementing Lead-Free Wave Soldering: Higher Levels of Copper and Iron Can Change the Alloy and Require New Guidelines. *Printed Circuit Design & Manufacture*, May 2005.

Barbini D., Wang P., 2007. Wave Solder: Process Optimization for Simple to Complex Boards. *Global SMT & Packaging*, vol. 7, no. 9, pp. 10–17.

Barreto G.A., 2008. Time Series Prediction with the Self-Organizing Map: A Review. *Studies in Computational Intelligence*, vol. 77, pp. 135–158.

Bernard C., 1977. Wave Soldering Joint Quality Trouble Shooting Guide. *Insulation/Circuits*, vol. 23, no. 12, pp. 23–25.

Bishop C., 1995. *Neural Networks for Pattern Recognition*. Clarendon Press, Oxford.

Blum A.L. and Langley P., 1997. Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence*, vol. 97, no. 1–2, pp. 245–271.

Borneman J.D., 1981. Quick Wavesoldering Troubleshooting. *Insulation/Circuits*, vol. 27, no. 8, pp. 38–39.

Boser B., Guyon I., and Vapnik V.N., 1992. A Training Algorithm for Optimal Margin Classifiers. In: Haussler D. (ed.), *Proc. the Fifth Annual Workshop on Computational Learning Theory*, pp. 144–152. ACM Press.

Boulos M., Hamilton C., Moreno M., Mendez R., Soto G., and Herrera J., 2009. Selective Wave Soldering DoE to Develop DfM Guidelines for Lead and Pb-free Assemblies. *Circuits Assembly*, vol. 20, no. 1, pp. 26–34.

Breiman L., Friedman J.H., R.A. Olshen, and C.J. Stone, 1984. *Classification and Regression Trees*. Wadsworth Inc., Monterey, CA.

Briggs A.J. and Yang C.M., 1990. Experiment Design, Defect Analysis, and Results for the Wave Soldering of Small Outline Integrated Circuits. In: *Proc. of the Electronic Manufacturing Technology Symposium*, pp. 361–365.

Brinkley P.A., 1993. Northern Telecom Achieves Improved Quality by Combining DOE and SPC. *Industrial Engineering*, vol. 25, no.5, pp. 63–65.

Broomhead D.S. and Lowe D., 1988. Multivariable Functional Interpolation and Adaptive Network. *Complex Systems*, vol. 2, pp. 321–355.

Charalambous C., 1992. Conjugate Gradient Algorithm for Efficient Training of Artificial Neural Networks. *IEE Proceedings – G Circuits Devices and Systems*, vol. 139, no. 3, pp. 301–310.

Cheng B. and Titterington D.M., 1994. Neural Networks: A Review from a Statistical Perspective. *Statistical Science*, vol. 9, no. 1, pp. 2–30.

Cho H. and Park W.S., 2002. Neural Network Applications in Automated Optical Inspection: State of the Arts. In: B. Javidi and D. Psaltis (eds.), *Algorithms and Systems for Optical Information Processing* VI, Proceedings of SPIE, vol. 4789, pp. 224–236.

Choudhary A.K., Harding J.A., and Tiwari M.K., 2009. Data Mining in Manufacturing: A Review Based on the Kind of Knowledge. *Journal of Intelligent Manufacturing*, vol. 20, pp. 501–521.

Chowdhury A.R. and Mitra S., 2000a. Reduction of Defects in Wave Soldering Process. *Quality Engineering*, vol. 12, no. 3, pp. 439–445.

Chowdhury K.K., Gijo E.V., and Raghavan R., 2000b. Quality Improvement Through Design of Experiments: A Case Study. *Quality Engineering*, vol. 12, no. 3, pp. 407–416.

Chunquan L., Dejian Z., and Zhaohua W., 2004. Study on SMT Solder Joint Quality Fuzzy Diagnosis Technology Based on the Theory of Solder Joint Shape. *China Mechanical Engineering*, vol. 15, no. 21, pp. 1967–1970.

Chunquan L., 2007. Study on Assembly Quality Fault Diagnosis System of Chip Components based on Fuzzy Analysis. *WSEAS Transactions on Systems*, vol. 6, no. 1, pp. 109–116.

Coit D.W., Billa J., Leonard D., Smith A.E., Clark W., and El-Jaroudi A., 1994. Wave Solder Process Control Modeling Using a Neural Network Approach. In: Dagli C.H., Fernandez B.R., Ghosh J., and Kumara R.T.S. (eds.), *Intelligent Engineering Systems Through Artificial Neural Networks*, vol. 4, ASME Press, New York, pp. 999–1004.

Coit D.W., Jackson B.T., and Smith A.E., 1998. Static Neural Network Process Models: Considerations and Case Studies. *International Journal of Production Research*, vol. 36, no.11, pp. 2953–2967.

Coit D.W., Jackson B.T., and Smith A.E., 2002. Neural Network Open Loop Control System for Wave Soldering. *Journal of Electronics Manufacturing*, vol. 11, no.1, pp. 95–105.

Cooil B., Winer R.S., and Rados D.L., 1987. Cross-Validation for Prediction. *Journal of Marketing Research*, vol. 24, pp. 271–279.

Davies D.L. and Bouldin D.W., 1979. A Cluster Separation Measure. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, vol. 1, no.2, pp. 224–227.

Diepstraten G., 2001. Analyzing Lead-Free Wavesoldering Defects. *Surface Mount Technology*, suppl. issue, June 2001, pp. 2–5.

Dietterich T.G., 1997. Machine-Learning Research: Four Current Directions. *AI Magazine*, vol. 18, no. 4, pp. 97–136.

Dodge H.F. and Torrey M.N., 1977. A Check Inspection and Demerit Rating Plan. *Journal of Quality Technology*, vol. 9, no. 3, pp. 146–153.

Dror H.A. and Steinberg D.M., 2006. Robust Experimental Design for Multivariate Generalized Linear Models. *Technometrics*, vol. 48, no.4, pp. 520–529.

Duch W. and Jankowski N., 1999. Survey of Neural Transfer Functions. *Neural Computing Surveys*, vol. 2, pp. 163–212.

Edinbarough I., Balderas R., and Bose S., 2005. A Vision and Robot Based On-line Inspection Monitoring System for Electronic Manufacturing. *Computers in Industry*, vol. 56, no. 8–9, pp. 989–996.

Engelbrecht A.P., 2007. *Computational Intelligence: An Introduction*, 2nd edition. John Wiley & Sons, West Sussex, England.

Eppinger S.D, Huber C.D., and Pham V.H., 1995. A Methodology for Manufacturing Process Signature Analysis. *Journal of Manufacturing Systems*, vol. 14, no. 1, pp. 20–34.

Erwin E., Obermayer K., and Schulten K., 1992. Self-organizing maps: Ordering, convergence properties and energy functions. *Biological Cybernetics*, vol. 67, no. 1, pp. 47–55.

Everitt B.S., 1974. *Cluster Analysis*. John Wiley & Sons, New York.

Fayyad U.M., Piatetsky-Shapiro G., Smyth P., and Uthuruswamy R. (eds.), 1996. *Advances in Knowledge Discovery and Data Mining*. AAAI/MIT Press, Menlo Park, California.

Feng C.-X. J., Gowrisankar A.C., Smith A.E., and Yu Z.-G. S., 2006. Practical Guidelines for Developing BP Neural Network Models of Measurement Uncertainty Data. *Journal of Manufacturing Systems*, vol. 25, no. 4, pp. 239–250.

Fidan I. and Kraft R.P., 2000. Inline Troubleshooting for Electronics Manufacturing Systems. In: Proc. IEEE/CPMT Intelligent Electronics Manufacturing Technology Symposium, pp. 338–343.

Fisher R.A., (1935). *The Design of Experiments*. Oliver and Boyd, Edinburgh.

Fountain T., Dietterich T., and Sudyka B., 2000. Mining IC Test Data to Optimize VLSI Testing. In: *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computer Machinery, New York, USA, pp. 18–25.

Gebus S., 2006. *Knowledge-based Decision Support Systems for Production Optimization and Quality Improvement in the Electronics Industry*. Doctoral thesis, University of Oulu, Finland.

Gebus S. and Leiviskä K., 2009. Knowledge Acquisition for Decision Support Systems on an Electronic Assembly Line. *Expert Systems with Applications*, vol. 36, pp. 93–101.

Gen M. and Cheng R., 1997. *Genetic Algorithms & Engineering Design*. John Wiley & Sons, United States of America.

Giaquinto A., Fornarelli G., Brunetti G., and Acciani G, 2009. A Neurofuzzy Method for the Evaluation of Soldering Global Quality Index. *IEEE Transactions on Industrial Informatics*, vol. 5, no. 1, pp. 56–66.

Guyon I. and Elisseeff A., 2003. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, vol. 3, pp. 1157–1182.

Hagan M.T. and Menhaj M., 1994. Training Feedforward Networks with the Marquardt Algorithm. *IEEE Trans. Neural Networks*, vol. 5, pp. 989–993.

Halgamuge S.K. and Glesner M., 1994. Neural Networks in Designing Fuzzy Systems for Real World Applications. *Fuzzy Sets and Systems*, vol. 65, no. 1, pp. 1–12.

Hand D., Mannila H., and Smyth P., 2001. *Principles of Data Mining*. MIT Press, Campbridge (MA).

Harding J.A., Shahbaz M., Srinivas, and Kusiak A., 2006. Data Mining in Manufacturing: A Review. *Journal of Manufacturing Science and Engineering*, vol. 128, no.4, pp. 969–976.

Havia E., Bernhardt E., Mikkonen T., Montonen H., and Alatalo M., 2005. Implementation of Lead-free Wave Soldering Process. In: *Proc. Electronics Production and Packaging Technology, ELTUPAK 2005*.

Haykin S., 2009. *Neural Networks and Learning Machines*, 3rd edition. Pearson Education Inc. Upper Saddle River, New Jersey.

Heikkinen M., Nurminen V., Hiltunen T., and Hiltunen Y., 2008. A Modeling and Optimization Tool for the Expandable Polystyrene Batch Process. *Chemical Product and Process Modeling*, vol. 3, no. 1, article 3.

Heikkinen M., Poutiainen H., Liukkonen M., Heikkinen T., and Hiltunen Y., 2009a. SOM-based Subtraction Analysis to Process Data of an Activated

Sludge Treatment Plant. In: Troch I. and Breitenecker F. (eds.), *Proceedings of MATHMOD 09 VIENNA*, Full Papers Volume [CD], pp. 1021–1026. Argesim Report No. 35. ARGESIM – Publishing House, Vienna.

Heikkinen M., Hiltunen T., Liukkonen M., Kettunen A., Kuivalainen R., and Hiltunen Y., 2009b. A Modelling and Optimization System for Fluidized Bed Power Plants. *Expert Systems with Applications*, vol. 36, no. 7, pp. 10274–10279.

Heikkinen M., Poutiainen H., Liukkonen M., Heikkinen T., and Hiltunen Y., 2010. Self-Organizing Maps in the Analysis of an Industrial Wastewater Treatment Process. *Mathematics and Computers in Simulation* (in press).

Helo P., 2004. Managing Agility and Productivity in the Electronics Industry. *Industrial Management & Data Systems*, vol. 104, no. 7, pp. 567–577.

Hiltunen Y., Heikkinen M., Hiltunen T., Räsänen T., Huhtinen J., and Kettunen A., 2006. A SOM-based Tool for Process State Monitoring and Optimization. In: Juuso E. (ed.), *Proceedings of the 47th Conference on Simulation and Modelling (SIMS)*, Finnish Society of Automation and SIMS, pp. 164–169.

Ho S.-L., Xie M., and Goh T.-N., 2003. Process Monitoring Strategies for Surface Mount Manufacturing Processes. *The International Journal of Flexible Manufacturing Systems*, vol. 15, pp. 95–112.

Hoe S.L., Toh K.C., and Chan W.K., 1998. A Thermal Model of the Preheat Section in Wave Soldering. In: *Proc. 2nd Electronics Packaging Technology Conference*, pp. 240–245.

Holland J.H., 1975. Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor.

Jacobs R., 1988. Increased Rates of Convergence Through Learning Rate Adaptation. *Neural Networks*, vol. 1, no. 4, pp. 295–307.

Jagannathan S., Seebaluck D., and Jenness J.D., 1992. Intelligent Inspection of Wave-Soldered Joints. *Journal of Manufacturing Systems*, vol. 11, no. 2, pp. 137–143.

Jagannathan S., 1997. Automatic Inspection of Wave Soldered Joints Using Neural Networks. *Journal of Manufacturing Systems*, vol. 16, no. 6, pp. 389–398.

Jain A.K. and Dubes R.C., 1988. *Algorithms for Clustering Data*. Prentice Hall, New Jersey.

Jain A.K., Murty M.N., and Flynn P.J., 1999. Data Clustering: A Review. *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323.

Jain A.K., Duin R.P.W., and Mao J., 2000. Statistical Pattern Recognition: A Review. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37.

Jämsä-Jounela S.-L., Vermasvuori M., Endén P., and Haavisto S., 2003. A Process Monitoring System Based on the Kohonen Self-Organizing Maps. *Control Engineering Practice*, vol. 11, pp. 83–92.

Judd M. and Brindley K., 1999. *Soldering in Electronics Assembly*, 2nd edition. Newnes, Oxford, MA.

Junninen H., Niska H., Ruuskanen A., Patama T., Tuppurainen K., Kolehmainen M., and Ruuskanen J., 2004. The Performance of Different Imputation Methods for Air Quality Data with Missing Values. *Atmospheric Environment*, vol. 38, pp. 2895–2907.

Juntunen P., Liukkonen M., Pelo M., Lehtola M. and Hiltunen Y, 2010a. Modelling of Residual Aluminum in Water Treatment Process. In: Šnorek, M., Buk, Z., Čepek, M. and Drchal, J. (eds.), *Proc. 7th EUROSIM Congress on Modelling and Simulation*, Vol.1: Book of Abstracts (p. 129) + Vol. 2: Full Papers [CD]. Czech Technical University in Prague.

Juntunen P., Liukkonen M., Pelo M., Lehtola M. and Hiltunen Y, 2010b. Modelling of Turbidity in Water Treatment Process. In: Juuso E. et al. (eds.), *Proc. SIMS 2010*, the 51st International Conference of Scandinavian Simulation Society.

Juuso E., Jokinen T., Ylikunnari J., and Leiviskä K., 2000. *Quality Forecasting Tool for Electronics Manufacturing*. Report A No. 12. University of Oulu, Control Engineering Laboratory.
Available via: http://herkules.oulu.fi/isbn9514275071/

Kadlec P., Gabrys B., and Strandt S., 2009. Data-driven Soft Sensors in the Process Industry. *Computers and Chemical Engineering*, vol. 33, pp. 795–814.

Kalteh A.M., Hjorth P., and Berndtsson R., 2008. Review of the Self-Organizing Map (SOM) Approach in Water Resources: Analysis, Modelling and Application. *Environmental Modelling and Software*, vol. 23, no. 7, pp. 835–845.

Kasslin M., Kangas J., and Simula O., 1992. Process State Monitoring Using Self-Organizing Maps. In: Alexander I. and Taylor J. (eds.), *Artificial Neural Networks 2*, Volume I, North-Holland, Amsterdam, Netherlands, pp. 1532–1534.

Khandpur R.S., 2005. *Printed Circuit Boards: Design, Fabrication, and Assembly*. McGraw-Hill, United States of America.

Kim J.H. and Cho H.S., 1995. Neural Network –Based Inspection of Solder Joints Using a Circular Illumination. *Image and Vision Computing*, vol. 13, no. 6, pp. 479–490.

Kim J.H., Cho H.S., and Kim S., 1996a. Pattern Classification of Solder Joint Images Using a Correlation Neural Network. *Engineering Applications of Artificial Intelligence*, vol. 9, no. 6, pp. 655–669.

Kim T.-H., Cho T.-H., Moon Y.-S., and Park S.-H., 1996b. An Automated Visual Inspection of Solder Joints Using 2D and 3D Features. In: *Proc. 3rd IEEE Workshop on Applications of Computer Vision (WACV '96)*, pp. 110–115.

Kim T.-H., Cho T.-H., Moon Y.S., and Park S.H., 1999. Visual Inspection System for the Classification of Solder Joints. *Pattern Recognition*, vol. 32, no. 4, pp. 565–575.

Kiviluoto K., 1996. Topology Preservation in Self-Organizing Maps. *IEEE International Conference on Neural Networks*, vol. 1, pp. 294–299.

Ko K.W. and Cho H.S., 2000. Solder Joints Inspection Using a Neural Network and Fuzzy Rule-Based Classification Method. *IEEE Transactions on Electronics Packaging Manufacturing*, vol. 23, no. 2, pp. 93–103.

Kohonen T., 1982. Self-Organized Formation of Topologically Correct Feature Maps. *Biological Cybernetics*, vol. 43, pp. 59–69.

Kohonen T., 1986. *Learning Vector Quantization for Pattern Recognition*. Report TKK-F-A601. Helsinki University of Technology, Espoo, Finland.

Kohonen T., 1990. The Self-Organizing Map. In: *Proc. of the IEEE*, vol. 78, no. 9, pp. 1464–1480.

Kohonen T., 1991. Self-Organizing Maps: Optimization Approaches. In: Kohonen T., Mäkisara K., Simula O., and Kangas J. (eds.), *Artificial Neural Networks*, pp. 981–990. Elsevier Science Publishers.

Kohonen T., Oja E., Simula O., Visa A., and Kangas J., 1996. Engineering Applications of the Self-Organizing Map. *Proceedings of the IEEE*, vol. 84, no. 10, pp. 1358–1384.

Kohonen T., 1999. Analysis of Processes and Large Data Sets by a Self-Organizing Method. *Proceedings of the Second International Conference on Intelligent Processing and Manufacturing of Materials*, IEEE Service Center, vol. 1, pp. 27–36.

Kohonen T., 2001. *Self-Organizing Maps*, 3rd edition. Springer-Verlag, Berlin Heidelberg.

Koji Y., Takatsu N., and Oh M., 1996. Visual Solder Inspection Using Neural Network. *Systems and Computers in Japan*, vol. 27, no. 1, pp. 92–100.

Kusiak A. and Kurasek C., 2001. Data Mining of Printed Circuit Board Defects, *IEEE Transactions on Robotics and Automation*, vol. 17, no. 2, pp. 191–196.

Kwak Y.H. and Anbari F.T., 2006. Benefits, Obstacles, and Future of Six Sigma Approach. *Technovation*, vol. 26, pp. 708–715.

Laine S., 2003. *Using visualization, variable selection and feature extraction to learn from industrial data*. Doctoral thesis, Helsinki University of Technology, Finland.

Lee N.-C., 2002. *Reflow Soldering Processes and Troubleshooting: SMT, BGA, CSP and Flip Chip Technologies*. Newnes, USA.

Li Y., Mahajan R.L., and Tong J., 1994. Design Factors and Their Effect on PCB Assembly Yield: Statistical and Neural Network Predictive Models. *IEEE Transactions on Components, Packaging, and Manufacturing Technology – Part A*, vol. 17, no. 2, pp. 183–191.

Lim T.E., 1989. Optimisation of Wave-Soldering Variables Using Factorial Experiment. *Quality Assurance*, vol. 15, no. 1, pp. 14–16.

Lim T.E., 1990. Quality Improvement Using Experimental Design. *International Journal of Quality & Reliability Management*, vol. 7, no. 1, pp. 70–76.

Lin Y.-H., Deng W.-J., Shie J.-R., and Yang Y.-K., 2007. Optimization of Reflow Soldering Process for BGA Packages by Artificial Neural Network. *Microelectronics International*, vol. 24, no.2, pp. 64–70.

Little R.J.A and Rubin D.B., 1987. *Statistical Analysis with Missing Data*. John Wiley & Sons, New York.

Liu H. and Motoda H. (Eds.), 2008. *Computational Methods of Feature Selection*. Chapman & Hall, United States of America.

Liu S., Ume I.C., and Achari A., 2004. Defects Pattern Recognition for Flip-Chip Solder Joint Quality Inspection With Laser Ultrasound and Interferometer. *IEEE Transactions on Electronics Packaging Manufacturing*, vol. 27, no. 1, pp. 59–66.

Liukkonen M., Havia E., Leinonen H., and Hiltunen Y., 2007. A SOM-based Approach for Analysing and Modelling a Wave Soldering Process. In: Jämsä-Jounela, S.-L. (ed.), *Proc. 14th Nordic Process Control Workshop*, pp. 126–129. Helsinki University of Technology, Laboratory of Process Control and Automation 12. Multiprint Oy, Espoo.

Liukkonen M., Hiltunen T., Havia E., Leinonen H., and Hiltunen Y., 2008. Selecting Variables for Quality Predictions in Wave Soldering by Using Multi-Layer Perceptrons. In: Tuominen A., Kantola J., Suominen A. and Hyrynsalmi S. (eds.), *Proc. NEXT 2008, the Fifth International New*

Exploratory Technologies Conference, pp. 373–382. TUCS General Publication, no. 50, Uniprint.

Liukkonen M., Havia E., Leinonen H., and Hiltunen Y., 2009a. Application of Self-Organizing Maps in Analysis of Wave Soldering Process. *Expert Systems with Applications*, vol. 36, no. 3, pp. 4604–4609.

Liukkonen M., Hiltunen T., Havia E., Leinonen H., and Hiltunen Y., Modeling of Soldering Quality by Using Artificial Neural Networks, 2009b. *IEEE Transactions on Electronics Packaging Manufacturing*, vol. 32, no. 2, pp. 89–96.

Liukkonen M., Heikkinen M., Hiltunen T., Hälikkä E., Kuivalainen R., and Hiltunen Y., 2009c. Modeling of Process States by Using Artificial Neural Networks in a Fluidized Bed Energy Plant. In: Troch I. and Breitenecker F. (eds.), *Proc. MATHMOD 09 VIENNA*, Full Papers Volume [CD], pp. 397–402. Argesim Report, no. 35. ARGESIM Publishing House, Vienna.

Liukkonen M., Heikkinen M., Hälikkä E., Kuivalainen R., and Hiltunen Y., 2009d. Emission Analysis of a Fluidized Bed Boiler by Using Self-Organizing Maps. In: Kolehmainen M., Toivanen P., and Beliczynski B. (eds.), *Lecture Notes in Computer Science: Proc. ICANNGA'09, International Conference on Adaptive and Natural Computing Algorithms*, vol. LNCS 5495, pp. 119–129. Springer-Verlag Berlin Heidelberg.

Liukkonen M., Heikkinen M., Hälikkä E., Hiltunen T., and Hiltunen Y., 2009e. Self-Organizing Maps for Analysis of Process Emissions in Fluidized Bed Energy Plant. In: Elmegaard B., Veje C., Nielsen M.P., and Mølbak T. (eds.), *Proc. SIMS 50: Modeling of Energy Technology* [CD-ROM], pp. 237–243. Technical University of Denmark.

Liukkonen M., Havia E., Leinonen H., and Hiltunen Y., 2010a. Quality-oriented Optimization of Wave Soldering Process by Using Self-Organizing Maps. *Applied Soft Computing*, Vol. 11, No. 1, pp. 214–220.

Liukkonen M., Havia E., Leinonen H., and Hiltunen Y., 2010b. Expert System for Analysis of Quality in Production of Electronics. *Expert Systems with Applications*, manuscript submitted for publication.

Liukkonen M., Hälikkä E., Hiltunen T., and Hiltunen Y., 2010c. Adaptive Soft Sensor for Fluidized Bed Quality. In: *Proc. Impacts of Fuel Quality on Power Production & Environment Conference* [CD-ROM], paper 19. Electric Power Research Institute (EPRI).

Liukkonen M., Hiltunen T., Hälikkä E., and Hiltunen Y., 2010d. Adaptive Approaches for Emission Modeling in Circulating Fluidized Beds. *International Journal of Computer Science & Emerging Technologies* (in press).

Liukkonen M., Hälikkä E., Kuivalainen R. and Hiltunen Y., 2010e. Modeling of Nitrogen Oxide Emissions in Fluidized Bed Combustion Using Artificial Neural Networks. *International Journal of Data Engineering*, Vol. 1, No. 2, pp. 26–35.

Liukkonen M., Heikkinen M., Hälikkä E., Hiltunen T. and Hiltunen Y., 2010f. Analysis of Flue Gas Emission Data from Fluidized Bed Combustion Using Self-Organizing Maps. *Applied Computational Intelligence and Soft Computing*, Vol. 2010, Article ID 932467. doi:10.1155/2010/932467.

Liukkonen M., Heikkinen M., Hiltunen T., Hälikkä E., Kuivalainen R. and Hiltunen Y., 2010g. Artificial Neural Networks for Analysis of Process States in Fluidized Bed Combustion. *Energy* (in press).

Lu S.-L., Zhang X.-M., and Kuang Y.-C., 2008. Neural Network –Based Inspecting Method of PCB Solder Joint. *Journal of South China University of Technology (Natural Science)*, vol. 36, no. 5, pp. 135–139.

Lulu M. and Rao R.L., 1990. Parameter Design for a Wave Solder Process. *Quality Engineering*, vol. 2, no. 3, pp. 301–318.

MacQueen J., 1967. Some Methods for Classification and Analysis of Multivariate Observations. *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, Volume I: Statistics. University of California Press.

Malave C.O. and Sastri T., 1992. Prediction of Wave Solder Machine Parameters Based on Printed Circuit Board Design Characteristics, In: Dagli C.H., Burke L.I., and Shin Y.C. (eds.), *Intelligent Engineering Systems Through Artificial Neural Networks*, vol. 2. ASME Press, New York, pp. 833–838.

Manko H.H., 2001. *Solders and Soldering – Materials, Design, Production, and Analysis for Reliable Bonding*, 4th edition. McGraw-Hill, United States of America.

Meireles M.R.G., Almeida P.E.M., and Simões M.G., 2003. A Comprehensive Review for Industrial Applicability of Artificial Neural Networks. *IEEE Transactions on Industrial Electronics*, vol. 50, no. 3, pp. 585–601.

Mendez R., Moreno M., Soto G., Herrera J., and Hamilton C., 2008. Design for Manufacturability in the Lead Free Wave Solder Process. *In: Proc. IPC APEX Conference 2008*.

Mesenbrink P., Lu J.-C., McKenzie R., and Taheri J., 1994. Characterization and Optimization of a Wave-Soldering Process. *Journal of the American Statistical Association*, vol. 89, no. 428, pp. 1209–1217.

Meyer S., Wohlrabe H., and Wolter K.-J., 2009. Data Mining in Electronics Packaging. *Proc. 32nd International Spring Seminar on Electronics Technology*, pp.1–7.

Mitchell T.M., 1997. *Machine Learning*. McGraw-Hill, United States of America.

Moganti M., Ercal F., Dagli C.H., and Tsunekawa S., 1996. Automatic PCB Inspection Algorithms: A Survey. *Computer Vision and Image Understanding*, vol. 63, no. 2, pp. 287–313.

Moller M.F., 1993. A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. *Neural Networks*, vol. 6, pp. 525–533.

Morris J. and Szymanowski R., 2009. Effect of Contact Time on Lead-Free Wave Soldering. *Dataweek – Electronics and Communications Technology*, Febr. 18, 2009.

Mosier C.I., 1951. Problems and Designs of Cross-Validation. *Educational and Psychological Measurement*, vol. 11, pp. 5–11.

Neubauer C., 1997. Intelligent X-ray Inspection for Quality Control of Solder Joints. *IEEE Transactions on Components, Packaging & Manufacturing Technology – Part C: Manufacturing*, vol. 20, no. 2, pp. 111–120.

Noble P.J.W., 1989. *Printed Circuit Board Assembly*. Halsted Press, New York.

Oja M., Kaski S., and Kohonen T., 2002. Bibliography of Self-Organizing Map (SOM) papers: 1998-2001 Addendum. *Neural Computing Surveys*, vol. 3, pp. 1–156.

Ong. T.Y., Samad Z., and Ratnam M.M., 2008. Solder Joint Inspection With Multi-Angle Imaging and an Artificial Neural Network. *The International Journal of Advanced Manufacturing Technology*, vol. 38, pp. 455–462.

Pal N.R. and Pal S., 2002. Editorial: Computational Intelligence for Pattern Recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no. 7, pp. 773–779.

Pande P. and Holpp L., 2002. *What Is Six Sigma?* McGraw-Hill, United States of America.

Pascoe G., 1982. Fault Finding in the Wave Soldering Process. *Circuit World*, vol. 8, no. 2, pp. 31–33.

Pawlak Z., 1982. Rough Sets. *International Journal of Computer and Information Sciences*, vol. 11, no. 5, pp. 341–356.

Pierce B.L., Shelton D.J., Longbotham H.G., Baddipudi S., and Yan P., 1994. Automated Inspection of Through Hole Solder Joints Utilizing X-ray Imaging. *IEEE Aerospace and Electronics Systems Magazine*, vol. 9, no. 2, pp. 28–32.

Poole D., Mackworth A., and Goebel R., 1998. *Computational Intelligence: A Logical Approach*. Oxford University Press, New York.

Prasad P. and Fitzsimmons D., 1984. Troubleshooting Wave Soldering Problems with Statistical Quality Control (SQC), *Circuit World*, vol. 10, no. 4, pp. 13–18.

Pupin S. and Resadi M., 1994. Solder Joints Inspection Machine VISA: Advanced Vision Inspection System. *Automazione e Strumentazione*, vol. 42, no. 11, pp. 107–112.

Quinlan J.R., 1986. Induction of Decision Trees. *Machine Learning*, vol. 1, pp. 81–106.

Randhawa S.U., Barton Jr. W.J., and Faruqui S., 1986. Wavesolder assistant: An Expert System to Aid Troubleshooting of the Wave Soldering Process. *Computers and Industrial Engineering*, vol. 10, no. 4, pp. 325–334.

Rawlings J.O., Pantula S.G., and Dickey D.A., 1998. Applied Regression Analysis: A Research Tool, 2nd edition. Springer-Verlag, New York.

Reed R.D. and Marks II R.J., 1999. *Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks*. MIT Press, Cambridge, MA.

Riedmiller M. and Braun H., 1993. A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm. In: Ruspini H. (ed.), *Proc. IEEE International Conference on Neural Networks (ICNN)*, San Francisco, pp. 586–591.

Riedmiller M., 1994. Advanced Supervised Learning in Multi-layer Perceptrons: From Backpropagation to Adaptive Learning Algorithms. *Computer Standards and Interfaces*, vol. 16, pp. 265–278.

Ritter H., Martinetz T., and Schulten K., 1992. *Neural Computation and Self-Organizing Maps: An Introduction*. Addison-Wesley, Reading, MA.

Roh Y.J., Park W.S., and Cho H., 2003. Correcting Image Distortion in the X-ray Digital Tomosynthesis System for PCB Solder Joint Inspection. *Image and Vision Computing*, vol. 21, pp. 1063–1075.

Rumelhart D.E., Durbin R., Golden R., and Cahuvin Y., 1995. Backpropagation: The Basic Theory. In: Chauvin Y. and Rumelhart D.E. (eds.), *Backpropagation: Theory, Architectures, and Applications*, Lawrence Erlbaum Associates, Hillsdale, New Jersey, pp. 1–34.

Ryu Y.K. and Cho H.S., 1997. A Neural Network Approach to Extended Gaussian Image Based Solder Joint Inspection. *Mechatronics*, vol. 7, no. 2, pp. 159–184.

Samad T. and Harp S., 1992. Self-Organization with Partial Data. *Network: Computation in Neural Systems*, vol. 3, no. 2, pp. 205–212.

Samson D. and Terziovski M., 1999. The Relationship Between Total Quality Management Practices and Operational Performance. *Journal of Operations Management*, vol. 17, no. 4, pp. 393–409.

Sankaran V., Embrechts M.J., Harsson L.-E., Kraft R.P., and Millard D.L., 1995. Applications of Backpropagation in Printed Wiring Board Inspection. In: Dagli C., Akay M., Chen C.L.P., Fernandez B.R., and Ghosh J., *Intelligent Engineering Systems Through Artificial Neural Networks*, Vol. 5: Fuzzy Logic and Evolutionary Programming. ASME Press, New York, pp. 931–936.

Sankaran V., Chartrand B., Millard D.L., Embrechts M.J., and Kraft R.P., 1998. Automated Inspection of Solder Joints – A Neural Network Approach. *European Journal of Mechanical and Environmental Engineering*, vol. 43, no. 3, pp. 129–135.

Santos D.L., Raj R.B., Lane M., Sissenstein D., and Testani M., 1997. Defect Reduction in PCB Contract Manufacturing Operations. *Computers & Industrial Engineering*, vol. 33, no. 1–2, pp. 381–384.

Sauer W., Oppermann M., Weigert T., Werner S., Wohlrabe H., Wolter K.-J., and Zerna T. (eds.), 2006. *Electronics Process Technology: Production Modelling, Simulation and Optimisation*. Springer-Verlag, London, U.K.

Schafer J.L., 1997. Analysis of Incomplete Multivariate Data. *Monographs on Statistics and Applied Probability*, vol. 72. Chapman & Hall, London.

Scheuhing R.B. and Cascini R., 1990. Wave Solder Thermal Profiling Using Multivariate Linear Regression Techniques. *Electronic Manufacturing*, vol. 36, no. 5, pp. 20–28.

Seiffert U. and Jain L.C. (eds.), 2002. *Self-Organizing Neural Networks: Recent Advances and Applications*. Springer Verlag, Heidelberg.

Shina S.G., 1989. Reducing Defects in a Printed Circuit Board Wave Soldering Process using Taguchi Methods. *Proc. NEPCON East'89*, pp. 205-224.

Sleeper A., 2005. *Design for Six Sigma Statistics: 59 Tools for Diagnosing and Solving Problems in DFFS Initiatives*. McGraw-Hill Professional Publishing, United States of America.

Smith J.A. and Whitehall F.B., 1997. *Optimizing Quality in Electronics Assembly: A Heretical Approach*. McGraw-Hill, United States of America.

Snitko C., 1998. Wave Solder Process Improvements Using Design of Experiments. In: *Proc. NEPCON West'98 Conference*, vol. 2, pp. 603–609.

Specht D.F., 1988. Probabilistic Neural Networks for Classification, Mapping, or Associative Memory. In: *Proc. IEEE International Conference on Neural Networks*, pp. 525–532.

Sterritt J.R., 1991. Optimizing the Wave Soldering Process. In: Lau J.H. (ed.), *Solder Joint Reliability: Theory and Applications*, Van Nostrand Reinhold, New York, pp. 117–142.

Stone M., 1974. Cross-validatory Choice and Assessment of Statistical Predictions. *Journal of the Royal Statistical Society*, vol. 36, pp. 111–147.

Stuart M., Mullins E., and Drew E., 1996. Statistical Quality Control and Improvement. *European Journal of Operational Research*, vol. 88, no. 2, pp. 203–214.

Tino de U.M., 2008. Analyzing Pb-free defects using partial DoE. *Circuits Assembly*, vol. 19, no. 12, pp. 34–35.

Tokutaka H., 1997. Condensed Review of SOM and LVQ Research in Japan. In: *Proceedings of WSOM'97*, pp. 322–329. Helsinki University of Technology, Neural Networks Research Centre.

Tollenaere T., 1990. Supersab: Fast Adaptive Backpropagation with Good Scaling Properties. *Neural Networks*, vol. 3, no. 5, pp. 561–573.

Tripathy B.K., 2009. Rough Sets on Fuzzy Approximation Spaces and Intuitionistic Fuzzy Approximation Spaces. In: Abraham A., Falcón R., and Bello R. (eds.), Rough Set Theory: A True Landmark in Data Analysis. Studies in Computational Intelligence 174, Springer-Verlag Berlin Heidelberg.

Tsai T.-N. and Yang T., 2005a. A Neuro-computing Approach to the Thermal Profile Control of the Second-side Reflow Process in Surface Mount Assembly. *Journal of Manufacturing Technology Management*, vol. 16, no. 3, pp. 343–359.

Tsai, T.-N., Yang, T., and Hou, P.-A., 2005b. Neurofuzzy Modeling of the Reflow Thermal Profile for Surface Mount Assembly. *International Journal of Systems Science*, vol. 36, no. 2, pp. 89–101.

Tsai, T.-N., 2005c. Development of an Integrated Reflow Soldering Control System Using Incremental Hybrid Process Knowledge. *Expert Systems with Applications*, vol. 28, pp. 681–692.

Tsaptsinos D., 1995. Back-propagation and Its Variations. In: Bulsari A.B. (ed.), *Neural Networks for Chemical Engineers*, pp. 33–75. Elsevier Publishing Company, Amsterdam.

Tsenev V. and Marinov L., 2004. Design of Experiments (DOE) of Automatic Soldering Process "Wave Soldering". In: *Proc. 27th International Spring Seminar on Electronics Technology: Meeting the Challenges of Electronics Technology Progress*, vol. 2, pp. 338–341.

Tseng T.-L., Jothishankar M.C., and Wu T., 2004. Quality Control Problem in Printed Circuit Board Manufacturing – An Extended Rough Set Theory Approach. *Journal of Manufacturing Systems*, vol. 23, no. 1, pp. 56–72.

Tsironis L., Bilalis N., and Moustakis V., 2005. Using Machine Learning to Support Quality Management – Framework and Experimental Investigation. *The TQM Magazine*, vol. 17, no. 3, pp. 237–248.

Twomey J.M. and Smith A.E., 1996. Artificial Neural Network Approach to the Control of a Wave Soldering Process. *Intelligent Engineering Systems Through Artificial Neural Networks*, vol. 6, pp. 889–894.

Ultsch A. and Siemon H., 1989. *Exploratory data analysis: Using Kohonen's topology preserving maps*. Technical Report 329, University of Dortmund, Germany.

Uriarte E.A. and Martín F.D., 2006. Topology Preservation in SOM. *Proceedings of World Academy of Science, Engineering and Technology*, vol. 15, pp. 187–190.

Vapnik V.N., 1998. *Statistical Learning Theory*. John Wiley & Sons, United States of America.

Venkatasubramanian V., Rengaswamy M., Yin K., and Kavuri S.N., 2003. A Review of Process Fault Detection and Diagnosis: Part I: Quantitative Model-based Methods. *Computers & Chemical Engineering*, vol. 27, no. 3, pp. 293–311.

Vermasvuori M., Endén P., Haavisto S., and Jämsä-Jounela S.-L., 2002. The Use of Kohonen Self-Organizing Maps in Process Monitoring. In: *Proc. First International Symposium on Intelligent Systems*, vol. 3, pp. 2–7.

Vesanto J., 1999a. SOM-based Data Visualization Methods. *Intelligent Data Analysis*, vol. 2, no. 3, pp. 111–126.

Vesanto J., Himberg J., Alhoniemi E., and Parhankangas J., 1999b. Self-Organizing Map in Matlab: the SOM toolbox. *Proceedings of the Matlab DSP Conference*, pp. 35–40.
Tool box available via: http://www.cis.hut.fi/projects/somtoolbox/

Vesanto J. and Alhoniemi E., 2000. Clustering of the Self-Organizing Map. *IEEE Transactions on Neural Networks*, vol. 11, no. 3, pp. 586–600.

Vesanto J., 2002. *Data Exploration Process Based on the Self-Organizing Map*. Doctoral thesis, Helsinki University of Technology, Finland.

Available via: http://lib.hut.fi/Diss/2002/isbn9512258978/

Voci F., Eiho S., and Sugimoto N., 2002. Fuzzy Interference Filter and Morphological Operators for Short Circuits Detection in Printed Circuit Board. In: *Proceedings of the International Symposium on Industrial Electronics*, pp. 672–677.

Wang K., 2007. Applying Data Mining to Manufacturing: the Nature and Implications. *Journal of Intelligent Manufacturing*, vol. 18, pp. 487–495.

Wei X. and Frantti T., 2002. Adaptive Real-Time Fuzzy X-ray Solder Joint Inspection System. *Journal of Manufacturing Systems*, vol. 21, no. 2, pp. 111–125.

Werbos P.J., 1974. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Doctoral thesis, Harvard University, Campbridge (MA).

Werbos P.J., 1994. *The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting*. John Wiley & Sons, New York.

Whitney A.W., 1971. A Direct Method of Nonparametric Measurement Selection. *IEEE Transactions on Computers*, vol. 20, no. 9, pp. 1100–1103.

Williams M. and Raaijmakers P., 1999. Applying Taguchi Analysis to Wave Soldering. *Circuits Assembly*, vol. 10, no. 8, pp. 58–64.

Willmott C.J., 1981. On the Validation of Models. *Physical Geography*, vol. 2, pp. 184–194.

Willmott C.J., 1982. Some Comments on the Evaluation of Model Performance. *Bulletin American Meteorological Society*, vol. 63, no. 11, pp. 1309–1313.

Yang K. and El-Haik B.S., 2009. *Design for Six Sigma: A Roadmap to Product Development*, 2nd ed. McGraw-Hill, United States of America.

Yang K. and Trewn J., 2004. *Multivariate Statistical Methods in Quality Management*. McGraw-Hill, United States of America.

Yang T. and Tsai T.-N., 2002. Modeling and Implementation of a Neurofuzzy System for Surface Mount Assembly Defect Prediction and Control. *IIE Transactions*, vol. 34, pp. 637–646.

Yun T.S., Sim K.J., and Kim H.J., 2000. Support Vector Machine –based Inspection of solder joints Using Circular Illumination. *Electronics Letters*, vol. 36, no. 11, pp. 949–951.

Zadeh L.A., 1965. Fuzzy Sets. *Information and Control*, vol. 8, pp. 338–353.

Zhou J.-C., Xiao X.-Q., En Y.-F., Chen N., and Wang X.-Z., 2008. Thermo-Mechanical Fatigue Reliability Optimization of PBGA Solder Joints Based on ANN-PSO. *Journal of Central South University of Technology*, English edition, vol. 15, no. 5, pp. 689–693.

Publications of the University of Eastern Finland
Dissertations in Forestry and Natural Sciences

1. Hassinen, Minna (2010) Thermal responses of inward and delayed rectifier potassium channels in fish cardiac myocytes
2. Huang, Xiang (2010) Water quality in the Tibetan Plateau : Chemical evaluation of the headwaters of four major Asian rivers
3. Ukishima, Masayuki (2010) Prediction and Evaluation of Halftone Print Quality Based on Microscopic Measurement
4. Tiilikainen, Raisa (2010) The effects of selective harvesting on life history traits of moose Alces alces
5. Lipponen, Pasi (2010) Elasto-Plastic Modelling Approach for Paper Cockling and Curling Phenomena
6. Pradhan, Surendra (2010) Yield and quality of vegetables fertilized with human urine and wood ash
7. Korhonen, Jenni (2010) Antibiotic Resistance of Lactic Acid Bacteria
8. Poutiainen, Hannu (2010) Tools for improved efficiency and control in wastewater treatment
9. Piipponen, Samuli (2010) Kinematic analysis of mechanisms using computational algebraic geometry
10. Matikka, Hanna (2010) The effect of metallic implants on the RF energy absorption and temperature changes in tissues of head : A numerical study
11. Xiaoguang, Qi (2010) Studies on value sharing for derivatives and difference operators of meromorphic functions
12. Laukkanen, Janne (2010) Fabrication of metallic micro- and nanostructures for optical solutions
13. Vilhunen, Sari (2010) UVC irradiation based water treatment : A study of UV light emitting diodes, atomic layer deposited TiO2 and novel applications
14. Myllymaa, Sami (2010) Novel micro- and nanotechnological approaches for improving the performance of implantable biomedical devices
15. Myllymaa, Katja (2010) Novel carbon coatings and surface texturing for improving the biological response of orthopedic implant materials
16. Jantunen Paula (2010) Role of Sorption in the Ecological Risk Assessment of Xenobiotics
17. Yang, Congli (2010) Studies on warphi-Bloch functions and different type operators between Qk spaces
18. Lähivaara, Timo (2010) Discontinuous Galerkin method for time-domain wave problems
19. Matilainen, Juha (2010) The regulation of inflammatory cytokines by vitamin D
20. Jetsu, Tuija (2010) Modeling color vision
21. Patrick Faubert (2010) Responses of Non-Methane Biogenic Volatile Organic Compound Emissions to Climate Change in Boreal and Subarctic Ecosystems
22. Hyvärinen, Heikki (2010) Studies on Partial Coherence and Electromagnetic Optics of Metamaterials
23. Karvinen, Petri (2010) Applications of resonant waveguide gratings

MIKA LIUKKONEN

Intelligent Methods in the Electronics Industry

Quality Analysis of Automated Soldering

Methods associated with computational intelligence such as artificial neural networks, fuzzy logic and evolutionary computation are nowadays used widely in different industrial environments. This work documents an application of intelligent data-based modeling methods to quality analysis of electronics production. These methods benefit from the useful characteristics of computational intelligence, of which the key elements are an ability to learn from experience, self-organize and adapt in response to dynamically changing conditions, and a considerable potential in solving real world problems. The results show that they provide an efficient way of analyzing quality in the electronics industry.

UNIVERSITY OF
EASTERN FINLAND

PUBLICATIONS OF THE UNIVERSITY OF EASTERN FINLAND
Dissertations in Forestry and Natural Sciences

ISBN 978-952-61-0281-8