Produktion und Logistik

İhsan Onur Yılmaz

Development and Evaluation of Setup Strategies in Printed Circuit Board Assembly

GABLER EDITION WISSENSCHAFT

İhsan Onur Yılmaz

Development and Evaluation of Setup Strategies in Printed Circuit Board Assembly

GABLER EDITION WISSENSCHAFT

Produktion und Logistik

Kontakt: *Professor Dr. Hans-Otto Günther, Technische Universität Berlin, FG BWL – Produktionsmanagement, Wilmersdorfer Str. 148, 10585 Berlin*

Diese Reihe dient der Veröffentlichung neuer Forschungsergebnisse auf den Gebieten der Produktion und Logistik. Aufgenommen werden vor allem herausragende quantitativ orientierte Dissertationen und Habilitationsschriften. Die Publikationen vermitteln innovative Beiträge zur Lösung praktischer Anwendungsprobleme der Produktion und Logistik unter Einsatz quantitativer Methoden und moderner Informationstechnologie.

İhsan Onur Yılmaz

Development and Evaluation of Setup Strategies in Printed Circuit Board Assembly

With a foreword by Prof. Dr. Hans-Otto Günther

GABLER EDITION WISSENSCHAFT

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Dissertation Technische Universität Berlin, 2007

D 83

1st Edition 2008

Editorial Office: Frauke Schindler / Jutta Hinrichsen

Gabler is part of the specialist publishing group Springer Science+Business Media.
www.gabler.de

Cover design: Regine Zimmer, Dipl.-Designerin, Frankfurt/Main
Printed on acid-free paper
Printed in Germany

ISBN 978-3-8349-1200-8

Foreword

During the last decade the electronics industry faced growth rates considerably higher than average due to innovative products and the comprehensive use of electronic control devices for almost all types of technical products. Hence, printed circuit board (PCB) assembly can be seen as one of the most dynamic branches of the electronics industry. In modern electronics manufacturing, highly automated assembly systems are used to mount the electronic components at pre-specified locations onto the PCBs. Because of the tremendous complexity of the process technology and huge capital investments, highly sophisticated planning and control strategies are needed for the operation of the assembly plants.

In the past both industry and academia became heavily involved in the development of tools and planning concepts which help to master the huge variety of customized electronic products. While previous research work has primarily been concerned with a high-volume, low-variety production environment, there is now an ongoing trend towards the use of highly flexible manufacturing equipment and the production of printed circuit boards in a mixed sequence with only small lot sizes.

In this study the author develops practical setup strategies for the assembly of PCBs especially in a medium-variety production environment. At the core of his principle approach are the identification of similarities between different types of PCBs and the generation of PCB clusters upon which group setup strategies are based. The developed setup strategies are also innovative in the sense that they integrate the optimization of detailed machine operations. This integration has not been achieved in the classical approaches which primarily rely on statistical clustering techniques.

The results and conclusions are of paramount importance for the development of planning and control software which accompanies the use of automated assembly machinery in industry. It is shown in a comprehensive numerical investigation that the group setup methodology developed by the author outperforms classical approaches known from the academic literature. The author specifically investigates so-called “collect-and-place” machines which have recently received increased popularity in industry. This study is indeed a new and creative contribution to solving complex planning and control problems arising in the PCB assembly industry.

Prof. Dr. Hans-Otto Günther

Acknowledgement

Electronics industry is one of the most important industrial sectors in the world. Rapid developments in electronics industry and globalization of the world markets demand for an increase of flexibility and efficiency in production. In addition, the size of the electronic products reduces with each innovation, which requires a high level of automation in order to assure the quality requirements. This study focuses on development and application of some advanced planning systems, which are essential for utilization of production facilities designed to respond to current market requirements.

The research presented in this study has been carried out during my occupation as a research assistant at the Department of Production Management at TU Berlin. I owe a debt of gratitude to my doctoral thesis supervisor Prof. Dr. Hans-Otto Günther for his valuable guidance in this study and his both academic and personal support during the previous years. I also thank Prof. Dr. Werner Jammernegg for his engagement as the second assessor.

Special thanks are also due to Prof. Dr. Martin Grunow for his contribution and valuable collaboration in the field of electronics assembly. I would also like to acknowledge the engagement of my colleagues Dr. Matthias Lehmann, Dr. Ulf Neuhaus, Dr. Gang Yang and Markus Meiler in our academic discussions. My special thanks go to Hanni Just and other student assistants who managed to create a familial atmosphere in the department. I also thank to my master thesis students who enabled me to carry out research in different areas of electronics assembly.

I am grateful to my parents for all the support I have got in my education. Without them I would not get the chance to be here. My most bountiful gratitude goes to my wife for her tolerance and forbearance, and enabling me a perfect working environment without any worries.

İhsan Onur Yılmaz

Table of Contents

List of Figures

List of Tables

List of Abbreviations

3G	Third generation
a_i	Number of assigned nozzles for nozzle type i
BAP	Bin assignment problem
CAD	Computer-aided design
COB	Chip-on-board
CPM	Critical path method
DCA	Direct chip attach
E(X)	Expected value of X
EDP	Electronic data processing
FMS	Flexible manufacturing system
GA	Genetic algorithm
GDP	Gross domestic product
GGA	Grouping genetic algorithm
GSU	Group setup
GT	Group technology
HGA	Hybrid genetic algorithm
IC	Integrated circuit
IP	Integer programming
KCNS	Keep component needed soonest
KTNS	Keep tool needed soonest
LP	Linear programming

MaST	Maximum spanning tree
MIP	Mixed-integer programming
ms	Makespan
MST	Minimal spanning tree
MUS	Makespan with unique setups
P	Number of total placement operations for observed PCB
P'	Remaining number of placement operations
PAR	Partition-and-repeat
PCB	Printed circuit board
PDA	Personal digital assistant
PERT	Program evaluation and review technique
p_i	Number of placement operations requiring nozzle type i
QAP	Quadratic assignment problem
R	Number of segments on placement head
R', R''	Remaining capacity of the placement head
RAM	Random access memory
RF	Radio frequency
RoHS	Restriction of certain hazardous substances
RPP	Rural postman problem
S_1, S_2	Partition sets for nozzles
SA	Simulated annealing
SDS	Sequence-dependent scheduling

S_{ij}	Savings value calculated for operations *i* and *j*
SIM_{pq}	Similarity between clusters *p* and *q*
SMD	Surface-mount device
SMT	Surface-mount technology
SOIC	Small outline integrated circuit
SOJ	Small outline J
SOT	Small outline transistor
SPP	Shortest path problem
TAB	Tape automated bonding
THD	Through-hole device
THT	Through-hole technology
$TMAG_{ij}$	Travel time between feeder positions *i* and *j* on the magazine
$TPCB_{ij}$	Travel time between locations *i* and *j* on the PCB
TR	Rotational cycle time of the placement head
TS	Tabu search
TSP	Traveling salesman problem
V_x, V_y	Velocity of the placement head in the x- and y-direction, respectively
W_{ij}	Weight of arc (*i*,*j*), i.e. travel time of the placement head between two locations on the board

WIP	Work-in-process
X_i	X-coordinate of the feeder location for component *i* in the magazine of the placement machine
x_i, y_i	X- and y-coordinate of the placement location for component *i* on the PCB, respectively

1. Introduction

In the last decades, electronic control units have become essential parts of many modern industrial and commercial products. Since 2000, electronics industry is the biggest industrial sector in the world ahead of the automobile industry.[1] Electronics is integrated into almost every product we use in our daily life – from the alarm clock we use to wake up every morning to the remote control for watching TV in the evening. Unfortunately, "people on the streets", although they use over more than 100 of them a day, are still not aware of the product called printed circuit board (PCB).

A long term prediction for the future of the electronics industry is complicated due to a high number of different influencing factors. One of them is the buying power of consumers in Europe and the USA, which relies on the development of the gross domestic product (GDP) of these countries. Currency exchange rates play another important role on the cost and the selling price of electronic equipments. Arbitrary factors like development of new software which demand new powerful hardware or personal needs for possessing the newest fashionable mobile phone are also not negligible. Legal regulations, e.g. new emission limits for automobiles and unpredictable incidences like 9/11, have a direct affect on the future volumes and the market structure.[2]

Similar to other industry sectors, the trend of globalization has also changed the whole production scheme in electronics industry (see figure 1.1). The recession in 2001 and 2002 will be remembered as a turning point and the electronics industry is expected to grow to be 2.5 times as big as it is today over 20 years. It is projected that China (34%) and other Asian countries (17%) (e.g. Taiwan, Korea, and India) will be the biggest manufacturers of electronics in 2020. This happens mainly at the expense of Europe and Japan, which have begun to outsource and move manufacturing plants to Asia, keeping only the high-end production inland. Important criteria, e.g. political stability, subventions, transport connections, currency exchange rates and educational level in order to get qualified personnel are important decision factors which will decide on the direction and volume of the global movement.[3]

---

1 Cf. Gasch (2002).
2 Cf. Gasch (2002).
3 Cf. Gasch (2002).

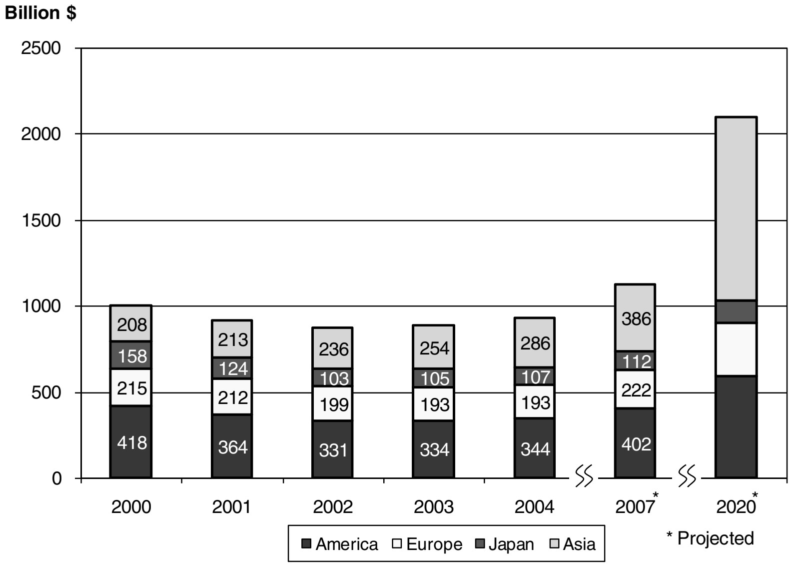


Figure 1.1: Changing regional manufacturing profiles[4]

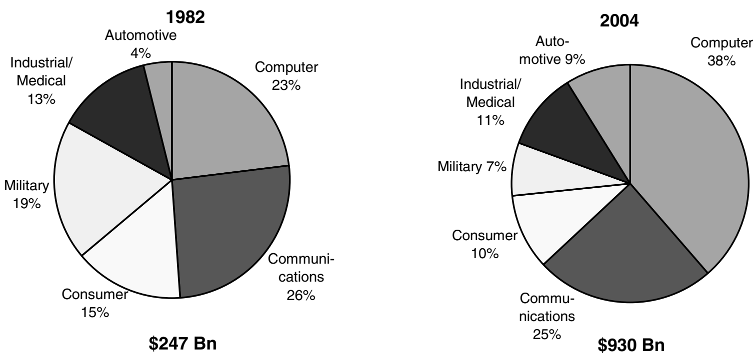


Figure 1.2: Growth of the global electronics industry market[5]

4 Based on the figures in Plonski (2004).

Figure 1.2 illustrates the growth of the global electronics industry. In 1982, the global market for electronic products was around $247 billions, which is approximately one fourth of the volume realized in year 2004. Due to the rapid developments in office and electronic data exchange systems in the previous decades, two third of the electronics industry is currently based on networked communications.[6] With the development of broadband and wireless communication systems and demand on more data storage and access equipments, the market for *computer* and *communication* equipment shows an inclining trend for the future. Precisely speaking, the market for computers comprising electronic data processing (EDP) and office equipments has increased tremendously from $56.81 billions in 1982 to $359 billions in 2004.

Digital audio and video equipments constitute the most significant group in *consumer* electronics. The market for this sector is expected to enlarge with next generation digital media, i.e. high definition video and recording systems. Applications of electronic equipment in *industrial* and *medical* sectors will be based more on integrated business solutions and control systems using cheapening sensors.

The applications for the *automobile* industry have gained much importance in the electronics industry. The volume of electronics produced for the automobile sector has increased in 22 years from $9.88 billions to $82 billions with an increase in total electronics market share from 4% to 9%. In automobile industry, many mechanical components are either controlled or replaced by electronic units. Stricter fuel economy and emission standards have motivated usage of electronics in power train and engine control systems. Greater emphasis on safety systems, and integration of information and entertainment devices are other main sources for usage of electronics in vehicles.

Military and *aerospace* industries, although have lost an important share in the electronics industry, also expect to grow in the oncoming years. Modern smart weapons and communication devices are equipped with complex electronic control systems. Wireless information and communication has also gained a lot of importance in military field. The usage of unmanned aerial vehicles with strike capabilities will drive the demand for more electronic devices in oncoming developments.

---

[5] Source: Plonski (2004).
[6] Cf. Plonski (2004).

Although electronic devices consist of many different components, PCBs are the core elements of each electronic product. Table 1.1 illustrates the rapid growth of the global PCB market which is expected to expand from $30.1 billions to $40 billions in only a five years period. The figures show that the EDP and office systems together with the communication equipment build up a significant share of the PCB market. Important is to realize that the usage of PCBs in packaging sector almost doubles its value.

Table 1.1: Changes in global PCB supply by application (in million $)[7]

Application	2002	2007*
EDP/Office	10,653	13,851
Communications	6,510	8,862
Consumer	4,590	5,248
Automotive	1,669	2,013
Industrial/Medical	2,255	2,773
Military/Aerospace	1,772	2,218
Packaging	2,639	5,138
Total	**30,088**	**40,103**

* projected

Table 1.2 shows the world PCB production, where real values until the end of 2006 are used to project the close future. Interestingly, the real figures from Nakahara (2007) present that the projection from Plonski (2004) for year 2007 (see table 1.1) is already passed beyond with $42.4 billions in 2005. This rapid increase is expected to continue and reach up to a level of $64.4 billions in 2009. Two main drivers for this unexpected positive growth in the last observation period of 2006 were the integrated circuit (IC) substrates and the cellular phones. During the market for mobile phones was 820 million units in 2005, nearly one billion mobile phone headsets are produced in 2006.

The high demand for flat screen TVs, mobile phones, digital cameras, games and various digital consumer products kept especially Japanese PCB manufacturers busy.[8] More than 95% of all mobile phone headsets sold in Japan are featuring 3G services requiring quite complex

7 Source: Plonski (2004).
8 Cf. Nakahara (2007).

boards. During Japan and Taiwan receive a good part of their growth from overseas investments, South Korea PCB manufacturers mainly served domestic customers. Countries in South East Asia benefited from high investments from Japanese producers. China has now almost 25% of the world market share with an output of $12.1 billions. However, 95% of this production volume is accomplished in foreign transplants.[9] North America had hardly any growth at all. The PCB business for the automotive sector is practically gone from North America, whereas military and medical electronics are still quite strong. Currently, more than 30% of the European PCB consumption is imported from Asia. With the exception of automotive boards, volume production in Europe is practically gone. According to Nakahara (2007), Asia will have 85% of PCBs manufactured in 2009 with a modest increase of its 82% enjoyed in 2006.

Table 1.2: World PCB production (in million $)[10]

Region	2005	Growth (%)	2006	Growth (%)	2007	Growth (%)	2008	Growth (%)	2009	Growth (%)
Total America	**4,692**	**0.0**	**4,718**	**0.6**	**4,803**	**1.8**	**4,938**	**2.8**	**5,051**	**2.2**
N. America	4,583	0.0	4,624	0.9	4,707	1.8	4,840	2.8	4,950	2.2
S. America	109	0.0	94	-0.1	96	2.3	98	2.4	101	2.4
Total Europe	**3,605**	**-7.9**	**3,660**	**1.5**	**3,697**	**1.0**	**3,748**	**1.4**	**3,786**	**1.0**
Germany	1,292	-6.1	1,402	8.5	1,458	4.0	1,524	4.5	1,577	3.5
Other Regions	430	-0.1	437	2.1	452	3.5	468	3.5	482	3.0
Total Asia Pacific	**33,675**	**11.2**	**39,738**	**18.0**	**43,969**	**10.6**	**49,421**	**12.4**	**55,107**	**11.5**
Japan	9,995	3.4	11,228	12.3	11,733	4.5	12,707	8.3	13,660	7.5
China	10,060	23.2	12,102	20.3	14,195	17.3	17,177	21.0	20,440	19.0
Taiwan	5,980	8.8	7,350	22.9	8,268	12.5	8,914	7.8	9,430	5.8
S. Korea	4,890	13.5	5,795	6.2	6,200	7.0	6,728	8.5	7,145	6.2
Other Asia	2,750	3.2	3,263	18.7	3,573	9.5	3,895	9.0	4,432	13.8
World Total	**42,402**	**4.8**	**48,553**	**14.5**	**52,921**	**9.0**	**58,575**	**10.7**	**64,426**	**10.0**
China Share (%)	23.7		24.9		26.8		29.3		31.7	

9 Cf. Nakahara (2007).
10 Source: Nakahara (2007).

Above explained advances in the electronics market have created a need for flexible production with high throughput rates. Decreasing prices and the globalization forces manufacturers to produce a diversity of products with low costs. Because of the high capital investment on automated electronics assembly lines, care must be taken to utilize the manufacturing equipment in the most economical manner.

By increasing usage of surface-mount devices and automation techniques in electronics assembly, researchers have focused on optimization problems in this field.[11] On one side, machine producers offer some software applications, which are based on basic methodologies to create solutions in a short period of time. On the other side, researchers create models and solutions, which are in many cases far away from representing the real-life situation, because of the high number of assumptions in simplifying the models which are usually *NP*-hard. The aim of this study is to propose some methodologies which represent solutions for real-case problems.

The remainder of this study is organized as follows. The technology of PCB manufacturing and details of the PCB assembly process are presented in chapter2. A categorization of surface-mount placement machines and their working principles are given in chapter 3. In chapter 4, a hierarchical decomposition of the planning problems is presented. The main focus of this chapter is the comparison of different setup strategies and discussing the advantage of using a group setup strategy against other strategies in the literature. In chapter 5, a comprehensive group setup strategy which employs different similarity measures (Jaccard, simple matching, and inclusion measure) and clustering techniques (conventional and inclusion-based) is presented. The novel group setup approach integrates machine-specific algorithms in each step of the agglomerative grouping process and comprises new aspects like determination of the actual makespan. The results of the detailed numerical investigation are presented in chapter 6. A conclusion of the presented study is given in chapter 7.

[11] Cf. section 2.2 for SMDs.

2. Technological Background

2.1 Printed Circuit Boards

PCBs are the key components in almost all electronic assemblies. The idea of the PCB originated from the need of placing components and devices on a non-conductive carrier and adding functional and electrical connections with conductive paths.[12] With the invention of PCBs, the former three-dimensional wiring of valves, coins and resistances has been replaced with a two-dimensional pattern on an insulating board.[13]

The history of the PCB begins with the end of the first quarter of the 20th century. The fundamentals of the PCB technology had originated from a patent taken by Cesar Parolini in 1925, but the idea of today's modern PCB was first created by Dr. Paul Eisler in 1930 and patented in 1943. The first applications of this concept delayed until the end of World War II when PCBs have been used in the USA for building electronically complex military devices like radars and missile controllers. The development of transistors by the late 1950s, which caused to a considerably smaller size and lower dissipation compared to thermionic tubes, and the usage of multi-layer boards made it possible to mount a variety of components and reduce the overall size of the equipments.[14]

A PCB is a substrate of a paper or glass fabric impregnated with a resin, commonly epoxy, phenolic or silicone.[15] It consists of one or more layers of metal conductors and insulating material which allow electronic components for being electronically interconnected and mechanically supported. The simplest form of PCB is the single-layer, single-sided board, which contains metalized conductors only on one side of the board. Greater levels of complexity and component density can be achieved by producing double-sided and multi-layered boards. In a double-sided assembly, the PCB is assembled with components on both sides of the substrate.

PCBs can be produced either in an additive or a subtractive process.[16] In an additive process, the conductive track (i.e. copper) is applied directly to the surface of the substrate. In a subtractive process, the copper foil is added first to the whole surface of the substrate. Next, the

[12] Cf. Bachmann et al. (1999), p. 51.
[13] Cf. Strauss (1994), p. 2.
[14] Cf. Sautter (1988), p. 11.
[15] List of substrates and their characteristics can be found in Brindley (1990), p. 15-16, Kear (1987), p. 32-33, and Prasad (1989), p. 120-126.
[16] Details of PCB production processes are given in Hanke (1994), chapter 4.

track pattern is defined and covered with an etch resist. Finally, an etchant is applied on the board removing the excess conductive material and leaving the required track. Applying the etch resist is referred to as *printing* and this is the reason why PCBs are actually called *printed* circuit boards.[17] Sometimes, both processes may be combined to produce PCBs with more than one layer of conductive track.[18]

2.2 Components

Components are inserted onto PCBs and connected with its designed circuit in order to facilitate PCBs with some functionality. These PCBs are also called as populated PCBs. PCB assembly technologies can be categorized into two main groups:

- the through-hole technology (THT) using through-hole devices (THDs), and
- the surface-mount technology (SMT) using surface-mount devices (SMDs).

Figure 2.1 illustrates the principles of both technologies listed above. After the invention of PCBs, THT has been applied for assembling resistors, capacitators, thermionic valves and three-legged transistors.[19] In this former type of PCB assembly, the leads of the components are plugged through the holes, and their protruding ends are then trimmed and soldered.

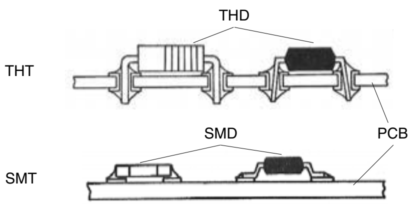


Figure 2.1: Assembly principles of THT and SMT[20]

[17] Cf. Brindley (1990), p. 22.
[18] Cf. Coombs (1988), p. 11.17 ff.
[19] Cf. Strauss (1994), p. 2.
[20] Source: Coombs (1988), p. 3.2.

The SMT has its roots in the 1950s.[21] The first SMDs – also called flatpacks – were used in high reliability military applications.[22] In the 1970s, with the emerging Japanese electronics industry, Japanese producers started using SMDs in consumer products. In SMT assembly, the components are mounted on the leads of the board circuits and soldered using a soldering paste. The details of the SMT assembly process are described in the next section.

In comparison to the THT, SMT assembly incorporates many advantages. SMDs are quite smaller than the THDs. This allows the designer to reduce the space by keeping the functionality or adding more functions without changing the dimensions. Additionally, SMT enables a two-sided assembly of the PCB, which gives the possibility to increase the integration rate. Hence savings up to 50% in PCB dimensions can be achieved compared to a THT design.[23] A high percentage of the THDs cannot be automated and must be manually assembled. In this respect, the SMT has big advantages in automation and shows a higher reliability. Additional costs caused by drilling, bending and lead cutting operations are omitted. The soldering process is simplified by using solder-reflow systems instead of wave soldering[24], which abolishes the need to worry about leads protruding on the other side of the substrate.[25] SMDs have shorter leads and soldering pads, which reduce parasitic inductances and capacitances. The lower mass of the SMDs provides shock and vibration resistance.

Today, almost all types of components are available as SMDs. SMDs are categorized as *passive* components, i.e. resistors, capacitors and inductors, and *active* components, i.e. ceramic and plastic chip carriers, small outline transistors (SOTs), small outline integrated circuits (SOICs), small outline J (SOJ) packages and fine pitch packages.[26] A logical consequence of increasing miniaturization, and thus increasing packaging density in fine-pitch SMDs is the increasing usage of bare chips on PCBs (the so-called chip-on-board (COB) technology). Hence, by adding connectors on the backside area of the bare chip, more connectors can be designed in smaller space. Bare chips are mounted either directly on the PCB (direct chip attach, DCA) or by placing them on subcarriers on the PCB (tape automated bonding, TAB). In DCA assembly, bare chips are either placed face up and connected functionally using wires

[21] Cf. Rowland and Belangia (1993), p. 2.
[22] Cf. Prasad (1989), p. 4.
[23] Cf. Sautter (1988), p. 14, Coombs (1988), p. 3.2, and Brindley (1990), p. 72.
[24] Soldering methods are explained in section 2.3.3.
[25] Cf. Gingsberg (1989), p. 5.
[26] Cf. Sautter (1988), chapter 5.

(chip-and-wire) or face down using soldering balls (flip chip).[27] Although SMDs made up 74.9% of the worldwide ICs in 2002 and bare dies only 11%, bare die technology shows an increasing trend and is expected to replace the ICs in PCB assembly field.[28]

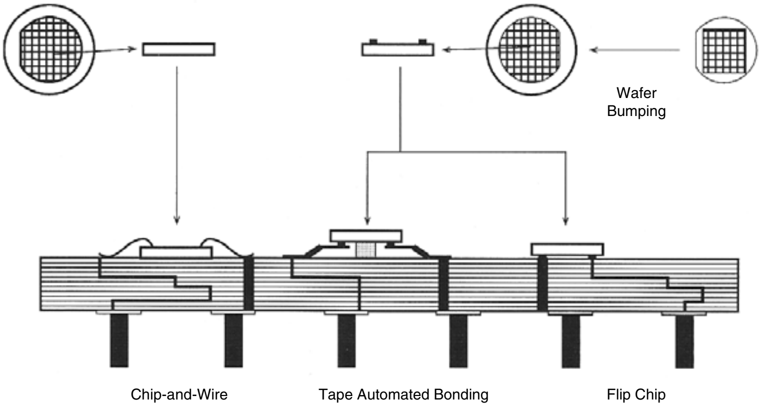


Figure 2.2: Applications of COB technology[29]

2.3 SMT Assembly Process

In the literature, PCB assembly process is classified into two main types depending on the location of the components (i.e. on one or both sides of the PCB), and three classes defining the type of parts used in the assembly (i.e. THDs, SMDs or both).[30] Depending on the type and the class of production, several assembly stages are required to assemble a PCB. This study focuses on the single-sided SMT assembly process which involves the stages described in figure 2.3.

---

[27] Cf. Scheel (1999), p. 13.

[28] Based on the figures in Bachmann (1999), p. 16.

[29] Source: Coombs (1988), p. 3.2.

[30] This kind of classification is common in PCB assembly literature, e.g. Prasad (1989), p. 7-10, and Rowland and Belangia (1993), p. 5-8.

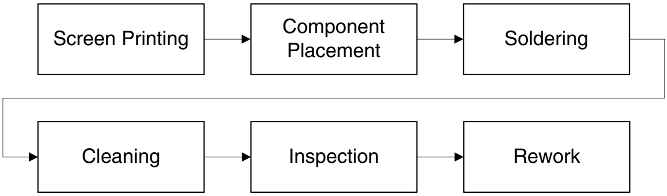


Figure 2.3: Process flow for assembly operations of a single-sided SMT PCB[31]

2.3.1 Screen Printing

The first process in SMT assembly is the application of soldering paste on the substrate, where solder paste must be deposited for electrical connections. There are two types of processes for applying solder paste on the PCB. If high flexibility is required or the same tool has to be used for different applications, solder paste can be dispensed by being squeezed through the needle of a syringe. However, since the soldering paste is dispensed on one point at a time, this is a very slow process. Nowadays, the most common method for solder paste application is the screen (stencil) printing, which is ideally suited to high production volumes of similar assemblies with a large number of surface mounted components. While a screen contains an open wire mesh around which solder paste must flow, a stencil opening is fully etched and does not obstruct the paste flow. Screens and stencils are stretched in a metallic frame and aligned above the PCB. After the paste is deposited, PCB is separated and transferred to the next processing stage.[32]

The selection of the appropriate solder paste is important to satisfy all production environment relevant conditions.[33] Choosing the appropriate solder paste for an application involves judicious selection of the components of the solder paste, i.e., the solder power, rosin/resin system, solvent, activator, and the agents needed to modify rheological characteristics[34].

2.3.2 Component Placement

After soldering paste is applied on the substrate, SMDs are placed on the predefined positions on the PCB. This process can be performed either manually or automatically. Although manual operations have constituted a great part of the assembly process in the beginning of the

---

[31] Based on Prasad (1989), p. 10, and Rowland and Belangia (1993), p. 6.
[32] Details and comparison of screen and stencil printing is given in Prasad (1989), chapter 9.3.
[33] Cf. Gingsberg (1989), p. 182.
[34] Rhelogical characteristics are: e.g. viscosity, slump, tackiness, and working life.

SMT, it is currently applied only for very small production amounts like labor models and prototypes. With the development of flexible machines with a wide range of component spectrum, most of the assembly process is currently carried out using fully automated SMT lines. The throughput of such an assembly line is primarily determined by the component placement machines which may constitute about 50% of the total capital investment required for a medium volume assembly line.[35] Hence, component placement machines, which mainly are the bottleneck of the SMT lines, must be utilized. The types of component placement machinery available in the market are presented in chapter 3.

2.3.3 Reflow Soldering

After the placement operations are completed, the populated PCBs pass through a reflow furnace where soldering is carried out for securely interconnecting the components to the board. Depending on the specific application, different soldering equipments working with infrared, hot vapor or laser technologies are available in the market.[36] Among these, infrared reflow process has prevailed in many applications because it provides higher yield and lower operating costs.[37] The so-called soldering cycle is used to define the temperature profile for different zones in the furnace for preheating, soldering and cooling phases.

Since the introduction of the EU regulation on Restriction of certain Hazardous Substances (RoHS), the use of poisonous substances (e.g. lead, quicksilver, and cadmium) in production of consumer electronics is forbidden. Thus, PCB manufacturers have to cope with the new specifications arising from lead-free pastes, which require up to 40˚C higher melting points, and hence cause more stress on PCBs and SMDs.[38]

2.3.4 Cleaning

SMT assemblies must be cleaned after reflow soldering to ensure the removal of flux and other induced contaminants.

2.3.5 Testing[39]

The smaller sizes and tighter tolerances associated with high-density surface mounting applications require increased inspection and process control. The defects encountered in SMT –

35 Cf. Prasad (1989), p. 387.
36 Cf. Siemens AG (1990), p. 238-241.
37 Cf. Reithinger (1994), p.13 and Prasad (1989), p. 468.
38 Cf. Walz (2004), p. 155.
39 Inspection methods for PCB assembly are given in Scheel (1999), chapter 5.

solder opens, solder wicking, tombstoning, bridges, misalignment, part movement, solder balls, and so on – are caused by solderability and lead coplanarity problems, poor paste printing, bad placement, and improper soldering process profiles.[40]

Soldering points are generally controlled via *visual* inspection. Automated inspection machines with optical cameras are essential parts of modern SMT lines and control the soldering quality after the reflow soldering process. Alternatively, laser and x-ray inspection equipment can be used, if three-dimensional control is required or light-optical unreachable points, e.g. back-side of bare dies, have to be inspected.[41]

Additional to visual inspection, electrical testing can also be carried out to assure electrical functionality. One of the electrical testing methods is the so-called *function* test, where components or component groups are connected to the testing equipment using sockets. The most common test method for assembled PCBs is however the *in-circuit* testing, which uses a matrix of probes contacting circuit nodes on the board. The signals used to stimulate an individual node are compared with the measured response to the expected one. Hence, the electrical value of every single component can be tested isolated from other signals in the circuit.

2.3.6 Rework

Statistical analyses reveal that 64% of process-related errors are caused by deficits in screen printing, which is followed by defects due to assembly, reflow soldering and components with 15%, 15% and 6%, respectively.[42] When the conformity with the requirements is not met by the PCB in the testing stage, rework is needed to ensure compliance. The main defects in SMT lines are either defective or misaligned components or incorrect conductor routing.[43] Incorrect conductor routing is a problem which arises in the PCB design process and must be adjusted with a new circuit design. If defects are caused by components or soldering points, these can be corrected at a rework station. Commonly, hot-air reflow equipments are used to replace defective components or realign functioning ones.

---

40 Cf. Prasad (1989), p. 556-557.
41 A list of testing processes are given in Scheel (1999), p. 594-596.
42 Cf. Scheel (1999), p. 19, and NN (2001a).
43 Cf. Gingsberg (1989), p. 195. Possible reasons for defects related to materials and processes are given in Prasad (1989), p. 522-529.

3. SMT Placement Machines

Placement machines are the most critical and expensive pieces of equipment in SMT, which commonly constitute the bottleneck of the assembly line, and hence determine the throughput. Growing production volumes, reducing size of components and complexity of the boards reinforced the usage of full-automated assembly lines. Depending on the placement type, full-automated placement systems can be observed in two main groups: *simultaneous* and *sequential* placement equipment.[44]

In *simultaneous* placement systems, the machine picks up and places all components at once, which requires the positioning of these components on a pattern before they are assembled. This method has been developed for producing PCBs including standard simple components in very large batch sizes. High production rates can be achieved with simultaneous placement, but it is not suitable for flexible automation due to long setup times and high level of customization.

Sequential assembly machines work with so-called pick-and-place principle, i.e. pick every single component and then place it on the PCB in a sequential manner. These machines are more flexible and can assemble a wide range of components but do not reach the placement rates of simultaneous machines. There are also some special types of machines, which allow simultaneous pick up of multiple components and sequential placement of each on the PCB. Details of these machines are described in section 3.3.

Another way to classify the placement equipment is by its design and functionality. In this study, placement machines are classified based on their working principles as following:

- pick-and-place machines,
- collect-and-place machines,
- chip shooter, and
- other forms of SMT machines.[45]

---

[44] Cf. Krups (1991), p. 158, and Siemens (1990), p. 242.

[45] More detailed classifications of placement machines can be found in Tirpak et al. (2002), and Ayob and Kendall (2008).

All of the above given placement machines consist of at least three kinematical elements: the magazine that provides the electronic components, the table that holds the PCB during the assembly, and the transfer system that collects the components from the magazine, transfers to the PCB, and performs the actual mounting. These kinematical elements may have different degrees of freedom depending on the particular machine.

3.1 Structure of SMT Placement Machines

3.1.1 Component Feeders

SMT components are supplied generally in three forms, namely in *tapes*, as *bulk material* in tubes or as *matrix trays*.[46] Currently, most of the SMDs are delivered in *tape* form. Tapes are standardized to fit tape feeders and are available in different widths. However, with the miniturization of components, the tape costs and recycling issues become more significant.[47] Hence, *bulk material* delivery is becoming more interesting as a delivery form for small SMDs, especially for passive components or small semiconductors. These bulk components are generally packed into tubes (bulk cases) and attached to the magazine using bulk case feeders. Because the components have tiny dimensions, a tube can occupy almost ten times more components compared to a standard reel, which reduces the refill effort and machine stoppages.[48] Another competitive advantage of the bulk case delivery against tape feeders is the savings in component storage area. Despite many advantages of bulk case feeders, they are not so flexible like tape feeders and cannot be used for the delivery of all SMD types. Hence, bulk case feeders will not be able to compensate tape feeders completely in the close future.

Matrix trays (also called waffle packs) are developed out of the necessity for handling quad flat packs and fine pitch components.[49] These hold the components securely without damaging the fragile leads. Modern fine-pitch placement machines are equipped with matrix tray feeders, which allow storage and automated delivery of multiple components.

46 Cf. Brindley (1990), p. 113, and Rowland and Belangia (1993), p. 42-45.
47 Cf. Bachmann et al. (1999), p. 85.
48 Cf. NN (2001b).
49 Cf. Rowland and Belangia (1993), p. 45.

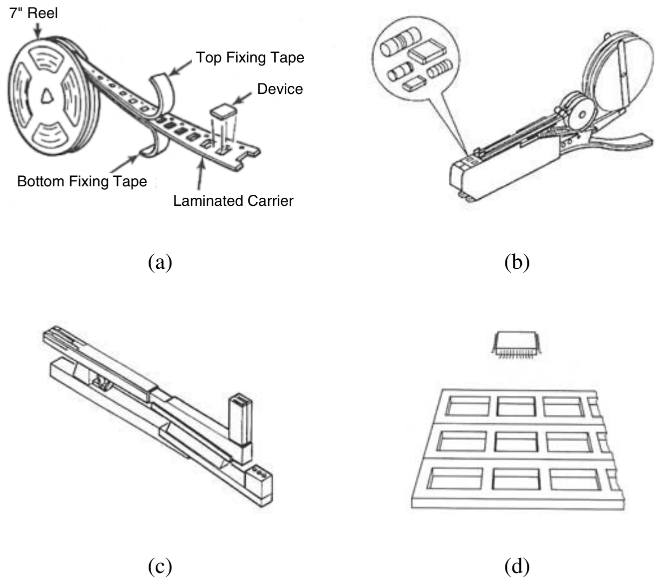


Figure 3.1: Different types of component packaging and feeders:[50]
(a) tape carrier (b) tape feeder (c) bulk case feeder (d) matrix tray

The feeders of today are high-tech equipments, which deliver SMDs with high precision below the pickup point and are designed to shorten component and feeder setup times.

3.1.2 Component Magazine

Component feeders are allocated on the slots of the component magazine. These slots are usually standardized to take one feeder with 8 mm width. If a component feeder is wider than one single slot, it occupies a number of adjacent slots depending on its width. During the setup of component feeders, the content of a feeder and its slot position has to be compared with the setup plan to avoid misplacement of components on the PCB. This is done efficiently using a barcode reader system and automatically comparing the content of a feeder with the component list loaded in the line control system.

50 Source: Coombs (1988), p. 19.26.

Depending on the machine type, a magazine either moves on an axis and delivers the next component to be assembled below a fixed pickup point, or stays stationary on the machine enabling a fixed coordinate for each component feeder. Stationary magazines enable refill of components by splicing, i.e. connecting the end of the old tape with the beginning of the new tape during the placement operations take place. Modern placement systems are equipped with feeder trolleys which enable the so-called offline component setup, i.e. component feeders of the next assembly job can be prepared off the line on another trolley during the placement of the current job takes place. Thus, setup of the next assembly job only requires exchange of feeder trolleys, which drastically reduces the setup time. Figure 3.2 illustrates detaching and attaching a feeder trolley on an example of a Siemens placement machine.

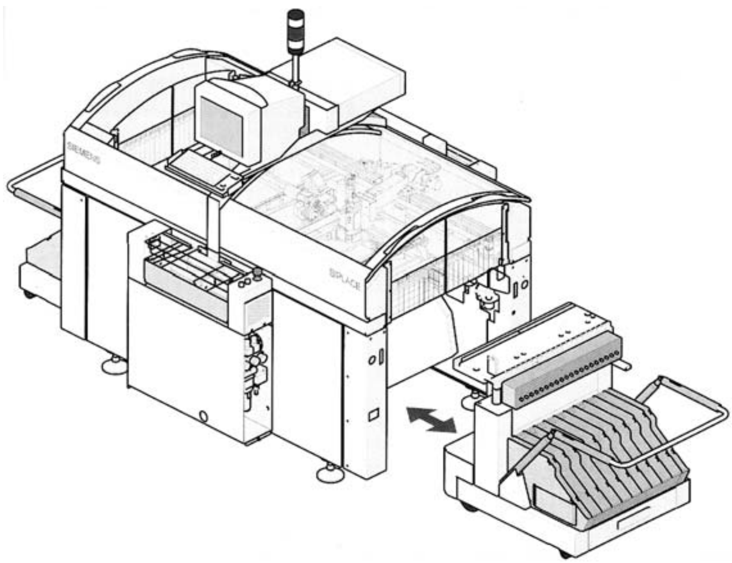


Figure 3.2: Exchange of a feeder trolley[51]

3.1.3 Placement Head

The placement head is responsible for removing the component from the feeder, orienting it correctly, and placing it on the PCB. Depending on the machine and component type, the placement head is equipped with either vacuum nozzles or grippers. Usually, three to five

[51] Source: Bachmann et al. (1999), p. 121.

different nozzles are enough to cover a broad range of components.[52] Vacuum or optical sensing is used to verify that the component is actually in place on the nozzle tip. The placement head must also have tactile sensing to control the vertical stroke on vertical axis, which is very important to prevent components from being crushed between the placement nozzle and the PCB.

Depending on the machine type, placement systems can be divided into two groups. Chip shooter type machines are equipped with rotary turrets, which include a number of placement heads. The so-called X-Y gantry machines, e.g. pick-and-place and collect-and-place machines, are equipped with placement heads which move on X-Y axis to transport components from feeders to corresponding placement locations on the PCB. The placement principles for different type of machines are given in sections 3.2 to 3.5.

3.1.4 Component Centering

Components are usually not delivered from the feeders in a perfectly centered form. In order to provide a precise positioning of a component on the PCB, the placement system must align the component picked up by the nozzle. The vision system determines the x, y, and γ (rotational) offset of each component prior to the placement operation. In addition to determining the component offset, the vision system can also inspect the component for dimensional integrity and lead damages.[53]

Centering can occur either internally or externally as indicated in figure 3.3. In internal centering, the centering mechanism is located on the placement head, while external centering means that the centering mechanism is located on the frame of the placement system.[54] Mechanical centering requires component-dependent centering mechanisms and can damage fine-pitch leads easily. Therefore, modern placement machines use vision centering and are occupied with cameras directly at the placement head. This allows component centering to be carried out parallel to the movement of the placement head, which has an improving effect on the placement time. In fine-pitch pick-and-place operations, external vision systems are commonly used for more precise inspection of component alignment. Hence, the component travels on the way to the placement operation over a fixed camera.

---

[52] Cf. Rowland and Belangia (1993), p. 124.
[53] Cf. Rowland and Belangia (1993), p. 126.
[54] Cf. Rowland and Belangia (1993), p. 126.

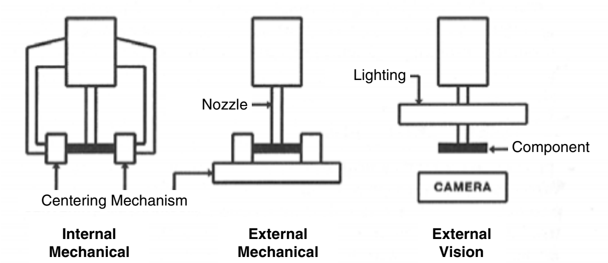


Figure 3.3: Component centering methods[55]

3.1.5 PCB Table

PCBs enter the placement machine using a conveyor transport system and are fixed on a so-called PCB table before they get assembled. PCB tables can either be fixed or moveable depending on the kinematics of the placement equipment. Tables movable on X-Y axes are usually parts of chip shooter type machines where the PCB table locates the next placement point below the placement head. Modern X-Y gantry systems, i.e. pick-and-place and collect-and-place machines, work generally with stationary tables.

3.1.6 Positioning System

Machines with a moveable placement head are occupied commonly with an X-Y gantry. This system allows the head to move in two axes simultaneously. A gantry system, in comparison to a standard robot arm, can carry more weight and reaches higher precision. Modern placement machines work with servo motors which continually advances the drive mechanism until the correct position is achieved. The main advantages of servo motors (compared to stepper motors used in the past) are the increased speed and accuracy of the placement and the ability to handle heavier loads.[56]

3.2 Pick-and-Place Machines

A general idea of the assembly principle in pick-and-place machinery can be easily gathered from the name given to these types of machine. It is in fact the automated version of a manual placement operation, where a worker picks up a component with tweezers, moves it with his

[55] Source: Rowland and Belangia (1993), p. 126.
[56] Cf. Rowland and Belangia (1993), p. 125.

arm to the PCB and places it on the predefined location. In the automated pick-and-place assembly, the transportation of components is carried out with a robot arm or more commonly with an X-Y gantry system in a sequential manner. The other two elements of the placement system, i.e. the PCB table and the magazine, are in many cases stationary.[57] The placement arm can usually move in x- and y-directions simultaneously. The pickup and placement of components are carried out by the movement of the nozzle in Z direction. Figure 3.4 illustrates a pick-and-place operation including picking up the component from the feeder, rotating and centering it simultaneous to the transport operation, and placing it on the predefined point on the PCB.

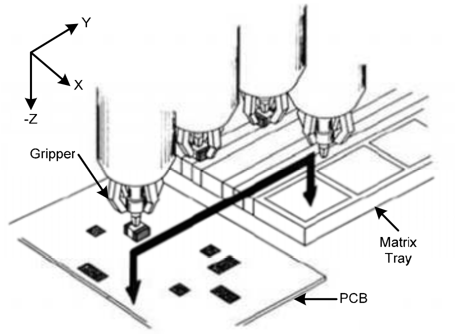


Figure 3.4: Pick-and-place operation[58]

Pick-and-place machines have a simple structure and reach a high assembly precision. The investment costs and the development and production requirements are lower in comparison with other placement systems. Production rates of this type of machines are lower than collect-and-place machines and chip shooters. Pick-and-place machines are mostly used in an assembly line in combination with other systems and perform the operations of fine-pitch components, which need precise placement or have abnormal shapes and/or dimensions. Pick-and-place machines are mostly equipped with nozzle magazines which allow fast change of vacuum nozzles or grippers and are installed with waffle pack changers for matrix tray supply

[57] Grunow (2000) describes other types of machines with moveable PCB tables and component magazines. However, these very specific types of machines have disappeared from the market, and thus will not be discussed here.

[58] Source: Krups (1991), p. 159.

of components. These machines are commonly equipped with external vision systems which control the visual conformity of the component, especially the leads of chip carriers, with the defined visual specifications. Some machines are even capable of testing electrical characteristics of components during the transfer operation. If an abnormal component determined, it is thrown into a trash box and a new component for the same operation is supplied from the component feeder. Thus, quality problems may increase the total number of placement tours and extend the placement time on the machine.

3.3 Collect-and-Place Machines

Collect-and-place machines work with the same principle of pick-and-place systems except the number of elements which can be picked up and placed in an assembly tour. There are two main types of placement heads used in the industry, namely beam-type and rotary placement heads.

Placement machines with beam heads include a number of nozzles which are located parallel to each other. Figure 3.5 illustrates a beam-type placement head with four nozzles. The nozzles can be selected independently, which allows different pickup and placement sequences. Another advantage of the beam-type placement head is its capability of picking up more than one component at a time. However, the component feeders should have the same width with the distance between the nozzles for enabling such a simultaneous pickup operation. The placement of components is done in most cases sequentially because the placement locations on the PCB are defined by the PCB layout and generally does not allow a parallel placement.

In case of collect-and-place machines with rotary placement heads, the nozzles are located radial on a turret (see figure 3.5). Each component is picked up sequentially from feeder locations until the capacity of the head is reached. As only the nozzle on the bottom side can perform the pickup or placement operation, the placement head has to rotate one segment in order to execute the next operation. Modern collect-and-place machines have integrated component control and calibration units on the placement head and carry out these operations directly after picking up the component to ensure a flawless assembly. Since the turret rotates only in one direction, the sequence of pickup and placement should be identical.

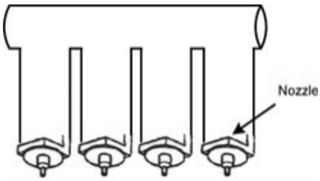


Figure 3.5: Beam-type placement head equipped with four nozzles

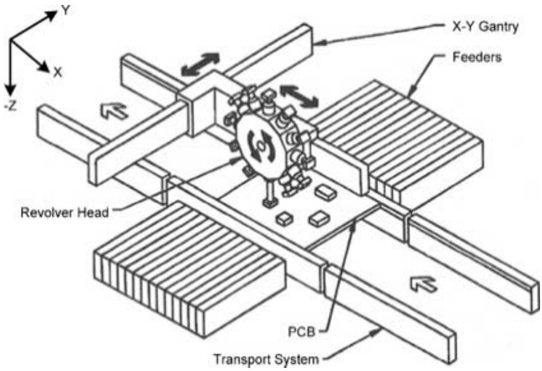


Figure 3.6: Single-gantry collect-and-place machine with a rotary placement head[59]

In practical applications, it is almost impossible to exploit the advantage of simultaneous operations in beam-type placement heads, because the arrangement of feeders in the component magazine and the composition of the placement locations on the board have to comply with the geometrical distances of the nozzles on the placement head. Additionally, the number of nozzles on a beam-type placement head is restricted compared to the revolver-type head due to its design. Thus, a considerably larger number of placement tours is needed to place the same number of components. Therefore, collect-and-place machines with rotary placement heads are favored in many industrial applications. For the rest of this work, the name collect-and-place will be used as a synonym for the machines with rotary placement heads. The working principles and the assembly cycle of these machines are described in section 5.4 in detail.

59 Source: Schiebel (1994), p. 90.

3.4 Chip Shooter

Chip shooters consist of three simultaneously working kinematical components: the component magazine, the PCB table and the turret. The main characteristic of this machine is the design of the transfer system as a turret with a rotary degree of freedom. The magazine brings the component to be assembled under the pickup point. The first nozzle picks this component and the turret rotates one segment to pick the next component to be assembled. On the other side, the table positions the placement point under the loaded head and the head mounts the component. Between two consecutive segment rotations a component is picked up and a component is placed simultaneously, but each pickup or placement operation is carried out sequentially. Since the turret rotates at high speed, substantial centrifugal forces work on components which cause the components to be dislocated or lost. The strength of these forces depends on components size, weight and type of packaging. Therefore, components are assigned to different rotation speed classes. The components with a slow rotational speed class reduce the rotational pace for the whole turret. In order to reduce the number of slow rotations, component types which are assigned to same rotation speed classes are generally placed subsequently. Rotary turret systems are equipped with multiple nozzles on each segment, which enables the selection of an appropriate nozzle for each component type. Thus, tool changes during placement of the same PCB are not employed in chip shooters.

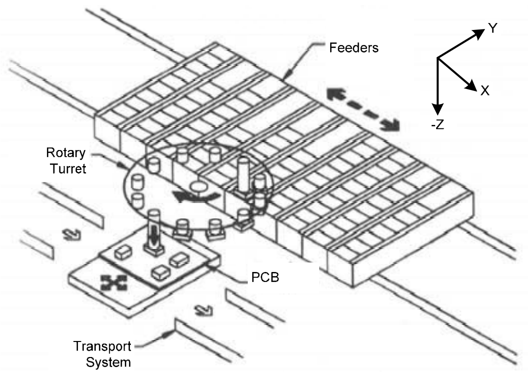


Figure 3.7: Chip shooter[60]

[60] Source: Schiebel (1994), p. 89.

Chip shooter machines are designed for high-speed placement of components which are supplied on tape reels and located on the slots of the magazine. Because of the moveable magazine principle, the setup of components has to be carried out online, i.e. the machine has to be stopped to assign each feeder required for the next job to the magazine slots. However, if a chip shooter is equipped with two magazines, one of the magazines can be setup for the next job during the other one supplies components for the placement operations. When the first job is completed, the next job is loaded by the software, which is called hot swapping.[61] Moveable magazines also prohibit splicing new component reels to empty ones during the assembly operations take place. Hence, the machine has to be stopped each time a component reel empties.

Another disadvantage of chip shooter type machines is the dependability of the pace on the dimensions of the components. Bigger components must be rotated slower in order to avoid component losses by centrifugal forces, which causes a decrease in the rotational speed of the whole turret, and hence the placement speed. Also the movement of the PCB table can dislocate mounted components depending on the retention force of the tackiness of the soldering paste and the component weight. Because of these reasons, chip shooters are mostly used in the assembly of small components and are generally not suitable to assemble a whole PCB. One of the other disadvantages is the extreme dimensions of the machine, which brings difficulties in transportation and layout planning.

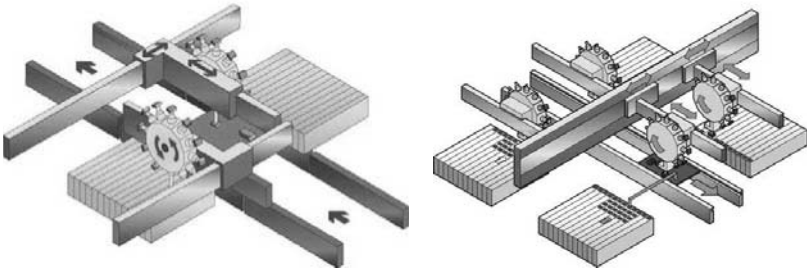
3.5 Other Forms of SMT Machines

PCB machine vendors have developed a number of different placement machines which are based on different configurations of the above described main placement principles. Depending on the required level of flexibility, placement speed, and budget, PCB manufacturers have the privilege to select among a various number of placement machines.

Due to its flexibility, the collect-and-place principle has prevailed in the market. Modularity and affordability of these machines allow different configurations enabling many options in number of gantry systems, nozzles on the placement head and transport systems. Figure 3.8 illustrates the double-gantry and four-gantry double-transport variations of collect-and-place machines, which arised recently in the market.

---

[61] Cf. Coble and Bohn (1997).



(a) (b)

Figure 3.8: Different configurations of collect-and-place machines:[62]
(a) double-gantry and (b) four-gantry collect-and-place machines

There are also other solutions which comprise advantages of different placement types. Commonly, there are only a few fine-pitch components to be assembled on a PCB which may cause underutilization of a fine-pitch placement machine. Because of the low placement speed of pick-and-place machines, the placement cost per component is very high for fine-pitch machines compared to other high-speed placement equipments. Hence, a combination of collect-and-place head for high-speed placement and fine-pitch pick-and-place equipment installed on the same gantry enables placement of a complete component spectrum in one single machine and reduces the per component placement cost significantly.[63] There are also other variations of this concept where a single placement machine is equipped with a rotary head on one gantry and the pick-and-place head on the other.

---

[62] Source: Siemens (2007).
[63] Cf. Bachmann et al. (1999), p. 82.

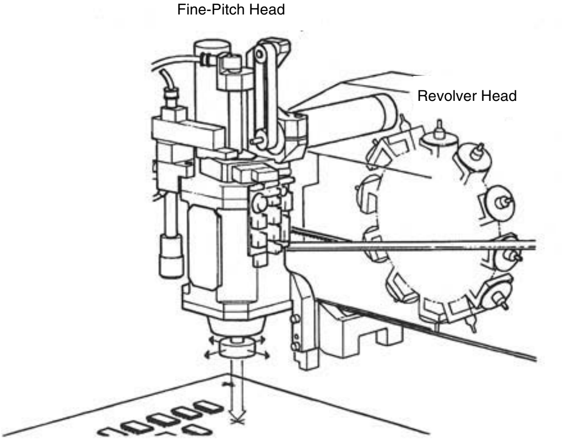


Figure 3.9: Collect-and-place head with fine-pitch pick-and-place module[64]

Another type of machine, the so-called modular placement machine, is used for mass production of PCBs where a high-speed placement is desired. This placement principle is also known as inline placement.[65] This equipment employs a series of pick-and-place heads allocated to fixed assembly stations. Each placement unit includes an own component magazine and a transfer system, and places only a very limited number of component types. The modules are linked with a conveyor system, which transports the PCBs from one station to the next. Hence, the number of placement units and the balance of the workload among placement heads define the placement speed. However, this type of machinery is only appropriate for mass production because the feeder setup has to be changed for each new product on all units, which is associated with high setup costs. A modular placement machine and its working principle are presented in figures 3.10 and 3.11.

---

64 Source: Schiebel (1994), p. 91.
65 Cf. Prasad (1989), p. 390.

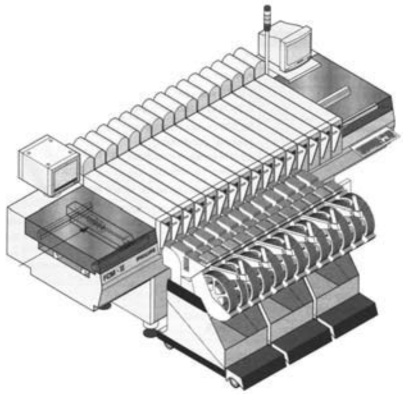


Figure 3.10: Modular placement machine[66]

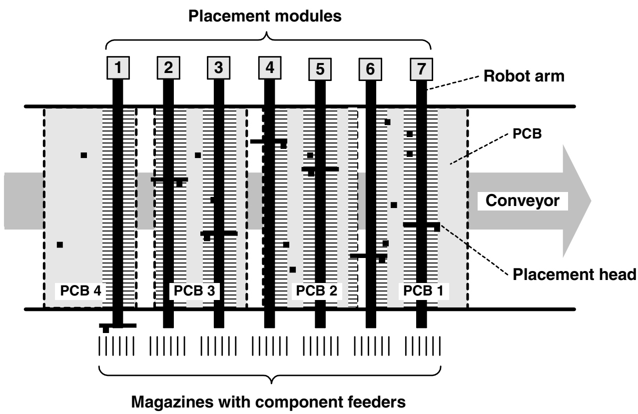


Figure 3.11: Working principle of a modular placement machine[67]

3.6 Selection of the Appropriate SMT Machine

For a long period of time, the most popular types of sequential placement machines have been the pick-and-place machine with single transfer of components from the magazine to the

[66] Source: Grunow et al. (2003).
[67] Source: Grunow et al. (2003).

board and the chip shooter type of machine, which uses a rotary turret to transfer the components from the magazine to the board. Meanwhile, the chip shooter has almost disappeared from the market and been replaced by different types of collect-and-place machines. Collect-and-place machines have become popular in industry mainly because they provide a high degree of flexibility with respect to the range of component types that can be assembled and, at the same time, allow a considerable placement speed compared to other types of machinery.

Collect-and-place machines have several advantages compared to chip shooters. One prime advantage is that the PCB resides on a stationary table during the placement process. Thus, defects are avoided, which are due to acceleration forces moving components away from their original placement position. A second advantage is the online splicing capability of collect-and-place machines, i.e. the possibility to refill component feeders without halting the machine. Additionally, collect-and-place machines acquire less space and are built modularly depending on the requirements of the PCB manufacturer.

The placement speed of chip shooters depend on the cycle time of components currently traveling on the turret. To avoid losing components due to centrifugal forces, the rotational speed of the turret cannot be higher than allowed limits for all components. Hence one single large component can slow down the assembly speed of all others. Additionally, chip shooters cannot assemble large and nonstandard components. In contrast to chip shooters, the cycle time of a collect-and-place machine is independent of the component type. Hence, collect-and-place machines can arrive between 70-90% of the theoretical placement speed whereas the real placement speed stays only at a level of 40-70% for chip shooters.[68]

Today, SMDs which are as small as 0.2×0.1 inches (i.e. the so-called 0201 components) are found in many products. The tendency for smaller end products introduces wider usage of 01005 SMDs in e.g. cellular phones, personal digital assistants (PDAs) or radio frequency (RF) modules.[69] Hence, only high-precision robot arm movements and a stationary PCB table enables the placement of such tiny components. Chip shooters are limited to assemble only over 0402 size components due to the kinematics of this machine type. Such problems do not arise in collect-and-place assembly. However, pick-and-place machines will stay as an important piece of equipment for surface mounting lines and will gain more importance with in-

[68] Dürr (1997a), p. 1.
[69] Cf. Kimman (2005).

creasing fine-pitch placement due to new packaging types (e.g. bare dies). Depending on the throughput and flexibility requirements of the manufacturer, pick-and-place machines will be combined in the future with collect-and-place machines giving manufacturers a chance to adjust their throughput via modularity. This is the only way to realize robust investments which will be able to satisfy placement requirements of the future components.

4. Production Planning in Electronics Assembly

4.1 Overview of Planning Problems

The general planning problem for manufacturing systems is too complex to be solved globally. Thus, it is common in the flexible manufacturing systems (FMS) literature to decompose the planning problem into hierarchical decision structures relating to a variety of decisions to be taken in long-, medium- and short-term periods.[70] In these decomposition schemes, problems are divided into solvable subproblems which are coupled with each other in the global scheme. Of course, this methodology cannot guarantee the global optimality of the final solution, even assuming that all subproblems are solved to optimality, because many subproblems are generally interrelated and the solution of one problem affects the solution of the next one. This is even more relevant for the case of PCB assembly, where most subproblems themselves turn out to be *NP*-hard, and hence can only be approximately solved by heuristic procedures.[71] Nevertheless, such hierarchical approaches have prevailed in the literature and previously proved to deliver good quality solutions to complex planning problems.

Planning decisions are generally categorized under *strategic*, *tactical* and *operative* decision levels.[72] *Strategic* planning includes long-term decisions and planning activities taken by the top management and influence the flexibility of the manufacturing system. The main problems are the design and selection of the equipment and of the products to be manufactured.

The decisions over the organization of the production can be classified under the *tactical* planning. These decisions guide to reach the targets stepwise, which are defined in the strategic planning. The reengineering or improvement activities of the production infrastructure can be classified under this level. Typical problems for tactical planning are dimensioning the capacity of an assembly system and layout planning. Also creation of product families and assigning these to different production segments are problems observed at the tactical level.

In the *operative* level, decisions required for carrying out real-time operations have to be taken. These problems are related with releasing the production orders into the system consider-

[70] A broad literature review on planning problems and classifications for FMS is given in Crama et al. (1996b), chapter 1.

[71] Cf. Crama et al. (1990).

[72] Cf. Crama et al. (1996b), p. 4-10, and Günther and Tempelmeier (2007), p. 27.

ing setup plans, batch sizes, and sequencing of the jobs. The target of operative planning is the economical usage of the available capacities planned within tactical decisions.

The *strategic* level does not play an essential role in the PCB assembly literature. The critical decisions in PCB assembly refer to the estimation of the required capacity, the selection of the appropriate equipment, and the organization of the assembly system. For the latter, assembly lines, machine groups, or configurations with a number of independently operating machines represent the major options. The choice between these options mainly depends on the particular manufacturing environment, which is either characterized by low production volumes of many specialized PCB types or high-volume production of a limited number of standard products. Hence, the following classification of planning problems focuses on *tactical* and *operative* levels.

Similar to other flexible manufacturing systems, the high complexity of the PCB assembly problem also suggests its decomposition into more manageable subproblems, and accepting the solution of each subproblem as the starting point for the next one.[73] **McGinnis et al. (1992)** propose the general definition of planning problems in PCB assembly in the following three decision levels:[74]

- *Grouping*: Generation of machine groups and PCB families and assignment of PCB families to machine groups.
- *Allocation*: Allocation of components to machines when a machine group consists of more than one machine.
- *Arrangement* and *Sequencing*: Assignment of component feeders to slots in the component magazine and sequencing of placement operations of each PCB at each placement machine.

These problems are connected to each other such that the solution of problems at higher level requires problems at lower level to be solved. For example, the allocation of components to each placement machine is generally decided under the workload balancing problem. However, the actual workload on each machine is directly affected by lower level decision problems, i.e. the assignment of component feeders to the magazine slots and the sequence of placement

[73] Cf. Crama et al. (1990).
[74] Cf. Crama et al. (1997), and Ammons et al. (1997).

operations of each PCB. Hence, the problems at the lower level have to be solved each time when decisions for higher level problems are taken.

In the following section, a novel hierarchical classification scheme is described in detail and the relevant literature for different planning problems will be reviewed.

4.2 Hierarchical Decomposition of Planning Problems

Figure 4.1 presents a novel and detailed hierarchical classification of PCB assembly problems according to the number of different board types and machines.[75] This kind of a classification scheme gives a better understanding of the planning problems depending on the observed manufacturing environment and serves as a common basis for software solutions.

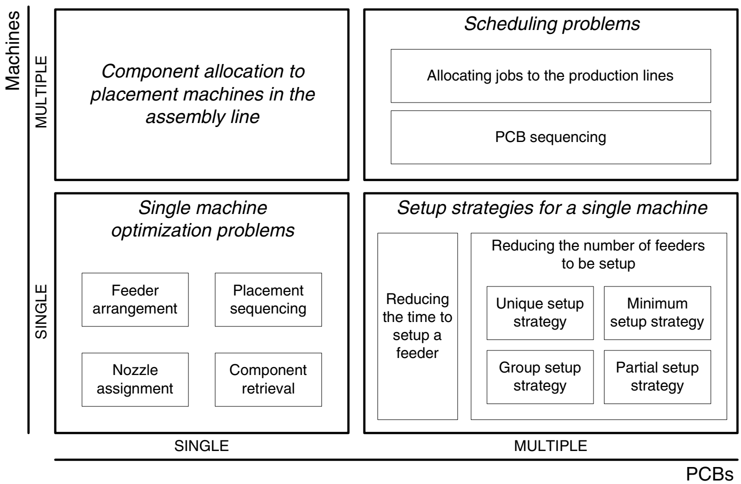


Figure 4.1: Hierarchical decomposition of problems in PCB assembly

As illustrated in figure 4.1, the PCB assembly problem involves a large number of intertwined problems. There are a various number of researches in the literature which focus only on a segment of these decision problems by assuming that either the solution of other interrelated

[75] Cf. Smed et al. (2000), and Johnson and Smed (2001).

problems are given or to be neglected in the planning approach. In the following sections, details of these problems and related literature will be presented.

4.2.1 Multiple PCB – Multiple Machine (M-M) Problems

This type of problems forms the topmost level in the planning hierarchy. Generally, the number of machines in each assembly line is determined and fixed prior to this problem stage. Hence, the main problem to be solved is the allocation of PCB types to product families and these to machine groups. This problem is similar to the cell formation problem in group technology (GT) or job grouping problem in FMS literature.[76] The difference between these two approaches is that in the GT framework, products are grouped by use of a clustering algorithm based on component commonality between the boards. However, the FMS grouping model explicitly takes the carrier capacity into account and tries to minimize the number of families to be formed. In contrast to FMS, modern PCB assembly lines consist of a serial allocation of placement machines.[77]

The *allocation* problem also involves problems of lot sizing and workload balancing between the lines, i.e. interline balancing. Hence, the target of the allocation problem is to minimize the workload of the most heavily loaded assembly line. The resulting model is similar to bin packing or parallel machine scheduling models which are presented in **Crama et al. (2002)**. In this model, the feeder capacity of each line is treated at an aggregated level. Because the single machine problems which deliver the actual assembly times are not solved yet, the assembly time for each PCB is estimated.

After PCB families are allocated to different assembly lines, the *line sequencing* problem must be solved for sequencing different board types within each family. Hence, different PCB jobs are sequenced on each line consisting of several placement machines. The sequencing problem generally focuses on either fulfilling the due dates of PCB jobs or minimizing the total setup effort.

Papers working on design issues are rare in the PCB assembly literature. **Ahmadi (1993)** deals with problems of system design, process configuration, product grouping, and line configuration. These problems are observed in three blocks. In the first block, a mathematical

[76] Cf. Crama et al. (2002).
[77] There are also some old approaches which focus on manual or semi-automated assembly and hence a flexible layout. However, these approaches involving the routing problem are not relevant for modern PCB assembly and hence not in the interest of this study.

model which deals with the process configuration is formalized. Given this solution, the optimal positioning of the feeders and the sequence of pick-and-place operations are determined on each machine. The solution of this stage is used in the third block to generate the solution to the grouping and mini-line production problems. **Ahmadi and Kouvelis (1999)** present an analytical framework for the design of PCB assembly lines. They analyze alternative line configurations and develop mathematical programming based optimization models which are solved with proposed algorithms.

To solve the PCB to assembly line assignment problem, **Ellis and Bhoja (2002)** develop a large mixed-integer programming (MIP) model which is solved by decomposing the problem in conjunction with a branch-and-bound approach and several improvement techniques. Similarly, **Rogers and Warrington (2004)** present a mathematical programming model to support the allocation of production lots to assembly lines. **Neammanee and Randhawa (2003)** apply group technology concepts on assignment of boards to production lines by simultaneously solving the PCB sequencing and component allocation problems. An alternative approach, namely genetic algorithm (GA), for solving the PCB-line-assignment problem arising in a PCB manufacturing company is pursued by **Ho and Ji (2005a)**.

Integration of design and scheduling issues is addressed in the paper by **Dessouky et al. (1995)**. The presented heuristic approach intents to group machines into workstations and applies efficient sequencing rules for scheduling the assembly jobs for different types of PCBs. The objective of this approach is to keep the work-in-process (WIP) inventory levels at a minimum. Scheduling tasks including determination of due dates of customer orders, assigning products to daily production plans, and releasing boards to production lines are observed in **Lin et al. (1997)**. Lot sizing and scheduling decisions are evaluated with a simulation module and detailed analyses are carried out under a variety of manufacturing scenarios. A comprehensive simulation study of SMD assembly lines is presented in **Lambert et al. (2006)**. The objective of their study is to assess the effect of setup and scheduling strategies on the performance of the assembly line.

4.2.2 Single PCB – Multiple Machine (1-M) Problems

After each board is assigned to the assembly lines and sequenced on each line, the next problem is the allocation of placement operations to different placement machines of a sequential assembly line - generally using the objective of balancing the workload, i.e. intraline balancing. Depending on the boards and components to be assembled, usually a placement machine constitutes the bottleneck of the assembly line which dictates the line efficiency. In the simplified case, each component feeder is assigned only to one placement machine in the assembly line. Hence, all placement operations requiring this component type are carried out by this specific machine. The workload is affected by the number of placement operations assigned to each machine and the length of the tours required for their placement. However, since single machine optimization problems (i.e. feeder arrangement and placement sequencing) have not been solved yet, placement times required for calculating the workload have to be estimated.

Crama et al. (1990) propose a hierarchical approach to the problem of optimizing the throughput rate of a line of several placement machines devoted to the assembly of a single product. The decisions are made so that the PCB can be mounted using all machines and the processing time on the bottleneck machine is minimized. The throughput problem is divided into a number of interrelated subproblems for which optimization models and their complexity are presented. **Ammons et al. (1997)** decompose the entire planning problem into three major subproblems: grouping (assignment of PCB families to machine groups), allocation (assignment of component types to machines) and arrangement and sequencing (assignment of component feeders to slots in the component magazine and sequencing of placement operations). To solve these problems, efficient heuristic procedures are proposed.

Askin et al. (1994) investigate the component allocation and workload balancing problems for an open-shop assembly cell consisting of a number of identical machines. **Lofgren et al. (1991)** investigate flexible routing of PCBs through an assembly cell consisting of a number of workstations and present a heuristic for component-to-station assignments. **Hillier and Brandeau (2001)** present exact and heuristic models for balancing the workload of an assembly line consisting of automated and manual insertion operations. The objective of this research is to allocate works to automated machines in order to reduce labor hours required.

Crama et al. (1997) observe the assembly of multiple board types in an assembly line by creating a so-called composite board. The composite board comprises placement locations of all individual board types and is used for assigning the feeder racks for all components required for the observed boards. A line balancing heuristic based on improvements on the existing SMT line setups is presented by **Watkins and Cochran (1995)**. Their procedure moves components from bottleneck to non-bottleneck machines to improve the existing setup.

Using a stochastic mixed-integer programming framework, **Lin and Tardif (1999)** investigate the line balancing problem in face of uncertainty in demand and capacity. The research direction of mathematical programming based solution approaches is further pursued by **Lapierre et al. (2000)** who apply Lagrangean relaxation techniques and by **Kodek and Krisper (2004)** who present a branch-and-bound-based optimal algorithm to solve the line balancing problem. Also based on mixed-integer programming, **Sawik (2002)** presents an integrated approach for simultaneously balancing the workload and scheduling jobs in a surface mount technology line. Exact and heuristic approaches for an SMT placement line consisting of several serial and/or parallel placement machines separated by finite intermediate buffers are given in **Kaczmarczyk et al. (2004)**.

Wilhelm and Tarmy (2003) propose a set of heuristics for the interrelated decisions that form a process plan for assembling a given type of PCB on a series of chip shooter machines. A genetic algorithm for allocating placement operations to machines in a line is developed by **Ji et al. (2001)** and the application of a new artificial intelligence technique known as the immune algorithm is proposed by **Khoo and Alisantoso (2003)**.

A major drawback of the aforementioned approaches for solving the PCB assembly line balancing problem can be seen in the assumption that the time needed to perform a placement operation is given in advance. However, it is well known from several investigations that actual placement times heavily depend on the machine setup, i.e. the assignment of component feeders to slots in the component magazine, and the sequence of the placement operations.[78] In solving these detailed optimization problems, it has to be regarded that automated placement machines employed in industry differ fundamentally by their mode of operation, speed, component feeder capacity and the feasible range of components that can be processed. **Kulak et al. (2007b)** show that a line balancing solution using estimated placement times may lead to considerable errors in the actual cycle time of the assembly line. The presented two-stage GA-based approach, which integrates the machine-specific optimization problems into the line balancing solution, performs better than other approaches which are purely based on estimating placement times.

[78] Cf. Grunow et al., 2000, Laakso et al., 2002, Duman, 2005, and Yilmaz et al., 2007.

4.2.3 Multiple PCB – Single Machine (M-1) Problems

In a single machine environment where multiple PCBs have to be produced, the main problem is the selection and definition of the best appropriate machine setup strategy. The placement time of a PCB depends on the feeder arrangement as well as the placement sequence for each individual PCB. In case of a single PCB, a unique feeder assignment can be determined for the specific PCB to optimize its placement operations. In a multiple PCB manufacturing environment, a decision must be taken on how feeders required for each PCB are to be allocated to the placement machine. Depending on the batch sizes and the number of different boards to be assembled on the same equipment, an appropriate setup strategy for balancing the tradeoff between total changeover effort and individual placement times should be determined. Hence, the setup problem becomes complex if machine-specific algorithms are integrated for calculating the makespan.

The setup time reduction can be achieved either by reducing the time to setup a feeder or by reducing the number of setup operations and the number of feeders to be setup.[79] Approaches focusing on reducing the time to set up a feeder are presented in **Coble and Bohn (1997)** and are out of the scope of this study. Hence, reducing the number of setup operations is the main focus presented here. Setup strategies are generally categorized under four groups which are explained in detail with relevant literature in section 4.3.

4.2.4 Single PCB – Single Machine (1-1) Problems

After higher level problems are solved, the PCB assembly problem can be reduced to a single machine optimization problem and can be observed for assigned placement operations of each PCB on each placement machine. The main problems of this decision level are:

- *feeder arrangement*,
- *placement sequencing*,
- *nozzle assignment*, and
- *component retrieval*.

[79] Cf. Coble and Bohn (1997).

These problems above are highly intertwined and must be solved in an integrative approach. The type and design of the placement machine has a direct effect on the solution of the above mentioned problems. Arrangement of feeders to the slots on the magazine and sequencing of placement operations are central planning elements for each machine type. The *feeder arrangement* problem is usually modeled as a quadratic assignment problem (QAP) while the *placement sequencing* resembles a traveling salesman problem (TSP). Both of these problems are known to be *NP*-hard problems and can only be solved with efficient heuristics for problems of practical size.

The problem of *nozzle assignment* plays a significant role for assemblies, where a single machine places a wide spectrum of component types. Hence, different nozzles may be required to pick up a component. Despite its practical relevance, this problem has been generally neglected in the PCB assembly literature.

The *component retrieval* problem becomes significant if several component feeders of a same type are assigned to more than one magazine slot. This strategy is usually applied for machines equipped with more than one placement head to balance the workload of gantry systems and increase productivity. Hence, it becomes necessary to decide which feeder should deliver the components for which placement operations.

Solving single machine problems is desired for increasing the effective operation rate, improving flexibility and for solving higher level problems more efficiently.[80] Despite the direct effect of these problems on actual placement times, these problems are mainly decoupled from the other problems in the hierarchy and observed commonly under unique setup strategies where minimizing the placement time is the main aspect. Thus, previous approaches on single machine optimization problems will be presented under unique setup strategies in section 4.3.1.

4.3 Setup Strategies

If more than one PCB type is assigned to an assembly line, a policy has to be developed for preparing the equipments for the new PCB type before its production. A setup operation on a PCB assembly line includes all of the machines and conveyor systems between them. Conveyors are adjusted to accommodate the width of the next board to be produced. The screen

[80] Cf. Johnsson (2001).

printer should be loaded with the appropriate screen and the soldering reflow furnace must be reprogrammed with the new temperature and duration settings. Additionally, the assembly line has to be equipped with all component feeders required to produce the new type of board. Hence, component feeders or feeder trolleys have to be removed and replaced with new ones. The new placement software for the new board type has to be uploaded to the placement machines to define component retrieval and placement sequences.

Off all these setup tasks, the preparation of the component feeders and their placement machines is the most time consuming.[81] This is a labor intensive process which includes tasks for setting up the reels on component feeders and loading these feeders to placement machines or feeder trolleys. Thus, the machine setup will be reduced to the observation of setting up the component feeders for the rest of this study.

In industrial PCB assembly, a variety of setup strategies is applied. Basically, these strategies differ by the relative priority which is given to the conflicting objectives of minimizing the setup effort and of minimizing the placement time per board. According to **Leon and Peters (1998)**, four types of setup strategies can be identified:

- *Minimum setup* strategies focus on minimizing the changeover times between different types of PCBs manufactured on the same machine. This is the predominant objective in small-lot PCB assembly where a high variety of PCB types is manufactured. In this application environment, fine-tuning the machine operations is only of minor importance.
- In high-volume production systems, however, *unique setup* strategies are preferred, which aim at minimizing the actual placement time per board for a single type of PCB. This goal is achieved by optimizing the machine operations, i.e. assigning the different component feeders to locations in the component magazine of the machine and sequencing the placement operations in a most effective way. These problems depend on the design and functionality of placement machines, and thus require machine-specific algorithms for their solutions.
- *Partial setup* strategies are characterized by the identification of a common set of component types which remain set up on the machine, and a PCB-specific set of com-

[81] Cf. Coble and Bohn (1997).

ponents which are exchanged in the component magazine whenever a change to another type of PCB takes place.

- *Group setup* strategies presented in the literature thus far are similar to minimum setup strategies except that changeover does not take place between individual PCB types, but between families of PCBs. Accordingly, setup families have to be defined so that no setup times are incurred for changing over between different boards within the same family.

In the following sections, previous approaches for each of these setup strategies will be discussed in detail.

4.3.1 Unique Setup Strategy

Unique setup strategy is adequate for a small variety of board types produced in high volumes. In this case, the productivity gains resulting from the optimal feeder assignment for a single PCB type offset the high setup times incurred. Thus, minimizing the placement time by arranging the best feeder allocation for each board type is the main focus of the unique setup strategy. All component feeders from the previous PCB batch are removed before starting with the setup of the new product which will be produced in high volumes. Sequencing the PCB jobs is not of significant importance in unique setup strategy.

Unique setup strategy consists of different measures for optimizing single machine problems which are already described in section 4.2.4. Because the structure of these problems depends on the specific design and kinematics of each placement machine, machine-specific algorithms are required for their solutions. Previous approaches for solving machine-specific problems are discussed for each machine category separately in the following.

The majority of the recent studies examine different types of pick-and-place machines used in different industrial applications. **Drezner and Nof (1984)** are among the first to describe the problems of component assignment and placement sequencing in robotic assembly cells with a robot arm to pick and place parts on the assembly place. They formulate the problem as a TSP consisting of the bin cells occupying parts to be delivered and the locations on the assembly place. They propose a heuristic approach which breaks the above described problem into two separate problems. The first problem, namely the bin assignment problem (BAP) decides on the assignment of parts into bins. When the assignment is given, the problem of ordering the movements of the robot arm is formulated as a TSP formulation. Hence, Drezner and Nof (1984) are among the first to decompose and sequentially solve single machine problems. For both problems, linear optimization models are presented.

Ball and Magazine (1988) focus on sequencing the insertion operations in a THT printed circuit board assembly. The feeder assignment problem is discussed but not solved. They observe a series of pick-and-place machines and apply Manhattan metric for calculating distances between insertion operations. The possibility of simultaneous movement of the placement arm, and hence modeling the movements with the Chebyshev metric rather than the Manhattan metric is also discussed. The insertion sequence problem is modeled as a rural postman problem (RPP).

Mettela and Egbelu (1989) develop a classification approach for creating a process plan for PCB component insertion depending on the kinematical dominance of the modules of a placement machine. The robot arm of the observed placement machine moves between fixed pickup and placement positions while the magazine and PCB table are busy with locating the next feeder and placement position below the placement arm. If the robot arm movement dominates other parallel operations, any technically feasible sequence of insertions is assumed to be optimal. If the PCB table dominance property holds, an optimal insertion sequence is computed by applying a nearest remaining unvisited neighbor heuristic. In order to reduce the cycle time in case of magazine dominance, all components of the same type will be inserted sequentially. A rule-based solution approach is also presented for the general assignment problem where no moveable module clearly dominates another one.

Van Laarhoven and Zijm (1993) observe a robot arm with three heads and feeders on both sides. The order in which the components are picked by the heads is fixed, however the order of placement can be chosen. The following problems are taken into consideration: the choice of heads to be mounted on the placement arm of each machine, the choice of components to be placed at each machine, an assignment of feeders to feeder positions on each machine, clustering of components to be placed in one pick-and-place move, and sequencing of component insertions at one machine. They propose a hierarchical approach and impress the need for an iterative approach for achieving the solution. Most of the problems discussed are well-known *NP*-hard problems and are solved using simulated annealing (SA) algorithms.

Kumar and Li (1995) observe the feeder assignment and insertion sequence problems on a pick-and-place machine. They obtain near-optimal solutions using minimum weight matching for determining an optimal assignment of pickup locations, and TSP for determining an optimal sequence of pickups and placements. The TSP is solved with standard heuristics and the initial solution is then improved with 2-opt, 3-opt or or-opt local search methods. The minimum weight matching problem is solved optimally using standard optimization software.

Su et al. (1995) observe a dynamic choice of pick-and-place points for retrieving and inserting components using a machine with moveable magazine and PCB table. They suggest that this new approach avoids robot waiting time. The experimental results show that the proposed dynamic approach is superior to fixed pick-and-place approach in nearly all cases. **Wang et al. (1998)** also observe the above explained hypothetical machine with a dynamic pick-and-place mode and present heuristics for assembly sequencing and magazine assignment problems based on an online feedback system to control movements of different parts of the machine. **Su and Fu (1998)** investigate the application of a simulated annealing method for the solution. **Ayob and Kendall (2005a)** apply a Chebyshev dynamic pick-and-place approach using a triple objective function with weighted sums of changes in placement cycle time, PCB table movement and feeder movement.

Lin et al. (1995) observe a system with two robot arms and two transfer systems with each having an own magazine. They focus also on maintaining a collision-free operation between the robots which work simultaneously at the same station. Each magazine contains all component types to be mounted and the slot location of each component type on each magazine is assumed to be known. Using this given magazine layout the sequence of assembly operations is determined.

Su and Srihari (1996) implement a prototype decision support system which uses artificial neural networks as a complement to expert systems in PCB assembly for finding solutions for magazine layout and insertion sequence problems. They observe a machine, which includes two workstations of the pick-and-place category. The insertion operations are allocated to one of these two subsystems. The components are sorted into groups depending on the type of nozzle required to carry out the mounting operation. Artificial neural networks are then applied to determine the sequence within a group of components. **Shih et al. (1996)** observe the same type of machine but use a nearest neighbor heuristic to determine the insertion sequence and present some numerical results for the achieved improvements.

Ahmadi et al. (1988) work on a dual delivery pick-and-place machine, which has a transfer system with two assembly heads and can pick and place simultaneously like a chip shooter. Each head has an own magazine, which is placed under the pickup point. The advantage of a concurrent design is defined as the concept of free move times, i.e. concurrent mechanical operations that must be performed parallel to pickup or placement operations regardless of the setup or sequences used in the process plan. Thus, the objective is not to minimize the distance traveled or the time required for alignment, but rather to minimize the delay time relative to the amount of free movement permitted by machine's design and control logic. In this paper, they present a hierarchy consisting of four problems and formulate mathematical models for them. **Ahmadi et al. (1995)** focus on the reel allocation problem for the above explained machine type and develop a heuristic approach. They present the problem in a network structure which can be solved optimally by dynamical programming. **Ahmadi et al. (1990)** present the details of the analyzed the CNC placement machine and explain their computer-aided model in detail. **Grotzinger (1992)** focus on the assembly sequence and reel allocation problems for the presented machinery. **Ahmadi and Kouvelis (1994)** deal with the allocation of component feeders to the given two feeder carriers by balancing the workload between them and reducing the required number of vacuum nozzle changes. The developed algorithms, which solve the given problem optimally using Lagrangian relaxation, are based on a branch-and-bound methodology.

There are also some GA-based approaches which have been developed for optimizing PCB assembly operations. **Wang et al. (1999)** apply GA to optimize the feeder assignment problem for a specific machine type with a single placement head and provide a comparison of GA against other optimization methods. **Deo et al. (2002)** provide a GA solution considering multiple setups for a specific PCB assembly machine which is equipped with a multi-purpose magazine. This magazine allows components to be retrieved both from tapes, which already hold the electronic components in the predetermined placement sequence, and regular component feeders. The objective of the GA is to minimize the total traveling distance, and thus the placement time of the PCB. Simultanouesly solving the feeder assignment and component sequencing problems are considered also by **Loh et al. (2001)**. The Manhattan distance is used in their methodology for simulating the movements of the Quad IIIc insertion machine. Their methodology performs better compared to the GA-based approaches of **Wong and Leu (1993)** and **Rubinovitz and Volovich (1994)**.

Some radial placement machines working with the pick-and-place principle have the potential to cause component damages due to the design of the placement head. **Duman and Or (2004)** present a precedence constrained TSP for solving the placement sequencing problem for this type of machine. The solution methodology ignores the precedence constraints in the initial stage, solves the problem as a pure Chebyshev TSP by applying convex-hull and or-opt algorithms, and finally applies a procedure to eliminate the damage conditions in the resulting TSP tour. **Duman and Or (2007)** observe the performances of different local search and metaheuristics – including tabu search (TS), simulated annealing and genetic algorithms – for the quadratic assignment problem arising in the feeder configuration. They are concerned with a class of placement machines encountered in insertion technology, in which the placement head is fixed, the board carrier is moveable in two dimensions, and the feeder carriage is linear and moveable in one dimension.

In the last two decades, chip shooter type machines have been broadly used in the PCB assembly industry. This trend has motivated researchers to focus on this type of machines and deal with optimization problems arising for this kinematics category. **Leipälä and Navalainen (1989)** discuss the insertion sequence and feeder assignment problems for a chip shooter machine. The insertion sequence problem is considered as a three-dimensional asymmetric TSP and the feeder assignment problem is presented as a QAP. Their model includes the assumption that only one feeder is used for each component type. The core of the algorithm is a basic improvement method for the magazine layout with an integrated farthest insertion heuristics for the insertion sequence. They observe three different methods for the creation of an initial magazine layout: a minimal spanning tree (MST) approach based on the Chebyshev metric, a TSP formulation, and a method based on the assignment of feeders according to their intensiveness on the board. **Sohn and Park (1996)** observe the similar machine and present an algorithm which is based on the one from Leipälä and Nevalainen (1989). The placement time for a component is considered as the maximum time required for the rotation of the turret, and the movements of the PCB table and the component magazine. In the first part of their heuristics, they find and an initial allocation of component feeders and a placement sequence of components on the board, but give more focus on the second step, where the initial solution is improved with pairwise exchange of the feeders.

Crama et al. (1996a) observe the component retrieval problem for the chip shooter. They generate an algorithm, which creates an optimal solution for the component retrieval problem by formulating it as a PERT/CPM problem with design aspects. This network problem is presented as an *NP*-hard problem and solved for very small problem instances with use of dynamical programming.

Or and Demirkol (1996) select an initial feeder configuration for a chip shooter machine either randomly or by using the current industry solution. They solve the problems of placement sequencing and magazine layout sequentially. The placement sequence is generated with a standard application for TSP with subsequent improvement heuristics. The problems of placement sequencing and magazine layout are iterated until a given limit or convergence point is arrived.

Horak and Francis (1995) take two subproblems into consideration: the placement sequencing problem and the sequence mending problem. The placement sequencing problem is formulated with Hamiltonian cycles using a Chebyshev metric. This is then used as an input for a greedy heuristic to solve the sequence mending problem, which determines both the order of placement of components on the board and the order in which the components should be arranged on the carrier. An adaptation of the well-known nearest neighbor heuristic is used for solving the sequence mending problem.

De Souza and Lijun (1995) deal with a chip shooter with 16 heads. They address the task of achieving high yield through determination of an optimal placement sequence given the constraints of feeder arrangement and machine design, and present a prototype planning system based on the artificial intelligence concept.

Moyer and Gupta (1996) focus on the magazine location problem and propose two heuristics for a chip shooter machine. Their goal is to minimize the magazine travel distance over an assembly assuming that the placement sequence is already given. The first heuristic presented in this paper assigns the feeders based on the transitions between the component types. The second methodology starts with an initial feeder assignment and applies pairwise exchange of feeders to achieve an improvement in the objective function.

The research in **Bard et al. (1994)** covers the component retrieval problem in addition to other problems discussed above for a chip shooter machine. Hence, the usage of more than one feeder per component type is allowed. A Lagrangian relaxation scheme is applied to the quadratic integer problem formulation, which is then divided into two subproblems and solved by a dynamical programming approach. The component insertion problem is formulated as a TSP model and solved with the nearest neighbor heuristics. Their results confirm that there is a strong relationship between the above mentioned subproblems.

Yeo et al. (1996) apply a rule-based frame system to generate the component feeder arrangement and placement sequence for a chip shooter. They use artificial intelligence programming and aim on minimizing the feeder movements in the applied one-pitch incremental feeder heuristics. The presented solution is based on the multi-feeder assignment, which is not always feasible to apply to real-life cases because of resulting high feeder investments.

There are also various GA applications on solving placement sequence and feeder assignment problems for chip shooters. **Khoo and Loh (2000)** formulate the sequencing problem as a multi-objective optimization problem under constraints. The aim of this work is to minimize the cycle time defined as the maximum time needed for PCB table movements, magazine movements, and turret indexing. **Ho and Ji (2003)** present a hybrid GA (HGA) comprising three different heuristics to solve the PCB assembly scheduling problem for a chip shooter machine. While the nearest neighbour heuristic is used to generate an initial solution for placement serquencing problem, 2-opt local search and iterated swap procedures are applied to feeder arrangement and component placement links for each new offspring generated by GA operators. The HGA presented in **Ho and Ji (2005b)** focuses only on placement sequencing problem and starts with an initial population in which chromosomes are generated either randomly or using the nearest neighbor heuristic. An iterated swap procedure is applied similarly to minimize the fitness function value of each chromosome. The results of the experiments reveal that creating initial chromosomes with a nearest neighbor heuristic is prior to random generation of initial placement sequences. The component retrieval problem, which arises in case of assigning two feeders of same type on the magazine, are solved by modifying the HGAs from **Ho and Ji (2006)**. However, all above described papers examine only small-sized PCBs with 10 component types and around 50 operations to evaluate the HGAs.

As described in section 3.4, the rotational speed, and hence the cycle time of a chip shooter machine depends on the weights of the components to be assembled. Hence, grouping and internally sequencing the components of each weight category in order to reach faster placements is observed by **Duman (2007).** Placement sequencing and feeder allocation problems are solved by defining TSP routes for each weight category. After placement sequencing is solved by applying convex-hull and or-opt algorithms from Duman and Or (2004), an internal arrangement of component feeders within each category is generated by applying a heuristic approach.

The main optimization problem for modular placement machines is assigning the component feeders to different modules with the objective of balancing the workload among the placement heads. The optimization of the operations of a modular placement machine has been investigated by **Grunow et al. (2003)**. They present an integer programming (IP) model and two different heuristic approaches. In the first stage of both heuristics, a feasible solution with respect to the limited component magazine capacity at each module of the placement machine is determined by using priority rules. One of the heuristic solutions applies a de-bottleneck approach by reassigning components in the second stage. Alternatively, an exact solution with an IP model is also formulated for the solution of this problem.

The optimization problems of a beam-type placement machine with parallel located nozzles have been observed by **Crama et al. (1996b)**. The optimization problem is divided into subproblems – from equipment and component allocation problems on an assembly line to the well-known subproblems of a single machine. They describe the complexity of these subproblems and offer different heuristic methods for their solutions.

Lee et al. (2000) use genetic algorithms for a joint-solution of the optimization problems in case of a multi-head beam-type placement machine. The multi-head placement problem is reduced to a single-head case by grouping component feeders and clustering components to be placed. Hence, Lee et al. (2000) apply standard single-head methods to the multi-head case. A partial-link concept is used for the structure of the chromosomes and the Euclidian distance is used for calculating the distances. **Hong et al. (2000)** investigate the implementation of a biological immune algorithm for the same problem.

Sun et al. (2005) propose a hybrid methodology including GA for optimizing operations of a dual-gantry multi-head placement machine. In their approach, GA is used for solving the component allocation and feeder assignment problems in combination with a simple greedy heuristic for balancing the workload between the placement heads. Their approach is to utilize simultaneous pickup operations by focusing on feeder arrangement. The placement positions for each work cycle are not considered explicitly and approximated to a center point.

The dual-gantry beam-type placement machine observed in **Choudhury et al. (2007)** is equipped with cameras mounted at the center of each magazine and a nozzle-change rack for each placement head. The objective of this research is to minimize the cycle time by balacing workloads assigned to heads which is solved by several list processing heuristics.

Although there has been a considerable interest in analyzing different types of assembly machinery, only a few research papers have been published which focus on optimizing the operations of collect-and-place machines with rotary placement heads. For instance, **Altinkemer et al. (2000)** have developed a model, which analyzes the feeder assignment and placement sequencing problems. They present an integrated approach which is solved with the Lagrangian relaxation of the optimization model. Unfortunately, this approach is based on some rigid assumptions. For instance, they assume that separate tours are created for each component type and neglect the rotational cycle time of the placement head. The principal limitations also hold for the paper by **Kazaz and Altinkemer (2003)** which expands the scope of Altinkemer et al. (2000) merely by allowing for multiple setups of the same component type.

Another investigation of collect-and-place machines has been conducted by **Grunow et al. (2004)**. This study presents a three-stage heuristic approach for efficiently solving the problems of a real single-gantry collect-and-place machine. In the first stage, feeders are assigned to magazine slots using the neighborhood values determined from a minimum spanning tree. Given this feeder assignment, heuristics based on well-known savings algorithm are used for finding a placement sequence. Finally, a 2-opt exchange procedure is applied to improve the obtained feeder assignment and component placement sequence. **Kulak et al. (2007a)** apply GA based methodologies for solving the machine optimization problems of both single- and dual-gantry collect-and-place machines. The feeder assignment and placement sequence for each machine type are presented in partial-link chromosome structures. Novel clustering algorithms are integrated into the GA solution in order to generate good initial chromosomes and increase the efficiency of the GA operators.

Some research on PCB assembly problems study different types of machines or are independent of a machine type, and hence find a general application. **Khoo and Ng (1998)** and **Khoo and Ong (1998)** propose GA approaches for optimizing the placement sequence in semi-automated PCB assembly. However, they do not consider the feeder assignment problem and the proposed algorithm is tested only on small problem instances. **Egbelu et al. (1996)** investigate four design alternatives of a robotic arm system. They first develop an initial layout of the magazine with simple algorithms and then solve the component insertion sequence problem with a composite procedure of farthest insertion heuristics. The obtained solution is improved with a 3-opt algorithm using the Euclidean distance measure. **Leu et al. (1993)** present a GA approach, which finds the sequence of component placements and arrangement of feeders simultaneously. This model is applicable on THD assembly machines, pick-and-place machines or chip shooters. **Nelson and Wille (1995)** concentrate on methods like genetic algorithms, evolutionary programming and simulated annealing and compare the performance of these with each other. Both Chebyshev and Euclidian distance measures are used in the experiments for different tests.

Nozzle allocation problem has been merely investigated for machines with beam-type placement heads. **Magyar et al. (1999)** deal with single machine optimization problems on a general surface mounting machine equipped with a beam-type placement head with four spindles. The component placement is organized into blocks called placement groups using local search algorithms. Given this placement sequence, a feeder setting minimizing the total time is calculated. The placement head picks up the components from the feeders in the same order as they are in the placement group. A proper nozzle-spindle combination is defined by using primary and secondary nozzle information for each component type to be mounted. If some of the nozzles attached to the spindles do not match the primary or secondary nozzle of the component in a placement tour, then the nozzle must be changed prior to the picking operation. Hence, a heuristic approach for sequencing placement groups to reduce number of nozzle changes is presented. The so-called gang-pick is also integrated, where several components are picked up from neighboring feeder slots simultaneously to reduce constant pickup costs.

Raduly-Baka and Knuutila (2007) address an optimal nozzle selection problem for the same type of machine described above. Each component type can be picked and placed by a certain nozzle type although a single nozzle type may support the placement of multiple component types. The original problem is extended to a cost limited nozzle selection procedure if the limited nozzle budget is exceeded by the optimal allocation of the nozzles. Efficient greedy algorithms are presented for optimally solving both described problems. Heuristic solutions are also investigated for the selection of nozzles in case of multiple PCB types.

Ayob and Kendall (2004) investigate the same problem for a beam-type placement head with two nozzles and four fixed feeder carriers on four sides of the machine. They present an approach which minimizes the nozzle changes in order to minimize the total assembly cycle time. This approach is extended in **Ayob and Kendal (2005b)** by using a weighted rank procedure where the nozzle pairs are ranked based on their effectiveness for enabling simultaneous operations, e.g. simultaneous pickup, same feeder bank pickup, same component feeder pickup, etc. Unfortunately, both approaches make use of the average processing times data given by the machine vendor, which disregards the actual allocation of the placement and pickup locations.

Knuutila et al. (2007) focus only on developing a nozzle usage policy with the assumption that the component placement order is already decided. They disregard the exchange of nozzles during the placement operations of a PCB because the placement arm can hold simultaneously one or several nozzles for each component type the PCB contains. The nozzle usage policy is formulated as a minimal length pickup sequence problem and solved with a greedy algorithm which tries to reduce the number of pickup tours.

All of the papers described above focus on the problem of selecting the number of nozzles to be assigned on a beam-type placement head. For the case of a collect-and-place machine equipped with a rotary placement head, the assignment of each nozzle to each specific head segment is of great importance, because it has a direct effect on the problems of pickup and placement sequence. Although this topic is relevant for practical problems where a variety of components have to be assembled by a single placement machine, it has been previously ignored in the literature dealing with the machines of this type. In this study, the problems of nozzle selection and assignment are integrated for the first time into the machine-specific algorithms for this type of machine. These algorithms are presented in section 5.4.

4.3.2 Minimum Setup Strategy

Minimizing the changeover time receives the main focus in planning activities for a high-variety low-volume production environment where the changeover effort is more dominant than the placement time. Placement machines are equipped only with a limited number of feeder slots. Therefore, not all the components required for different PCB jobs can be loaded on the machine at once. This requires the allocation of missing feeders on an assembly machine before starting the assembly of a new product. Minimum setup strategy tries to sequence boards by producing similar products after each other such that the total changeover effort is minimized. Given the feeder assignment for each product, the placement sequence and herewith the placement time can be determined for each board.

The main idea of job sequencing in minimum setup strategy has arised similarly from the FMS field. **Bard (1988)** has developed an efficient procedure for sequencing jobs on a single machine capable of automatically interchanging tools. In FMS, the processing times are assumed to stay constant. This does not hold for the PCB assembly where placement time is affected directly by the decision of feeder allocation. Hence, the objective of this paper, namely minimizing the makespan for a fixed set of jobs, translates into minimizing the overhead associated with changing over from one job to the next. The author formulates a nonlinear model formulation which proves to be too complex to solve for large problems. Therefore, a heuristic approach, which determines a job sequence by applying Lagrangian relaxation on the optimization model and keep tool needed soonest (KTNS) policy for finding a feasible loading, is presented.

Tang and Denardo (1988) address the same problem like Bard (1998) and select the minimization of the total number of tool switches as the performance criterion for the presented mixed-integer model formulation. They observe the tool replacement problem, i.e. how to find the best tool sequence with a fixed job sequence, and present a heuristic procedure finding a solution which is locally optimal in the sense that improvements cannot be found by varying either the tool sequence or the job sequence. Their heuristic consists of three major steps. The first step generates a good job schedule by finding a short Hamiltonian path on a complete graph where each job is defined as a node and the length of each arc represents an approximation for the number of tool switches. In the second step, the total number of tool switches required by the determined job schedule is calculated by using the KTNS policy. The third step consists of finding a job schedule that may incur a fewer number of tool switches than the current best job schedule. This can be achieved by selecting the best job sequence out of all job sequences which can be performed by the same sequence of tools used to process the current best job schedule. They present an iterative greedy procedure for the above described process which terminates when no improvements in job sequence or number of tool switches can be found.

Jain et al. (1996) focus on assembly lines where a high-mix of boards is produced in low volumes. They observe the high-speed pick-and-place machine on the line, which usually requires far more time to set up than the others. It is assumed that the feederbank is completely occupied in the beginning. The problem is to find a sequence that minimizes the number of component changes. The sequencing problem is formulated as a nonlinear binary integer problem similar to Bard (1988) and Tang and Denardo (1988). However, Jain et al. (1996) criticize some assumptions of the previous models and formulate a new solution approach which allows using component feeders of different sizes. To model this problem, they extend the former formulation by adding a feeder slot subscript and by breaking the feeders into three categories. Because the initial feeder assignment is defined by the last assignment from the previous set of boards, a rolling horizon approach is applied. A fast heuristic which can handle large industrial problems is developed as a compensation for the presented optimization model. The same general multiphase approach from Tang and Denardo (1988) is followed with several important modifications including application of local search improvements and a fourth step for generating a good initial setup for the next rolling horizon.

Crama et al. (1994) represent the minimum setup problem as a tool-job matrix. The tool switching problem is defined as determining a job sequence and an associated sequence of loadings for the tool magazine, such that all tools required by the *j*th job are present in the *j*th loading, and the total number of tool switches is minimized. They assume that the tool magazine is always fully loaded and each tool fits in one slot. The time to remove or insert each tool is constant and same for all tools. They prove that this problem is already *NP*-hard for the magazine capacity of two slots and an optimal sequence of tool loadings can be found in polynomial time for each fixed job sequence. Hence, the problem is decomposed into two interdependent issues, namely computing a job sequence, and determining which tools should be loaded in the tool magazine at each moment for the given sequence in order to minimize the total number of setups required. They propose six basic heuristic approaches for solving the *NP*-hard tool switching problem which fall into two main categories: construction and improvement strategies. The cost of each job sequence is calculated using a KTNS heuristic. These heuristics are tested on random instances for evaluating the performance of different proposed heuristics.

Kim et al. (1996) propose several heuristics – including tabu search and simulated annealing methods – for solving a generalized flowshop scheduling problem with the objective of minimizing mean tardiness, i.e. total tardiness of jobs for final products using the due dates given by the input schedule of the assembly line. An appropriate parameterization is determined for each metaheuristic after a number of initial tests. The performance of metaheuristic approaches is then compared with basic heuristics (e.g. neighborhood search) for finding initial job sequences in a series of computational tests. They also introduce a new algorithm called the rolling block optimization where job sequences are segmented into a number of blocks for improving the search for better job sequences. In each iteration, the sequence within a block is modified while the sequences in other blocks are fixed. This procedure is repeated for all blocks sequentially. Also a backward rolling method where a sequence obtained from forward rolling is examined starting from the last block is presented. The performance of the algorithms is evaluated using randomly generated test problems. In this research, the SA approach was found to be superior compared to other approaches.

Two alternative linear programming (LP) formulations for workload planning of a bottleneck station in small-lot printed circuit board assembly are addressed in **Günther et al. (1997)**. The primary objective of both models is to maximize the throughput. The detailed model tends to underestimate the number of setups involved because it is assumed that each component type is set up at most once for the daily mix of jobs. Since it is computationally cumbersome to consider the detailed component setup, an aggregate model which does not include decision variables for component setup is presented as an alternative. The aggregate model does not pay attention to the fact that a certain component type may have been set up previously for a different PCB type, and thus overestimates the number of setups. In order to reduce the aggregational error, a fuzzy LP model has been developed. This approach specifically exploits the component commonality among jobs in the job pool and tends to select board types which are rather similar to each other. A heuristic solution procedure based on priority rules and the keep tool needed soonest (KCNS) policy is used as a benchmark for observing aggregational errors of the LP formulation.

Gronalt et al. (1997) develop a heuristic for component switching on SMT machines. In the first stage, they determine the component setup for a given sequence of board types to be processed on a single placement machine. In the second stage, component feeders are assigned to slots in the magazine of the placement machine. They also consider the fact that component feeders may require a varying number of slots in the component magazine of the placement machine, which has been ignored in previous literature despite its significant practical relevance. Due to small-lot assembly, they aim at increasing machine utilization by focusing on the scheduling variables which influence the setup and rearrangement operations. The job sequencing problem is not outlined in greater detail and assumed to be given since this is discussed in Günther et al. (1998). In their two stage approach, a recursion is necessary if no feasible solution is obtained for the assignment problem. The heuristic then reverts to the first stage, tightens the capacity restriction by decreasing the magazine capacity by one slot, and tries to resolve the feeder slot assignment problem of the second stage. The recursion process is continued until a feasible solution for feeder assignment can be found. The effect of reducing the capacity in the first stage is the reduction in the number of consecutive jobs in which feeders remain unexchanged in the magazine. The components are first tried to set up on empty slots. When no empty slots are available, new feeders are switched with feeders which are not required for any of the remaining jobs. If these policies are not sufficient for assigning the required feeders, the KCNS policy is applied. Above described procedures are executed for all remaining jobs in sequence.

Günther et al. (1998) focus on the problems of job sequencing and component setup with the objective of minimizing the makespan by generating a good job sequence and reducing the total changeover effort. Similar to Gronalt et al. (1997), component feeders may allocate more than one magazine slot. Both the arrival time for each job and the precedence constraint for the primary and secondary sides of the same board are considered in this paper. The job sequencing process is coupled with the individual arrival time of each board. Moreover, some jobs reenter the SMD station since components have to be mounted on both sides of the board. A pair of jobs belonging to this class requires a delay between the completion of the first job and the start of the second job. Accordingly, the objective of this paper is not only to minimize the setup time for the replacement of component feeders, but also to minimize the makespan for processing the entire batch of the assembly jobs. To solve this scheduling problem, the decision problem is decomposed into subproblems of job sequencing and component setup. Additionally, an efficient feeder assignment heuristic is developed to solve the third subproblem, i.e. assignment of component feeders to individual positions in the magazine for the case of varying feeder widths. The problems of job sequencing and component placement resemble finding the shortest path in the TSP except that the distances are not known yet. Hence, the component commonality between any pair of jobs is used as an estimate for the setup effort incurred when switching between two jobs. Both an optimistic (lower bound) and a pessimistic (upper bound) estimation of changeover times between two consequent jobs are presented and the upper bound estimation is selected for the presented application. The estimation of changeover times allows using a standard TSP approach for modeling the job sequencing problem. The setup time required for the reflow soldering process is also considered in the solution approach. To solve the resulting TSP with additional arrival time and precedence constraints, a heuristic procedure which finds an initial solution greedily and improves it with 2-opt search is presented. Given the sequence of jobs to be processed, the component setup between two jobs is determined by applying a KCNS policy which is adjusted for the case of varying widths. In addition to the heuristic which applies these three steps sequentially, an iterating heuristic is also presented.

Similar to Günther et al. (1998), **Hirvikorpi et al. (2006)** also investigate the component switching problem as a reconsideration of the KTNS rule enabling component feeders with different widths. The problem is to minimize the total cost of replacing component feeders when the sequence of processing PCBs is known in advance. In the first step, a heuristic is presented for solving the general problem with feeders of identical width. Given the job sequence, a distance measure is defined for each component type which gives the number of periods until it will be used again. A heuristic selection procedure returns a set of feeders, which is valid for removal and delivers the maximum weighted distance. The best fit method, i.e. selecting the smallest space possible for the feeder to be inserted, is used for selecting the best position for the insertion. These procedures are integrated into a heuristic solution, which checks for each new job if all required component feeders are allocated on the magazine. The heuristic approach for the general problem is extended for the case of varying feeder widths with additional "remove" and "move" operations. Remove operations are required when there is not enough empty space for a feeder to be inserted. Moves are necessary when it is not possible to remove enough feeders because they are also needed during the current time period. The first heuristic approach proposed for solving the multi-width case allocates a large continuous area for the feeders to be inserted adjacently, whereas the second heuristic makes a distributed stepwise allocation which suits better if the magazine is fragmented. The performance of the presented heuristics is compared with a lower bound and two primitive heuristics.

Ashayeri and Selen (2007) consider job sequencing and component allocation problems for the daily production of a machine line consisting of several types of placement machines. Different feeder types and widths are used in the observations of this paper. Two different strategies with distinctive objectives are presented. The first strategy which aims at minimizing changeovers consists of three stages. In stage one, a sequence that maximizes the sum of commonalities between successive jobs is found by applying a maximal insertion method for the TSP formulation. Next, components of each job are allocated to machines and feeders using an IP formulation. The objective of this model is the maximum resemblance with the previous allocation. In the last stage, the KTNS policy is applied for each job-allocation combination in order to save changeovers. The second strategy aims to minimize processing times and comprises similarly three stages. In the first stage, the same IP allocation model is solved, but this time the objective is to minimize the assembly time per job. Next, balancing procedures are applied to each job to reallocate components in order to achieve a better processing time. Job sequencing problem is solved in the final stage by redefining the commonality as the number of shared component-machine-feeder allocations.

Rajkumar and Narendran (1998) use basic group technology principles for the problem of sequencing PCB assembly jobs. Existing methods, which group and then sequence jobs (so-called group-and-sequence approaches) and allow splitting jobs into different groups with a multiple-loading strategy, are evaluated using small examples. In contrast to previous approaches which compute similarities between pairs of PCBs, group technology rules are used for searching similarities between the current setup and remaining PCBs to be sequenced. Hence, two PCBs which are most similar in terms of component type requirement are initially chosen as head and tail PCBs for the job sequence.[82] The PCB requiring more component feeders is selected for the head position because of the assumption that it may cover more component feeders required for the subsequent jobs, and hence reduce the number of component switches. Other PCBs are appended progressively using a PCB index which is calculated by dividing the size of the PCB (number of component types) by the number of extra components to be mounted. The real component assignment is calculated using the KTNS policy. Splitting a PCB job is only allowed if it results in a net saving of the makespan.

Van Hop (2003) criticizes the sequential solution methods for board sequencing and component loading problems and develops a new approach which simultaneously tackles these interdependent problems. In this paper, different heuristics are presented and their performances are evaluated. The first heuristic is a TSP-based approach which modifies the algorithm from Günther et al. (1998) merely by solving a cost matrix based TSP for board sequencing instead of selecting the first board from the actual component setup. The second heuristic, the so-called improvement heuristic, consists of two phases. In the construction phase, an initial job sequence is determined using the heuristic approach from Rajkumar and Narendran (1998). This is improved in the second phase via deleting the link in the sequence showing the highest component change and reinserting this job into the sequence where it has the highest number of similar components. The magazine status for each position in the queue is determined by the KTNS rule. The constructive heuristic which is developed in this paper establishes the board sequence and component loadings simultaneously. This is a greedy procedure which sequentially adds boards into the sequence and calculates the component loading with the KTNS policy for each candidate. If no more slots exist for a required component, components with a lower degree of requirement for the next jobs are removed first. This is apparently only possible under the strong assumption of using only feeders of a constant width. Finally, a

[82] Jaccard's similarity measure is used for calculating the similarities.

composite heuristic which distinguishes from the improvement heuristic in terms of alternating starting solutions is presented.

Some of the minimum setup investigations focus on assembly systems with parallel machines. Hence, grouping approaches are mostly applied first to allocate jobs to different alternative machines. **Rajkumar and Narendran (1997)** extend the algorithms from Rajkumar and Narendran (1998) to the case of multiple machines by adding some new rules into their heuristics. Hence, a similarity based heuristic using objectives of minimizing the makespan and balancing the workload is presented for the problem of loading PCBs on alternative placement machines. Splitting a PCB batch between several machines is not permitted. The number of groups is decided by the number of placement machines available during the planning period. The proposed approach comprises the steps of choosing initial PCBs, and allotment of other PCBs based on a clustering scheme. Jaccard's similarity coefficient is used for defining the similarities. Seeds for each group are determined by picking PCBs with large component type domains, which at the same time are distinctive from each other, in order to achieve a better feeder distribution among machines and reduce required duplications of component feeders. After the seeds are selected, remaining PCBs are sorted in the descending order of number of component types and allotted to the placement machines. The feeder allocation problem is solved similarly using the KTNS rule for each determined job sequence on each machine.

Van Hop and Nagarur (2004) observe the scheduling problem of multiple boards on several non-identical parallel machines at a single production stage. The task of minimizing the number of component switches is divided into three problems. The first problem, i.e. grouping, separates n boards into m groups to be allocated to m machines. Similar boards are grouped together with the objective of reducing the setup time and balancing the workload between machines. The second level problem deals with sequencing a given set of boards for a machine. The component switching problem, i.e. component unloading/loading on the magazine, constitutes the last problem. A composite genetic algorithm is developed to solve this multi-objective problem. The integrated solution is encoded as a string of pair values for each group of boards. The first number indicates the board membership in a group, and the second one represents the sequencing position of a board in that group. A new population of solutions is generated by using different genetic operators for grouping and board sequencing. The fitness function consists of a weighted sum of multiple objectives, i.e. workload balancing, board similarity and total setup time. Numerical analyses show that the presented solution methods are efficient, and results are obtained within a reasonable amount of time.

Reducing the total changeover time for an online setup system in a high-variety low-volume production environment can also be achieved by grouping similar PCB jobs. Hence, the main difference between previous approaches and the following literature is that job sequencing is applied to groups of jobs rather than single PCBs. Although the following approaches resemble the application of a group setup strategy, they focus on the problem of job sequencing with the objective of minimizing only the setup effort. Therefore, these approaches will be discussed as a special form of the minimum setup strategy.

Sule (1992) developed a heuristic procedure to minimize the changeover of component tapes on sequencers. The sequencers have the task to arrange components on an output tape in a sequence to assemble a certain PCB. Each sequencer has a feeding mechanism called dispenser which has slots occupied with input tapes of a component type. Hence, the problem of minimizing the changeover of input tapes is similar to the feeder allocation problem in modern PCB assembly. The solution procedure is divided into two phases. In the first phase, using the concept of group technology, similar components are grouped together using closeness ratios for defining relationships between component types in a group. In the second phase, heuristics based on usage of planning and loading tables are applied for scheduling the PCB production and assigning component tapes to sequencer positions.

Crama (1997) focuses on the tool switching problem in flexible manufacturing systems and addresses several objective functions for one-machine loading problem. He tackles three of the most basic objective functions, namely maximizing the number of parts in a feasible batch, minimizing the number of tool switching instants, and minimizing the total number of tool switches. Both linear models and a nonlinear knapsack model for solving the above described problems are presented. The objective of the optimization problem is formulated as selecting a feasible batch, i.e. a subset of parts that could be produced without any tool switches. A broad literature review of this problem and different model formulations are given in this paper.

Garetti et al. (1996) observe a scheduling system, which aims at minimizing the makespan. This goal is achieved by reducing setup and idle times of the machines. The assembly times are considered as deterministic and the setup time as the mean value of a statistical distribution. They create groups of board mixes (no-setup mixes) where no setup is required between the batches within the same group. The goals of the scheduling system are the minimization of the maximum operating time and the system setup time.

Hashiba and Chang (1991) present a decomposition approach including steps of grouping and ordering PCB jobs and assignment of their components. The objective is to generate the minimum number of groups using component commonalities between different board types. These groups are then ordered by solving a TSP with arcs defined using the Hamming distance. A heuristic approach similar to KTNS is proposed for the component assignment problem. In **Hashiba and Chang (1992)**, a simulated annealing approach with an embedded component assignment heuristic is presented as an alternative to the previous research.

Similar to approaches above, **Rosetti and Stanford (2003)** also developed a method based on grouping and sequencing PCB jobs on the bottleneck machine of an assembly line. In the particular application area, the solution to the component-feeder assignment problem is assumed to be given. The expected number of setups between each PCB is calculated both using the Hamming distance and an estimation with a probability function. Given a fixed number of groups, PCBs are assigned to each group and sequenced with a simple nearest neighbor heuristic based approach using sequence-dependent setup times. Detailed analyses show that the sequences based on the expected setup distance measure perform significantly better than the sequences utilizing the Hamming distance.

Bhaskar and Narendran (1996) also focus on reducing the total PCB and component setup times and present a linear binary integer programming model, which is an adaptation of the nonlinear MIP presented by Maimon and Shtub (1991). The problem of ordering PCB groups is approximated to a TSP. In this formulation, PCB groups are considered as cities and the distance between two PCB groups is taken as the number of component changeovers required while changing from one group to another. Because of the *NP*-complete nature of this problem, a heuristic based on a maximum spanning tree (MaST) approach is presented for grouping a set of PCBs. The component-PCB incidence matrix with the entries denoting the number of different types of components to be assembled on each PCB is the input to the problem. Using this incidence matrix, distances between each PCB is calculated by the cosine similarity coefficient. The resulting MaST problem is solved by Prim's algorithm. Initially, all PCBs form a single group which results in an infeasible solution violating the capacity constraint. An iterative procedure increases the number of groups progressively by deleting the arc with minimum similarity and checking the capacity constraint at every iteration. Obtaining the first feasible solution with this approach constitutes the first stage. In the second stage, possibilities of splitting PCBs among groups are explored. If any of the initial groups contains only one PCB, the possibility of splitting the components of this PCB into two other groups in order to save one group setup time is evaluated. Splitting a PCB is allowed only if the extra feeder exchange effort caused by adding split PCBs into existing groups is less than a group setup. This procedure is repeated until the results converge.

Smed et al (1999) also divide boards into groups which are sequenced by the production engineer on the bottleneck machine for a low-volume high-variety manufacturing environment. The objective is to minimize the number of setup occasions (i.e. the number of instants the setup operations occur), and hence reducing the number of job groups. The observed placement machine is equipped with four feeder carriages from which two are set up with most frequently used components for the complete board mix and fixed for the assembly of total PCBs. Products are classified into groups according to their closeness, i.e. the amount of mutual components. The rest four carriages are then allocated with the custom components required for assembling each group. While one group is assembled, the remaining two carriages are occupied with the custom components of the next job. The groups are sequenced based on the due dates. The grouping problem is formulated as a mathematical optimization problem. Several heuristics are presented which are based on different clustering techniques and local search methods. The effectiveness of the new system is demonstrated on a real industrial application and significant improvements achieved in comparison to the previous unique setup strategy are presented.

Smed et al. (2003) discuss a hybridization of the group setup and minimum setup strategies in their paper. While the number of product groups are intended to be kept low, they also try to minimize the number of required changeovers. The objective function is a weighted sum of these conflicting objectives, i.e. the number of job groups and required feeder changes. They present three group setup and four minimum setup algorithms based on well-known approaches from literature and modify these for developing a group setup with minimum feeder changes.[83] The grouping approaches are modified by calculating the amount of feeder changes between the groups, which is determined for a fixed permutation of groups after applying the KTNS rule. In contrast to the grouping-based approaches, presented minimum setup strategies first define a sequence from which groups can be identified. By scanning the PCB permutation, these heuristics next fill up the magazine with component feeders until its capacity is exhausted. A new hybrid algorithm is also presented in this paper, which creates PCB groups using the objective of reducing the number of groups and sequences them using a minimum

83 Details of these approaches are presented in Smed et al. (2000).

setup heuristic. Performances of the heuristics are tested under different parameterization for the parts of the objective function.

Salonen et al. (2006) consider the same problem and extend the heuristic approaches from Smed et al. (2003) with two new hybrid algorithms. In contrast to the hybrid approach from Smed et al. (2003), the new hybrid approach terminates the grouping process when similarities between groups drop below a threshold value. In the next stage, created groups are sequenced and the required feeders for each job are assigned using the same minimum setup heuristic presented in previous papers. The second hybrid approach calls the minimum setup heuristic each time after merging two groups and evaluates the cost of grouping. If the new grouping proves to be better, the grouping solution is accepted. The advantage of this method is that it searches more globally than the first hybrid approach, which proves to yield better solutions in the numerical experiments.

4.3.3 Group Setup Strategy

Group setup strategy is adequate for a high- to medium-variety of PCB types produced in small to medium batch sizes. In such a production environment, the savings in placement time by organizing the component feeders for each single PCB type individually may not compensate the changeover effort. Frequent setups are also not favored because of qualitative reasons which may lead to losses in productivity. In this case, similar PCBs are grouped together into families, components of which are setup at once similar to the unique setup strategy. An assumption of the group setup approach is that no major setup is required when changing from one board to the other within a family.

Although conflicting objectives of minimizing placement times for each PCB and total setup effort have to be in the focus of group setup approaches, there are a number of studies in the literature which only focus on reducing the number of groups, and hence the number of setup occasions. **Carmon et al. (1989)** are among the first to apply group technology concepts on PCB assembly and discuss the term group setup strategy for high-mix low-volume environment. The idea is to classify products into groups which make use of similar components. They refer to the term sequence-dependent scheduling (SDS) which is based on sequencing jobs requiring same resources for eliminating much of the setup between them. Because the sequencing problem resembles the TSP, which is *NP*-complete, they propose a group setup (GSU) method as a new approach to reduce the overall setup time and to increase the production flow in PCB assembly. In GSU, products are divided into groups, each of which is produced in two stages. In the first stage, the common components are set up and assembled for the whole group. The next stage, which is referred as residual setup and production, requires the separate setup and assembly of the remaining components on each product. The traditional production strategy (unique setup) and the GSU method are compared using throughput, labor time and production makespan as performance measures. The complex SDS problem discussed in Carmon et al. (1989) is investigated also in **Maimon et al (1993).** They compare the unique setup and GSU methodologies from Carmon et al. (1989) with a new SDS method which sequences PCB jobs to reduce the WIP levels. Three performance measures are used for comparing these approaches: the line throughput, the average WIP inventory level, and the implementation complexity.

Maimon and Shtub (1991) present an exact MIP formulation and a fast heuristic for grouping a set of PCBs. In the mathematical formulation, a PCB or a component can be a member of more than one group, i.e. the procedure allows for multiple loading of PCBs and components. The objective is to minimize the total setup – sum of each component and PCB setup – subject to the machine capacity constraint. A multi-stage heuristic approach searches the components/boards matrix and assigns components to groups using a greedy procedure. Hence, some PCBs may be split into more than one group, which is controlled with a threshold value.

Shtub and Maimon (1992) investigate the problem of grouping PCBs as an extension of the set-covering problem. A general approach based on cluster analysis and measure of similarity between PCBs is suggested as a basis for PCB grouping. The objective of the PCB set-covering problem is to minimize the total setup time for PCBs and components by reducing the number of groups. Although an application of an offline setup strategy is not discussed, sequencing of groups is not considered. Because the set-covering binary integer program is complicated to solve, cluster analysis and similarity measures from GT are applied. The presented approach is based on Jaccard's similarity measure and applies a single linkage rule for clustering boards. Similarities between PCBs are represented as a component/board incidence matrix. PCBs are formed into groups in descending order of pairwise similarities considering the capacity constraint of the magazine.

Luzzatto and Perona (1993) present a heuristic approach which aims at concurrent optimization of setup operations and load balancing. The grouping procedure takes into account both qualitative (types of PCBs to be produced and types of components required by each PCB) and quantitative (production volumes of each PCB and the number of each component type required) aspects of the PCB mix. They consider a production line consisting of several work phases, however, the methodology obtains the solution of each work phase separately from others. The objective is to group PCBs in a number of cells which must be less than or equal to the number of machines available in each stage of the assembly process.

Daskin et al. (1997) present the partition-and-repeat (PAR) strategy which focuses on clustering component feeders rather than PCB jobs.[84] Hence, the component feeders required by a family of PCBs are partitioned into subsets satisfying the staging capacity of the magazine. The groups are configured to run each subset in turn, requiring the accumulation of all partially completed PCBs in the family. The advantage of this strategy is the reduced number of setups, whereas the disadvantage is that it requires every PCB in the family to be accumulated as a partially completed WIP. The objective is to minimize a weighted sum of the total number of times at which PCB types are switched, and the total of subset cardinalities. If it is both feasible and optimal to load each PCB onto the machine only once, the PAR strategy resembles the group setup approach. This case holds for the presented approach. The presented heuristic model adds unassigned PCBs to emerging groups in the ascending number of additional required components. A new group is initiated if the magazine capacity is exceeded. This procedure is repeated until all PCBs are assigned to a group. The initial grouping solution is improved using alternating swap and move operations until no cost saving can be achieved. Using the solution of this approach as a lower bound, they develop a branch-and-bound algorithm which optimally solves moderate instances. The main assumption made is that a group of components is completely removed from the machine before a new group is loaded. They also discuss about the possibility of leaving common component feeders on the machine for the next setup, but underline the fact that this would further complicate the problem by adding a sequencing problem on top of it.

Knuutila et al. (2001) discuss the job grouping problem for a bottleneck chip shooter machine in a high-variety low-volume environment. This paper discusses the problem of arranging the jobs of one machine into groups in such a way that job change costs, which depend on the number of job groups, will be minimized. They show how real-life instances can be solved by three different methods. The heuristic methods determine an initial grouping by merging singular groups using similarity criteria. After initial groupings are formed, further

[84] Cf. McGinnis et al. (1992) for details of the PAR strategy.

improvements can be achieved by swapping, merging and hill-climbing methods, which try to fill up the free space left in some groups. Several heuristics combining these activities are presented and compared with optimal solutions to prove their efficiency. In addition to efficient heuristics and binary programming approaches, they introduce the constraint programming approach into PCB job grouping problem. The heuristic approaches presented in this paper are fast and produce optimal or near-optimal solutions, whereas the binary model is only capable of finding optimal solutions for small problem instances and serves as a benchmark. The constraint programming approach can solve moderately large problem instances to optimality and has the great advantage of changing the problem formulation relative easily. Similar to Smed et al. (1999), a physical subdivision of feeder units into a number of feeder carriages is investigated for achieving further reductions in changeover time.

Modern placement equipments are capable of occupying different feeder types.[85] **Knuutila et al. (2004)** investigate the job grouping problem with multiple feeders of several types including tape, track, multi-tube and matrix tray feeders. The problem of job sequencing vanishes with the use of changeable feeder banks. The proposed heuristics are based on the job grouping algorithms presented in Smed et al. (1999) and Knuutila et al. (2001). Several similarity functions are evaluated and heuristic improvement rules are adjusted accordingly. These rules comprise basic modules which are integrated into greedy clustering, global and local search, and tabu search algorithms. Detailed computational analysis is given for comparison of search algorithms. A generalization of the grouping problem by formulating different feeder types as a special case of the standard grouping problem is given in **Hirvikorpi et al. (2005)**.

Similarly, **Yu et al. (2005)** have developed an integer programming approach to PCB grouping problem with the objective of minimizing the total setup time. A column generation approach is presented to solve the IP group setup model for a single pick-and-place machine. The original problem is decomposed into a master problem and a column generation subproblem. Starting with a few columns in the master problem, new columns are generated successively by solving the subproblem optimally. To solve the master problem, a branch-and-bound algorithm is used with the generated columns. Additionally, a new branching strategy is introduced for maintaining the consistency in the column generation procedure after branching. Detailed experimental studies are carried out in order to compare the results with the solutions from previous studies.

---

[85] Types of component feeders are presented in section 3.1.1.

Hillier and Brandeau (1998) focus on insertion of components using manual or semi-automated processes but their solution approaches can be applied to more general problems. They observe a high mix of boards produced in relatively low volume. The objective is not to minimize the makespan but rather to minimize the total number of labor hours required. This is achieved by an optimal allocation of jobs to different processes where components have different assembly times. This paper extends the approaches from **Brandeau and Billington (1991)** by finding an optimal solution to the problem and presents a heuristic that gives better results than either of the heuristics presented in Brandeau and Billington (1991). The heuristic approach starts with the initial feasible solution of the binary integer program, which is determined by a Lagrangian relaxation removing the capacity constraint. The objective of the model formulation is to minimize the total amount of time required to assemble the boards. It is assumed that the processing cost, i.e. setup and insertion times, is known for each board and component type in advance. The presented algorithm applies a linear relaxation to the model in order to obtain a lower bound, and a Lagrangian relaxation in order to obtain a feasible solution – an upper bound for the branch-and-bound approach. They propose also an extension of their algorithm for solving the multiple-machine problem.

Application of fuzzy approaches enables integration of some other soft criteria into the grouping problem. **Johtela et al. (2000)** describe a job grouping model for determining a loading strategy. The total setup time depends on the number of tool switching instants. Hence, the main objective is to minimize the setup times by grouping the products efficiently. The job grouping problem is extended by adding other soft criteria. They present a multiple-criteria decision making model consisting of hard constraints, which define the space of admissible solutions, and soft constraints, which characterize the quality of scheduling decisions. The main criteria examined in this study are track widths, double-sided PCBs, setup size, urgency, oven temperature, number of groups, and total setup. Soft criteria are used for selecting the best solution within a crisp set of feasible job grouping solutions. Each criterion is represented as a fuzzy set and aggregated to give an overall optimality measure of the solution.

Van Hop and Tabucanon (2000) present also a fuzzy multi-attribute decision-making model for grouping electronic components. They consider a number of component types which carry some design or assembly attributes. Each component is expressed by a vector of functions. Each PCB is assigned to a group using a specific selection process presented in this paper.

Williams and Magazine (2007) discuss the tradeoff between reducing the setup time and increase in processing time when batching PCB jobs. The observed pick-and-place machine consists of a placement head, which moves between fixed pickup and placement points, and a moveable component magazine and a PCB table, which position the next components to be picked up and placed, respectively. The movement of the table is ignored because it requires less time than the retrieval of the component. Hence, the placement time depends only on the locations of component feeders. They describe the optimal feeder assignment as an organ pipe configuration, i.e. the most frequently occurring components are allocated to the feeder slots closest to the home position with decreasing frequency implying further distance from the home position. During a setup, all component types for the previous group are removed from the machine and the component types required for the next group are installed in the feeder slots. The time to set up the machine is assumed to be constant and therefore independent of the set of jobs to be processed, although the authors describe operations of an online component setup. They develop four families of heuristics to solve the problem of job grouping. The clustering family is based on a set of hierarchical agglomerative clustering algorithms known from the cluster analysis literature. The bin packing family is analogous to the best-fit decreasing bin packing approximation algorithm. The remaining two heuristics are genetic algorithm based approaches. The so-called GASPP family utilizes a sequencing GA and a shortest path problem (SPP) algorithm for fitness evaluation. The GGA family is based on a grouping genetic algorithm. The objective of all these approaches is to create a partition of PCB jobs so that the total manufacturing time (setup time plus processing time) is minimized for the entire set of jobs. The performance of the presented approaches is evaluated within a series of experiments.

4.3.4 Partial Setup Strategy

As the name of this strategy implies, partial setup strategy focuses on removing only a part of the component feeders from the machine when changing from one job to another. A component feeder is removed from its current feeder slot only if the associated changeover penalty is less than the reduction in placement time.[86] Hence, both placement times and magazine layout has to be observed in detail for evaluating possible gains in total makespan. The current state of the machine and the sequence of jobs play a significant role in this decision, which increases the complexity of the problem additionally. A distinction between permanent and tempo-

[86] Cf. Leon and Peters (1998).

rary feeder sections must also be evaluated within a partial setup strategy. Because of its high complexity, the partial setup strategy has not attracted much interest in the PCB assembly literature.

Sadiq et al. (1993) develop an intelligent slot-assignment algorithm to sequence a number of PCB assembly jobs on a placement machine using the objective of minimizing the total production time. The algorithm consists of two stages: the assignment and reassignment stages. In the first stage, new parts are assigned on the machine with the objective of minimizing setup time, while in the second stage parts are reassigned to different locations on the machine to minimize the runtime. Their research focuses on a chip shooter type machine. In the assignment stage, new feeders are added in available empty slots or by replacing feeders having a history of minimal usage. Hence, PCBs to be populated in the near future are considered and feeders with highest usage are assigned to slot positions promoting placement speed. A database system is developed to store several essential parameters required for the presented algorithm. The slot-assignment algorithm generates all possible combinations of job sequences and applies the slot-assignment process on each. Several search rules for processing double- and single-slot feeders are given in this paper. In the reassignment stage, the algorithm compares the benefit of reduced run time versus the cost of setup time to relocate the parts in a contiguous group of slots. This is calculated by the assumption that the setup time for a part change is approximately equal to the run time of 27 slot moves. This break-even criterion between setup and run time is used for each job in the sequence.

Peters and Subramanian (1996) analyze an electronic assembly system with multiple placement machines operating in parallel in a single stage of a flexible flow line. The partial setup strategy attempts to determine the balance between processing and changeover times during system operation. Four primary issues observed in this study are: assignment of products to machines, sequence of products on each machine, assignment of components to feeder locations for each product, and component placement sequence for each product. They introduce the term "state", i.e. the current component types on the machine and specific feeder location for each component type. Given a particular machine state, the changeover time to configure the machine for a product can be analyzed as a required or an optional setup. The required setup consists of loading all component types that are not already on the machine but are needed for assembling a product. The optional setup is the rearrangement of feeders on the feeder carriage to reduce the processing time. Each product is assumed to be produced on only one of the machines and processing of a batch of the same product cannot be split between different machines. Their approach is based on the strong assumption that all feeders are of the same size and any component feeder can be assigned to any feeder slot. Because the presented partial setup problem is *NP*-hard, four heuristic strategies are developed for solving the setup problem. The first strategy is a unique setup strategy and serves as a benchmark. The second heuristic is a sequence-dependent setup strategy, and uses the machine state and the placement sequence specified by the unique setup strategy in order to allocate the products to the machines, and sequence each board on each machine to minimize the changeover time. The third strategy is the minimum setup strategy, which performs only the required setups needed to transform the current state to a feasible state for the next board. The last strategy, namely the so-called tradeoff strategy, intends to choose specific optional setups that balance the tradeoff between processing and changeover times. The tradeoff setup strategy begins with the minimum setup solution and systematically proceeds to modify the feeder assignment for each product. Thus, the placement time is reduced as long as improvements in makespan can be obtained. Systematic modification of the feeder assignment for a product is achieved by moving the feeder having the highest frequency of concurrence to the position it would occupy if a unique setup were used for the product. A suggested rearrangement, even if it appears advantageous for the current product, could increase the contribution of some subsequent products in the makespan, thus increasing the overall makespan. Furthermore, a different product sequence might be warranted after a feeder rearrangement.

Leon and Peters (1996) observe partial setup strategies in a medium-volume, medium-variety manufacturing environment. The presented concepts are applied for optimizing operations of a single pick-and-place machine producing multiple products. The results of partial setups are compared with other commonly used strategies. The presented partial setup strategy explicitly considers the tradeoff between changeover time and placement time with the objective of minimizing the total production time. Similar to Peters and Subramanian (1996), they impress the complicated characteristic of the changeover time depending on not only the direct predecessor but all preceding products. This is partially due to the residual components that may be left on the machine after a product is produced, even if those components are not needed for the next product. Therefore, the problem under consideration is more complicated than the sequence-dependent setup problem. The problem is defined as state-dependent instead of sequence-dependent, and an approach with the objective of minimizing the makespan subject to the constraints of component placement, component-feeder assignment and product sequencing is presented. A simple heuristic solution procedure is developed for partial setups, which is then compared with the unique, minimum, and group setup strategies. The solutions for unique and minimum setups are obtained as special cases of the partial setup problem. The presented group setup approach assumes that either all boards are produced in the same family or families of boards have already been determined. Given the family, the procedure creates a composite board, determines the component feeder assignment for the family and the component placement sequence for each board in the family. In case of partial setup, an initial board sequence is determined arbitrarily. For each board in this sequence, feeder assignment and placement sequence problems are solved by applying shortest augmenting path and nearest neighbor heuristics, respectively. The resulting component feeder assignment is used in calculating a new job sequence by solving the sequence-dependent setup scheduling problem.

Above described setup approaches are also presented in **Leon and Peters (1998)**. In this paper, fundamentals of different setup strategies are explained in detail and evaluated for a single pick-and-place machine in a series of detailed experiments. The authors criticize the fundamental GT concept of doing similar things similarly since it relies on the assumption that little or no setup penalty is incurred when changing from one product to another belonging to the same product family. They discuss the fact that this argument is weak in context of PCB assembly, because the time to complete a given assembly depends on the length of the placement tour, which arises from the component feeder assignment as well as the specific placement coordinates. Therefore, two different board types requiring exactly the same components show a perfect similarity but may require different component feeder assignments to minimize the placement time for each board. Hence, the group setup approach from Leon and Peters (1996), where groups are assumed to be given, is extended with a preceeding hierarchical grouping approach for determining groups before problems of feeder assignment and placement sequence are determined for each board in the family. The objective of the grouping approach is similar to the approach presented in Leon and Peters (1996) (i.e. reducing the number of groups under a given feeder capacity constraint). An agglomerative clustering approach is applied to the grouping problem using Jaccard's similarity measure to calculate similarities between merging families. However, machine-specific optimization problems are solved only after the grouping process, and are hence decoupled from the grouping problem. The original partial setup approach from Leon and Peters (1996) is also improved by using the solution of the newly presented group setup approach as a basis to construct an initial board sequence. Hence, a partial board sequence is determined by specifying the order in which the families will be loaded on the placement machine. The ordering within the family is left as an arbitrary one. Additionally, two new minimum setup heuristics which arise from the procedure proposed by Lofgren and McGinnis (1986) are presented. Results show that the presented partial setup strategy outperforms other strategies due to its flexibility in adapting to different conditions and underline the necessity of developing a clustering procedure that considers the changes in placement time.

4.3.5 Evaluation of Setup Strategies

Traditionally, PCB manufacturers used to work with long production runs and large batch sizes to be efficient. Although high machine utilization can be secured in such a production environment, large inventory levels and long lead times are the consequences. Today's electronics market is driven by the products with shorter life cycles. Flexibility and quick response to orders are key elements for being competitive in the market. Hence, more and more PCB manufacturers are forced to produce in smaller batch sizes, which makes the application of a traditional unique setup strategy impractical in many cases because of the high setup effort and feeder reassignment complexity related with it.

The selection of the appropriate setup strategy depends on the characteristics of the manufacturing environment. In figure 4.2, setup strategies are classified according to production volumes and variety of boards to be manufactured. Unique setup strategy, which focuses on optimizing the placement time for each individual PCB, is the appropriate strategy for high-volume production. If several boards are produced in small batch sizes, reducing the changeover effort, and hence applying a minimum setup strategy is adequate. Both partial and group setup strategies seem appealing for applications in a medium-variety medium-volume environment. Like minimum setup and unique setup, partial setup is an incremental setup strategy, i.e. it requires the exchange of component feeders for each individual PCB type to be produced.

Group setup has mostly been understood as creation of setup families with respect to component similarity. Common to almost all of the previously mentioned setup strategies is that actual PCB assembly times are either based on rough estimates or assumed to remain constant irrespective of the composition of the PCB families.[87] Hence, the effects of grouping PCBs on

---

[87] Changes in individual placement times are introduced only in the group setup strategy by Williams and Magazine (2007) and in some of the presented partial setup strategies under some assumptions.

the global makespan, i.e. the total time required to assemble all of the PCBs and the impact on the actual placement times, have been neglected. This is due to the understanding of the conventional group technology concept, which relies on the assumption that adding a similar product to a group will not effect the unit production time of the individual product types. However, this is not the case in PCB assembly.[88] The time required to perform an assembly operation highly depends on the assignment of component feeders to slots in the magazine from where the components are retrieved. Adding a new PCB type to a group requires a modified setup of the component magazine, which must be determined for the whole group of PCBs with the objective of minimizing the total production time of the group. Thus, actual placement times per PCB are expected to be higher than in the case of a unique setup, where the magazine is set up for each type of PCB individually.

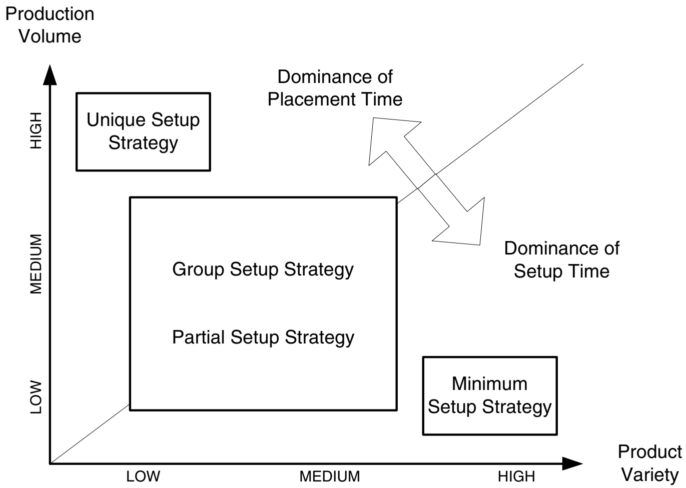


Figure 4.2: Classification of setup strategies depending on the manufacturing environment

This effect is illustrated in figure 4.3. As an example, batches of PCB types 1, 2, and 3 are to be produced. In a unique setup, each batch is preceeded by a specific setup time S1, S2, and S3, during which the placement machine is fine-tuned to the specific PCB type by optimizing the feeder arrangement in the component magazine and the sequence of the individual place-

88 Cf. Carmon et al. (1989), McGinnis et al. (1992), Ammons et al. (1997), Leon and Peters (1998), Smed et al. (1999), Smed et al. (2003), Williams and Magazine (2007), and Yilmaz et al. (2007).

ment operations accordingly. In a group setup, however, one tries to generate setup families of different PCB types, e.g. to produce the entire sequence of batches with a joint setup operation S1/2/3. Generally, the placement time for each batch will be higher as in the case of a unique setup. This effect is indicated in figure 4.3 by the hatched areas inside the bars indicating the PCB assembly times. Specifically, figure 4.3 illustrates the desired result of a reduced makespan. This outcome, however, can only be achieved if the increase in placement time does not exceed the saving in setup time and the magazine capacity is sufficient to accommodate all component feeders required for the entire setup family.

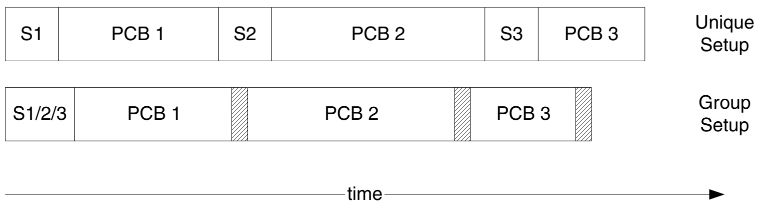


Figure 4.3: Makespan reduction achieved through group setup strategy

Lately, placement machines with interchangeable feeder trolleys have become popular in production environments with significant setup times.[89] They allow the bulk of the setup processes to be performed offline. Feeder trolleys can be detached from the placement machine and replaced with another one in a very short period of time.[90] Hence, the setup time between PCB jobs can be assumed as constant and sequence-independent in case of an offline setup strategy.

Even for machines with a conventional magazine technology, group setup strategies appear to be superior to partial setup, when the number of common component types between different PCBs is high or the changeover criterion receives higher attention than the reduction of placement time.[91] Additionally, partial setup strategies are considerably more complex because a decision on the set of component feeders to reside in the magazine and on those to be removed has to be made for each batch of PCBs. In this case, sequence-dependent changeover times do not only depend on the immediate predecessor of a PCB type, but on all of its prede-

[89] Feeder trolleys are dicussed in section 3.1.2.
[90] According to Dürr (1997b), exchange of a trolley takes around 5 minutes. Another 5 minutes is required for uploading the placement data for the new PCB job.
[91] Cf. Leon and Peters (1998).

cessors. Another disadvantage of incremental setup strategies is the requirement for frequent exchange of feeders, which is not favored in practice because of quality reasons.[92]

Because of the above described advantages of the group setup approach, a novel group setup methodology which focuses on improving the global makespan for a given number of batches of different PCB types is presented in this study. However, the presented group setup methodology distinguishes from previous approaches via the integration of machine-specific algorithms into the grouping process for evaluating the actual makespan in each step of the grouping procedure. For each group of PCBs, problems of placement sequencing and feeder assignment are solved using efficient heuristics. In contrast to the commonly used composite PCB approach, batch sizes of each PCB are taken into account in order to achieve the best magazine layout for the whole group of PCBs. The problems of nozzle selection and assignment are examined for the first time for a collect-and-place machine with a rotary placement head.

[92] Cf. Smed et al. (1999).

5. Development of Group Setup Strategies

Figure 5.1 illustrates the planning problems which are investigated in the developed group setup approaches. In this study, two different approaches are proposed for the job grouping problem. In the first approach, grouping is performed by use of well-known similarity measures and agglomerative linkage methods (see section 5.1). The second approach employs the so-called "inclusion measure" as a similarity coefficient, which is more appropriate for PCB assembly and generates setup families using a novel hierarchical clustering technique which is based on the inclusion tree representation scheme due to Raz and Yaung (1994). This approach is laid out in section 5.2. Because of the hierarchical nature of the presented grouping processes, initial grouping results are then improved using heuristic procedures which are described in section 5.3.

Conventional clustering techniques are modified in order to comply with the characteristics of the job grouping problem in PCB assembly. Both group setup approaches follow the objective of minimizing makespan of the PCB jobs to be assembled and take the limited capacity of the component magazine into account. Hence, machine-specific algorithms for solving problems of feeder assignment, placement sequencing and nozzle assignment are integrated into the solution approach. These approaches are presented in detail in section 5.4.

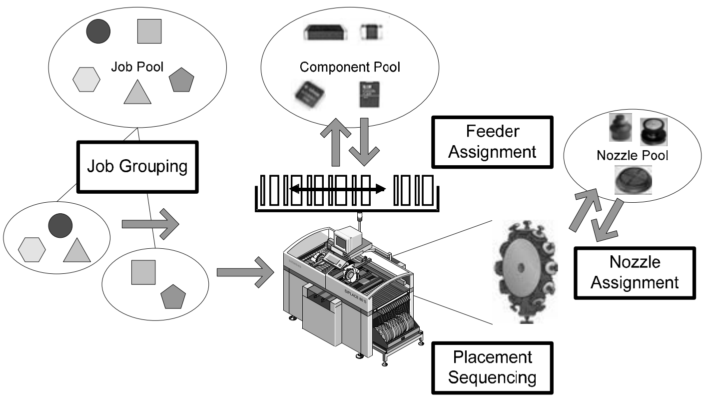


Figure 5.1: Structure of the proposed group setup approaches

5.1 Hierarchical Clustering Based on Conventional Approaches

5.1.1 Clustering Approaches

Clustering is the process of classifying objects into subsets that have a meaning in the context of a particular problem.[93] A general question facing researchers in many areas of inquiry is how to organize observed data into meaningful structures, that is, to develop taxonomies.

Cluster analysis classifies objects (e.g. PCBs) so that each object is very similar to others in the cluster with respect to some predetermined selection criteria. The resulting clusters of objects should then exhibit high internal (within-cluster) homogeneity and high external (between-cluster) heterogeneity.[94] Central to all of the goals of cluster analysis is the notion of degree of similarity (or distance) between the individual objects being clustered. Clustering approaches are divided into two major categories, i.e. *partitional*[95] and *hierarchical* clustering.

The main idea of *partitional* clustering approaches is to assign a set of cases into *k* clusters so that the within-cluster errors are minimized. In contrast to hierarchical clustering, the number of clusters must be defined in advance. In the first step, cluster seeds are identified and objects are assigned to the nearest seeds. The so-called optimization procedures recalculate cluster seeds after generating initial clusters and reassign objects to new cluster seeds if these are closer than the original seeds. This procedure is repeated until no more improvement can be achieved.[96]

Hierarchical clustering approaches involve the construction of a tree-like hierarchical structure and build clusters stepwise. Hierarchical clustering methods are categorized into *divisive* and *agglomerative* methods which are illustrated in figure 5.2. *Divisive* methods start with a clustering solution consisting of one large cluster containing all objects. Objects which are most dissimilar are then split off stepwise until each object is a separate cluster itself.

[93] Cf. Jain and Dubes (1988), p. 55.
[94] Cf. Hair et al. (1998), p. 473.
[95] Also known as nonhierarchical or *k*-means clustering.
[96] Cf. Jain and Dubes (1988), section 3.3 for an overview of partitional clustering methods.

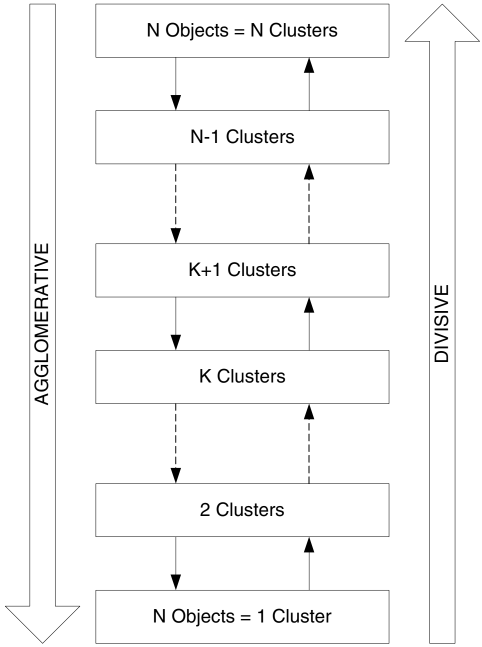


Figure 5.2: Agglomerative vs. divisive clustering

Agglomerative methods start inversely, with the assumption that each object forms a cluster. In each clustering step, two clusters showing the highest similarity are merged into a new aggregate cluster, thus reducing the total number of clusters by one. This procedure continues until all objects are assigned into one single cluster.

An important characteristic of hierarchical clustering approaches is that the results of an earlier stage are always nested within the results of a later stage creating a similarity to a tree.[97] Because clusters are formed only by joining existing clusters, any member of a cluster can trace its membership in an unbroken path to its beginning as a simple observation.

This study applies ***hierarchical agglomerative*** clustering approaches for grouping batches of PCBs. The advantage of using an agglomerative approach is the ability to observe different

[97] Cf. Hair et al. (1998), p. 493.

criteria at each clustering decision node on the tree-like structure called dendrogram.[98] Hence, machine-specific optimization problems can be solved at each clustering step to observe performance of new clustering solution in means of makespan improvement and magazine capacity constraints. Using a hierarchical scheme reduces the number of times machine-specific algorithms are called, and thus decreases the computational burden. Another advantage of hierarchical clustering against partitional clustering is that the number of groups does not need to be defined in advance and can be obtained flexibly during the clustering procedure using a termination criterion.

5.1.2 Similarity Measures

Anderberg (1973) presents several similarity measures used for different variable classifications. In PCB assembly, PCB data are best represented with binary nominal variables in a so-called PCB-component type incidence matrix (see table 5.1).[99] Hence, each PCB type is represented by a string of binary values, where '1' indicates the presence and '0' the absence of a component type. Using these data, similarities between PCB types can be examined.

Table 5.1: Example of binary representation of PCB data

	Component Type											
	a	b	c	d	e	f	g	h	i	j	k	l
PCB 1	1	1	0	0	1	1	1	1	1	0	1	0
PCB 2	1	1	0	1	0	1	1	1	0	0	1	0
PCB 3	1	0	1	0	1	1	0	0	1	0	0	1
PCB 4	1	0	1	1	0	0	0	1	0	0	1	0
PCB 5	0	1	0	1	0	0	0	0	0	1	0	1

There are a number of different similarity measures available in the literature. Similarity measures for binary variables are best illustrated with the help of a 2×2 contingency table. The value of a in table 5.2 indicates the number of component types present in both PCB types i and j while d expresses the number of component types absent in both. b and c represent the numbers of component types, which are only present in one of the PCBs i and j, respectively.

---

[98] Examples of dendrograms are presented in section 5.1.3.

[99] Cf. Anderberg (1973), p. 48-52 for an overview of variable classifications.

Anderberg (1973) overviewed general matching coefficients used in conjunction with binary data. According to Anderberg (1973), similarity measures with *d* matches in the denominator and no *d* matches in the numerator, as well as measures including arbitrarily double-weighted matches or mismatches in the denominator with no corresponding double-weighted matches or mismatches in the numerator are not recommended. Table 5.3 illustrates some similarity measures which can be logical to use according to the definition of Anderberg (1973), and recommended by Shafer and Rogers (1993a) for use of clustering PCBs based on binary data.[100]

Table 5.2: 2×2 contingency table

		PCB *j*	
		Presence (1)	Absence (0)
PCB *i*	Presence (1)	*a*	*b*
	Absence (0)	*c*	*d*

Table 5.3: General similarity measures for FMS based on binary data[101]

Equal weights	Russel and Rao	$\frac{a}{a+b+c+d}$
	Simple matching	$\frac{a+d}{a+b+c+d}$
	Jaccard	$\frac{a}{a+b+c}$
Double weight for matches	Sokal & Sneath 1	$\frac{2(a+d)}{2(a+d)+b+c}$
	Dice	$\frac{2a}{2a+b+c}$
Double weight for mismatches	Sokal & Sneath 2	$\frac{a}{a+2(b+c)}$
	Rogers - Tanimoto	$\frac{a+d}{a+d+2(b+c)}$

[100] Cf. Shafer and Rogers (1993a) and (1993b), Kusiak and Cho (1992), Mosier et al. (1997), Sarker and Islam (1999), Yin and Yasuda (2005) and (2006) for overview and comparison of similarity measures for FMS.
[101] Adapted from Anderberg (1973) and Shafer and Rogers (1993a).

Among these, Jaccard's similarity measure has probably gained the greatest attention. However, Jaccard's similarity measure may not be the most suitable for PCB assembly, as this measure ignores the conjoint absence (*d*) of a component type between PCBs. For instance, in the example of table 5.1, the similarity between PCB types 1 and 2 is given by:

$$J(1,2) = \frac{6}{6+2+1} = 0.67$$

To overcome the deficiencies of Jaccard's similarity measure, the simple matching coefficient can be defined which considers both conjoint absence (*d*) and presence (*a*) of component types. Hence, PCBs with less number of component types show a higher similarity and receive a higher chance to be grouped. This measure tends to reduce the variety of component types within a group. For example, using the sample PCBs of table 5.1, the similarity between PCB types 1 and 2 is given by:

$$SM(1,2) = \frac{6+3}{6+2+1+3} = 0.75$$

In this study, both Jaccard's and simple matching measures are integrated into the hierarchical grouping approach. Thus, experiments might reveal a performance difference between measures that concern conjoint absence and the ones that disregard it. For this purpose, similarities between each pair of PCBs are calculated and stored in a similarity matrix. Tables 5.4 and 5.5 illustrate the initial similarity matrixes for both Jaccard's and simple matching measures.

Table 5.4: Similarity matrix for the observed data using Jaccard's measure

	PCB 1	PCB 2	PCB 3	PCB 4	PCB 5
PCB 1	1				
PCB 2	0.67	1			
PCB 3	0.4	0.18	1		
PCB 4	0.3	0.5	0.22	1	
PCB 5	0.09	0.22	0.11	0.13	1

Table 5.5: Similarity matrix for the observed data using simple matching measure

	PCB 1	PCB 2	PCB 3	PCB 4	PCB 5
PCB 1	1				
PCB 2	0.75	1			
PCB 3	0.5	0.25	1		
PCB 4	0.42	0.67	0.42	1	
PCB 5	0.17	0.42	0.33	0.42	1

5.1.3 Linkage Methods

The general procedure for agglomerative clustering is presented in figure 5.3. During the clustering procedure, a linkage method has to be selected to define the measure of similarity between clusters and revise the similarity matrix accordingly. There are several linkage methods for describing the similarities between groups of objects.[102] Among these, *single*, *complete*, and *average* linkage are the most commonly used methods in FMS literature.[103] It is important to figure out that different clusters may be obtained for the same data if different linkage methods should be applied.

Step 1. Begin with *n* clusters each consisting of exactly one object. Let the clusters be numbered from 1 to *n*.

Step 2. Search the similarity matrix for the most similar pair of clusters. Let the chosen clusters be labeled *p* and *q* and let their associated similarity be SIM_{pq}, $p>q$.

Step 3. Reduce the number of clusters by one through merger of clusters *p* and *q*. Label the product of merger *q* and update the similarity matrix entries in order to reflect the revised similarities between cluster *q* and all other existing clusters. Delete the row and column of *SIM* pertaining to cluster *p*.

Step 4. Perform *steps 2* and *3* a total of *n*-1 times (at which point all objects will be in one cluster). At each stage, record the identity of the clusters which are merged and the value of similarity between them in order to have a complete record of the results.

Figure 5.3: Agglomerative hierarchical clustering procedure[104]

In the *single* linkage method, similarities between a newly formed cluster and other clusters are determined as the similarity between the two most similar objects in both clusters. This method is known as the single linkage because clusters are joined at each stage by the single strongest link between them.[105] Hence, single linkage method is also known as nearest neighbor method. For any cluster of two or more objects produced by the single linkage method,

[102] Other linkage methods are described in e.g. Anderberg (1973), section 6.2, and Jain and Dubes (1988), section 3.2.
[103] According to Backhaus et al. (1990), p. 136, other common proximity measures, e.g. centroid, median and Ward, are not appropriate for similarity measures and can only be used for clustering with distance measures.
[104] Cf. Anderberg (1973), p. 232.
[105] Cf. Anderberg (1973), p. 239.

every member is more similar to some other member of the same cluster than to any other entity not in the cluster.[106]

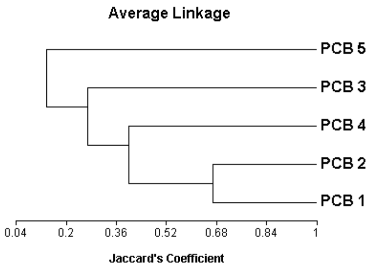
Complete linkage (farthest neighbor) method is closely related to the single linkage method except that now most dissimilar objects in two observed clusters determine the cluster similarity. Hence, the minimum of all pairs of objects between both clusters is selected. Since updating both above described linkage methods involve choosing only minimum or maximum values, these are invariant to any transformation, and thus leave the ordering of similarities unchanged.[107]

Average linkage method defines similarity between groups as the average of similarities between all pairs of objects in the two groups.[108] Such a calculation does not depend on the extreme values. Hence, average linkage method tends to combine clusters with small within-cluster variation.[109]

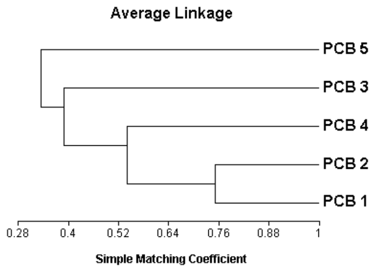
Selection of the appropriate linkage method is not a trivial issue. Single linkage method is referred to cluster objects at a relatively low level by linking chains of intermediates which is also seen in figure 5.4.[110] This may be useful, if optimally connected clustering is more significant than generating homogeneous clusters. However, because of the chaining effect, single linkage may fail to resolve relatively distinct clusters if a small number of intermediate points are present between the clusters. On the contrary, complete linkage may result in dilatation and may produce too many but more homogeneous clusters.[111]

---

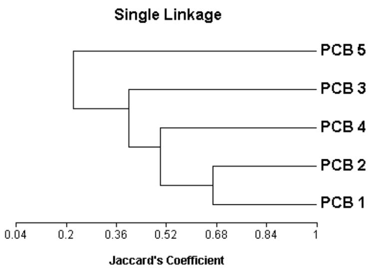
106 Cf. Anderberg (1973), p. 239.
107 Cf. Anderberg (1973), p. 239.
108 Cf. Everitt (1980), p. 31.
109 Cf. Hair et al. (1998), p. 496.
110 Based on Everitt (1980), p. 67-68.
111 Cf. Gordon (1999), p. 88



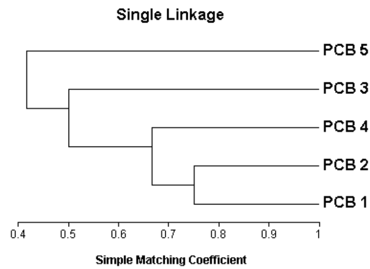
(a)



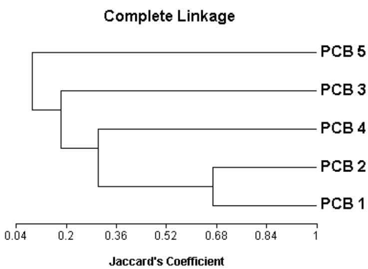
(b)



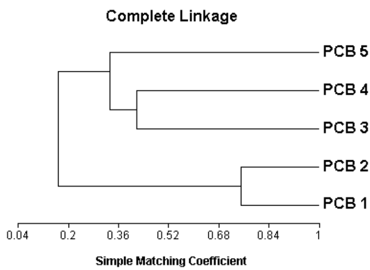
(c)



(d)



(e)



(f)

Figure 5.4: Results of different similarity and linkage methods for the observed PCB data

In hierarchical clustering, dendrograms are used to present a picture of the clustering structure. These are two-dimensional diagrams illustrating the fusions or divisions which have been made at each successive stage of the analysis.[112] Cutting a dendrogram at any level defines a clustering and identifies contents of clusters. Figure 5.4 illustrates clustering results of the binary data from table 5.1 by using Jaccard's and simple matching similarity measures in combination with average, single and complete linkage methods. For the observed small problem case, Jaccard's similarity measure gives same results independent of the selected linkage method. Only the similarity values change slightly. This also holds for the first two linkage results of simple matching measure which has exactly the same clustering result. One of the obvious observations is that cutting these dendrograms in a stage results in one big cluster and many single PCB clusters. This is not always beneficial for job grouping in PCB assembly, where reducing the number of groups by increasing the number of PCBs per group is essential to utilize component staging capacity of the placement machine.

In this small example, only the complete linkage clustering using simple matching measure delivers a different result than other methods. Generally, similarity measures cannot be defined as superior or inferior to another and no one methodology can be judged to be "best" in all circumstances.[113] Thus, applying different similarity measures and clustering approaches is essential to find the best methodology for the observed problem.

Another observation from figure 5.4 is that two PCBs having the highest number of component types are directly assigned into the first group formed independent of the methodology used. If the placement machine can occupy only e.g. 8 different component feeders, the component staging capacity would be exceeded even in the first clustering stage of each procedure leading to an infeasible grouping solution. Hence, conventional hierarchical clustering approaches must be modified by integrating PCB assembly specific constraints into each clustering decision step.

5.1.4 Group Setup Using Conventional Hierarchical Clustering

The application of the above described hierarchical clustering schemes is quite common in the group setup literature. In conventional group setup approaches, the common objective is to

---

[112] Cf. Everitt (1980), p. 25.
[113] Cf. Everitt (1980), p. 104.

minimize the number of setups by reducing the number of groups to be produced.[114] Hence, only the component staging capacity of the placement equipment is considered as a restriction for defining the groups. However, the increase of individual placement times for each PCB plays a significant role on the global makespan especially for medium-volume production. Thus, setup savings achieved with a feasible grouping may not compensate the increase in placement times. In this study, a novel hierarchical clustering scheme is designed by adding additional issues into consideration. In particular, adding a PCB type to a group or merging two groups is assumed feasible only if two conditions are satisfied:

- the component magazine capacity is sufficient to accommodate feeders for all component types within the (enlarged) group of PCB types, and
- grouping reduces the global makespan.

To verify the latter condition, the overall assembly time for the candidate group is determined as the sum of the group setup time and component placement times for all PCBs in the group. This solution is compared with the current situation comprising two separate groups. It should be noted that in the presented approach, the machine setup (group feeder assignment and individual placement sequences) is adjusted whenever a new PCB type is added to the group or two groups are merged.

The proposed algorithm employs both Jaccard's similarity coefficient and the simple matching coefficient. For recalculation of the similarity matrix after a merging operation, average, single and complete linkage methods are applied and evaluated for each similarity measure.

At the initial stage of agglomerative clustering, each cluster consists of just one PCB type. After all similarity coefficients are calculated, the two clusters which show the maximum similarity to each other are determined. These clusters are only merged into a composite group if the component magazine capacity constraint is not violated and the global makespan is reduced. If one of these conditions is not satisfied, the value of the similarity coefficient is set as infeasible and the next pair of groups with maximum similarity is selected. This procedure is repeated until no more clustering is possible. The steps of this approach are presented in figure 5.5.

[114] Cf. sections 4.3.3 and 4.3.5.

Step 1. Calculate the similarities of PCBs and/or groups using the selected measures.

Step 2. Find the candidate PCBs/groups by determining the maximum-similarity pair.

Step 3. Check capacity and makespan reduction constraints.

Step 4. IF both conditions are satisfied, THEN group the selected PCBs/groups and update the similarity values using the selected linkage rule, ELSE set the similarity between selected PCBs/groups as infeasible.

Step 5. IF grouping was successful, THEN check capacity constraint for any further mergers of the new group with all other groups. IF no more groupings is possible, THEN remove the new group from the grouping process.

Step 6. Repeat s*tep 2* until no more grouping is possible.

Figure 5.5: Proposed group setup approach based on conventional hierarchical clustering

In this study, the group setup algorithm is tested on a single-gantry collect-and-place machine. However, the grouping algorithm is a general purpose approach and can be applied to any machine type if fast machine-specific algorithms are available. The machine-specific optimization algorithms applied in this study are presented in section 5.4. Whenever a new group is formed, all possible future mergers of this new group with the remaining groups are observed in a search process. Checking component slot requirements for each possible merger enables determining whether there is a chance to enlarge this new group further. If this is not the case, the new group is taken out of the grouping process to avoid any unnecessary calculations in the grouping process.

5.2 Hierarchical Clustering Based on Inclusion Trees

5.2.1 Inclusion Measure

Raz and Yaung (1994) present a clustering technique based on the criterion of inclusion, rather than on the more commonly used concept of similarity between pairs of objects. Conventional similarity measures do not provide information about which PCB type best comprises or includes another one. However, this information might be quite valuable in scheduling PCB assembly. For instance, a so-called perfect subset situation exists if one type of PCB uses a subset of component types from another PCB. Combining these two PCB types does not require any additional slots in the component magazine (i.e. no additional setup of a component feeder is required), and thus may prove to be also advantageous in terms of makespan.

This analysis can be performed by calculating so-called asymmetric inclusion measures, which can be defined by using the definition of *a* and *c* from table 5.2 as follows:[115]

$$IM(i,j) = \frac{a}{a+c} \tag{1}$$

IM(*i*,*j*) indicates the extent to which PCB *j* is included in PCB *i*. For instance, in the example of table 5.1, PCBs 1 and 2 have 6 components in common, i.e. 6 of the 7 components of PCB 2 are included in the component set of PCB 1, resulting in an inclusion measure of

$$IM(1,2) = \frac{6}{7} = 0.86$$

Table 5.6 illustrates the inclusion measure for all pairs of PCBs. In contrast to conventional similarity measures, inclusion measure is an asymmetric measure, and hence must be calculated bidirectionally.

Table 5.6: Similarity matrix for the observed data using inclusion measure

	PCB 1	PCB 2	PCB 3	PCB 4	PCB 5
PCB 1	-	0.86	0.67	0.60	0.25
PCB 2	0.75	-	0.33	0.80	0.50
PCB 3	0.50	0.29	-	0.40	0.25
PCB 4	0.38	0.57	0.33	-	0.25
PCB 5	0.13	0.29	0.17	0.20	-

5.2.2 Clustering Using Inclusion Trees

The Raz and Yaung (1994) clustering heuristic applies the concept of inclusion to generate a tree-like hierarchy of objects which is called an inclusion tree. The notion of inclusion implies that each object has a size such that larger entities include or contain smaller ones. For the case of grouping different PCBs, PCB types can be defined as objects for which the number of component feeders required to assemble the specific PCB is defined as the size of it. The procedure for constructing an inclusion tree is based on two fundamental principles:[116]

[115] Cf. Raz and Yaung (1994).
[116] Cf. Raz and Yaung (1994).

- An object cannot be included by another object of smaller size.
- In the hierarchy, an object should be assigned under the object that best includes it which is measured by the inclusion measure.

Including each object by the object of larger or equal size with which it has the largest inclusion measure is assured by applying these two principles. Before generating an inclusion tree, objects are sorted in the descending order of their sizes. In the first step, the largest object is defined as the root of the inclusion tree. All remaining objects are assigned in the predefined order to the node with which it has the highest inclusion measure. The original construction algorithm from Raz and Yaung (1994) is presented in figure 5.6.

Step 1. Assign the largest object as the root of the tree.

Step 2. Assign the largest still unassigned object as a descendent node attached to the node in the tree to which it shows the largest value of inclusion measure. In case of ties, assign the object to a lower level object in the tree which is at the same time smaller in size.

Step 3. Repeat *step 2* until all objects are assigned.

Figure 5.6: Raz and Yaung (1994) hierarchical clustering procedure

Figure 5.7 illustrates the inclusion tree for the binary data presented in table 5.1. The types of PCBs are represented with the nodes during the weights of each arc connecting the entities indicate the value of the inclusion measure. After the complete inclusion tree is constructed, clusters are created by disconnecting portions of the hierarchy tree. Hence, a threshold value for the inclusion measure is defined and the inclusion tree is cut at “weak” branches. Figure 5.8 illustrates the grouping solutions obtained for threshold values of 0.6 and 0.7, respectively.

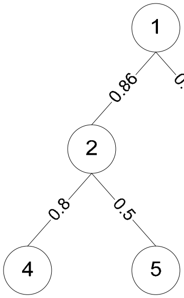


Figure 5.7: The inclusion tree for the observed PCBs

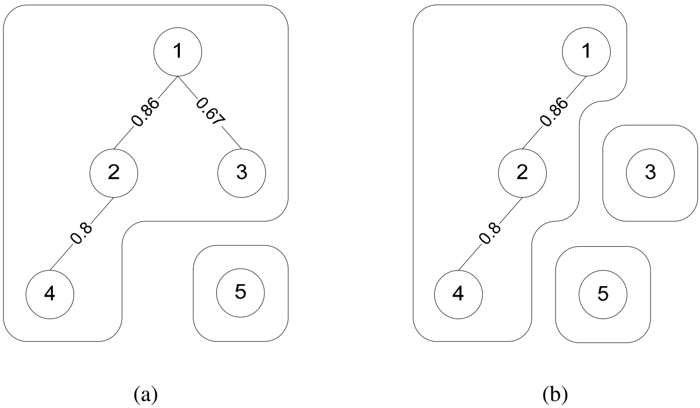


Figure 5.8: Formation of clusters using threshold values of: (a) 0.6 and (b) 0.7

5.2.3 Group Setup Using Inclusion Trees

In the original algorithm, clusters are formed by cutting arcs in the tree connecting nodes with an inclusion measure less than a user defined threshold value. In the specific case of PCB assembly, however, this procedure could frequently lead to groups of PCBs which violate the magazine capacity constraints and do not promise a reduction in global makespan. Therefore, the original procedure, which uses threshold values for cutting the arcs, is not applicable here. Instead, the original algorithm is modified in order to explicitly consider the constraints arising from group setup strategies in PCB assembly.

The construction of the initial inclusion tree is similar to the Raz and Yaung (1994) clustering approach. In the first step, inclusion measures are calculated for all pairs of PCBs. Because a PCB can not be included in a smaller PCB, i.e. one with a smaller number of component types, PCBs are sorted in descending order of their size. The largest PCB is assigned to the root of the inclusion tree. Other PCBs, in descending order of their size, are assigned as a descendant of the node in the tree to which they show the highest value of the inclusion measure. In case of ties, the PCB type is assigned as a descendant of the node with the smallest size among those in the tie group. This procedure is repeated until all PCB types are assigned in the tree.

After the complete inclusion tree has been constructed, the arc with the maximum weight in the tree is identified. In case of ties, the arc at the lowest level in the tree is chosen. The rationale behind this tie-breaker rule is that the nodes assigned to lower levels in the tree require a smaller number of component types, and thus occupy less component magazine capacity. For the selected arc, two conditions must be satisfied: (1) the magazine capacity constraint must not be violated through the common component type requirements and (2) grouping is only allowed if the global makespan is reduced. If any of these conditions are not fulfilled, the arc connecting the PCBs is marked as infeasible and the inclusion tree is reconstructed (see figure 5.9 (a)). Hence, more than one inclusion tree may be constructed parallelly if the selected PCB cannot be assigned to any other node in the initial inclusion tree. If both conditions are met, the two corresponding PCB types are merged into a group. Similar to the previous grouping approach, a search algorithm controls the feasible groupings of this new group with others according to the magazine capacity constraint. If no other PCB types can be added to that group, it is excluded from the inclusion tree (see figure 5.9 (b)). Otherwise, the newly created group constitutes a new node in the inclusion tree and inclusion measures of all other PCB types with respect to this group of PCBs are updated (see figure 5.9 (c)). In the subsequent steps of the clustering procedure, this group may again be joined with PCB types represented by neighboring nodes in the tree. Thus, groups consisting of more than two PCB types may be created. This procedure is repeated until no further groups can be formed. The details of this grouping procedure are given in figure 5.10. Again, it should be noted that the calculation of makespan requires specific algorithms for scheduling the machine operations. These algorithms are explained in section 5.4.

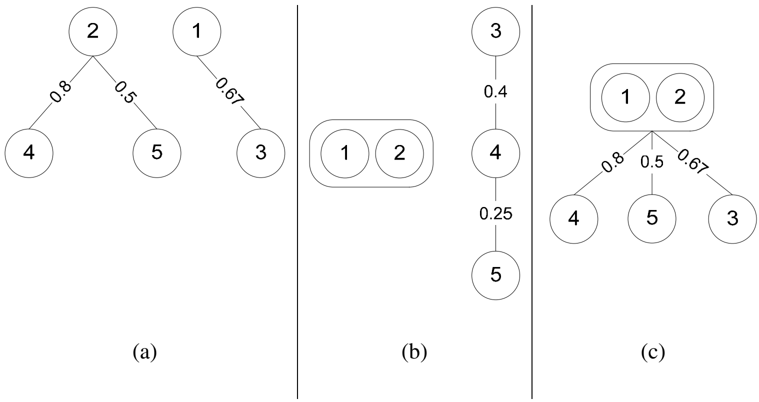


Figure 5.9: Recalculated inclusion tree: (a) after infeasible grouping
(b) after feasible grouping without any further grouping possibility
(c) after feasible grouping with further grouping possibility

Step 1. Calculate the asymmetric inclusion measures for all pairs of PCBs/groups.

Step 2. Assign the largest still unassigned PCB/group as a descendent node attached to the node in the tree to which it shows the largest value of inclusion measure. In case of ties, assign the PCB/group to a lower level PCB/group in the tree which is at the same time smaller in size.

Step 3. Repeat *step 2* until all PCBs/groups are assigned. IF an inclusion tree cannot be constructed, THEN grouping is completed.

Step 4. Select the pair of PCBs/groups with the maximum inclusion measure. In case of ties, select the pair at the lower level in the inclusion tree.

Step 5. Check capacity and makespan reduction constraints.

Step 6. IF both conditions are satisfied, THEN group the selected PCBs/groups and update the inclusion measures, ELSE mark the similarity between selected PCBs/groups as infeasible and return to *step 2.*

Step 7. IF the newly formed group can be assigned any other PCBs/groups according to the capacity constraint, THEN create a new node from this group and return to *step 1*, ELSE exclude this new group from the inclusion tree and return to *step 1* to create a new inclusion tree from remaining PCBs/groups.

Figure 5.10: Proposed group setup approach using inclusion trees

5.3 Improving Group Setup Solutions

Hierarchical clustering techniques have a general disadvantage since they contain no provision for reallocation of objects which may have been poorly classified at an early stage in the analysis.[117] Thus, improvement heuristics based on local search techniques must be applied to the solutions of hierarchical clustering.

In this study, two improvement heuristics are implemented. After initial groupings are gathered, the so-called move heuristic tries to further reduce the number of setup groups until the global makespan cannot be improved anymore or no more reduction in the number of groups is achievable. After applying the move heuristic to the initial grouping solution, each member of initial groups is swapped by another one to further improve the global makespan. In the following, both heuristics are explained in detail.

5.3.1 Move Heuristic

The move heuristic is designed to further reduce the number of PCB groups in case of a weak initial grouping solution. The proposed heuristic approach selects the group with the smallest feeder slot requirement and tries to reinsert its contents to other groups to save one more grouping. Hence, the number of required additional slots for inserting each PCB from the selected group to any of the remaining groups is investigated. In the first step, the PCB which generates the smallest additional slot usage is moved to the group which performs the best fit. This procedure is repeated for all PCBs of the selected group until the selected group is emptied. The new group formation with reduced setup effort is only accepted if the savings compensate the increase in total placement time. The search procedure is carried out for every group until no more moves are feasible or no makespan improvement can be achieved. The steps of the proposed move procedure are illustrated in figure 5.11. It should be noted that machine-specific heuristics are applied after a group is emptied in order to calculate the new global makespan.

5.3.2 Swap Heuristic

After the move heuristic is applied to the initial group setup solution, the global makespan can be further improved by swapping PCBs between different groups. Because machine optimization problems must be solved to calculate changes of each swap operation, each PCB is

[117] Cf. Everitt (1980), p. 68.

swapped only with a randomly selected PCB from another group in order to reduce the computational effort. The details of the proposed swap algorithm are given in figure 5.12.

Step 1. Select the next available group with the lowest magazine slot usage. Stop the move search if no more groups are available for application of the move heuristic.

Step 2. FOR all PCBs in the selected group and all remaining groups, calculate the number of additional slots if this PCB is moved to another group.

Step 3. IF a candidate PCB can be found, THEN move the PCB with the lowest additional slot requirement to the best candidate group, ELSE mark the selected group as infeasible, discard the current move solution and return to *step 1*.

Step 4. Repeat *steps 2* and *3* until the selected group is emptied.

Step 5. Calculate the new global makespan with the reduced setup using machine-specific algorithms.

Step 6. IF the global makespan is improved, THEN accept the new grouping and return to *step 1*, ELSE mark the selected group as infeasible, discard the current move solution and return to *step 1*.

Figure 5.11: Move heuristic procedure

Step 1. Select for each PCB a random PCB from another group and swap these PCBs.

Step 2. Calculate the improvement in global makespan. IF an improvement is achieved, THEN accept swapping these PCBs, ELSE discard the swap solution.

Step 3. Repeat *steps 1* and *2* until all PCBs are examined.

Figure 5.12: Swap heuristic procedure

5.4 Machine-Specific Operations Scheduling

As mentioned before, the presented approach attempts to model the placement times of automated placement machines more realistically than other approaches known from the academic literature. Thus, machine-specific algorithms for optimizing operations of a placement machine have to be integrated into the solution procedure.[118] In particular, *assignment of component feeders* to slots in the component magazine and *placement sequence* of components must be determined at each step of the grouping procedure considering the individual operation mode of the placement machine. In addition to these optimization problems, the *assignment of nozzles* is investigated for the first time for a collect-and-place machine with a rotary placement head. The *component retrieval* problem arises from assigning multiple feeders of the same type on the same placement machine. Thus, the idea of multiple feeder assignment is generally used for balancing the workload between gantries of a double-gantry placement machine and is not relevant for the observed single-gantry machine.

The author's research group has worked out a variety of scheduling algorithms for different types of placement machinery. For instance, Grunow (2000) has developed an algorithm for feeder assignment and placement sequencing for chip shooter machines. Component allocation problems, which play a key role for modular placement machines, have been investigated by Grunow et al. (2003). An efficient heuristic solution procedure for scheduling operations of collect-and-place machines with a single rotary placement head has been developed by Grunow et al. (2004). Kulak et al. (2007a) propose GA-based solution approaches for the same type of placement machines and extend the research on double-gantry collect-and-place machines. In the sequel, the integration of machine scheduling algorithms into the proposed hierarchical clustering approach is exemplified for the case of a single-gantry collect-and-place machine.[119] Nevertheless, the proposed approach can be applied to any other type of machinery.

In general, the assembly cycle time, i.e. the time required to complete each PCB, consists of a fixed setup time, which includes loading/unloading and the time to adjust the position of the PCB on the work table, and a variable placement time. The latter is significantly affected by

(i) the allocation of component feeders to positions in the magazine,

(ii) the allocation of nozzles to positions in the rotary placement head,

(iii) the assignment of placement operations to the various tours of the placement head,

(iv) the order of the placement operations within each tour, and

[118] Cf. section 4.2.4 for an overview of machine optimization problems.

[119] The structure and working principle of this type of placement machine is presented in detail in section 3.3.

(v) the sequence of these tours in an overall tour.

The outline of the heuristic solution procedure and interaction of solution steps between each other are shown in figure 5.13. In the first stage of the proposed methodology, problem (i) is solved, i.e. feeders (component types) of each PCB/group are assigned to locations in the magazine of the machine using a greedy algorithm adapted from Grunow (2000). Additionally, the number of nozzles of each type is determined for each PCB and allocated to segments on the placement head (problem (ii)). In the second stage, based on the assignment of component feeders to magazine positions and nozzles to segments on the placement head, the component placement sequence is determined. Apparently, for a collect-and-place machine this problem is similar to the well-known vehicle routing problem with the placement head corresponding to the (single) vehicle with a limited loading capacity. Therefore, a standard method for vehicle routing problems, namely the savings heuristic introduced by Clark and Wright (1964), is adapted for its solution. The placement sequencing method yields the assignment of placement operations to the various tours of the placement head (problem (iii)) and the order of the placement operations within each tour (problem (iv)). Additionally, the sequence of the placement tours has to be determined (problem (v)). It is important to note that the problems of feeder assignment and placement sequencing are highly interdependent. Due to the design of the collect-and place machine, the component retrieval sequence is the same as the placement sequence. Hence, both the movements of the placement head for the retrieval of components from the magazine and the movements for the actual placement, have to be considered in the heuristic procedure. Finally, local search principles are applied in the third stage in order to improve the PCB/group feeder assignment and component placement sequences obtained.

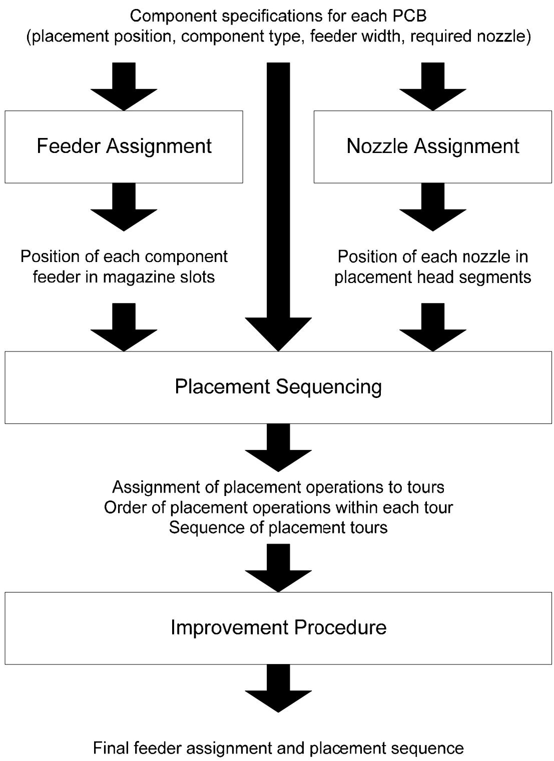


Figure 5.13: Outline of the heuristic solution procedure

There are some major assumptions and operation specifications of the collect-and-place machine, upon which the investigations in this study are based:

- This study considers a collect-and-place machine with a single component magazine and a single rotary placement head equipped with a given number of nozzles (e.g. 12).
- The capacity of the component magazine is sufficient to accommodate all feeders (component types) required to assemble a single PCB. However, the total number of feeder slots required for assembling all PCB types may exceed the component magazine capacity.
- Each magazine setup includes only one feeder per component type.

- Component feeders are of variable width, i.e. they occupy a variable number of slots in the component magazine depending on the size of the individual component type.
- Each component type can be picked with a specific type of nozzle. Hence, placement of all components of a PCB may require more than one type of nozzle.
- For each assembly tour, the placement sequence is the same as the pickup sequence.
- The placement head performs stepwise rotational movements only in one (forward) direction, i.e. reverse rotation is not possible.
- The time needed to pick up a component from the magazine and to place it onto the board is the same for all components. Travel times between individual locations, however, depend on their distance and the given speed of the robot arm.

5.4.1 Feeder Assignment

Assigning Feeders of a Single PCB

The first subproblem to be examined is the assignment of component feeders to slots in the magazine of the placement machine. For the solution to this subproblem, Grunow et al. (2004) have developed a heuristic approach which analyzes the neighborhood relations between the different types of components and the corresponding placement locations on the board. The basic idea behind this heuristic approach is to arrange component types, which are characterized by strong neighborhood relations, adjacent to each other in the component magazine.[120]

The coordinates of the individual placement locations and the corresponding component types to be assembled are given in the computer-aided design (CAD) of the board. Based on these data, a complete graph can be generated, in which the placement locations are represented by nodes and all of the nodes are completely connected by arcs. With C placement locations, this graph contains $C \cdot (C-1)$ arcs (see figure 5.14 (a)). Each arc represents a possible movement of the placement head between two placement locations on the board. Weights on the arcs indicate the distance between a pair of placement locations. Since a single tour linking all placement locations consists of only $C-1$ arcs, a more aggregate representation of the placement locations is needed. This is achieved by transforming the complete graph into a corresponding

[120] Cf. Grunow (2000), section 5.3.1.2.

minimal spanning tree. An MST is a nondirected graph which links all nodes such that the sum of the corresponding arc weights are at minimum. For the determination of the MST, Kruskal (1956) has proposed a very efficient algorithm which has been adopted in the presented solution approach. The number of arcs by which a node is linked to other nodes in the resulting MST can be considered as a measure of the strength of its neighborhood relations. This measure is used as a priority index for the assignment of feeders (component types) to slots in the component magazine.

Unfortunately, in the application considered, the derivation of arc weights is not straightforward. In particular, the operation specifications of the collect-and-place machine have to be considered. Since a typical placement machine is equipped with independent drives for moving the placement head in the x- and y-direction, the time required for a movement between components *i* and *j* on the board is determined by the maximum of the travel time in x- and y-direction. Moreover, the rotary placement head must rotate one segment to the subsequent nozzle, before the next placement operation can be performed. Considering all three concurrent movements of the placement head, arc weights can be determined as follows:

$$W_{ij} = \max\left\{TR, \left|\frac{x_i - x_j}{V_x}\right|, \left|\frac{y_i - y_j}{V_y}\right|\right\} \quad (2)$$

W_{ij} Weight of arc (i,j), i.e. travel time of the placement head between two locations on the board

TR Rotational cycle time of the placement head

x_i, y_i X- and y-coordinate of the placement location for component *i* on the PCB, respectively

V_x, V_y Velocity of the placement head in the x- and y-direction, respectively

In PCB design, it sometimes occurs that placement locations for certain component types are concentrated in an area so that they can be reached within the rotational cycle time of the placement head. As a result, the corresponding arcs in the MST take the same weight. To further discriminate the neighborhood relations in such cases, outgoing arcs from same placement position which carry the same weights are further ordered with an advanced weighing procedure suggested by Grunow (2000).[121]

For the illustration of the MST procedure from Kruskal (1956), an example with five placement locations is given in table 5.7. In the first stage, a complete graph is constructed (see figure 5.14 (a)) and all arcs of the complete graph are sorted in the ascending order of their weights as in table 5.8. The arc (2,3) which carries the smallest weight is selected as first the arc of the MST. Similarly, the arcs (4,5) and (3,4) are also inserted into the MST. The fourth candidate, i.e. arc (2,4), would result in a cycle with the previously selected arcs (2,3) and (3,4), and thus cannot be added into the MST. This holds also for arc (3,5), which would similarly create a cycle. Finally, arc (1,2) is observed and taken into the MST as it does not result in any closed cycle with any of the previously selected arcs. Hence, the termination criterion (*C*-1) is arrived and the MST is constructed as in figure 5.14 (g).

Table 5.7: Travel time matrix for a PCB with five placement operations

Placement location	1	2	3	4	5
1	--	35	55	40	50
2		--	10	25	45
3			--	20	30
4				--	15
5					--

Table 5.8: Ordered arcs of the complete graph

Order	1	2	3	4	5	6	7	8	9	10
Arc	(2,3)	(4,5)	(3,4)	(2,4)	(3,5)	(2,1)	(1,4)	(2,5)	(1,5)	(1,3)

121 Cf. Grunow (2000), p. 145-152 for details of the advanced approach.

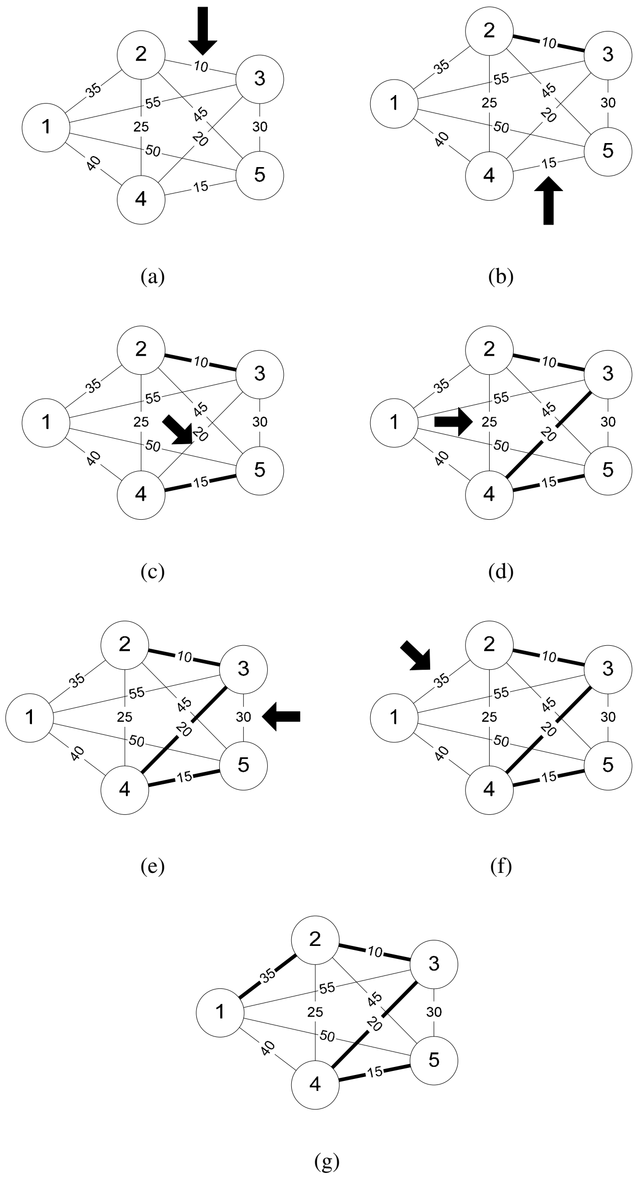


Figure 5.14: An example for the MST solution approach from Kruskal (1956)

Very often, a specific type of electronic component is assembled more than once at different locations on the PCB, and thus has to be represented by multiple nodes. In this case the neighborhood index of a pair of component types is defined as the total number of arcs linking the nodes (i.e. corresponding placement locations) of the two component types in the MST.

Basically, the feeder assignment problem is equivalent to a quadratic assignment problem, which is known to be *NP*-hard.[122] To solve this problem for assembling a single PCB on a collect-and-place machine, the heuristic solution procedure in figure 5.15 is developed. The basic idea behind this approach is to arrange component types, which are characterized by strong neighborhood relations, adjacent to each other in the component magazine. Very often, a specific type of electronic component is assembled more than once at different locations on the PCB. Hence, the neighborhood index of a pair of component types is defined as the total number of arcs linking the nodes of two component types in the MST. This measure is used as a priority index for the assignment of feeders (component types) to slots in the component magazine. In the initial step, the pair of component types with the highest value of neighborhood index is assigned to a central position. Subsequently, from the set of yet unassigned component types, the one which shows the highest value of neighborhood index with respect to the component type either assigned to the left- or rightmost occupied position in the magazine is assigned adjacent to its counterpart. This procedure is repeated until all feeders (component types) are assigned. Finally, the determined feeder layout is assigned to the center of the component magazine.

The general principles of the proposed heuristic procedure can be explained with the same small example of five placement operations. Using the additional data from table 5.9, the neighborhood indexes between pairs of component types are calculated as in table 5.10. Component types A and B appear to have the best neighborhood value among all component types, and hence the feeders of these component types are selected as the initial pair. Next, the best neighborhood value of the remaining component types with any of the left- or rightmost occupied feeders has to be selected from the neighborhood matrix. For this small example, only component C is left. The feeder of component type C is thus assigned adjacent to component type B with which it shows the best neighborhood. Figure 5.16 illustrates the feeder assignment solution.

[122] Cf. section 4.3.1.

Step 1. Set up the complete graph of the placement locations on the board and determine the arc weights.

Step 2. Generate the MST from the complete graph using the algorithm from Kruskal (1956).

Step 3. For each pair of component types, determine the neighborhood index as the total number of arcs linking the nodes of the two component types in the MST.

Step 4. Assign the pair of component types with the highest value of neighborhood index to a position in the center.

Step 5. From the set of yet unassigned component types determine the one which shows the highest value of neighborhood index with respect to the component type either assigned to the left- or rightmost occupied position. Assign the selected component type adjacent to its counterpart. (Note that in this step there are always two possibilities, left or right, for assigning a new component type. Both options have to be considered in making the possible assignment.)

Step 6. Repeat *step 5* until all feeders (component types) are assigned.

Step 7. Assign the determined feeder layout to the center of the component magazine.

Figure 5.15: Feeder assignment procedure

Table 5.9: Component types of each placement operation

Placement operation	1	2	3	4	5
Component type	A	B	A	B	C

Table 5.10: Neighborhood values between component types

Placement location	A	B	C
A	0	3	0
B		0	1
C			0

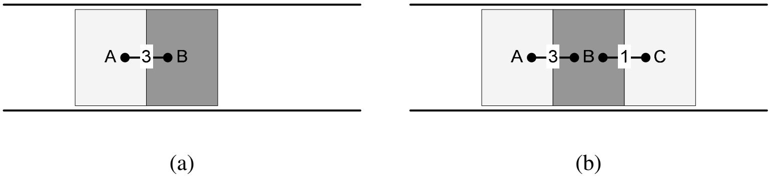


Figure 5.16: Assignment of component feeders

Assigning Feeders of a Group of PCBs

In this study, the above described basic single-PCB approach from Grunow et al. (2004) is extended for the case of determining the magazine setup for a group of different PCB types. Figure 5.17 illustrates the first step of the procedure, in which the MST-based neighborhood matrix is determined for each individual PCB type.

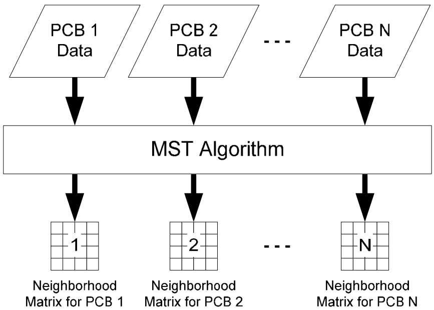


Figure 5.17: Creation of neighborhood matrices

An issue of considerable importance in determining the magazine setup is the consideration of batch sizes. In the electronics assembly literature, the composite (super) PCB approach has been widely used. However, composite PCBs do not take batch sizes of the different PCB types into account. In case the batch sizes of the various PCB types are significantly different, the composite PCB approach leads to a magazine setup which does not properly reflect the usage of the individual component types.

In the proposed feeder assignment approach for a group of PCBs, the batch sizes of each PCB type are taken into account. For a pair of component types, the elements in the neighborhood matrix indicate the expected number of immediately successive placement operations. Therefore, the matrix elements are weighted by the batch size of the corresponding PCB. By summing up all weighted PCB-specific neighborhood matrices, the aggregate neighborhood matrix for the entire group of PCBs is obtained. Based on this aggregate matrix, the component setup is determined much more realistically compared to the conventional composite PCB approaches. Figure 5.18 illustrates the suggested procedure.

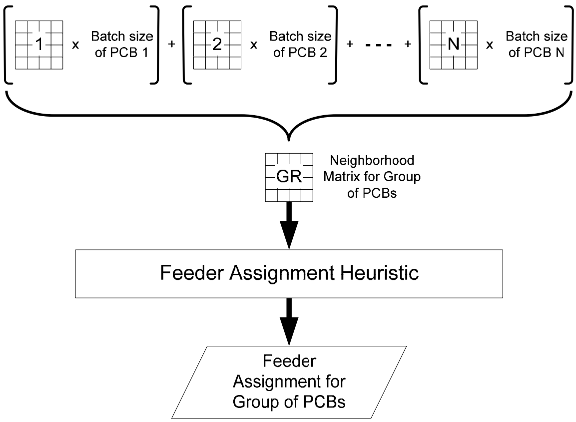


Figure 5.18: Determination of the feeder assignment for a group of PCBs

5.4.2 Nozzle Assignment

Although the nozzle assignment problem is of great significance for industrial applications, it has received only rare interest in PCB assembly literature. Analog to other problems in PCB assembly, this problem is intertwined with the solution of other machine optimization problems. Most of the previous solution approaches are based on tool switching algorithms known from FMS literature which overcome this problem by allowing nozzle changes during the assembly cycle of the same PCB. Hence, the objective is to reduce the number of tours and required nozzle setup operations. However, this is only practical for fine-pitch placement machines which are equipped with one or few nozzles and can execute fast nozzle exchange in case of screwed grippers. Collect-and-place machines with rotary placement heads are generally equipped with vacuum nozzles which can assemble small components in a high production rate. The exchange of these vacuum nozzles requires a significant setup time because of the high number of nozzles on the rotary head (e.g. 12) and the need for calibrating each vacuum nozzle.

In this study, the nozzle assignment problem is considered for the first time for a collect-and-place machine equipped with a rotary placement head. In contrast to beam-type placement heads, the sequences of pickup and placement operations are identical for the observed machine, Hence, the problem is not only to find the optimal number of nozzles but also the optimal assignment of nozzle types to several segments in the head. Because of the above described specifications of rotary head placement machines, a nozzle setup is carried out before each new batch of same PCB type is started. Hence, the nozzle assignment problem is solved for each single PCB separately. Nozzle exchange time is considered to be constant for each PCB type and independent of the number of groups determined in the group setup approach. Therefore, the setup time required for nozzle exchange is not required to be observed in the global makespan calculations.

The assignment of nozzles on the placement head has a direct effect on the pickup and placement sequence, because each component can only be handled by a specific type of nozzle which has to be available on the head segment selected to execute its placement. Thus, the nozzle assignment problem has to be solved prior to the placement sequencing heuristic. The heuristic procedure presented in the following comprises two stages. In the first stage, the optimal set of nozzles is calculated for each PCB type. The next problem deals with assigning the determined nozzle set to segments in the rotary placement head of the collect-and-place machine.

Determining the Set of Nozzles for Each PCB

Raduly-Baka and Knuutila (2007) have developed several optimal policies for determining the number of nozzles for assembling a PCB on a beam-type placement machine. In their assumptions, each component type can be picked up and placed by a certain nozzle type, although a nozzle type may support the placement of different component types. The so-called optimal nozzle selection problem is solved to optimality using a three-phase greedy procedure.[123] For better understanding the algorithm, the following notations are used:

R Number of segments on the placement head

[123] The proof of optimality is presented in Raduly-Baka and Knuutila (2007).

p_i Number of placement operations requiring nozzle type i

P Number of total placement operations for the observed PCB

a_i Number of assigned nozzles for nozzle type i

In the first phase, nozzles are partitioned into two sets, i.e. S_1 and S_2. Nozzle types which fulfill the following formula are inserted into S_1:

$$R \times p_i < P \tag{3}$$

Hence, it is assured that nozzle type i, which is required for less than one placement operation per tour, is represented on the placement head by setting its a_i value to 1. The remaining capacity of the placement head (R') and remaining number of placement operations (P') are adjusted accordingly using equations (4) and (5). Nozzle types which are not considered in S_1 are assigned to S_2 and observed in the second phase.

$$R' = R - |S_1| \tag{4}$$

$$P' = P - \sum_{i \in S_1} p_1 \tag{5}$$

The algorithm assigns a_i nozzles for each nozzle type i in S_2, which is calculated using equation (6). The remaining capacity of the placement head is stored in a variable called R'' which is calculated according to equation (7) as in the following.

$$a_i = \left\lfloor \frac{R' \times p_i}{P'} \right\rfloor \tag{6}$$

$$R'' = R' - \sum_{i \in S_2} a_i \tag{7}$$

If there are still more free segments left on the placement head ($R''>0$), the algorithm distributes the remaining R'' free segments to nozzle types having the highest value of p_i/a_i. Hence, the algorithm searches in S_2 for nozzle type i with the maximum value for p_i/a_i, increments a_i by one, and removes nozzle type i from S_2. This process is repeated R'' times until the remaining capacity is completely allocated.

The third phase of the algorithm focuses on improving the results of the nozzle assignment. This is done by searching for the maximum p_i/a_i value among all nozzle types in S_2 and trying to decrease the p_i/a_i ratio by increasing the value of a_i. For this purpose, the algorithm of Raduly-Baka and Knuutila (2007) searches for two different nozzle types i and j such that $p_j/(a_j-1)<p_i/a_i$ and $a_j>1$. If there is more than one such pair, nozzle types i and j with the maximum p_i/a_i and minimum $p_j/(a_j-1)$ values are selected. If such two nozzles are found, the algorithm decreases a_j by one and increases a_i by one, and removes nozzle type i from S_2. This process is repeated until there are no nozzle types left in S_2 which satisfy the mentioned above conditions. The detailed flow of the algorithm is presented in figure 5.19.

Step 1. FOR each nozzle type i, calculate Rxp_i. IF $Rxp_i<P$, THEN add nozzle type i into S_1 and set a_i=1, ELSE add nozzle type i into S_2.

Step 2. FOR each nozzle type in S_2, calculate a_i using equation (6) and assign it a_i nozzles. IF there is no more nozzle capacity left on the placement head, THEN the optimal set of nozzles is determined.

Step 3. Select nozzle type i giving the highest value for p_i/a_i. Increment a_i by one and remove it from S_2.

Step 4. Repeat *step 3* until no more nozzles can be assigned.

Step 5. Select a pair of nozzles i and j in S_2 such that $p_j/(a_j-1)<p_i/a_i$ and $a_j>1$. IF such a pair cannot be found, THEN the optimal set of nozzles is determined, ELSE redistribute one nozzle from type j to i and remove nozzle type i from S_2.

Step 6. Repeat *step 5* until S_2 is empty.

Figure 5.19: Procedure for determining the optimal set of nozzles

The procedure for determining the optimal set of nozzles is demonstrated in the following using the same small example given before. Assume that component types A, B, and C from table 5.9 require nozzle types a, b, and c for their placement, respectively, and the placement head is equipped with three segments. In the first step, the values in table 5.11 are calculated using (3) and nozzle types are separated into two sets resulting in S_1={c}and S_2={a,b}. Hence, nozzle type c which belongs to S_1 is assigned one position on the placement head (a_c=1). Before starting with the second phase of the algorithm, R' and P' are updated using equations (4) and (5), respectively (R'=2 and P'=4). The a_i values are calculated for both nozzles in S_2 as follows:

$$a_b = \left\lfloor \frac{2\times 2}{4} \right\rfloor = 1 \qquad \text{and} \qquad a_c = \left\lfloor \frac{2\times 2}{4} \right\rfloor = 1$$

Hence, each nozzle type in S_2 gets one position assigned on the placement head. The value of R'' is calculated using equation (7). Because there are no more segments left on the placement head ($R''=0$), the algorithm terminates. The optimal set of nozzles for the observed example consists of one nozzle for each type.

Table 5.11: Calculations for determining number of nozzles for each type

Nozzle type	a	b	c
Number of placement operations	2	2	1
$R\times p_i$	6	6	3

Assignment of Nozzles to Rotary Head Segments

The optimal algorithm presented in the previous section determines the optimal number of segments per nozzle type required for assembling each PCB. For the case of beam-type placement machines, the position of nozzles on the placement head is of minor importance. However, collect-and-place machines with rotary placement heads have the restriction that pickup and placement sequences are identical and highly dependent on the configuration of nozzles on the placement head. Hence, an algorithm has to be developed to allocate the predetermined set of nozzles to segments in the rotary head.

The nozzle assignment approach presented in this study applies a similar neighborhood approach developed for the assignment of component feeders to magazine slots. Hence, the MST created by the algorithm from Kruskal (1956) is now evaluated in terms of neighborhoods between types of nozzles (see figure 5.20). The neighborhood index of a pair of nozzle types is defined as the total number of arcs linking the nodes (i.e. corresponding placement locations) of the two nozzle types in the MST. After the neighborhood matrix is generated, the heuristic solution procedure in figure 5.21 is suggested for assigning the optimal set of nozzles to segments on the placement head. The basic idea behind this approach is – similar to the feeder assignment approach – to arrange nozzle types, which are characterized by strong neighborhood relations, adjacent to each other on the placement head. In the initial step, the pair of nozzle types with the highest value of the neighborhood index is assigned. The number of remaining nozzles for each type is updated accordingly. Subsequently, the remaining nozzle types are assigned stepwise either to the left- or rightmost adjacent positions with the descending values of their neighborhood indexes with respect to one of the edge positions. This procedure is repeated until all nozzles of the determined set are assigned. Finally, the generated sequence of nozzles is allocated to the placement head starting with the assignment of the first nozzle in the first placement segment.

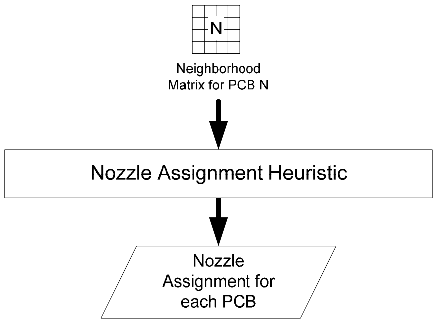


Figure 5.20: Determination of the nozzle assignment for a PCB

Step 1. Set up the complete graph of the placement locations on the board and determine arc weights.

Step 2. Generate the MST from the complete graph using the algorithm of Kruskal (1956).

Step 3. For each pair of nozzle types, determine the neighborhood index as the total number of arcs linking the nodes of the two component types in the MST.

Step 4. Assign the pair of nozzles with the highest value of the neighborhood index to a position in the center.

Step 5. From the set of yet unassigned nozzles, determine the one which shows the highest value of the neighborhood index with respect to the nozzle either assigned to the left- or rightmost occupied position. Assign the selected nozzle adjacent to its counterpart. (Note that in this step there are always two possibilities, left or right, for assigning a new nozzle. Both options have to be considered in making the possible assignment.)

Step 6. Repeat *step 5* until all nozzles are assigned.

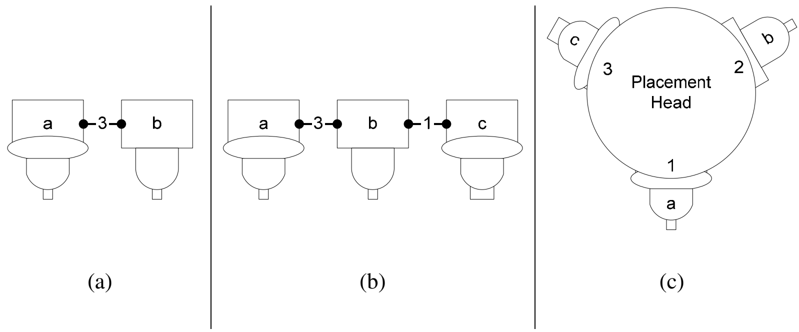
Step 7. Allocate the determined nozzle layout in the same order starting with the assignment of the first nozzle to the first segment on the placement head.

Figure 5.21: Nozzle assignment procedure

The procedure of assigning a set of nozzles to placement segments is presented with the same example consisting of five placement operations. Using the MST solution of figure 5.14 and data from table 5.10, the neighborhood matrix for the nozzle types is calculated as shown in table 5.12. Similar to the feeder assignment procedure in section 5.4.1, the set of nozzles are first assigned in a sequence (see figure 5.22 (a) and (b)) which is then allocated to the segments of the rotary placement head (see figure 5.22 (c)).

Table 5.12: Neighborhood values between nozzle types

Placement location	a	b	c
a	0	3	0
b		0	1
c			0



(a) (b) (c)

Figure 5.22: Assignment of nozzles to placement segments

5.4.3 Placement Sequencing

After the feeder and nozzle assignments have been determined, the next step is to sequence the individual placement operations for each type of PCB. Considering the special case of a collect-and-place machine, a major constraint arises from the component staging capacity of the revolver-type placement head. As the total number of components to be placed on a board generally exceeds the capacity of the revolver by far, several tours of the placement head have to be established. The number of placement operations within a tour is restricted by the number of nozzles on the revolver. Principally, the placement sequencing problem consists of three interrelated subproblems:[124] (iii) assigning components (placement operations) to the various tours, (iv) ordering the placement operations within each tour, and (v) determining the sequence of the tours in an overall tour. Problems (iii) and (iv) can be considered as a vehicle routing problem in micro-dimensions. The components to be mounted are being collected from the warehouse (the magazine) and will be distributed to different customers (placement locations on the PCB). Hence, algorithms originally designed to solve vehicle routing problems are applied in order to sequence the placement operations. Additional steps are incorporated into the algorithms in order to solve problem (v), i.e. to sequence the various tours.

Determining an exact solution to the vehicle routing problem is mathematically an extremely complex task. Therefore, efficient heuristic algorithms have been developed. In particular, the so-called savings heuristic due to Clarke and Wright (1964) is considered as a very efficient standard method which has been adapted to many application environments.[125] The savings heuristic is based on the idea that, instead of driving back and forth from the warehouse to a pair of customers, the truck takes loads to be delivered to both customers in a combined tour. For each pair of customers, the resulting savings value is calculated, which expresses the reduction in travel time achieved by substituting the shuttle tours through a combined tour. The savings value is used as a priority index in sequentially constructing the complete tour while observing the capacity limits of the truck. In fact, many variations of the basic savings heuristic have been proposed in the literature.

Grunow et al. (2004) present four different savings-based heuristic approaches for solving the above described problems of placement sequencing. Of these four methods, heuristic 2 outperforms the others according to the numerical analysis conducted.[126] Therefore, this method is selected as a basis for the developed placement sequencing procedure and employed in determining the placement time of PCBs during the presented clustering algorithms. One deficit of the solution procedures from Grunow et al. (2004) is the assumption that nozzles are capable of picking up any component type. They assume that nonstandard components which require a special type of nozzle are assembled in a separate cycle before or after regular components are placed. The placement sequencing heuristic from Grunow et al. (2004) is hence fur-

[124] Cf. section 5.4 in order to review the machine optimization problems.
[125] Savings algorithm is considered to be the best construction heuristic for the vehicle routing problem according to Ball et al. (1995), p. 244.
[126] Cf. Grunow et al. (2004) for detailed performance analysis of heuristic methods.

ther developed in this study enabling the execution of all placement operations of a specific PCB which requires different types of nozzles for its completion.

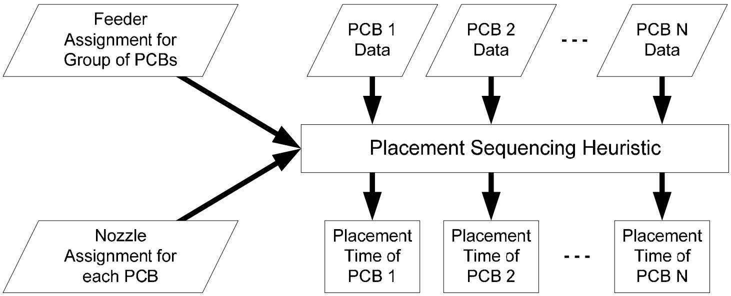


Figure 5.23: Determination of the component placement sequence and the placement time for each individual PCB

Generating the Placement Tours

The basic heuristic from Grunow et al. (2004) is almost a straightforward implementation of the fundamental savings principle. This heuristic considers savings in travel time of the placement head with respect to pickup and placement locations of each component. For calculating savings values, all tours are assumed to start and end at a given park position. The rotational cycle time of the rotary placement head is neglected during the savings calculations. However, the final placement time per board is calculated by simulating the movements of the placement head. Hence, both the travel time between the PCB table and the component magazine as well as the rotational cycle time of the placement head are considered in final calculations. The following notations are required in addition to the notations given in section 5.4.1 to formulate some functions:

X_i X-coordinate of the feeder location for component *i* in the magazine of the placement machine

$TPCB_{ij}$ Travel time between locations *i* and *j* on the PCB

$TMAG_{ij}$ Travel time between feeder positions *i* and *j* on the magazine

S_{ij} Savings value calculated for operations *i* and *j*

The travel time between two locations i and j on the board is taken as the maximum of the travel time in the x- and the y-direction:

$$TPCB_{ij} = \max\left\{\left|\frac{x_i - x_j}{V_x}\right|, \left|\frac{y_i - y_j}{V_y}\right|\right\} \quad (8)$$

The time required for traveling between two feeder positions i and j in the component magazine is given below in equation (9). Because of the serial arrangement of feeders in the magazine, only movements in the x-axis are needed to be considered in the calculation of the travel time.[127] Each placement location on the PCB always corresponds to a feeder position in the magazine which holds the particular type of component.

$$TMAG_{ij} = \left|\frac{X_i - X_j}{V_x}\right| \quad (9)$$

Based on the above simplifications the savings value for a pair of placement locations (i,j) simply expresses the reduction in assembly time that can be gained by replacing a shuttle tour to both locations i and j by a combined tour starting from the origin 0 and then traveling to i, from there to j and back to 0. Because the pickup and placement sequences are identical for collect-and-place machines equipped with rotary placement heads, the components have to be collected from the feeders in the magazine in the same sequence of placement. Hence, savings achieved by adding components i and j into same tour is calculated as follows:

$$S_{ij} = TPCB_{i0} + TPCB_{j0} - TPCB_{ij} + TMAG_{i0} + TMAG_{j0} - TMAG_{ij} \quad (10)$$

The terms $TPCB_{i0}$ and $TPCB_{j0}$ ($TMAG_{i0}$ and $TMAG_{j0}$) in equation (10) express the savings that can be achieved from eliminating one leg of the shuttle tours between $(i,0)$ and $(j,0)$, respectively, while $TPCB_{ij}$ ($TMAG_{ij}$) indicates the additional time required for the cross leg from i to j. Using equation (10), a matrix including savings for all pairs of placement operations is generated.

The above described original algorithm is extended with the capability of handling different nozzle types on the placement head. The output of nozzle assignment procedure is used as an input to the placement sequencing problem. Hence, a pair with the highest savings value is

[127] Cf. section 3.3 to review the structure of a collect-and-place machine.

only selected if there are adjacent slots on the placement head which are equipped with the nozzles required to pick up the selected component types. If there are multiple adjacent segments on the placement head for locating the selected pair, the most centered position is selected for assigning the initial pair. Because a savings tour can only be expanded on the edges, this approach gives the algorithm more flexibility to expand the placement tour into both directions. Next, the placement operation which gives the best savings value with one of the edge elements of the previous tour is selected from the savings matrix. If the observed head segment is equipped with an appropriate nozzle, the placement operation is assigned to that position. Otherwise, the search is continued until the best feasible placement operation is determined. This process is repeated until all head segments are loaded with placement operations.[128] The details of this procedure are given in figure 5.24.

Depending on the board to be assembled, the placement operations are usually not evenly distributed between different nozzles on the placement head. Therefore, savings tours may not be constructible for the last placement tour(s) because of inappropriate nozzles on adjacent candidate positions. In such cases, a random assignment procedure given in figure 5.25 is used for avoiding creation of additional tours. This procedure assigns remaining components to feasible positions on the placement head randomly in order to utilize the placement head capacity.

To illustrate the construction of a complete tour, an elementary example is used which considers a placement head equipped with three nozzles in order to populate a PCB with five different components. The travel times between placement and pickup positions and the park position of the placement head (referenced by the index *0*) are indicated in tables 5.13 and 5.14. The savings values calculated using equation (10) are given in table 5.15.

[128] Movements between magazine and PCB are more significant than the tour internal movements. Hence, reducing number of tours is of great importance (Cf. observations in Kulak et al. (2007a)). This can be achieved by utilizing placement tours. Therefore, the algorithm does not allow any empty segments as long as a candidate position can be selected.

Step 1. Calculate the savings matrix for all pairs of placement operations using equation (10).

Step 2. Select a pair of placement operations which are not assigned and deliver the best savings value. IF no more placement operations are left, THEN stop.

Step 3. Search for a pair of adjacent head segments which are equipped with appropriate nozzles for assembling selected components. IF no adjacent positions could be found, THEN stop and run the random assignment procedure.

Step 4. IF there are more than one candidate positions, THEN select the one which is closest to the center of rotary placement head in order to enable tour construction in both directions.

Step 5. Select a placement operation which can be assembled with the nozzle type of a candidate position and delivers the best savings value with one of the placement operations on one of the ends of the constructed tour. IF the selected placement operation can be assigned to any of the adjacent locations, THEN assign it to the selected position, ELSE run the random assignment procedure, open a new placement tour and return to *step 2*. IF all head segments are allocated a placement operation, THEN open a new placement tour and return to *step 2*.

Step 6. Return to *step 5*.

Figure 5.24: Placement sequencing procedure

Step 1. Select an empty segment on the placement head.

Step 2. Search for an unassigned component, which can be placed with the nozzle of the selected segment. IF a component could be found, THEN assign the placement operation to the selected segment

Step 3. Return to *step 1* until all empty segments are investigated.

Figure 5.25: Random assignment procedure

Table 5.13: Travel time matrix for placement positions

Placement location	0	1	2	3	4	5
0	--	150	130	120	110	100
1		--	35	55	40	50
2			--	10	25	45
3				--	20	30
4					--	15
5						--

Table 5.14: Travel time matrix for pickup positions[129]

Pickup location	0	1	2	3	4	5
0	--	120	100	120	100	80
1		--	20	0	20	40
2			--	20	0	20
3				--	20	40
4					--	20
5						--

Table 5.15: Savings matrix

Saving	1	2	3	4	5
1	--	445	455	420	360
2		--	440	415	345
3			--	410	350
4				--	355
5					--

The savings-based heuristic proceeds as follows. Using the previously determined nozzle assignment of figure 5.22, three segments of the placement head are assumed to be equipped with nozzles of a, b, and c, respectively. The savings values are then sorted in descending order as in table 5.16. The highest savings value of 455 is achieved for the pair of placement locations (1,3). An appropriate location is searched on the placement head which is equipped with adjacent nozzles to assemble the selected pair of components. Because no adjacent segments equipped with the required nozzles (a and a) to assemble the selected pair can be found on the placement head, the arc (1,3) is discarded. The next biggest saving is achieved by component pair (1,2). The nozzles required are a and b. This pair of components is allocated in segments 1 and 2. It is an essential feature of the savings heuristics that new nodes are only

[129] Note that feeders of 1 and 3, and 2 and 4 are the same and require nozzle types of a and b, respectively. Component 5 requires nozzle type c.

inserted at the start or the end of a tour as long as no capacity constraints are violated, but never between already assigned nodes. Due to the previous allocation, only segment 3 which is adjacent to segment 2 is left in the first placement tour. Thus, candidates (2,3), (2,4), and (2,5) remain in the descending order of their savings values. However, as segment 3 is allocated with nozzle type c, (2,3) and (2,4) cannot be assigned. The only feasible candidate (2,5) is inserted into the first tour. As the capacity limit of the placement head is reached, no further placement operations can be added to the tour. The first tour consisting of placement operations 1, 2, and 5 is shown in figure 5.26. There are only two components left, namely 3 and 4, which have to be assigned to the second tour. These can only be assigned to segments 1 and 2 which are adjacent and equipped with required nozzles. Figure 5.27 illustrates the movement of placement head to complete second placement tour.

Table 5.16: Ordered savings values

Pair of placement locations	(1,3)	(1,2)	(2,3)	(1,4)	(2,4)	(3,4)	(1,5)	(4,5)	(3,5)	(2,5)
Savings value	455	445	440	420	415	410	360	355	350	345

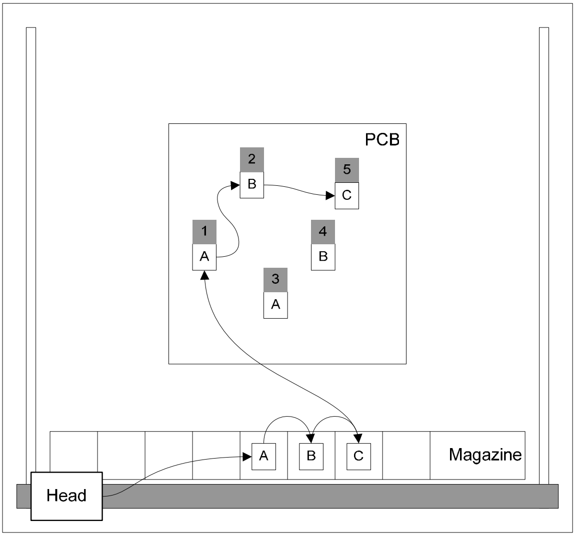


Figure 5.26: First placement tour comprising components 1, 2, and 5

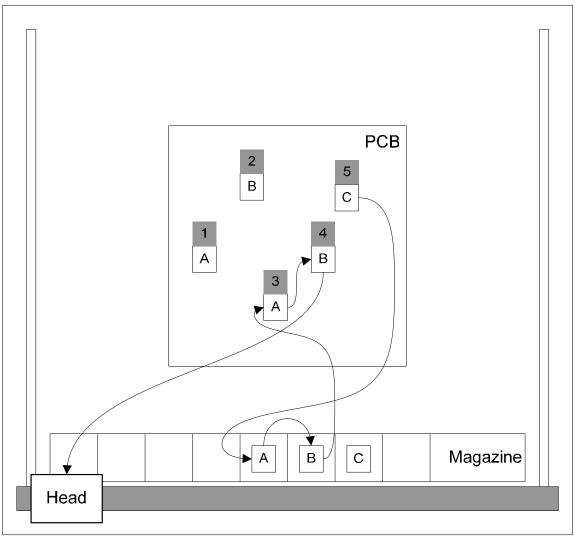


Figure 5.27: Second placement tour comprising components 3 and 4

Sequencing the Placement Tours

As discussed before, the presented savings-based heuristic does not model the movements of the placement head correctly during the construction phase, since it is assumed that all tours start and end at the given park position. In real operation, however, only the very first tour starts at the park position. All other tours actually start at the magazine position from where the first component is picked up and end at the location where the final component is placed onto the board. From there, the placement head traverses to the component magazine to pick up components for the next tour. In addition to the construction of tours, the individual tours must be combined into one overall tour of the placement head, i.e. problem (v) must be solved. To solve the sequencing problem, the heuristic solution approach employs an adapted version of the well-known nearest neighbor heuristic given in figure 5.28 in order to sequence the placement tours defined with the savings-based heuristic. In this stage, realistic travel times are assumed, i.e. including travel times between magazine locations as well as rotational cycle times of the placement head.

Step 1. Assign the first tour selected by placement heuristic to first position.

Step 2. FOR all remaining tours, select the tour which shows the minimum distance from its first pickup position to last placement position of the previous tour.

Step 3. Repeat *step 2* until all tours are sequenced.

Figure 5.28: Tour sequencing procedure

5.4.4 Improving the Feeder Assignment and Component Placement Sequence

In our heuristic solution approach, the assignment of component feeders to positions in the component magazine is determined first, followed by the sequencing of the individual placement operations. As explained in the previous sections, solutions of each of these subproblems are dependent from each other. Because of the sequential approach, the final solution obtained can be improved by an adapted application of the 2-opt-exchange procedure which is one of the best known improvement methods for combinatorial optimization problems.[130] The presented exchange procedure successively swaps two elements in the solution to a combinatorial decision problem. In case, the underlying objective function is improved, the swap is

[130] Cf. Lin (1965) for the 2-opt exchange procedure, and Ball et al. (1995), p. 245-255, for an overview of other improvement heuristics.

accepted and a new intermediate solution is obtained; otherwise, the swap is rejected and the next pair of elements is investigated. In its classical form, all possible pairwise swaps are considered. However, in order to reduce the computational burden associated with large-scale combinatorial optimization problems, either a given iteration limit is set and/or the number of swaps is reduced in a controlled manner.

Grunow et al. (2004) propose a modified 2-opt exchange procedure for optimizing operations of a single placement machine for a unique setup strategy. For the group setup approach presented in this study, an extended version of this exchange procedure is developed. For each group determined by the group setup approach, the exchange heuristic is applied to improve the feeder assignment obtained for the group of PCBs and the placement sequence for each individual PCB type. In order to reduce the computational effort, the extended exchange procedure is used only in the final step of the grouping solution. The details of the exchange procedure are illustrated in figures 5.29 and 5.30.

Step 1. Select the first group in the final solution.

Step 2. Consider the given feeder assignment. Select for every component feeder randomly another feeder holding a different component type. IF a reduction in total assembly time for the whole group is achieved, THEN exchange the two corresponding positions in the magazine. (Note that in this step the number of exchange operations considered equals the total number of feeders in the magazine.)

Step 3. Consider the given sequence of placement operations for the first PCB in the group. Select for every placement operation randomly another placement operation as a counterpart for performing a pairwise exchange. IF a reduction in total assembly time is achieved, THEN swap the two operations and modify the corresponding tours of the placement head accordingly. (Similarly to *step 2*, the number of exchange operations considered equals the total number of placement operations required.)

Step 4. Select the next PCB in the group and repeat *step 3* until all PCBs in the selected group are considered.

Step 5. Return to *step 2* unless a given iteration limit is reached.

Step 6. Select the next group in the final solution and return to *step 2* unless all groups in the final solution are considered.

Figure 5.29: Exchange procedure

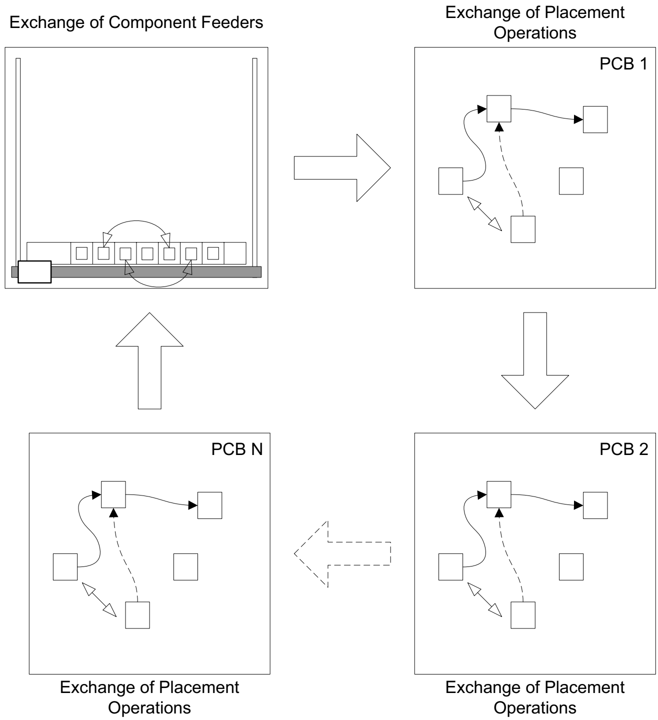


Figure 5.30: Illustration of the exchange procedure applied to a group of *N* PCBs

5.4.5 Determination of the Group Makespan

The heuristic procedures indicated in sections 5.4.1 and 5.4.3 are integrated into the clustering approaches presented in sections 5.1 and 5.2 in order to consider the group magazine setup and the placement time per PCB more realistically. Given the assembly time and the corresponding batch sizes of all PCBs $i \in I$, which are element of group j ($i \in Ij$), as well as the time required for setting up the magazine for group j, the total assembly time for the entire PCB group j can easily be derived. The global makespan is determined by adding up the total assembly times for all groups j:

$$\begin{aligned} \textit{global makespan} &= \sum_{j \in J} \textit{total assembly time}_j \\ &= \sum_{j \in J} \left(\textit{setup time}_j + \sum_{j \in I_j} \left(\textit{placement time}_i \times \textit{batch size}_i \right) \right) \end{aligned} \qquad (11)$$

This way, it is possible to examine if a reduction of the global makespan is achieved in the grouping procedure. In this study, an offline setup strategy is pursued where the setup time for each group corresponds to the time required for exchanging the feeder trolley. Hence, the time required for each setup operation can be considered to be constant.

In a group setup strategy it is assumed that, when changing over to a new group of PCB types, all feeders are removed from the magazine and each individual feeder required by the new group is assigned to the position in the magazine according to the predetermined magazine setup. Frequently, feeders which have already been used in the previous group must be relocated in the magazine in order to ensure the least possible assembly time for the new group of PCBs.

The proposed grouping approaches can also be used for the case of an online setup strategy where machine operations have to be stopped to assign feeders to magazine slots. However, in contrast to minimum and partial setup strategies, all component feeders required for the next group of PCBs have to be relocated even if these are available in the previous setup. Hence, the setup time for a group of PCBs can be calculated as the time to load and unload all component feeders required for its assembly. Thus, the setup time in equation (11) has to be calculated for each group.

The setup time for the exchange of nozzles is not considered in equation (11) because the nozzle setup operations are required before assembly of each PCB type, and thus can be taken as constant during the complete grouping process.

6. Numerical Investigation

In the following, a comprehensive numerical investigation is presented, which evaluates the performance of the proposed group setup approaches and highlights the effects of the new features integrated into the heuristic methods. The main questions addressed through the computational experiments are:

- Which conventional agglomerative linkage method performs best in combination with different similarity measures (section 6.2.1)?
- How effective is the inclusion-based clustering approach compared against conventional clustering methods (section 6.2.2)?
- What is the performance of the implemented improvement heuristics (section 6.2.3)?
- How do the proposed group setup approaches perform in comparison with a unique setup strategy (section 6.3.1)?
- How does the check for makespan reduction integrated into the group setup procedures improve their performance (section 6.3.2)?
- How does the consideration of batch sizes in the magazine setup heuristic of the group setup procedures improve their performance (section 6.3.3)?

Additional numerical experiments are conducted using two different sets of industrial PCBs from two different PCB manufacturers (section 6.4) and detailed analyses of group setup approaches are presented.

6.1 Experimental Design

6.1.1 Integrated Group Setup Solution System

In order to evaluate the performance of the group setup strategies presented in chapter 5, an integrated group setup solution system has been developed. Figure 6.1 illustrates the main components of the solution system including the required input and generated output data.

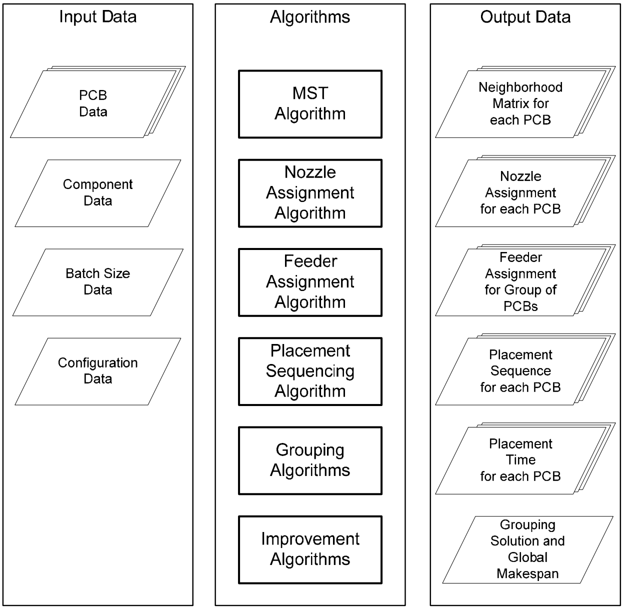


Figure 6.1: Structure of the integrated group setup solution system

For solving the PCB grouping problem, detailed data are required. X- and y-coordinates and the component type of each placement operation are stored for each PCB in a separate file. The component data file includes information on the required number of magazine slots and the nozzle type for each type of component to be assembled. The batch size data file feeds the group setup solution system with additional information on the batch size of each PCB type. Finally, grouping-specific parameters, e.g. setup time and grouping method, and machine-specific parameters, e.g. velocity of the gantry system and magazine slot capacity, are entered into the system using a configuration file.

The algorithms used in the grouping approach communicate between each other during the complete grouping process. The initial clustering solution of the proposed agglomerative approaches includes *n* clusters each consisting of a single PCB. Hence, each PCB is assembled with a unique magazine setup which includes only the component feeders required to carry out the placement operations of the specific PCB. For this purpose, the MST algorithm investigates the components on the PCB and generates the required neighborhood values required for determining the nozzle and feeder assignments. Next, the placement sequencing algorithm is applied to each single PCB to determine the actual placement time. The placement time of each PCB is then multiplied with the corresponding batch size. Adding the fixed setup time, assembly times for each PCB batch size are then determined.[131] The global makespan for the initial unique setup solution consists of the assembly times of each individual PCB batch.

After the global makespan of the initial solution is calculated, the grouping algorithm generates the initial similarity matrix and starts with the grouping procedure. At each grouping decision, the above described algorithms for optimizing machine operations are called in order to generate the feeder assignment for the new group and determine the individual placement operations for each PCB included in the group. This information is essential for calculating the group makespan value at each agglomeration stage realistically and comparing it with the previous solution including two groups with a separate setup for each. The grouping algorithm merges groups until no more groupings are feasible.[132]

Given the initial group setup solution, move and swap heuristics are applied for further improving the group setup solution. Subsequently, the adapted 2-opt exchange procedure is applied to each group in order to improve the group feeder assignment and the placement sequence of each PCB belonging to that group. Similarly, each improvement trial is evaluated in terms of makespan, and thus requires the execution of the machine-specific algorithms.

For the best setup solution determined, the integrated group setup solution system delivers the best placement sequence and nozzle assignment for each single PCB and the feeder assignment for each group of PCBs. Finally, the global makespan for the group setup solution as well as the placement time for each individual PCB are calculated.

6.1.2 Technical Parameters

The proposed heuristic procedures have been programmed using the C programming language and numerical experiments have been performed on a PC with 1.8 GHz AMD Athlon processor and 1 GB random access memory (RAM). In order to evaluate the performance of grouping approaches, the grouping problem is investigated for the case of a single gantry collect-and-place machine equipped with a 12-nozzle rotary placement head. The main characteristics of this machine which is considered throughout the detailed analysis are given in table 6.1.

---

[131] Cf. equation (11) on page 124.
[132] Cf. sections 5.1.4 and 5.2.3 for feasibility conditions.

Table 6.1: Characteristics of the placement machine

Number of placement segments	12
Segment rotation time (sec)	0.05
Pickup time (sec)	0.04
Placement time (sec)	0.04
Velocity in x-direction (mm/sec)	800.0
Velocity in y-direction (mm/sec)	800.0
Slot width (mm)	8.0

6.1.3 Generation of Test Data

Table 6.2 summarizes the experimental design parameters. For each experiment, a group of PCBs is generated from a set of 25 different PCB types. In order to investigate the effects of component commonality on grouping performance, two different scenarios have been investigated throughout the experiments. In the so-called low-commonality scenario, 25 PCBs are generated from a pool of 60 different electronic components. The component pool size is then reduced to 50 components in order to generate PCBs with higher component commonality. For both of the commonality scenarios, the number of component types per PCB is determined using a uniform distribution U[20,40] with an expected value (E) of 30.[133]

[133] Cf. Hogg and Tanis (1988), p. 19, for the definition of the expected value for uniform distributions.

Table 6.2: Experimental design

No. of PCB types in each experiment	25
No. of experiments	10
PCB dimensions (mm)	200×200
No. of component types	U[20,40]
Component commonality level	
Low	
Size of component type pool	60
High	
Size of component type pool	50
Component classes	
A	
Ratio of component types in pool	1/3
No. of operations per component type	U[5,11]
No. of slots required per feeder	1
B	
Ratio of component types in pool	1/3
No. of operations per component type	U[1,5]
No. of slots required per feeder	U[1,2]
C	
Ratio of component types in pool	1/3
No. of operations per component type	U[1,2]
No. of slots required per feeder	U[2,3]
Batch size scenarios	
Low	U[50,150]
Medium	U[50,450]
High	U[50,950]
Magazine capacity (no. of feeder slots)	
Limited	80
Unlimited	120
Setup time (sec)	
Low	300
High	900

The average pairwise commonality is selected as an index to measure the commonality between different PCBs used in the experiments. Average pairwise commonality is calculated for *n* PCBs as follows:

$$average\ pairwise\ commonality = \frac{\sum_{i=1}^{n}\sum_{j=i+1}^{n} pairwise\ commonality_{ij}}{n\times(n-1)/2}$$

$$= \frac{\sum_{i=1}^{n}\sum_{j=i+1}^{n} no.\ of\ common\ component\ types\ between\ PCB_i\ and\ PCB_j}{n\times(n-1)/2} \qquad (12)$$

Using the formula above, the average pairwise commonality is calculated as 14.48 and 18.26 for low- and high-commonality scenarios used in the experiments, respectively.

For each commonality scenario, the component pool is divided evenly into three classes. The so-called class A represents standard components which occupy only one single slot in the component magazine and which are assembled in a high number. The required number of slots for B and C class components is determined by using uniform discrete distributions U[1,2] and U[2,3], respectively. In order to achive a similar component distribution presented by the data generator of Föhrenbach (2002), uniform distributions of U[5,11], U[1,5], U[1,2] are selected for classes A, B and C, respectively. Hence, the expected ratio for the total number of placement operations per component class is determined as 64/24/12 (see table 6.3 for detailed calculations).

Table 6.3: Expected values for each class of components

Component class	A	B	C
E(no. of operations per component type)	8	3	1.5
Ratio of component types in the pool	1/3	1/3	1/3
E(percentage of placement operations)	64%	24%	12%

Placement locations for each component are randomly generated within the 200x200 mm dimensions of the PCB. Using the uniform distributions for the number of placement operations from table 6.2, the average number of components per PCB can be determined as 125 (see table 6.4). The expected values for the required number of feeder slots are 100 and 83.3 for each PCB used in the low- and high-commonality scenarios, respectively.

Table 6.4: Expected values for PCBs used in the experiments

E(no. of component types per PCB)	30
E(no. of components per PCB)	(8+3+1.5)*/3x30=125
E(no. of required feeder slots per PCB)	
Low-commonality	(1+1.5+2.5)**/3x60=100
High-commonality	(1+1.5+2.5)/3x50=83.3

* Sum of E(no. of operations per component type)
** Sum of E(no. of required feeder slots per component type)

Additional experimental factors include the average batch size, component magazine capacity, and setup time. Three batch size scenarios are analyzed: low, medium, and high with average batch sizes of 100, 250, and 500, respectively. Individual batch sizes are drawn from the uniform distributions indicated in table 6.2. The parameters of these distributions are chosen such that the variability of the batch sizes increases with the average batch size leading to more heterogeneous batch size structures.

The experiments carried out in this study assume that the observed placement machine is equipped with an interchangeable feeder trolley. In order to investigate the impact of different setup times on the performance of the grouping procedures, setup times of 300 and 900 seconds are considered.

The single-gantry collect-and-place machine, which is used throughout the experiments, is equipped with 80 feeder slots. In order to evaluate the effect of the component magazine capacity on the grouping performance, all problem instances are solved for both limited and unlimited capacity scenarios. During the data generation phase, all of the generated PCB instances are observed to require less than 120 magazine slots. Therefore, the unlimited capacity level is selected as 120 magazine slots.

Altogether 24 combinations of experimental factors are investigated for each group setup approach.[134] The random generation of the PCB types is repeated 10 times, meaning that each time a set of 25 different types of PCBs is created for which the grouping has to be performed. Thus, a total of 250 PCB types are generated for each, the low and high component commonality scenario. New batch sizes are generated randomly using the uniform distributions given in table 6.2 for each repetition and batch size scenario.

6.2 Comparison of Group Setup Approaches

In the first experimental phase, the performance of the proposed group setup approaches is compared against each other. Additionally, the effects of applying improvement heuristics on grouping results are also investigated.

6.2.1 Determining the Best Conventional Setup Approach

The first series of experiments are conducted to find out the best performing group setup approach based on conventional agglomerative clustering presented in section 5.1. The performances of Jaccard's and simple matching measures are investigated using different linkage

[134] 2 component commonality level scenarios x 3 batch size scenarios x 2 magazine capacity scenarios x 2 setup time scenarios.

methods, i.e. single, complete and average linkage. Thus, a total of 1440 experiments are carried out.[135] CPU times required to determine the group setup solutions for a problem instance comprising 25 PCB types are in the range of 12-20 seconds.

In order to select the best group setup approach based on conventional clustering techniques, the results of the experiments are evaluated under different criteria. Figure 6.2 illustrates the results based on the first criterion selected, namely average deviation of each method from the best solution obtained. Although the differences in the performance of clustering methods are minor, Jaccard's similarity measure is observed to be more robust to changes in the linkage method compared to the simple matching clustering.

Both similarity measures perform poorly if the complete linkage method is selected. However, the deterioration in the performance of Jaccard's measure is more significant. In complete linkage clustering, the similarity between two groups is determined by the lowest similarity connection. It is known that Jaccard's measure penalizes dissimilar products more than the simple matching measure because of disregarding the conjoint absence of components.[136] Hence, this may be the main reason for the observed effect regarding Jaccard's measure. The simple matching approach, on the other hand, delivers similar results for the average and complete linkage solutions. The most important observation is the good results achieved in general by single linkage methods which are based on defining the similarity between two clusters as the highest similarity between pairs of PCBs. This method has performed best with both similarity measures.

---

[135] 24 combinations of experimental factors x 10 PCB sets x 6 group setup approaches.
[136] Cf. section 5.1.2 for a comparison of similarity measures.

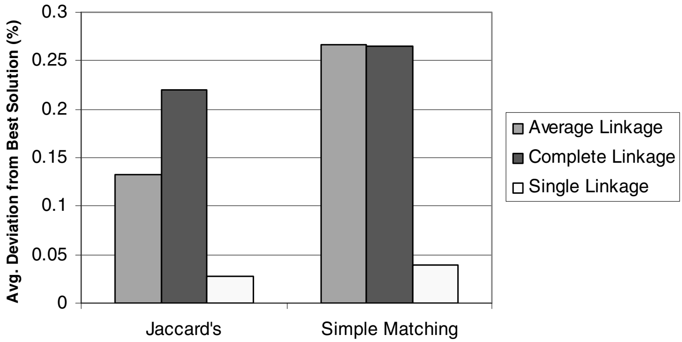


Figure 6.2: Average deviation of conventional clustering based approaches from the best solution found

Another investigation can be made by observing the average number of groups generated by each clustering approach (see table 6.5). As expected, the clustering methodologies generate less number of groups (around 3-4) for the case of unlimited capacity. This result reveals the fact that further grouping has generally not been feasible due to the magazine capacity limitations. If more magazine slots are available, more setup operations can be saved to achieve further makespan reductions. For the case of limited capacity, the algorithms generate similar results. However, in the unlimited case where the real performance of the linkage methods can be investigated, the differences in experimental results become more apparent. Analogous to previous observations based on the makespan deviation, the single linkage method performs best for both Jaccard's and simple matching measures.

Table 6.5: Average number of groups

Magazine capacity	Similarity measure	Linkage method		
		Average linkage	Single linkage	Complete linkage
Limited	Jaccard's	12.5	12.7	12.9
	Simple matching	12.9	12.7	12.8
Unlimited	Jaccard's	9.3	8.7	9.6
	Simple matching	9.5	8.7	9.4

In order to distinguish the effectiveness of the observed grouping approaches, the number of best solutions achieved by each methodology is considered as the final performance criterion. The results based on the number of best solutions achieved by each grouping approach are presented in figure 6.3. The results reveal that the simple matching measure performs best with single linkage clustering, i.e. this combination performed best in 111 and 109 of the 360 experiments conducted for each of the limited and unlimited capacity scenarios, respectively.[137]

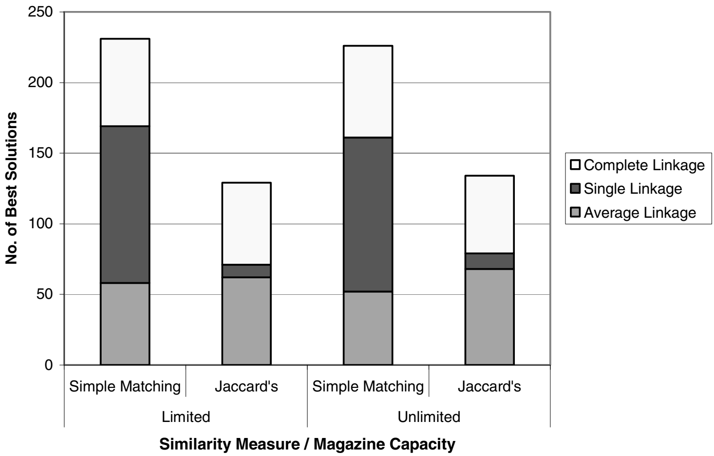


Figure 6.3: Number of best solutions obtained

Observing results of the detailed analysis above, the simple matching measure using single linkage clustering is selected as the best grouping approach based on the conventional agglomerative clustering. Hence, this approach will be used as a benchmark for evaluating the performance of the novel grouping approach based on the inclusion measure.

[137] 2 component commonality level scenarios × 3 batch size scenarios × 2 setup time scenarios × 3 linkage methods × 10 instances.

6.2.2 Comparison of Best Group Setup Approach Based on Conventional Clustering Against Inclusion-Based Grouping

In a second series of experiments, the best performing conventional approach, namely the single linkage clustering using the simple matching measure, is compared against the inclusion-based group setup strategy. Figure 6.4 illustrates the average deviation from the best solution determined for different magazine capacity and component commonality scenarios.[138]

The results reveal that the new grouping approach presented in this study, i.e. the inclusion-based group setup approach, performs better than the conventional group setup approaches in three out of four categories. The conventional approach based on the simple matching measure is only slightly better for the experiments with high component commonality and unlimited magazine capacity. Obviously, the performance of the inclusion-based approach increases if less similar PCBs are investigated. This is due to the implicit consideration of the available magazine capacity, as this method, by its nature, favors PCB pairs with a higher degree of inclusion regardless of conjoint absence or presence of component types. In contrast, the proposed simple matching clustering method tends to generate clusters with a higher number of component types in order to create better similarities, and thus approaches the magazine capacity limit much faster. For example, consider the unlimited magazine capacity for both low- and high-commonality scenarios. While the inclusion measure performs better in the low-commonality case (rightmost part of the figure), the performance of the simple matching approach deteriorates compared to its high-commonality results. Hence, simple matching improvements are inferior to improvements achieved by the inclusion-based approach for the unlimited capacity case. However, it has to be remarked that differences between both group setup approaches are just marginal. Therefore, further investigation based on a second criterion, namely the percentage of best solutions obtained, is essential in order to better distinguish the performance of both group setup approaches.

[138] 240 experiments are conducted for each group setup strategy (24 combinations of experimental factors × 10 PCB sets).

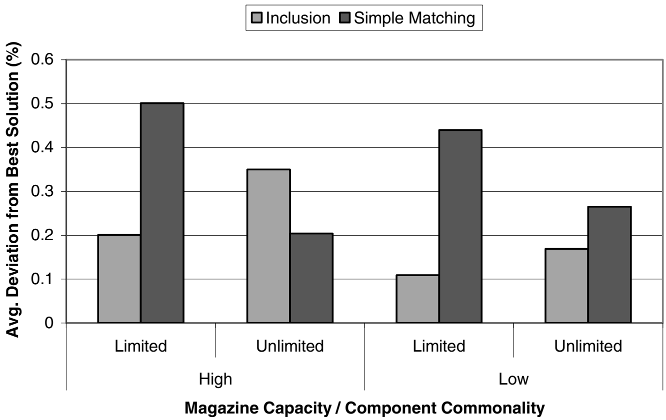


Figure 6.4: Average deviation of group setup approaches from the best solution found

The percentage of best solutions obtained for both examined group setup strategies are given in figure 6.5. The results demonstrate similar figures at first glance. In terms of the best solution found, the inclusion-based grouping approach dominates the simple matching approach under almost all conditions. Analogous to previous observations made on average deviation from best solution, the simple matching approach performs only slightly better in the unlimited capacity case with high component similarity.

Both of these figures recommend the usage of the inclusion-based approach, which focuses mainly on how components of one PCB are included in the other. This characteristic, which reduces the magazine slot usage, is more appropriate for the PCB assembly environment where the total number of slots required for assembling a group of PCBs usually greatly exceeds the magazine capacity. Because of the results given above, the inclusion-based group setup strategy will be used in further detailed analyses in the following sections.

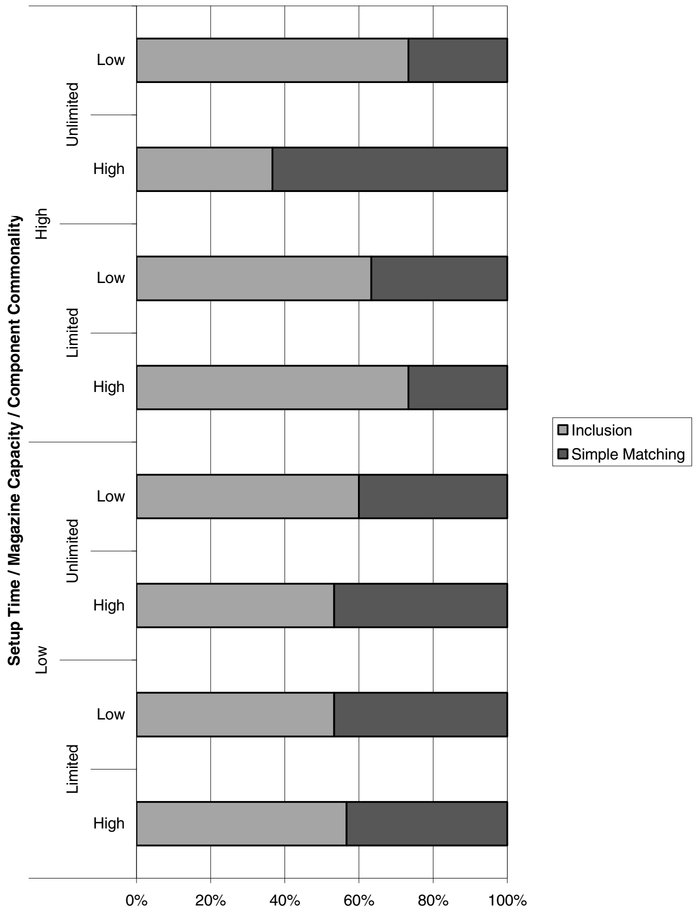


Figure 6.5: Percentage of best solutions found

6.2.3 Performance of Improvement Heuristics

The proposed approaches presented in this study differ from traditional clustering approaches by updating similarity measures and grouping feasibility at each decision node, and taking makespan and magazine capacity constraints into consideration. Nevertheless, hierarchical clustering schemes create only a single solution which is the disadvantage of these approaches against partitional clustering.[139] Therefore, the initial group setup solution of an hierarchical clustering approach can be further improved by applying local search heuristics.

In this study, two types of improvement heuristics are developed. For improving the initial grouping solutions of the hierarchical approaches, move and swap procedures, which are presented in section 5.3, are integrated into the solution process. After the best group setup solution is determined, the adapted 2-opt procedure is applied (see section 5.4.4) in order to improve the magazine setup for each group and the sequence of placement operations for each PCB. The move and swap heuristics are terminated after a structured search is conducted for the observed experiment. However, an iteration limit has to be defined in advance for the adapted 2-opt heuristic which improves the machine operations. In a series of experiments, a sufficient convergence of 0.89% improvement was reached after about 100 iterations. Thus, the iteration counter was set to 100 in following numerical experiments.

Results of improvements achieved by the above described improvement heuristics are presented in table 6.6. The average improvement presented for each heuristic is based on the improvements achieved by only using the selected approach on the results of the previous solution. The additional computational time for improvement heuristics is in range of 21-143 seconds for each experiment consisting of 25 PCBs.

Table 6.6: Results of improvement heuristics

		Average improvement (%)				Average total
Component commonality	Magazine capacity	Move heuristic	Swap heuristic	Adapted 2-opt heuristic	Total improvement	improvement (min)
High	Limited	0.00	0.10	0.88	0.98	43.56
	Unlimited	0.19	0.09	1.03	1.31	50.93
Low	Limited	0.00	0.10	0.82	0.92	38.63
	Unlimited	0.11	0.04	1.01	1.15	44.28

Considering the results given in table 6.6, the move heuristic has achieved a reduction in global makespan only for the case of unlimited magazine capacity. The main reason for the inefficiency of the move heuristic is the high utilization of the magazine capacity by each group generated throughout the experiments. Thus, the proposed move heuristic either could not further reduce the number of groups, or reducing one more group did not achieve any im-

[139] Cf. section 5.1.

provements in global makespan. However, the move heuristic was capable of improving the results slightly if there was enough magazine capacity available.

The move heuristic performs slightly better if the component commonality between PCBs increases. This is mainly due to the value of the global makespan selected as the reference point for the improvement. Because more similar PCBs can use the magazine capacity more efficiently, and thus generate fewer number of groups (less setup effort), the global makespan achieved for a similar PCB set is lower than the makespan for PCBs with low-commonality. Hence, an improvement made by reducing one group using the move heuristic is more effective for the high-commonality case.

After the move heuristic is applied to the initial groups, the swap heuristic, which exchanges each PCB with another randomly selected PCB from another group, is executed. For the observed case of 25 PCBs, 25 swap operations are considered (one random swap for each PCB) in order to reduce the computational burden. The results reveal that swap operations were successful to improve the makespan although the total improvement achived was quite small.

Considering the performance of both group improvement heuristics for the case of unlimited capacity, it is observed that the move heuristic performs better than the swap heuristic. This is because saving additional setup times usually result in more improvements than merely swapping PCBs. On the other hand, the swap heuristic is capable of generating better group configurations for many scenarios. The search process of the presented agglomerative grouping approaches are directed by board similarities and controlled by makespan and capacity constraints. However, using similarity alone as the only information for constructing the dendrogram fails in considering other relevant information, i.e. positions of the placement operations on each PCB and its batch size, which have an important effect on the global makespan. Results reveal that heuristic grouping solutions can further be improved by swapping PCBs, which probably show less similarity with the other PCBs of the observed groups, but still would make sense to exchange in terms of actual placement times.

In the conducted experiments, the makespan of the final group setup solution is further enhanced by the adapted 2-opt procedure, which tries to improve the feeder assignment for each group and placement sequencing of each PCB in the observed group. However, the improvement achieved by the adapted 2-opt procedure is very small. Similar to previously observed improvement heuristic results, achievements for the unlimited capacity scenario are slighty better than the capacitated case, which similarly depends on the lower value of the global makespan taken as the reference point for improvement. Nevertheless, the pure orientation of this approach on the placement time improves the solutions much better than other improvement approaches presented above.

The results demonstrate that improvement heuristics are crucial for further improving the solutions of the hierarchical clustering process which may oversee some potential makespan improvements because of its main focus on similarity. Although total improvements achieved by the presented heuristics seem to be minor at first sight, even one percent improvement may yield savings of several minutes or hours depending on the size of global makespan. In the investigated experiments, absolute improvements achieved were around 7-119 minutes with an average of 44.35 minutes.

6.3 Comparison of the Proposed Group Setup Approach Against Other Approaches

In the previous section, the performance of the proposed group setup approaches has been investigated and the best performing approach has been determined. In the following, group setup solutions are compared against the unique setup strategy and other conventional group setup approaches from literature. Detailed experiments are conducted for analyzing the effects of new aspects introduced into group setup.

6.3.1 Group Setup Strategy vs. Unique Setup Strategy

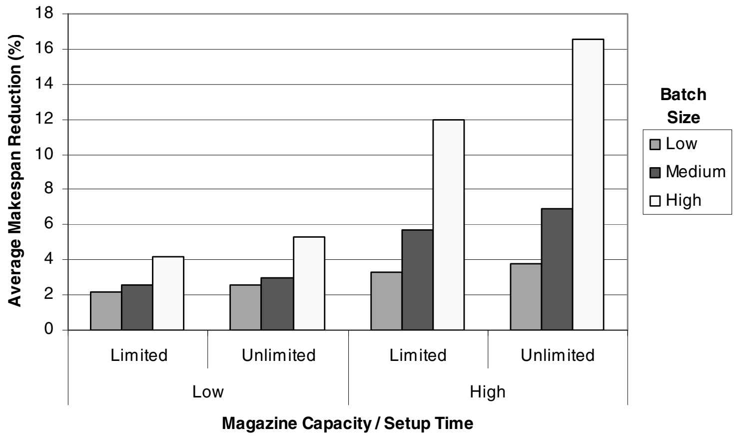
The general goal of the proposed group setup strategies is to reduce the global makespan by merging different groups of PCBs, and thus saving setup effort. The unique setup strategy, which fine-tunes the machine setup for each PCB type, constitutes the starting point for the agglomerative clustering procedure and provides a benchmark for evaluating the effectiveness of the grouping procedures. Clearly, a unique setup strategy is favorable in a mass production environment, while in a medium-variety assembly system, grouping of PCBs promises higher utilization of the placement equipment. As a reference measure for evaluating group setup strategies, the Makespan with Unique Setups (MUS) is used, which expresses the global makespan for a set I of PCB types, each one produced according to the unique setup strategy. The MUS measure is given in the following:

$$MUS = |I| \times setup\ time + \sum_{i \in I} \left(batch\ size_i \times unique\ setup\ \ placement\ time_i \right) \tag{13}$$

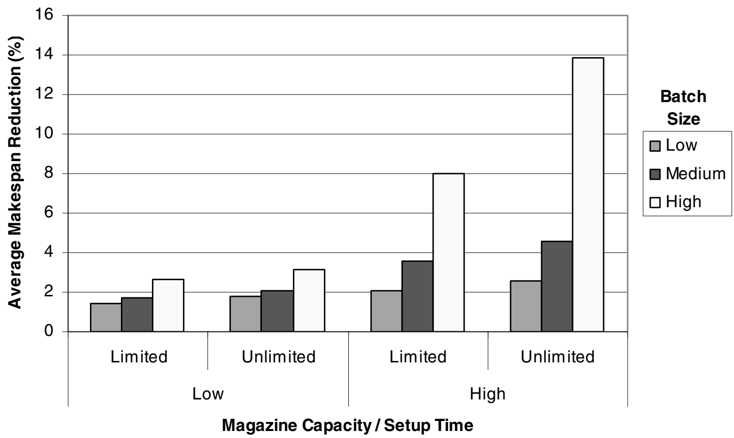
Figure 6.6 illustrates the effectiveness of the group setup strategy compared with the unique setup strategy. Results shown are based on the hierarchical clustering procedure using the inclusion measure, which has performed best in the initial experiments, and thus selected for the rest of the experiments.

It is experienced that the possible reduction of the global makespan is affected by the degree of component commonality. This is generally based on the capability of reducing the number of groups due to the reduced component pool size required to assemble the PCBs. Each PCB inserted into a group requires less extra feeders, and hence more PCBs can be added to existing groups before the magazine slot capacity is exceeded. This effect, similar to previous examinations, becomes more significant when the setup time increases. Clearly, the portion of total setup time in the global makespan is higher in the case of high setup time scenario. Thus, savings achieved by reducing the number of setup operations result in a more significant reduction of the global makespan.

External factors like magazine capacity, setup time, and batch size also have a considerable impact on the possible makespan reduction. In the extreme case of small batch sizes, unlimited component magazine capacity, and high setup times, the global makespan can be reduced by more than 16% with regard to MUS for the high-commonality case. With increasing batch sizes, however, the improvement over the unique setup strategy gets considerably smaller for each of the setup time / magazine capacity investigations. This is an expected result since the share of the total placement time in global makespan increases in contrast to the setup time if batch sizes become larger. Clearly, group setup strategy appears to be more effective in the case of high setup times, which can be seen in the right half of the figures 6.6 (a) and (b). Regarding the component magazine capacity, the proposed grouping method performs much better in the case of unlimited capacity, since larger groups of similar PCB types can be created.



(a)



(b)

Figure 6.6: Makespan reduction of group setup strategy over unique setup: (a) high- and (b) low-commonality scenarios

The average number of groups created by the proposed group setup approach for each different setting of setup time, batch size, magazine capacity and component commonality is given in figure 6.7. The results seen in this figure support the previous analysis on the reduction of the makespan. The unique setup solution, which is the starting point for the agglomerative clustering approach, considers 25 PCB groups each consisting of a single PCB type. The results in figure 6.7 reveal that in the worst case, namely the assembly of PCBs in large batch sizes with low setup times and limited magazine capacity, the group setup approach generates around 14 groups on average. As expected, reducing the number of groups, and hence saving more setup time becomes more profitable when batch sizes get smaller. Consider the results of the limited and unlimited magazine capacity scenarios in the left half of the figure. The number of groups is reduced significantly if enough magazine slots were available as this would result in further makespan improvements. However, such an improvement was not feasible for the case of limited capacity.

Clearly, similar effects are observed regarding the component commonality. Considering the limited magazine capacity, high-commonality PCBs achieve to generate less number of groups, in order to balance the increase in placement times with the savings from setup times. However, a detailed analysis of the results reveals that many favored groupings have been neglected for the low-commonality tests due to the limited magazine capacity. Comparing the right half of both component commonality scenarios (i.e. the unlimited capacity scenarios), the results between low- and high-commonality PCBs diminish. Hence, a similar number of groups can be achieved for both cases because of the missing magazine capacity limit. However, adding additional PCBs into an existing group leads to less additional component feeders for the case of high-commonality scenario. This causes less detoriation of the previous group magazine layout compared to the low-commonality case. Hence, the placement time, which heavily depends on the feeder assignment, worsens in the case of low-commonality groupings due to the higher number of feeders which are not required for assembling the current PCB. Thus, merging more PCBs into groups becomes meaningful if PCBs are similar.

In order to better understand the above described effects of the group magazine layout, the results of the unlimited capacity scenario from figure 6.6 should be compared with the results from figure 6.7. Although the number of groups given in figure 6.7 is not drastically reduced for the case of unlimited capacity, more improvement is achieved due to a better organization of feeders which yields a better placement time.

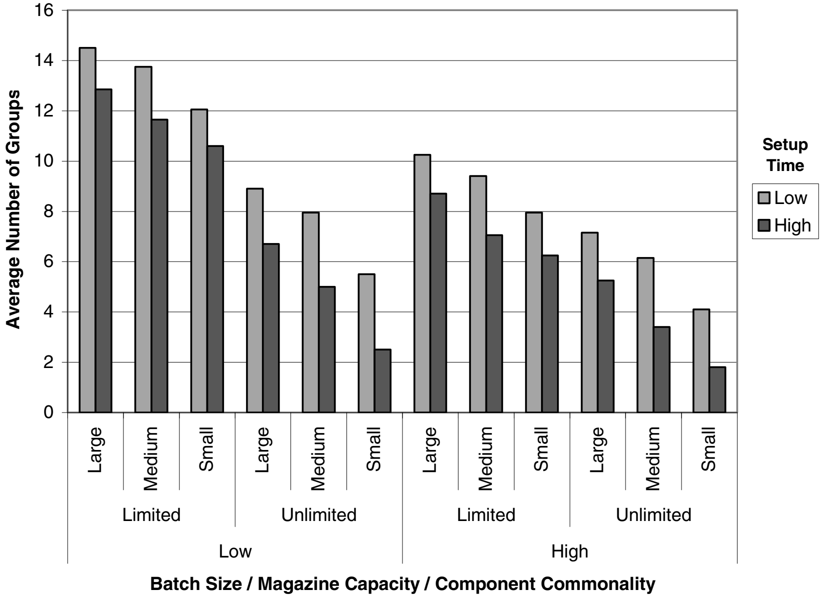


Figure 6.7: Average number of groups

The effect of magazine setup on the placement time is best obtained in figure 6.8 which illustrates the results of a selected experiment with small batch sizes and setup time of 900 seconds. As explained in section 4.3, group setup strategies require a joint machine setup for all PCBs in the group, while a unique setup strategy allows fine-tuning the machine for each individual PCB type. Hence, the actual placement times per PCB are expected to be higher than in the case of a unique setup. However, savings in setup time can be achieved, and grouping of PCBs is advantageous as long as the increased placement times do not overcompensate the savings in setup time. In figure 6.8, the group setup strategy merges five specific PCB types into one group. In the upper part of the figure, the resulting setup and placement times according to a unique setup strategy are shown, while the lower part of the figure indicates the total assembly time of the group for the case of the group setup strategy. Although four setup operations, i.e. 3600 seconds of setup time, could be saved, the overall reduction is only 3174 seconds (16.84%) due to an increase of the actual placement time caused by the group-oriented assignment of component feeders in the magazine. This simple example demonstrates the importance of integrating machine-specific algorithms for optimizing the operations of the placement machine into the grouping procedure.

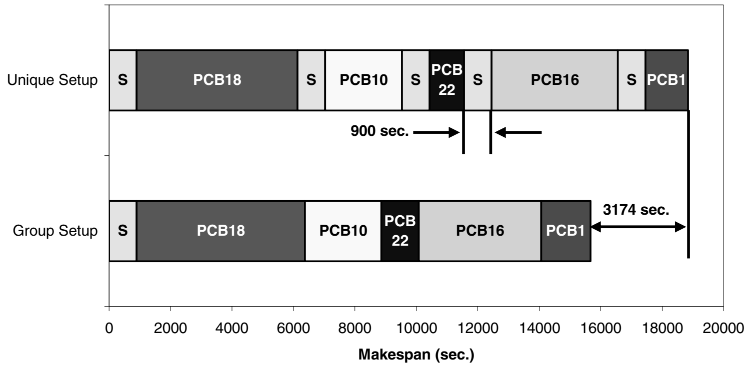


Figure 6.8: Reduction in setup times vs. increase in individual placement times

6.3.2 Effect of the Makespan Constraint in the Grouping Procedure

One of the significant contributions of the proposed group setup approach is the integration of machine-specific algorithms into the grouping procedure to calculate the improvement in makespan more realistically. Conventional group setup approaches ignore this effect and mainly concentrate on reducing the number of setups. This may only be true if adding more products into existing structures would not deteriorate the placement times of all PCBs. In order to show the vitality of considering the makespan in PCB grouping, a series of tests have been conducted on the 240 instances, both with the makespan criterion for accepting a grouping decision at each decision node, and without the makespan constraint as it is done in the conventional PCB grouping. The results in figure 6.9 reveal the essentiality of applying makespan the improvement criterion to the group setup approach. The results for both component commonality levels are aggregated because there has not been a significant difference with respect to the degree of component commonality.

The level of ‘0’ represents the MUS value achieved using a unique setup approach. The results clearly demonstrate that even a unique setup solution would outperform a conventional group setup approach in 7 out of 12 examinations. The negative deviation of a conventional group setup strategy from a unique setup approach becomes more significant for the case of large batch sizes. In such a production environment, a unique setup strategy may be more beneficial instead of applying a conventional group setup approach which is merely based on similarities and the magazine capacity constraint. A conventional group setup approach tries to iteratively group PCBs until a given magazine capacity limit is reached. In case of unlimited magazine capacity, the final solution would result in a single group consisting of all PCB types, which would further worsen the situation (see the first column in the right half of the figure for the unlimited capacity scenario). Hence, the conventional group setup approach would only perform well if setup times are high and batch sizes are small enough so that the savings achieved by reducing the setup effort compensates the worsening in individual placement times.

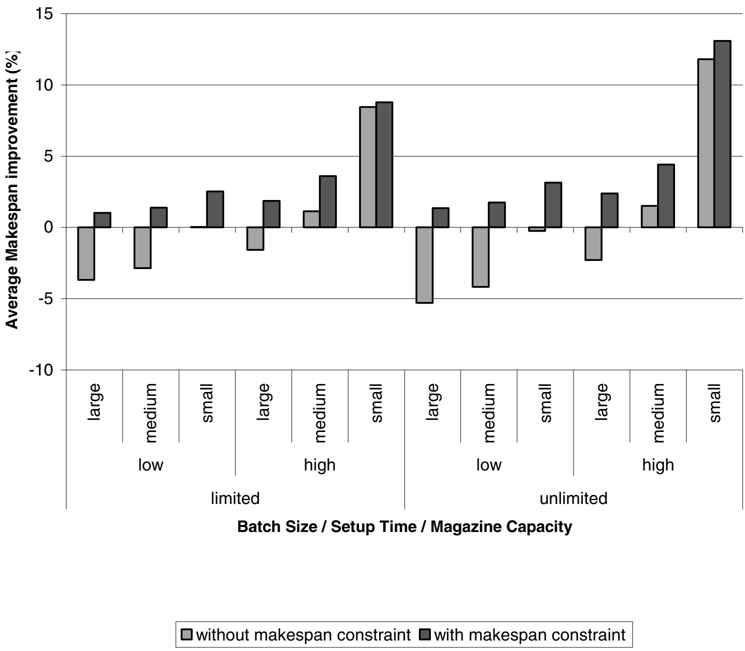


Figure 6.9: Effect of the makespan constraint on average makespan improvement

As seen in the yukarıda given figure, the proposed group setup approach performs best in all cases compared to the unique setup solution and the conventional group setup approaches known from literature. This is because a grouping would only be allowed if there is an improvement in the makespan. If merging two groups including a number of PCBs does not improve the global makespan, these groups are not considered further for any mergers, because joining them later with more PCBs in each group would deteriorate the placement times even more significantly. This allows an efficient novel search in the dendrogram and an update of nodes in each stage where the makespan criterion results in infeasibility. Thus, the proposed group setup approach results in a unique setup solution in the worst case if no improvement can be achieved by any grouping.

The effect of batch sizes is also integrated into the solution indirectly using the makespan constraint. Clearly, adding two small batches of PCBs in a heteregenous batch size environment would be more meaningful than joining two big batches of PCBs. Saving one setup time may clearly compensate the deterioration of the placement time for PCBs with small batch sizes, but may not be meaningful for boards assembled in large batches. A conventional group setup approach clearly ignores this critical issue and tries to group PCBs merely based on the similarities, which is one of the reasons for its poor performance.

The results of the conducted analysis reveal that the proposed group setup approach is quite flexible in finding the best solution for the best production environment. The predetermination of the appropriate setup strategy based on the characteristics of the observed production environment vanishes if the proposed group setup approach is applied. Hence, the planner does not need to decide on if the production environment is a high-, medium- or low-volume environment. The decision on the production environment depends additionally on the duration of the setup times. If setup times are relatively large, grouping would even be meaningful for a production environment with large batch sizes. If the possibility of fast feeder exchange (e.g. by using trolleys) is given, grouping PCBs, despite the sufficient magazine capacity, may not be meaningful anymore. The main advantage of the presented approach is the flexibility in fulfilling the requirements of a production environment. Hence, the proposed approach reacts as a unique setup strategy or a family setup (a single setup for all observed PCBs) in extreme cases. This is because grouping is only allowed if there is an improvement in the global makespan.

6.3.3 Effect of Integrating Batch Sizes in the Feeder Assignment Procedure

Finding the best magazine layout for a group of PCBs is essential for optimizing the placement times of each PCB. This is not a trivial issue as a change in the feeder assignment has an effect on placement times of all PCBs within the group. Hence, both initial feeder assignment and later improvement of the magazine layout must consider this interdependency.

As explained before, composite PCBs, which comprise all placement operations of individual PCBs to be grouped, are widely used in PCB assembly literature to determine the feeder assignment. However, the use of the composite PCB approach has some deficits. Firstly, a composite PCB represents each component on each PCB only once, and hence does not reflect the effect of batch sizes, i.e. the total number of placement operations for the observed component. As a result, each PCB is given equal weight on deciding the location of feeders in group magazine layout. In a composite PCB approach, the assignment problem is solved using placement location data from several PCB types. However, placement operations of each board type are carried out separately. Hence, solving the assignment problem using the composite PCB would result in a solution which generates infeasible subsequent placement operations.

The proposed approach based on integrating the batch sizes in group magazine formation, however, observes each PCB separately and weights the neighborhood solutions with corresponding batch sizes. Thus, PCBs with larger batch sizes and/or including larger number of placement operations (both result in a longer placement time) dominate the decision on the assignment of feeders in the group magazine layout. In other words, a “large” PCB gets the chance of locating its most neighboring component feeders as adjacent as possible. This concept enables giving larger PCB batches more chance to benefit from the feeder assignments while placement times for much smaller PCBs might detoriorate. However, this tradeoff serves the main objective of reducing the global makespan.

Figure 6.10 illustrates the effect of integrating batch sizes in the group magazine setup procedure. Under all circumstances, the group setup approach integrating batch sizes in the feeder assignment process performs better than approaches which evaluate PCBs with equal weights. Although improvements achieved are not so significant like the effect of the makespan constraint, the integration of batch sizes in the feeder assignment procedure is still essential.

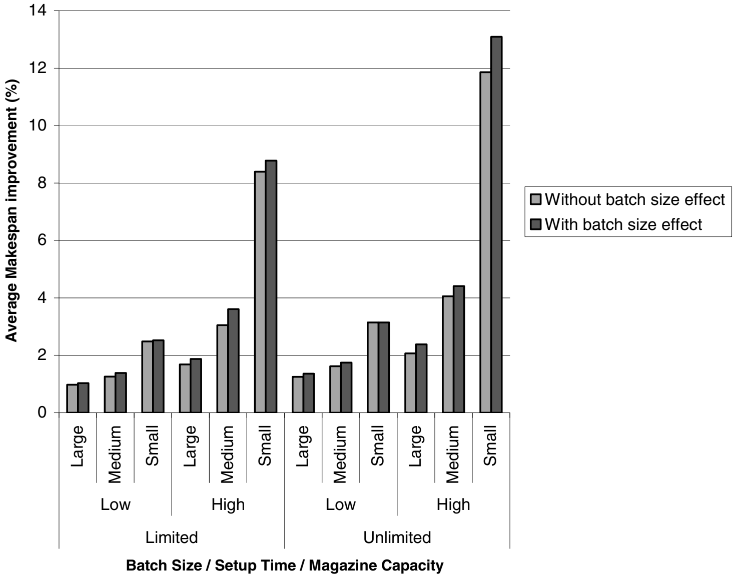


Figure 6.10: Effect of integrating batch sizes in the feeder assignment procedure

6.4 Numerical Tests on Industrial PCBs

Numerical results presented in the previous subsections have been derived from a comprehensive experimental design using randomly generated PCBs. In this procedure, experimental settings were systematically varied in order to test the performance of the clustering approaches under different conditions. In the following, several investigations on industrial data from two different PCB manufacturers are presented. In the industrial applications considered, the component magazine capacity of the key placement machine turns out to be a bottleneck. Hence, the hierarchical clustering procedure using inclusion trees, which performed best in the previous experiments, has been applied for grouping PCBs.

6.4.1 Tests on Industrial PCBs with Low Component Commonality

Test Data

The data from the first PCB manufacturer consist of 8 PCBs assembled for use in automation control equipments. The main characteristics of the industrial PCBs used in the experiments are given in table 6.7. Component feeders require one or more slots in the magazine of the placement machine. The total number of feeder slots required for each type of PCB is given in the fourth column of table 6.7. The component magazine capacity of 80 slots is sufficient to accommodate all feeders for a single PCB type, but not for the entire set of PCBs. Totally, five different types of nozzles are required to assemble the given group of PCBs.

Because actual batch sizes were not available, the batch size scenarios defined in table 6.2 are applied to 10 replications for both low and high setup times. Each replication corresponds to a set of randomly generated batch sizes for each of the PCB types. The initial group setup solution for each replication is determined between 49-227 seconds of CPU time with an average of 148 seconds. Additional time of 173-242 (avg. 207) seconds is required for the improvement heuristics.

Table 6.7: Characteristics of the first set of industrial PCBs

PCB	No. of components	No. of component types	Total no. of feeder slots required	PCB dimensions (mm)
1	420	21	52	246×327
2	152	10	30	253×178
3	448	11	22	253×181
4	424	26	63	266×211
5	501	21	48	380×333
6	537	25	56	363×332
7	460	18	38	223×343
8	416	17	35	245×329

Analyses on Makespan Improvement

The average improvement achieved by applying the proposed group setup strategy against the unique setup strategy is presented in table 6.8. The results reveal that the group setup strategy achieves only small improvements compared to the unique setup strategy. This is due to the low level of commonality (average pairwise commonality of 2.32) and the high magazine slot usage, i.e. the entire set of PCBs requires 241 slots in total which greatly exceeds the available machine capacity of only 80 or 120 slots. Hence, the number of groups could only be reduced to 6 in the best case as shown in table 6.9. It has to be emphasized that the improvement of the global makespan is mainly based on the reduction of the setup time. Nevertheless, detailed analyses have shown that the makespan constraint has also avoided some possible groupings. Because component feeders of the observed PCBs are quite different, producing PCBs with a unique setup was generally preferable because of the high number of additional feeders to be added into the new group magazine. Allocation of several new feeders significantly deteriorates the performance of the gantry movements for pickup operations, and thus exceeds the savings achieved by an additional setup operation.

In order to further investigate the above described effects, a detailed analysis is carried out on a selected specific experiment.

Table 6.8: Average improvement of group setup strategy against MUS

		Batch size scenario		
Setup time	Magazine capacity	Large	Medium	Small
Low	80	0.477	0.5	0.5
	120	0.461	0.5	0.6
High	80	0.517	0.7	1.5
	120	0.545	0.8	1.6

Table 6.9: Average number of groups formed

		Batch size scenario		
Setup time	Magazine capacity	Large	Medium	Small
Low	80	7.7	7.5	7
	120	7.7	7.3	6.8
High	80	7.4	7	6
	120	7.3	6.8	6

Detailed Analysis of a Selected Experiment

For further investigating the above described reasons for the low grouping performance, an experimental run with small batch sizes (see table 6.10), high setup time and 80 available magazine slots is selected. The initial inclusion matrix for the observed PCBs is given in table 6.11.

Table 6.10: Small batch sizes scenario applied to the observed experiment

PCB	1	2	3	4	5	6	7	8
Batch size	112	278	622	807	236	560	423	683

Table 6.11: Inclusion matrix for the observed PCB data

PCB	1	2	3	4	5	6	7	8
1		0.3	0.182	0.115	0.0476	0	0.0556	0.0588
2	0.143		**0**	0.115	0.0476	0.04	0	0
3	0.0952	**0**		0.192	0.0952	0.08	0.0556	0.0588
4	0.143	0.3	0.455		0.143	0.12	0	0
5	0.0476	0.1	0.182	0.115		0.44	0.167	0.176
6	0	0.1	0.182	0.115	**0.524**		0.167	0.176
7	0.0476	0	0.0909	0	0.143	0.12		**0.529**
8	0.0476	0	0.0909	0	0.143	0.12	0.5	

The largest inclusion measure of 0.529 found for PCBs 7 and 8 is taken as the first candidate. Both makespan reduction and magazine capacity constraints have to be satisfied for a feasible grouping. For the group magazine layout of these two PCBs, 26 component feeders occupying 54 slots in the component magazine are required. Hence, the magazine capacity constraint is satisfied as all feeders of both PCBs can be fit on 80 available slots. In addition, the makespan improvement also has to be assured. In the initial solution where each PCB is assembled using a unique setup, the total placement time for the batches of PCB 7 and 8 is calculated as 11831.82 and 15733.2 seconds, respectively. The total makespan for producing these two PCBs separately including setup times (2×900 seconds) is determined as 29365.02 seconds (see table 6.12). In order to evaluate the changes on the placement times after a group setup, machine-specific algorithms for the group feeder assignment and individual placement sequences are applied. The total assembly time for both batches of PCBs including a single group setup time is calculated as 28963.189 seconds. Because the global makespan is improved by 1.4%, the observed grouping is allowed. In the subsequent search process, it is observed that PCB 3 is a feasible candidate to join the new group of {7,8} in terms of the magazine capacity restriction. Therefore, group {7,8} is left in the grouping procedure for possible further mergers and all inclusion values between group {7,8} and other individual PCBs are updated.

Next, PCBs 6 and 5 which have the second biggest inclusion measure of 0.524 are observed. However, 81 magazine slots are needed to allocate all feeders required for their assembly.

Therefore, it is not feasible to add them together into a group. In the following grouping process, no more groupings were feasible either in terms of makespan reduction or magazine capacity until PCBs 2 and 3 are observed. Although both of the asymmetric inclusion measures of these PCBs are ‘0’, i.e. none of the PCBs includes any component type required for the other, it is feasible to add them into a group which would only require 52 magazine slots. The resulting makespan improvement is calculated as 381.932 seconds. Hence, a second group {2,3} is generated from these PCBs.

The final group setup solution consists of 2 groups with 2 PCBs each, and 6 groups including single PCBs with unique setups. The group setup solution brings a total improvement of 783.77 seconds (0.69%) compared to MUS. Because of the strict magazine capacity limit, move and swap heuristics did not show any improvements for the observed case. However, the total makespan improvement is calculated as 1742.9 seconds (1.54%) after applying the improvement heuristic for feeder assignment and placement sequencing.

The results in table 6.12 depict that the level of makespan detorioration increases with decreasing similarity. Hence, this proves that the placement time, which heavily depends on the feeder assignment, deteriorates if the number of additional feeders in the component magazine increases drastically. A conventional grouping approach, which neglects the makespan effect, would further group PCBs although the global makespan would not benefit from the savings achieved by setup reduction. The presented novel grouping approach avoids such cases which occurred in seven decision nodes for the observed experiment.

Table 6.12: Details of the grouping process for the observed experiment

Observed groups	Inclusion measure	Total makespan for groups: Before grouping	After grouping	Feasibility conditions: Magazine capacity (required no. of slots)	Makespan reduction (% improvement for observed groups)	New group
{7} & {8}	0.529	29365.02	28963.189	fulfilled (54)	fulfilled (1.4%)	{7,8}
{6} & {5}	0.524			failed (81)		
{4} & {3}	0.455	34354.719	34461.719	fulfilled	failed (-0.3%)	
{1} & {2}	0.3	12291.75	12336.75	fulfilled	failed (-0.37%)	
{4} & {2}	0.3			failed (84)		
{7,8} & {5}	0.19			failed (94)		
{5} & {3}	0.182	28521.5	28686.48	fulfilled	failed (-0.58%)	
{1} & {3}	0.182	22508.102	22955.578	fulfilled	failed (-1.99%)	
{6} & {3}	0.182	37106.73	37843.133	fulfilled	failed (-1.99%)	
{4} & {5}	0.143			failed (104)		
{4} & {1}	0.143			failed (108)		
{7,8} & {6}	0.12			failed (104)		
{4} & {6}	0.12			failed (112)		
{5} & {2}	0.1	18305.15	18797.352	fulfilled	failed (-2.69%)	
{6} & {2}	0.1			failed (83)		
{7,8} & {3}	0.091	43332.289	43790.328	fulfilled	failed (-1.06%)	
{5} & {1}	0.048			failed (97)		
{7,8} & {1}	0.048			failed (104)		
{3} & {2}	0	18521.852	18139.92	fulfilled (52)	fulfilled (2.06%)	{3,2}
{6} & {1}	0			failed (108)		
{7,8} & {4}	0			failed (117)		

6.4.2 Tests on Industrial PCBs with High Component Commonality

Test Data

In a second series of experiments, a set of 12 PCBs from a manufacturer of telecommuncation equipment is investigated. The characteristics of these PCBs are given in table 6.13. The average pairwise commonality for the observed PCBs is calculated as 15. Similar to the previous industrial case, component feeders require one or more slots in the magazine of the placement machine. The total number of feeder slots required for each type of PCB is given in the fourth column of table 6.13. The component magazine capacity of 80 slots is sufficient to accommodate all feeders for a single PCB type, but not for the entire set of PCBs. Since the nozzle information is missing from the manufacturer, an assumption is made that each component requiring a different slot width requires a different nozzle type. Totally, three nozzle types are assumed to be sufficient for the production of the observed PCB types.

Because actual batch sizes were not available, the batch size scenarios defined in table 6.2 are applied to 10 replications for both low and high setup times. Each replication corresponds to a set of randomly generated batch sizes for each of the PCB types. Each experimental run takes 13-45 seconds of CPU time with an average of 26 seconds for finding the initial groups. Additional time of 33-90 (avg. 63) seconds are required for improving the solutions.

Table 6.13: Characteristics of the second set of industrial PCBs

PCB	No. of components	No. of component types	Total no. of feeder slots required	PCB dimensions (mm)
1	336	33	44	312×211
2	121	41	59	220×153
3	165	31	49	217×212
4	149	11	11	227×201
5	149	11	12	227×201
6	121	41	59	220×153
7	109	24	45	227×203
8	79	28	49	223×105
9	422	25	37	226×217
10	197	31	50	117×213
11	238	57	64	222×209
12	180	53	59	223×209

Analyses on Makespan Improvement

Figure 6.11 shows the average improvement achieved by applying the group setup strategy based on hierarchical clustering over the makespan resulting from the unique setup strategy (MUS). Both the initial makespan improvement and the final group setup solution after applying the improvement heuristics are presented for different batch size, magazine capacity and setup time scenarios. The results reveal that the group setup strategy achieves significant improvements compared to the unique setup strategy. Especially for small batch sizes and large setup times, the average improvement reaches up to 6-7%.

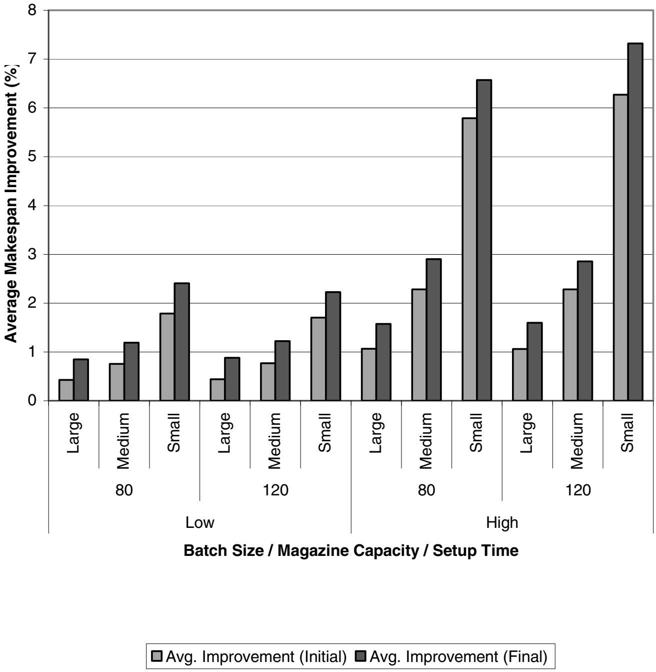


Figure 6.11: Average makespan improvement of group setup strategy against MUS

The analysis from figure 6.11 can be further detailed with the results in table 6.14 representing the average number of groups formed. The results show that the number of groups reduces when batch sizes get smaller. This is an expected result because smaller batch sizes result in shorter assembly times for each PCB job which increases the percentual savings achieved by reducing the setup operations. Hence, grouping becomes more advantageous for small batches.

Another important observation is that the number of groups stays relatively constant for the low setup time scenario independent of the magazine capacity. This result demonstrates that the main termination criterion for the grouping approach is the makespan constraint, i.e. reducing the number of groups would not yield any more makespan improvements although the magazine capacity would allow further groupings. The situation changes for the high setup time scenario in the low-capacity case, where grouping was terminated mainly by the magazine capacity constraint, especially for medium and small batch sizes. When more magazine slots (120 slots) are available, the number of groups can be further reduced to achieve more improvements. In other words, the high setup time of 900 seconds becomes more significant in global makespan and more savings in setup time are favored.

Table 6.14: Average number of groups formed

Setup time	Magazine capacity	Batch size		
		Large	Medium	Small
300	80	8.5	7.7	5.8
	120	8.5	7.6	5.8
900	80	6.9	5.2	5
	120	6.9	4.8	3.2

Detailed Analysis of Makespan Improvement

In contrast to the conventional clustering approaches known from the literature, the group setup strategies proposed in this study integrate a makespan constraint into the grouping procedure. In order to evaluate the effectiveness of this makespan constraint, two series of experiments, i.e. with and without consideration of the constraint, are conducted on the observed industrial data. Figure 6.12 displays the experimental results of the industrial PCB set using small batch sizes scenario. An examination of the results reveals the following:

- Consider the case of high component magazine capacity in figure 6.12 (a). Obviously, a simple grouping algorithm, which merely analyses the component similarity and magazine capacity without taking the influence of grouping decisions on the global makespan into account, merges all PCBs types into the lowest feasible number of groups. As a result, PCB types are assembled only in two large groups each using the same magazine setup for a high number of individual PCBs. However, the setup time savings are outweighed by the increase in placement time. Thus, this approach is outperformed by the proposed clustering procedures with integrated makespan consideration.

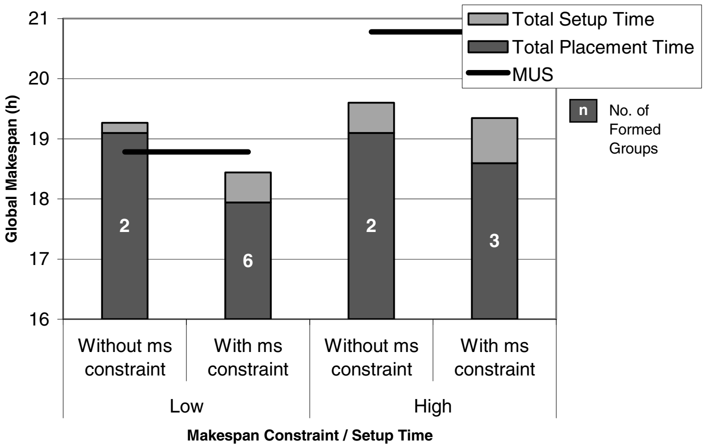
- The leftmost bar shown in figure 6.12 (a) even demonstrates the extreme case of a total group makespan being larger than the MUS resulting from a unique setup strategy. Hence, an appropriate unique setup strategy may be more suitable than the application of a simple group setup approach. Comparing the leftmost bars in figures 6.12 (a) and

(b) reveals that the consideration of a low component magazine capacity improves the resulting makespan, but results are still slightly worse compared to a unique setup strategy. This outcome is surprising at first glance, but it clearly demonstrates the consequences of ignoring the makespan constraint in the conventional grouping approach.

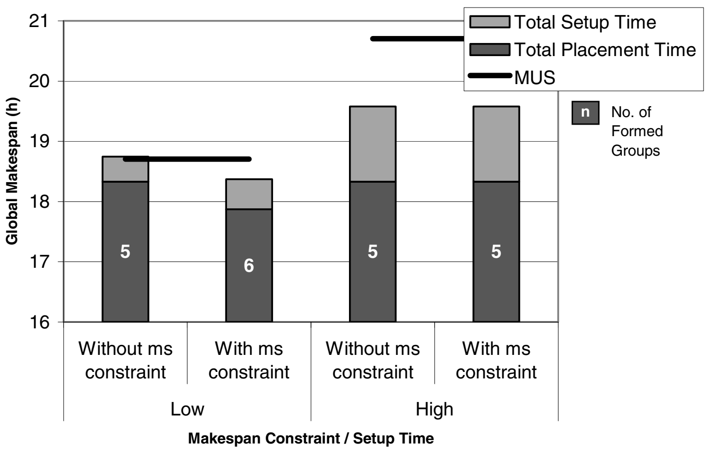
- As can be seen from all combinations of component magazine capacity and setup times shown in figure 6.12, the conventional group setup approach, which does not consider the makespan constraint, performs worse than the proposed novel group setup approaches, which integrate the check for possible makespan reduction. Merely for the case of low component magazine capacity, both approaches perform the same. Hence, the magazine capacity limitation has obviously outweighted the makespan criterion in the grouping process.

- The resulting number of groups varies considerably depending on the setup time and the component magazine capacity. In the case of high setup times (rightmost bars in figures 6.12 (a) and (b)), the proposed procedure tends to create a smaller number of groups, i.e. groups containing more PCB types. However, more groups are obtained for the case of high setup time when a lower limit for component magazine capacity is selected.

- Finally, it should be noted that the proposed group setup strategy includes the unique setup strategy as a special case if no savings in makespan can be achieved with any grouping.



(a)



(b)

Figure 6.12: Effect of the makespan (ms) constraint in the clustering procedure for: (a) high and (b) low component magazine capacity scenarios

6.5 Concluding Remarks

Experimental results underline the importance of integrating machine-specific algorithms into the group setup strategy solutions, where both setup and placement times play a significant role in minimizing the global makespan. Thus, approaches which merely focus on reducing the number of setup operations might perform even worse than a unique setup strategy.

The group setup strategy based on the inclusion measure, which reduces the magazine slot usage, performs best for the PCB assembly environment where the total number of slots required for assembling a group of boards usually greatly exceeds the magazine capacity. The adoption of the original clustering algorithm from Raz and Yaung (1994) to the PCB assembly environment has been proven to deliver good results with high magazine capacity utilization.

Experiments reveal that batch sizes and makespan observations should constitute fundamental parts of the group setup solutions in addition to the component magazine capacity. A group setup solution, which pursues the objective of minimizing the global makespan, has the ability of flexibly determining the number of groups depending on the production environment. Hence, it may even perform like unique or family setup in extreme cases.

7. Conclusions

The rapidly developing market for electronic products requires the use of flexible and highly automated assembly systems. Placement machines constitute the heaviest investment of an electronics assembly system which usually defines its throughput. Several analyses demonstrate that the real productive capacity of a placement machine remains much below the theoretical capacity given by the machine vendors. Hence, development and implementation of advanced planning and control systems are essential for exploiting the potentials of highly automated assembly lines.

In order to conceive the problems arising in the electronics assembly, one should understand the SMT technology and its future trends. The categorization of placement machines given in the **technological background** section assists the reader to apprehend how machine-specific optimization problems depend on the kinematics of the placement machine. Thus, an appropriate solution approach has to be tailored for each placement machine individually. The so-called X-Y gantry systems appear to be promising in responding to the requirements of the future electronics assembly because they provide a high degree of flexibility and modularity. Depending on the requirements of the production environment, these machines can be integrated into an assembly line in different modular configurations.

An overview of the planning problems is presented using a novel **hierarchical decomposition** based on the characteristics of the production environment. The previous studies which are discussed in the **literature review** have focused mainly on the setup strategies without tackling the intertwined problems of other hierarchy levels. On the contrary, the **novel group setup strategy** developed in this study integrates the machine-specific optimization problems into the solution approach and covers the new aspects of modern placement machines, e.g. utilizes the advantages of an offline setup.

The term **group setup** is considered as a synonym for creation of setup families in order to reduce the changeover effort. However, several analyses presented in this study have shown that the placement time of an individual PCB increases if different board types are assembled using a joint group setup. This is due to the new group feeder assignment which cannot be optimized for each single PCB as in the case of a unique setup strategy. This study focuses on reducing the **global makespan** by observing both the savings in setup time and the increase of individual placement times by integrating detailed **machine-specific algorithms** into the solution methodology.

Two different approaches based on **hierarchical agglomerative clustering** schemes are presented. The hierarchical clustering methodology enables observing different criteria at each decision stage for merging groups of PCBs. Machine-specific algorithms are used to evaluate the quality of the new solution at each agglomerative clustering step in terms of **makespan improvement** and **magazine capacity constraints**. Using a hierarchical scheme also reduces the number of times machine-specific algorithms are called, and thus the computational burden. The experimental results reveal that fast solutions can be generated in short computational times.

In the first group setup approach presented in this study, grouping is performed by use of well-known **similarity measures** (Jaccard's and simple matching) and **linkage methods** (single, complete, and average). The second group setup approach employs **inclusion measure** as a similarity coefficient and generates setup families using a novel hierarchical clustering technique which is based on the **inclusion tree** representation scheme. Conventional clustering techniques are modified in order to comply with the characteristics of the job grouping problem in PCB assembly. Hence, proposed group setup approaches allow grouping only if both conditions for **makespan improvement** and **magazine capacity constraint** are satisfied.

Because of the hierarchical structure of the grouping process, two **improvement heuristics** are implemented and applied to the initial group setup solution. The **move heuristic** attempts to decrease the number of setup groups until no more reduction in the number of groups is achievable. After applying the move heuristic, the **swap heuristic** exchanges each PCB job with another randomly selected one to further reduce the makespan. Both heuristics could improve the initial results of the conducted experiments only slightly, which demonstrates the quality of the solutions achieved by the novel group setup approaches.

In order to realistically determine the global makespan, **machine-specific optimization problems** have to be solved at each grouping stage. A hierarchical solution scheme is presented for tackling all problems involved in optimizing the operations of a single **collect-and-place machine** equipped with a **rotary placement head**. In the first stage, the **feeder assignment** problem is solved for determining the magazine layout for each group of PCBs. The neighborhood-based greedy assignment heuristic considers the batch sizes of each individual PCB and is also capable of allocating feeders of different width. Experimental results prove that integrating batch sizes into the feeder assignment heuristic performs better than applying the composite PCB approach which is commonly used in the literature. The **nozzle allocation** problem is discussed for the first time for this type of placement machine. The optimal nozzle set determined for each board type is assigned to the segments on the placement head considering the specific operational characteristics of the collect-and-place machine. The **placement sequencing** problem is solved with an adapted version of the savings algorithm. Since the above described interdependent problems are handled within a hierarchical scheme, a **local search** heuristic based on the well-known 2-opt algorithm is applied to the final group setup solution for further improving the global makespan.

A comprehensive **numerical investigation** is conducted in order to investigate the performance of the proposed group setup approaches in detail. Additional tests are carried out using industrial data from two different PCB manufacturers. The results reveal that the **simple matching** approach using **single linkage** method performs best among the group setup approaches based on conventional clustering techniques. The novel group setup approach using the **inclusion measure** performs even better than the best conventional approach. Hence, the implicit consideration of the available magazine capacity by favoring PCB pairs with a higher degree of inclusion is more appropriate than observing only the conjoint absence or presence of component types. This applies especially if the total number of slots required greatly exceeds the magazine capacity.

The proposed **group setup approaches** have realized high makespan reductions compared against a **unique setup strategy** for the observed medium-volume medium-variety production environment. As expected, best savings are achieved if batch sizes are small and setup times are large enough. PCBs showing a high level of similarity allow the efficient usage of available magazine slots, and thus reduce the number of required setup groups.

Detailed investigations illustrate the significance of integrating the **makespan effect** into the clustering procedure. A conventional group setup approach, which merely investigates the magazine capacity level, performs worse than the proposed group setup approach considering the improvements of makespan. Results also clearly demonstrate that a conventional group setup approach can perform even worse than a unique setup approach while the proposed novel group setup strategy is determined to deliver always the best result.

The presented study emphasizes the essentiality of handling the intertwined optimization problems arising in PCB assembly integratively. Hence, solving a single optimization problem to optimality does not assure reaching the best global solution. Development of fast efficient heuristics enables creating an **integrated solution approach** which can solve real-case problems in very short computational times. Optimization models in the literature usually involve rigid assumptions in order to simplify the model formulation, and hence disregard some relevant practical issues. This study proves that good solutions can be achieved if problem-tailored efficient heuristics, which consider the characteristics of the observed production environment, are applied in an integrated solution approach dealing with a wider area of problems.

The future SMT lines will comprise placement machines which are modularly designed for fulfilling the specific requirements of the PCB manufacturer. Modern placement stations consist of multiple gantry and transfer systems in a custom configuration of different types of placement heads. Hence, problems considering multiple machine systems, e.g. line and gantry workload balancing issues, will become more significant. Investigating the group setup strategy for such manufacturing environments is an interesting research topic for the future. The increasing cross-relations between subproblems of PCB assembly will further require development of integrated solution methodologies.

References

Ahmadi, R.H., A hierarchical approach to design, planning, and control problems in electronic circuit card manufacturing, in: Sarin, R.K. (ed.), *Perspectives in Operations Management*, Kluwer Academic Publishers, Dordrecht, 1993, 409-429.

Ahmadi, J., Ahmadi, R., Matsuo, H., Tirupati, D., Component Fixture Positionong/Sequencing for Printed Circuit Board Assembly with Concurrent Operations, *Operations Research*, 43, 1995, 444-457.

Ahmadi, J., Grotzinger, S., Johnson, D., Component allocation and portioning for a dual delivery placement machine, *Operations Research*, 36, 1988, 176-191.

Ahmadi, J., Grotzinger, S., Johnson, D., Emulating concurrency in a circuit card assembly system, *International Journal of Flexible Manufacturing Systems*, 3, 1990, 45-70.

Ahmadi, R., Kouvelis, P., Staging problem of a dual delivery pick-and-place machine in printed circuit card assembly, *Operations Research*, 42, 1994, 81-91.

Ahmadi, R.H., Kouvelis, P., Design of electronic assembly lines: An analytical framework and its application. *European Journal of Operational Research*, 115, 1999, 113-137.

Altinkemer, K., Kazaz, B., Köksalan, M., Moskowitz, H., Optimization of printed circuit board manufacturing: Integrated modeling and algorithms, *European Journal of Operational Research*, 124, 2000, 409-421.

Ammons, J.C., Carlyle, M., Cranmer, L., De Puy, G., Ellis, K., McGinnis, L.F., Tovey, C.A. Xu, H., Component allocation to balance workload in printed circuit card assembly systems, *IIE Transactions*, 29, 1997, 265-275.

Anderberg, M.R., Cluster Analysis for Applications, Academic Press, New York, 1973.

Ashayeri, J., Selen, W., A planning and scheduling model for onsertion in printed circuit board assembly, *European Journal of Operational Research*, 183, 2007, 909-925.

Askin, R.G., Dror, M., Vakharia, A.J., Printed Circuit Board Family Grouping and Component Allocation for a Multimachine, Open-Shop Assembly Cell, *Naval Research Logistics*, 41, 1994, 587-608.

Ayob, M., Kendall, G., A Nozzle Selection Heuristic to Optimise the Hybrid Pick and Place Machine, *Procedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems*, Singapore, 2004.

Ayob, M., Kendall, G., A Survey on Surface Mount Deice Placement Machine Optimisation: Machine Classification, *European Journal of Operational Research*, 186, 2008, 893-914.

Ayob, M., Kendall, G., A triple objective function with a Chebychev dynamic pick-and-place point specification approach to optimise the surface moutn placement machine, *European Journal of Operational Research*, 164, 2005a, 609-626.

Ayob, M., Kendall, G., Component Pick-and-Place Scheduling using a Weighted Nozzle Rank Approach, *Computer Science Technical Report*, No. NOTTCS-TR-2005-7, School of Computer Science and Information Technology, University of Nottingham, UK, 2005b.

Bachmann, Buchholz, Dürr, Eggerer, Geiger, Herzberg, Krause, Liegel, Messmer, Schiebel, Wormuth, Die Welt der Surface Mount Technology aus der Praxis für die Praxis (in German), Technologiehandbuch, Ausgabe 10/99, Siemens AG, 1999.

Backhaus, K., Erichson, B., Plinke, W., Weiber, R., Multivariate Analysemethoden (in German), Springer-Verlag, Berlin et al., 6. Auflage, 1990.

Ball, M.O., Magazine, M.J., Sequencing of insertions in a printed circuit board assembly, *Operations Research*, 36, 1988, 192-201.

Ball, M.O., Magnanti, T.L., Monma, C.L., Nemhauser, G.L. (eds.), Handbooks in Operations Research and Management Science, Network Models, Volume 7, Elsevier, Amsterdam et al., 1995.

Bard, J.F., A Heuristic for Minimizing the Number of Tool Switches on a Flexible Machine, *IIE Transactions*, 20, 1988, 382-391.

Bard, J.F., Clayton, R.W., Feo, T.A., A Machine Setup and Component Placement in Printed Circuit Board Assembly, *International Journal of Flexible Manufacturing Systems*, 6, 1994, 5-31.

Bhaskar, G., Narendran, T.T., Grouping PCBs for setup reduction: a maximum spanning tree approach, *International Journal of Production Research*, 34, 1996, 621-632.

Brandeau, M.L., Billington, C.A., Design of Manufacturing Cells: Operation Assignment in Printed Circuit Board Manufacturing, *Journal of Intelligent Manufacturing*, 2, 1991, 95-106.

Brindley, K., Newnes Electronics Assembly Handbook, Heinemann Newnes, Oxford et al., 1990.

Carmon, T.F., Maimon, O.Z., Dar-El, E.M., Group set-up for printed circuit board assembly, *International Journal of Production Research*, 27, 1989, 1795-1810.

Choudhury, N.D., Wilhelm, W.E., Rao, B., Gott, J., Khotekar, N., Process planning for circuit card assembly on a series of dual head placement machines, *European Journal of Operational Research*, 182, 2007, 626-639.

Clark, G. Wright, J.W., Scheduling vehicles from a central delivery depot to a number of delivery points, *Operations Research Quarterly*, 12, 1964, 568-581.

Coble, S.B., Bohn, R.E., Setup time reduction for electronics assembly: Combining simple (SMED) and sophisticated methods, *Technical Report 1997-03*, University of California, San Diego, 1997.

Coombs, C.F. (ed.), Printed Circuits Handbook, McGraw-Hill, Inc., 3rd Edition, New York et al., 1988.

Crama, Y., Combinatorial optimization models for production scheduling in automated manufacturing systems, *European Journal of Operational Research*, 99, 1997, 136-153.

Crama, Y., Flippo, O.E., van de Klundert, J., Spieksma, F.C.R., The assembly of printed circuit boards: A case with multiple machines and multiple board types, *European Journal of Operational Research*, 98, 1997, 457-472.

Crama, Y., Flippo, O.E., van de Klundert, J., Spieksma, F.C.R., The Component Retrieval Problem in Printed Circuit Board Assembly, *The International Journal of Flexible Manufacturing Systems*, 8, 1996a, 287-312.

Crama, Y., Kolen, A.W.J., Oerlemans, A.G., Spieksma, F.C.R., Minimizing the Number of Tool Switches on a Flexible Machine, *International Journal of Flexible Manufacturing Systems*, 6, 1994, 33-54.

Crama, Y., Kolen, A.W.J., Oerlemans, A.G., Spieksma, F.C.R., Throughput rate optimization in the automated assembly of printed circuit boards, *Annals of Operations Research*, 26, 1990, 455-480.

Crama, Y., Oerlemans, A.G., Spieksma, F.C.R., Production Planning in Automated Manufacturing, Springer-Verlag, 2^{nd} Ed., Berlin et al., 1996b.

Crama, Y., van de Klundert, J., Spieksma, F.C.R., Production planning problems in printed circuit board assembly, *Discrete Applied Mathematics*, 123, 2002, 339-361.

Daskin, M.S., Maimon, O., Shtub, A., Braha, D., Grouping components in printed circuit board assembly with limited component staging capacity and single card setup: problem characteristics and solution procedures, *International Journal of Production Research*, 35, 1997, 1617-1638.

De Souza, R., Lijun, W., Intelligent optimization of component onsertion in multi-head concurrent operation PCBA machines, *Journal of Intelligent Manufacturing*, 6, 1995, 235-243.

Deo, S., Javadpour, R., Knapp, G.M., Multiple setup PCB assembly planning using genetic algorithms, *Computers & Industrial Engineering*, 42, 2002, 1-16.

Dessouky, M.M., Adiga, S., Park, K., Design and scheduling of flexible assembly lines for printed circuit boards, *International Journal of Production Research*, 33, 1995, 757-775.

Drezner, Z., Nof, S.Y., On Optimizing Bin Picking and Insertion Plans for Assembly Robots, *IIE Transactions*, 16, 1984, 262-270.

Duman, E., A note on 'balancing printed circuit board assembly line systems', *International Journal of Production Research*, 43, 2005, 3955-3957.

Duman, E., Modelling the operations of a component placement machine with a rotational turret and stationary component magazine, *Journal of the Operational Research Society*, 58, 2007, 317-325.

Duman, E., Or, I., Precedence constrained TSP arising in printed circuit board assembly, *International Journal of Production Research*, 42, 2004, 67-78.

Duman, E., Or, I., The quadratic assignment problem in the context of the printed circuit board assembly, *Computers & Operations Research*, 34, 2007, 163-179.

Dürr, N., SIPLACE Bestückautomaten im Vergleich zu herkömmlichen Chipshootern (in German), *SIPLACE Fachinformation*, 2, Siemens AG, München, 1997a.

Dürr, N., Optimale Rüststrategie zur Erhöhung der Verfügbarkeit von SMD Bestückautomaten (in German), *SIPLACE Fachinformation*, 3, Siemens AG, München, 1997b.

Egbelu, P.J., Wu, C.-T., Pilgaonkar, R., Robotic assembly of printing circuit boards with component feeder location consideration, *Production Planning & Control*, 7, 1996, 162-175.

Ellis, K.P., Bhoja, S., Optimization of the assignment of circuit cards to assembly lines in electronics assembly, *International Journal of Production Research*, 40, 2002, 2609-2631.

Everitt, B., Cluster Analysis, John Wiley & Sons, Inc., New York, 2nd Edition, 1980.

Garetti, M., Pozzetti, A., Tavecchio, R., Production scheduling in SMT electronic boards assembly, *Production Planning & Control*, 7, 1996, 197-204.

Gasch, M., Hat die Leiterplatte eine Zukunft? Und wenn ja – wo und wie? Marktanalysen und Technologietrends, Teil 1: Markatanalysen (in German), *Fachverband Elektronik-Design*, FED-50-06, Berlin, 2002, 1-8.

Gingsberg, G., Surface Mount and Related Technologies, Marcel Dekker, New York, 1989.

Gordon, A.D., Classification, Chapman & Hall, London, 2nd Edition, 1999.

Gronalt, M., Grunow, M., Günther, H.O., Zeller, R., A heuristic for component switching on SMT placement machines, *International Journal of Production Economics*, 53, 1997, 181-190.

Grotzinger, S., Feeder Assignment Models for Concurrent Placement Machines, *IIE Transactions*, 24, 1992, 31-41.

Grunow, M., Günther, H.O., Föhrenbach, A., Simulation-based performance analysis and optimization of electronics assembly equipment, *International Journal of Production Research*, 38, 2000, 4247-4259.

Grunow, M., Günther, H.-O., Schleusener, M., Component allocation for printed circuit board assembly using modular placement machines, *International Journal of Production Research*, 41, 2003, 1311-1331.

Grunow, M., Günther, H.-O., Schleusener, M., Yilmaz, I.O., Operations planning for collect-and-place machines in PCB assembly, *Computers & Industrial Engineering*, 47, 2004, 409-429.

Grunow, M., Optimierung von Bestückungsprozessen in der Elektronikmontage (in German), Deutscher-Universitätsverlag, Wiesbaden, 2000.

Günther, H.O., Gronalt, M., Zeller, R., Job sequencing and component set-up on a surface mount placement machine, *Production Planning & Control*, 9, 1998, 201-211.

Günther, H.-O., Grunow, M., Schorling, C., Workload planning in small lot printed circuit board assembly, *OR Spectrum*, 19, 1997, 147-157.

Günther, H.-O., Tempelmeier, H., Produktion und Logistik (in German), 7. Auflage, Springer Verlag, Berlin et al., 2007.

Hair, J.F., Anderson, R.E., Tatham, R.L., Black, W.C., Multivariate Data Analysis, Prentice-Hall, Inc., New Jersey, 5th Edition, 1998.

Hanke, H.-J. (ed.), Baugruppen Technologie der Elektronik, Leiterplatten (in German), Verlag Technik GmbH, Berlin, 1994.

Hashiba, S. Chang, T.C., PCB Assembly Setup Reduction Using Group Technology, *Computers & Industrial Engineering*, 21, 1991, 453-457.

Hashiba, S., Chang, T.C., Heuristic and simulated annealing approaches to PCB assembly setup reduction, in: Olling, G.J., Kimura, F. (eds.), *Proceedings of the Eight International PROLAMAT Conference on Human Aspects in Computer Integrated Manufacturing*, North-Holland Publishing Co., Amsterdam, 1992, 769-777.

Hillier, M.S., Brandeau, M.L., Optimal Component Assignment and Board Grouping in Printed Circuit Board Manufacturing, *Operations Research*, 46, 1998, 675-689.

Hillier, M.S., Brandeau, M.L., Cost minimization and workload balancing in printed circuit board assembly, *IIE Transactions*, 33, 2001, 547-557.

Hirvikorpi, M., Knuutila, T., Johnsson, M., Nevalainen, O.S., A general approach to grouping PCB assembly jobs, *International Journal of Computer Integrated Manufacturing*, 18, 2005, 710-720.

Hirvikorpi, M., Salonen, K., Knuutila, T., Nevalainen, O.S., The general two-level storage management problem: A reconsideration of the KTNS-rule, *European Journal of Operational Research*, 171, 2006, 189-207.

Ho, W., Ji, P., Component Scheduling for Chip shooter Machines: A Hybrid Genetic Algorithm Approach, *Computers & Operations Research*, 30, 2003, 2175-2189.

Ho, W., Ji, P., PCB assembly line assignment: a genetic algorithm approach. *Journal of Manufacturing Technology Management*, 2005a, 16, 682-692.

Ho, W., Ji, P., A hybrid genetic algorithm for sequencing PCB component placement, *International Journal of Knowledge-Based and Intelligent Engineering Systems*, 9, 2005b, 129-136.

Ho, W., Ji, P., A genetic algorithm approach to optimizing component placement and retrieval sequence for chip shooter machines, *International Journal of Advanced Manufacturing Technology*, 28, 2006, 556-560.

Hogg, R.V., Tanis, E.A., Probability and Statistical Inference, Macmillan Publishing Company, New York, 3rd Edition, 1988.

Hong, J., Lee, W., Lee, S., Lee, B. Lee, Y., An Efficient Production Planning Algorithm for Multi-Head Surface Mounting Machines Using the Biological Immune Algorithm, *International Journal of Fuzzy Systems*, 2, 2000, 45-53.

Horak, T., Francis, R.L. Utilization of machine characteristics in PCB board assembly, *Working Paper*, Rutgers University, Management Science and Computer Information Systems Department, 1995.

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

Jain, S., Johnson, M.E., Safai, F., Implementing Setup Optimization on the Shop Floor, *Operations Research*, 43, 1996, 843-851.

Ji, P., Sze, M.T., Lee, W.B., A genetic algorithm of determining cycle time for printed circuit board assembly lines, *European Journal of Operational Research*, 128, 2001, 175-184.

Johnson, M., Smed, J., Observations on PCB Assembly Optimization, in: *The APEX Files: Proceedings of Technical Conference*, San Diego, 2001, SM6-3 1-6.

Johtela, T., Smed, J., Johnsson, M. Nevalainen, O., Fuzzy approach for modeling multiple criteria in the job grouping problem, *TUCS Technical Report*, 227, Turku Centre for Computer Science, Turku, Finland, 2000.

Kaczmarczyk, W., Sawik, T., Schaller, A., Tirpak, T.M., Optimal versus heuristic scheduling of surface mount technology lines, *International Journal of Production Research*, 42, 2004, 2083-2110.

Kazaz, B., Altinkemer, K., Optimization of multi-feeder (depot) printed circuit board manufacturing with error guarantees, *European Journal of Operational Research*, 150, 2003, 370-394.

Kear, F.W., Printed Circuit Assembly Manufacturing, Marcel Dekker, Inc., New York, 1987.

Khoo, L., Loh, K., A genetic algorithms enhanced planning system for surface mount PCB assembly, *International Journal of Advanced Manufacturing Technology*, 16, 2000, 289-296.

Khoo, L.P., Alisantoso, D., Line balancing of PCB assembly line using immune algorithms, *Engineering with Computers*, 19, 2003, 92-100.

Khoo, L.P., Ng, T.K., A Genetic Algorithm-based Planning System for PCB Component Placement, *International Journal of Production Economics*, 54, 1998, 321-332.

Khoo, L.P., Ong, N.S., PCB Assembly Planning Using Genetic Algorithms, *International Journal of Advanced Manufacturing Technology*, 14, 1998, 363-368.

Kim, Y.-D., Lim, H.-G., Park, M.-W., Search heuristics for a flowshop scheduling problem in a printed circuit board assembly process, *European Journal of Operational Research*, 91, 1996, 124-143.

Kimman, C., 01005-Chips einfach bestücken (in German), *Productronic*, 10, 2005, 22-23.

Knuutila, T., Hirvikorpi, M., Johnsson, M., Nevalainen, O., Grouping PCB Assembly Jobs with Feeders of Several Types, *The International Journal of Flexible Manufacturing Systems*, 16, 2004, 151-167.

Knuutila, T., Puranen, M., Johnsson, M., Nevalainen, O., Three perspectives for solving the job grouping problem, *International Journal of Production Research*, 39, 2001, 4261-4280.

Knuutila, T., Pyöttiälä, S., Nevalainen, O.S., Minimizing the number of pickups on a multi-head placement machine, *Journal of the Operational Research Society*, 58, 2007, 115-121.

Kodek, D.M., Krisper, M., Optimal algorithm for minimizing production cycle time of a printed circuit board assembly line, *International Journal of Production Research*, 42, 2004, 5031-5048.

Krups, R., SMT-Handbuch: vom CAD-Entwurf zur Platinfertigung (in German), Vogel Verlag, Würzburg, 1991.

Kruskal, J.B., On the shortest spanning subtree of a graph and the traveling salesman problem, *Proceedings of the American Mathematical Society*, 7, 1956, 48-50.

Kulak, O., Yilmaz, I.O., Günther, H.-O., PCB assembly scheduling for collect-and-place machines using genetic algorithms, *International Journal of Production Research*, 45, 2007a, 3949-3969.

Kulak, O., Yilmaz, I.O., Günther, H.-O., A GA-based solution approach for balancing printed circuit board assembly lines, *OR Spectrum*, 30, 2007b, 469-491.

Kumar, R., Li, H., Integer programming approach to printed circuit board assembly time optimization, *IEEE Transactions on Components, Packaging, and Manufacturing Technology – Part B*, 18, 1995, 720-727.

Kusiak, A., Cho, M., Similarity coefficient algorithms for solving the group technology problem, *International Journal of Production Research*, 30, 1992, 2633-2646.

Laakso, T., Johnsson, M., Johtela, T., Smed, J., Nevalainen, O., Estimating the production times in PCB assembly, *Journal of Electronics Manufacturing*, 2002, 11, 161-170.

Lambert, S., Abdulnour, G., Drolet, J., Cyr, B., Flexibility analysis of a surface mount technology electronic assembly plant: An integrated model using simulation, *International Journal of Flexible Manufacturing Systems*, 17, 2006, 151-167.

Lapierre, S.D., Debargis, L., Soumis, F., Balancing printed circuit board assembly line systems, *International Journal of Production Research*, 38, 2000, 3899-3911.

Lee, W., Lee, S., Lee, B., Lee, Y., An efficient planning algorithm for multi-head surface mounting machines using a genetic algorithm, *Journal of Universal Computer Science*, 5, 2000, 833-854.

Leipälä, T., Navalainen, O., Optimization of the movements of a component placement machine, *European Journal of Operational Research*, 38, 1989, 167-177.

Leon, V.J., Peters, B.A., Replanning and Analysis of Partial Setup Strategies in Printed Circuit Board Assembly Systems, *The International Journal of Flexible Manufacturing Systems*, 8, 1996, 389-412.

Leon, V., Peters, B., A comparison of setup strategies for printed circuit board assembly, *Computers and Industrial Engineering*, 34, 1998, 219-234.

Leu, M.C., Wong, H., Ji, Z., Planning of component placement/insertion sequence and feeder setup in PCB assembly using genetic algorithm, *Journal of Electronic Packaging*, 1993, 115, 424-432.

Lin, F.-R., Shaw, M.J., Locascio, A., Scheduling Printed Circuit Board Production Systems Using the Two-Level Scheduling Approach, *Journal of Manufacturing Systems*, 16, 1997, 129-149.

Lin, H.C., Egbelu, P.J., Wu, C.-T., A two-robot printed circuit board assembly system, *International Journal of Computer Integrated Manufacturing*, 8, 1995, 21-31.

Lin, S., Computer Solutions of the Traveling Salesman Problem, *Bell System Technical Journal*, 44, 1965, 2245-2269.

Lin, W.-L., Tardif, V., Component Partitioning Under Demand and Capacity Uncertainity in Printed Circuit Board Assembly, *International Journal of Flexible Manufacturing Systems*, 11, 1999, 159-176.

Lofgren, C.B., McGinnis, L.F., Dynamic scheduling for flexible printed circuit card assembly, *Proceedings of the 1986 IEEE Conference on Systems, Man, and Cybernetics*, USA, 1986, 1294-1297.

Lofgren, C.B., McGinnis, L.F., Tovey, C.A., Routing Printed Circuit Cards Through an Assembly Cell, *Operations Research*, 39, 1991, 992-1004.

Loh, T.S., Bukkapatnam, S.T.S., Medeiros, D., Kwon, H., A Genetic Algorithm for Sequential Part Assignment for PCB Assembly, *Computers & Industrial Engineering*, 40, 2001, 293-307.

Luzzatto, D., Perona, M., Cell formation in PCB assembly based on production quantitative data, *European Journal of Operational Research*, 69, 1993, 312-329.

Magyar, G., Johnsson, M., Nevalainen, O., On Solving Single Machine Optimization Problems in Electronics Assembly, *Journal of Electronics Manufacturing*, 9, 1999, 249-267.

Maimon, O., Shtub, A., Grouping methods for printed circuit board assembly, *International Journal of Production Research*, 29, 1991, 1379-1390.

Maimon, O.Z., Dar-El, E.M., Carmon, T.F., Set-up saving schemes for printed circuit boards assembly, *European Journal of Operational Research*, 70, 1993, 177-190.

McGinnis, L.F., Ammons, J.C., Carlyle, M., Cranmer, L., DePuy, G.W., Ellis, K.P., Tovey, C.A., Xu, H., Automated process planning for printed circuit card assembly, *IIE Transactions*, 24, 1992, 18-30.

Mettela, E.G., Egbelu, P.J., Alternative approaches to sequencing robot moves for PCB assembly, *International Journal of Computer Integrated Manufacturing*, 2, 1989, 243-256.

Mosier, C.T., Yelle, J., Walker, G., Survey of Similarity Coefficient Based Methods Applied to the Group Technology Configuration Problem, *Omega: International Journal of Management Sciences*, 25, 1997, 65-79.

Moyer, L.K., Gupta, S.M., SMT feeder slot assignment for predetermined component placement paths, *Journal of Electronics Manufacturing*, 6, 1996, 173-192.

Nakahara, H., Consumer Electronics Reign Supreme, *Printed Circuit Design & Manufacture*, 3, 2007, 32-36.

Neammanee, P., Randhawa, S.U., Integrated Methodology for Board Assignment and Component Allocation in Printed Circuit Board Assembly, *International Journal of Production Research*, 41, 2003, 919-937.

Nelson, K.M., Wille, L.T., Comperative study of heuristics for optimal printed circuit board assembly, *Southcon/95, Conference Record*, USA, 1995, 322-327.

NN (2001a), Reicher durch Reinigen? (in German), *Productronic*, 10, 2001, 54-59.

NN (2001b), Kosteneinsparungen durch Schüttgutzuführungen (in German), *Productronic*, 10, 2001, 16-17.

Or, I. Demirkol, E., Optimization issues in automated production of printed circuit boards: Operations sequencing and feeder configuration problems, *Transactions on Operational Research*, 8, 1996, 9-23.

Peters, B.A., Subramanian, G.S., Analysis of partial setup strategies for solving the operational planning problem in parallel machine electronics assembly systems, *International Journal of Production Research*, 34, 1996, 999-1021.

Plonski, J.P., The Economics of Electronic Assembly – Global Trends, Facts and Future, in: Wolfgang Scheel (ed.), Elekronische Baugruppen, Aufbau- und Fertigungstechnik, Die Trends von heute – die Chancen von morgen, *GMM-Fachbericht*, 44, VDE Verlag GmbH, Berlin, 2004, 17-26.

Prasad, R.P., Surface Mount Technology: Principles and Practice, Van Nostrand Reinhold, New York, 1989.

Raduly-Baka, C., Knuutila, Selecting the nozzle assortment for a Gantry-type placement machine, *OR Spectrum*, 30, 2007, 493-513.

Rajkumar, K., Narendran, T.T., A bi-criteria model for loading on PCB assembly machines, *Production Planning & Control*, 8, 1997, 743-752.

Rajkumar, K., Narendran, T.T., A heuristic for sequencing PCB assembly to minimize set-up times, *Production Planning & Control*, 9, 1998, 465-476.

Raz, T., Yaung, T., Heuristic clustering based on a measure of inclusion, *International Journal of Industrial Engineering*, 1, 1994, 57-65.

Reithinger, M., SMT-Montage in der Flachbaugruppenfertigung – heute und morgen, Tagungsbericht Leiterplatte, Bestückung, Baugruppenfertigung – Hybride, Bd. 2, *VDI-Berichte 1133*, VDI Verlag, 1994, 1-20.

Rogers, P., Warrington, R., Production planning for surface mount technology lines, *International Journal of Production Research*, 42, 2004, 2693-2718.

Rosetti, M.D., Stanford, K.J.A., Group sequencing a PCB assembly system via an expected sequence dependent setup heuristic, *Computers & Industrial Engineering*, 45, 2003, 231-254.

Rowland, R., Belangia, P., Applied Surface Mount Assembly, Van Nostrand Reinhold, New York, 1993.

Rubinovitz, J., Volovich, D., Genetic Algorithm for the Robotic Assembly Plan Problem, *SME Technical Paper MS940200*, 1994.

Sadiq, M, Landers, T.L., Don Taylor, G., A heuristic algorithm for minimizing total production time for a sequence of jobs on a surface mount placement machine, *International Journal of Production Research*, 31, 1993, 1327-1341.

Salonen, K., Smed, J., Johnsson, M., Nevalainen, O., Grouping and sequencing PCB assembly jobs with minimum feeder setups, *Robotics and Computer-Integrated Manufacturing*, 22, 2006, 297-305.

Sarker, B.R., Islam, K.M.S., Relative performances of similarity and dissimilarity measures, *Computers & Industrial Engineering*, 37, 1999, 769-807.

Sautter, R., Leiterplatten mit oberflächenmontierten Bauelementen (in German), Vogel-Buchverlag, Würzburg, 1988.

Sawik, T., Balancing and scheduling of surface mount technology lines, *International Journal of Production Research*, 40, 2002, 1973-1991.

Scheel, W. (ed.), Baugruppentechnologie der Elektronik, Montage (in German), Verlag Technik GmbH, Berlin, 2. Auflage, 1999.

Schiebel, G., Wichtige Kriterien bei SMD-Bestückautomaten in Richtung Verarbeitung externer Gehäuseformen (in German), Tagungsbericht Leiterplatte, Bestückung, Baugruppenfertigung – Hybride, Bd. 2, *VDI-Berichte 1133*, VDI Verlag, 1994, 1-20.

Shafer, S.M., Rogers, D.F., Similarity and distance measures for cellular manufacturing. Part I. A survey, *International Journal of Production Research*, 31, 1993a, 1133-1142.

Shafer, S.M., Rogers, D.F., Similarity and distance measures for cellular manufacturing. Part II. An extension and comparison, *International Journal of Production Research*, 31, 1993a, 1315-1326.

Shih, W., Srihari, K., Adriance, J., Expert system based placement sequence identification for surface mount PCB assembly, *International Journal of Advanced Manufacturing Technology*, 11, 1996, 413-424.

Shtub, A., Maimon, O., Role of similarity measures in PCB grouping procedures, *International Journal of Production Research*, 30, 1992, 973-983.

Siemens AG (ed.), Halbleiter: technische Erläuterungen und Kenndaten für Studierende (in German), Siemens Aktiengesellschaft, Berlin et al., 1990.

Siemens, Siplace, http://www.siplace.com, 2007.

Smed, J., Johnson, M., Nevalainen, O., A Hierarchical Classification Scheme for Electronics Assembly Problems, in: Yliniemi, L., Juuso, E. (eds.), *Proceedings of TOOLMET2000 Symposium – Tool Environments and Developments Methods for Intelligent Systems*, Oulu, Finland, 2000, 116-119.

Smed, J., Johnsson, M., Puranen, M., Leipälä, T. Nevalainen, O., Job Grouping in Surface Mounted Component Printing, *Robotics and Computer-Integrated Manufacturing*, 15, 1999, 39-49.

Smed, J., Salonen, K., Johnsson, M., Johtela, T., Nevalainen, O., A Comparison of Group and Minimum Setup Strategies in PCB Assembly, *TUCS Technical Report*, 327, Turku Centre for Computer Science, Turku, Finland, 2000.

Smed, J., Salonen, K., Johnsson, M., Johtela, T., Nevalainen, O.S., Grouping PCBs with minimum feeder changes, *The International Journal of Flexible Manufacturing Systems*, 15, 2003, 19-35.

Sohn, J., Park, S., Efficient operation of a surface mounting machine with a multihead turret, *International Journal of Production Research*, 34, 1996, 1131-1143.

Strauss, R., Surface Mount Technology, Butterworth-Heinemann Ltd., Oxford et al., 1994.

Su, C.T., Fu, H.-P., A simulated annealing heuristic for robotics assembly using the dynamic pick-and-place model, *Production Planning & Control*, 9, 1998, 795-802.

Su, Y.-C., Wang, C., Egbelu, P.J., Cannon, D.J., A dynamic point specification approach to sequencing robot moves for PCB assembly, *International Journal of Computer Integrated Manufacturing*, 8, 1995, 448-456.

Su, Y.-Y., Srihari, K., Placement sequence identification using neural networks in surface mount PCB assembly, *International Journal of Advanced Manufacturing Technology*, 11, 1996, 285-299.

Sule, D.R., A heuristic procedure for component scheduling in printed circuit pack sequencers, *International Journal of Production Research*, 30, 1992, 1191-1208.

Sun, D.-S, Lee, T.-E., Kim, K.-H., Component allocation and feeder arrangement for a dual-gantry multi-head surface mounting placement tool, *International Journal of Production Economics*, 95, 2005, 245-264.

Tang, C.S., Denardo, E.V., Models Arising From a Flexible Manufacturing Machine, Part I: Minimization of the Number of Tool Switches, *Operations Research*, 36, 1988, 767-777.

Tirpak, T.M., Mohapatra, P.K., Nelson, P.C., Rajbhandari, R.R., A Generic Classification and Object-Oriented Simulation Toolkit for SMT Assembly Equipment, *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 32, 2002, 104-122.

Van Hop, N., Board sequencing and component loading problem for a single machine in PCB assembly planning, *International Journal of Production Research*, 41, 2003, 4299-4315.

Van Hop, N., Nagarur, N.N., The scheduling problem of PCBs for multiple non-identical parallel machines, *European Journal of Operational Research*, 158, 2004, 577-594.

Van Hop, N., Tabucanon, M.T., Fuzzy multi-attribute decision-making for grouping of electronic components in process planning, *Production Planning & Control*, 11, 2000, 677-688.

Van Laarhoven, P.J.M., Zijm, W.H.M., Production Preparation and Numerical Control in PCB Assembly, *International Journal of Flexible Manufacturing Systems*, 5, 1993, 187-207.

Walz, D., Bleifreies Löten – Eine Herausforderung für die Endoberfläche von Leiterplatten (in German), in: Scheel, W. (ed.), Elekronische Baugruppen, Aufbau- und Fertigungstechnik, Die Trends von heute – die Chancen von morgen, *GMM-Fachbericht*, 44, VDE Verlag GmbH, Berlin, 2004, 155.

Wang, C., Ho, L.-S., Cannon, D.J., Heuristics for assembly sequencing and relative magazine assignment for robotic assembly, *Computers & Industrial Engineering*, 34, 1998, 423-431.

Wang, W., Nelson, P.C., Tirpak, T.M., Optimization of High-Speed Multi-station SMT Placement Machines using Evolutionary Algorithms, *IEEE Transactions on Electronics Packaging Manufacturing*, 22, 1999, 137-146.

Watkins, R.E., Cochran, J.K., A Line Balancing Heuristic Case Study For Existing Automated Surface Mount Assembly Line Setups, *Computers & Industrial Engineering*, 29, 1995, 681-685.

Wilhelm, W.E., Tarmy, P.K., Circuit card assembly on tandem turret-type placement machines, *IIE Transactions*, 35, 2003, 627-645.

Williams, S.K., Magazine, M.J., Heuristic approaches for batching jobs in printed circuit board assembly, *Computers & Operations Research*, 34, 2007, 1943-1962.

Wong, H., Leu, M.C., Adaptive Genetic Algorithm for Optimal PCB Assembly Planning, *CIRP Annals*, 42, 1993, 17-20.

Yeo, S.H., Low, C.W., Yong, K.H., A rule-based frame system for concurrent assembly machines, *International Journal of Advanced Manufacturing Technology*, 12, 1996, 370-376.

Yilmaz, I.O., Grunow, M., Günther, H.-O., Yapan, C., Development of group setup strategies for makespan minimisation in PCB assembly, *International Journal of Production Research*, 2007, 45, 871-897.

Yin, Y., Yasuda, K., Similarity coefficient methods applied to the cell formation problem: a comparative investigation, *Computers & Industrial Engineering*, 48, 2005, 471-489.

Yin, Y., Yasuda, K., Similarity coefficient methods applied to the cell formation problem: a taxonomy and review, *International Journal of Production Economics*, 101, 2006, 329-352.

Yu, S., Kim, D., Park, S., Integer programming approach to the printed circuit board grouping problem, *International Journal of Production Research*, 43, 2005,1667-1684.