ABSTRACT

OPTIMAL AND INTELLIGENT DECISION MAKING IN SUSTAINABLE DEVELOPMENT OF ELECTRONIC PRODUCTS

by
Meimei Gao

Increasing global population and consumption are causing declining natural and social systems. Multi-lifecycle engineering and sustainable development address these issues by integrating strategies for economic successes, environmental quality, and social equity. Based on multi-lifecycle engineering and sustainable development concepts, this doctoral dissertation aims to provide decision making approaches to growing a strong industrial economy while maintaining a clean, healthy environment. The research develops a methodology to complete both the disassembly leveling and bin assignment decisions in demanufacturing through balancing the disassembly efforts, value returns, and environmental impacts. The proposed method is successfully implemented into a demanufacturing module of a Multi-LifeCycle Assessment and Analysis tool. The methodology is illustrated by a computer product example.

Since products during the use stage may experience very different conditions, their external and internal status can vary significantly. These products, when coming to a demanufacturing facility, are often associated with incomplete/imprecise information, which complicates demanufacturing process decision making. In order to deal with uncertain information, this research proposes Fuzzy Reasoning Petri nets to model and reason knowledge-based systems and successfully applies them to demanufacturing process decision making to obtain the maximal End-of-Life (EOL) value from discarded products.

Besides the EOL management of products by means of product/material recovery to decrease environmental impacts, the concepts of design for environment and sustainable development are investigated. Based on Sustainability Target Method, a sensitivity analysis decision-making method is proposed. It provides a company with suggestions to improve its product's sustainability in the most cost-effective manner.

OPTIMAL AND INTELLIGENT DECISION MAKING IN SUSTAINABLE DEVELOPMENT OF ELECTRONIC PRODUCTS

by
Meimei Gao

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Computer Engineering

Department of Electrical and Computer Engineering

August 2003

APPROVAL PAGE

OPTIMAL AND INTELLIGENT DECISION MAKING IN SUSTAINABLE DEVELOPMENT OF ELECTRONIC PRODUCTS

Meimei Gao

Dr. Mengchu Zhou, Dissertation Advisor — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Reggie J. Caudill, Dissertation Co-Advisor — Date
Professor of Industrial and Manufacturing Engineering, NJIT

Dr. Nirwan Ansari, Committee Member — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Edwin Hou, Committee Member — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Ying Tang, Committee Member — Date
Assistant Professor of Electrical and Computer Engineering, Rowan University

BIOGRAPHICAL SKETCH

Author: Meimei Gao

Degree: Doctor of Philosophy

Date: August 2003

Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Engineering
  New Jersey Institute of Technology, Newark, NJ, 2003

- Master of Science in Automatic Control
  Northwestern Polytechnic University, Xi'an, P. R. China, 1997

- Bachelor of Science in Automatic Control
  Northwestern Polytechnic University, Xi'an, P. R. China, 1994

Major: Computer Engineering

Presentations and Publications:

M. Gao, M. C. Zhou, X. Huang, and Z. Wu,
"Fuzzy Reasoning Petri Nets," IEEE Transactions on Systems, Man, and Cybernetic, Accepted for publication, July 2002.

M. Gao, M. C. Zhou, and R. J. Caudill,
"Integration of Disassembly Leveling and Bin Assignment for Demanufacturing Automation," IEEE Transactions on Robotics and Automation, vol. 18, No. 6, pp. 867-874, December 2002.

M. Gao, and Z. Wu,
"The temporal reasoning method of fuzzy time Petri nets and its application in process supervision," Control and Decision, vol. 16, No. 3, pp. 371-374, 2001.

M. Gao, and Z. Wu,
"Modeling FMS Based on UML and OPNs," Chinese Journal of Mechanical Engineering, vol. 13, No. 2, pp. 90-95, 2000.

M. Gao, and Z. Wu,
"An object-oriented simulation environment for FMS," Journal of System Simulation, vol. 122, No. 6, pp. 634-637, 2000.

M. Gao, and Z. Wu,
"Temporal reasoning method based on linear logic," Journal of Computer Research and Development, vol. 37, No. 12, pp. 1452-1456, 2000.

M. Gao, and Z. Wu,
"Fuzzy reasoning Petri nets and its application in diagnosis system," Acta Automatica Sinica, vol. 26, No. 5, pp. 677-680, 2000.

M. Gao, and Z. Wu,
"Fuzzy time Petri nets and its temporal reasoning method," Computer Science, vol. 27, No. 7, pp. 11-14, 2000.

M. Gao, and Z. Wu,
"Deadlock detection of FMS based on OO model," Mechanical Science and Technology, vol. 19, No. 5, pp. 780-782, 2000.

M. Gao, Z. Wu, X. Pang, and Y. Yi,
"A distributed simulation system for FMS," Journal of Shanghai Jiaotong University, vol. 34, No. 5, pp. 604-607, 2000.

M. Gao, and Z. Wu,
"An analysis method of Petri nets based on linear logic," Computer Science, No. 4, pp. 18-21, 2000.

M. Gao, and Z. Wu,
"A Fuzzy Reasoning Petri Net Model and Its Reasoning Algorithm," Journal of Shanghai Jiaotong University, vol. E-4, No. 2, pp. 5-9, 1999.

M. Gao, and M. C. Zhou, "Fuzzy Intrusion Detection Based on Fuzzy Reasoning Petri Nets," To appear at Proceeding of the 2003 IEEE International Conference on Systems, Man, and Cybernetics, Washington D. C., October 8-15, 2003.

M. Gao, M. C. Zhou, and F. Wang,
"Improvement of Product Sustainability," To appear at Proceeding of the 2003 IEEE International Conference on Robotics and Automation, Taipei, Taiwan, Sept. 2003.

M. Gao, and M. C. Zhou,
"Product Sustainability Improvement Based on Performance of Product Components," Proceeding of the 2003 IEEE International Symposium on Electronics and the Environment & the IAER Electronics Recycling Summit, Boston, MA, May 19-22, 2003, pp. 224-229.

M. Gao, M. C. Zhou, and R. J. Caudill,
"A Bin Assignment Algorithm for Demanufacturing Automation," Proceeding of the 2002 IEEE International Conference on Robotics and Automation, Washington D. C., May 11-15, 2002, pp. 2351-2356.

M. Gao, M. C. Zhou, and R. J. Caudill,
"Development of Demanufacturing Module for Multi-Lifecycle Assessment and Analysis of Products," Proceeding of the 2001 IEEE International Conference on Systems, Man, and Cybernetics, Tucson, AZ, Oct. 7-10, 2001, pp. 1856-1861.

M. Gao, and M. C. Zhou,
"Fuzzy reasoning Petri nets for demanufacturing process decision," Electronics and the Environment, Proceedings of the 2001 IEEE International Symposium on, 2001, Denver, CO, May 7-9, 2001, pp. 167 –172.

M. Gao, Z. Wu, and M. C. Zhou,
"A Petri net-based formal reasoning algorithm for fuzzy production rule-based systems," Proc. 2000 IEEE International Conference on Systems, Man, and Cybernetics, Nashville, TN, USA, Oct. 10, 2000, pp. 3093 –3097.

M. Gao, X. Huang, and Z. Wu,
"Linear logic as a tool for presentation and temporal reasoning of time Petri nets," Proc. American Control Conference 2000 (ACC00)-IEEE, Chicago, Illinois, June 28-30, 2000, pp. 3177 –3181.

M. Gao, Z. Wu, and X. Huang,
"Fuzzy time Petri nets and its application in supervision of DEDS," The Third World Congress on Intelligent Control and Automation (WCICA'2000), vol. 4, 2000, pp. 2327 –2331.

X. Huang, M. Gao, Y. Wang, and S. Weng,
"Application of Petri nets in fault diagnosis of gas turbine system," Journal of Airspace Power Engineering of China, vol. 15, No. 3, pp: 311-314, 2000.

X. Huang, Y. Wang, S. Weng, and M. Gao,
"An on-line monitoring and communication system development for gas turbine system," Journal of Engineering for Thermal Energy and Power, vol. 15, No. 5, pp. 537-540, 2000.

To my beloved family

ACKNOWLEDGEMENT

I would like to express my sincere appreciation to my advisor, Dr. Mengchu Zhou, and co-advisor, Dr. Reggie J. Caudill, for their guidance and support through this research. Special thanks are given to Dr. Nirwan Ansari, Dr. Edwin Hou, and Dr. Ying Tang for serving as my doctoral committee.

I would like to acknowledge the support of the New Jersey Commission on Science and Technology and such industrial firms as AT&T, IBM, Lucent Technology and Panasonic etc. through the Multi-lifecycle Engineering Research Center (MERC) at NJIT.

I would like to thank all of the fellows in the Discrete Event System Laboratory and the Multi-lifecycle Engineering Research Center for their assistance, collaboration and friendship.

TABLE OF CONTENTS

Chapter Page

1 INTRODUCTION 1

1.1 Motivation 1

1.2 Objectives 5

1.3 Organization 5

2 LITERATURE REVIEW 7

2.1 Life Cycle Assessment (LCA) and Multi-Life Cycle Engineering (MLCE) 7

2.2 Demanufacturing 9

2.2.1 Disassembly Process 10

2.2.2 Bin Assignment 15

2.2.3 Performance assessment of demanufacturing at product design stage 16

2.3 Environmental Performance Metrics 19

2.4 Decision Making Methods 23

2.5 Summary 28

3 INTEGRATION OF DISASSEMBLY LEVELING AND BIN ASSIGNMENT FOR DEMANUFACTURING ANALYSIS 29

3.1 Product Tree Structure 30

3.2 Bin Assignment Algorithm 31

3.2.1 Complete Disassembly 31

3.2.2 Integration of Disassembly Leveling and Bin Assignment 33

TABLE OF CONTENTS
(Continued)

Chapter **Page**

3.3 Algorithm Implementation and Complexity Analysis 39

3.3.1 Algorithm Implementation.. 39

3.3.2 Complexity Analysis of The Algorithm... 42

3.4 MLCA Tool 45

3.4.1 Brief Description of MLCA Tool 45

3.4.2 Demanufacturing Module 47

3.4.2.1 Bin Assignment... 47

3.4.2.2 Process Specification and Definition 48

3.4.2.3 Result Display.. 49

3.5 A Computer Product Example 50

3.6 Summary.. 52

4 FUZZY REASONING PETRI NETS.. 54

4.1 Introduction.. 55

4.2 Fuzzy Production Rules and Their Representation with FRPN..................... 57

4.2.1 Formats of Fuzzy Production Rules... 57

4.2.2 Rule Representation 58

4.2.3 Negation Issue.. 58

4.2.4 Features of FRPN... 61

4.3 Formal Definition of FRPN 64

4.3.1 Definition of FRPN.. 64

TABLE OF CONTENTS
(Continued)

Chapter Page

4.3.2 Execution Rules of FRPN 66

4.4 A Reasoning Algorithm Based on FRPN 70

4.4.1 Reasoning Algorithm 70

4.4.2 Complexity of The Algorithm 74

4.4.3 Modularization of FRPN 77

4.4.4 Procedure to Apply FRPN to Fuzzy Rule-based Reasoning 77

4.5 FRPN for A Gas Turbine Diagnosis System 78

4.5.1 Rule Base 78

4.5.2 FRPN Modeling 82

4.5.3 Algorithm Execution Results 83

4.6 Summary 84

5 FUZZY REASONING PETRI NETS FOR DEMANUFACTURING PROCESS DECISION MAKING 86

5.1 Demanufacturing System Model 87

5.2 Demanufacturing Process Modeling with FRPN 88

5.3 An Electronic Appliance Example 90

5.4 Summary 93

6 SUSTAINABILITY IMPROVEMENT 94

6.1 Sustainability Target Method 95

6.2 Sensitivity Analysis 100

6.3 Sensitivity Analysis of STM Indicators 102

TABLE OF CONTENTS (Continued)

Chapter Page

6.3.1 Recourse Productivity Analysis 102

6.3.2 Eco-Efficiency Analysis 103

6.3.3 Decision Making Based on RPs of Components 103

6.3.4 Decision Making Based on EEs of Components 104

6.3.5 Results for General Cases 105

6.4 An Illustration Example 107

6.4.1 Based on RP and Costs Per Unit Change of RP 107

6.4.2 Based on RP Information of Components 110

6.4.3 Based on EE Information of Components 111

6.4.4 Discussions 112

6.5 Application to A Handy Board 111

6.6 Summary 115

7 CONCLUSIONS AND FUTURE WORK 116

7.1 Conclusions 116

7.2 Future Research 120

APPENDIX DATA OF ZENITH LAPTOP CASE STUDY 122

REFERENCES 127

LIST OF TABLES

Table — Page

3.1 Value Gains in Three Cases 51

6.1 Components of a Product, and Their Production Rates per Unit Environmental Impact (RP), Costs per Unit Change of RP (C), Eco-Efficiency (EE), and Unit Value (V_{PR}) 107

6.2 Main Components of a Handy Board 114

A.1 Product Tree Structure, Material Composition and the Results 122

A.2 Reuse Value (R_i) and Hazard Mitigation Value (H_i) 123

A.3 Unit Material Value in Bin β (C_β), and Recycling Yield of Material j in Bin β ($\Phi_{\beta j}$) 123

A.4 Disassembly Procedure and Data 125

A.5 Disassembly Results for Case 3 126

LIST OF FIGURES

Figure **Page**

2.1 The total life cycle engineering framework 8

2.2 Applying AI concepts with a computer.. 26

3.1 Tree structure of a product.. 30

3.2 The flow diagram of the bin assignment algorithm 40

3.3 A Zenith laptop computer 50

3.4 Demanufacturing screen of MLCA tool 51

4.1 The type-1 fuzzy rule presentation with FRPN.. 59

4.2 The type-2 fuzzy rule presentation with FRPN.. 59

4.3 The type-3 fuzzy rule presentation with FRPN (after being normalized)........... 60

4.4 A simple FRPN example.. 63

4.5 A kind of rule representation with Petri nets 64

4.6 The type-1 fuzzy rule presentation with FRPN (after firing rule R_j) 67

4.7 The type-2 fuzzy rule presentation with FRPN (after firing rule R_j) 68

4.8 The type-3 fuzzy rule presentation with FRPN
(after firing rules R_{j1}, R_{j2}, …, and $R_{j,k-1}$) 68

4.9 A rule-based system composed of several modules.. 77

4.10 The FRPN model of a turbine fault diagram expert system................................ 82

4.11 The execution result of the algorithm for the example 83

5.1 A demanufacturing system.. 87

LIST OF FIGURES
(Continued)

Figure **Page**

5.2 An example for making demanufacturing decisions 88

5.3 The FRPN for demanufacturing process decision 89

5.4 Disassembly Petri net for a handlight 91

6.1 The Handy Board 112

A.1 Mating diagram of Zenith laptop computer 124

CHAPTER 1

INTRODUCTION

1.1 Motivation

With the rapid development of computer and communication technologies, the growth rate of the electronic industry has been much more dramatic than ever expected. Electronic products have the following features:

1) Their volume is very large;
2) They become obsolete very quickly; and
3) Many of them contain both valuable and toxic materials.

Computers and other electronic devices are as common in offices as telephones, and their quantity is growing. Faster, more powerful computers quickly replace their predecessors. A 1999 study by the National Safety Council estimated that nationally, approximately 20.6 million personal computers would become obsolete in the United States in 1998 and nearly 500 million computers would become obsolete between 1997 and 2006. Most old electronic products are either in storage or thrown in landfill rather than being recycled [NSC, 1999]. End-of-Life (EOL) electronic products may contain a significant amount of contaminants, including mercury, lead, cadmium, and polychlorinated biphenyls (PCBs). These toxic materials are not a problem when consumers use electronic products, but they can create health and environmental hazards if they are not properly disposed of at the end of their life.

The need to develop products that minimize environmental damage has become increasingly evident. Products are important in fulfilling human needs, but the side effects of production and consumption - pollution and depletion of natural resources - must also be of concern to designers, manufacturers, and even users. Based on the U.S. Environmental Protection Agency's municipal solid waste (MSW) characterization and the durable goods fraction of MSW, the scrap electronics wastestream may be as high as 5 to 10 million tons per year [Mahecdra, 2002]. Consumer electronics and computers contribute significantly to the environmental burden placed on the municipalities across the nation. Due to the customer's increased interest in "green products" and government legislation, companies have been consistently pressured to increase the pace to develop more and more green products through green manufacturing processes.

Electronic products contain valuable secondary materials - metals and engineered plastics - that can be used by the marketplace in the production of new products. If discarded electronic products and wastestreams, such as computers, televisions, and telephones, could be recovered and reengineered into valuable feedstreams, then the negative trend is broken to achieve sustainability.

Based on the principle of sustainable economy where competitiveness is balanced with environmental responsibility, Multi-Life Cycle Engineering (MLCE) approach was proposed [Caudill, 1997]. MLCE concentrates on investigating, developing and creating new applications for materials and components from discarded products. The objective is to enhance the use of materials and components from discarded products so that they could be used over more than one product lifecycle.

As an integral part of a product life cycle, demanufacturing is a process of disassembling a product and assigning bins for the resultant subassemblies and parts, which are then reused, re-manufactured, reengineered, or disposed of [Shyamsundear et al, 1997]. The process of disassembly, the first step in demanufacturing, attempts to break up a product into several pieces, with expectation that the pieces together have a net value greater than that of the discarded product. Then, the disassembled subassemblies and components are sorted and assigned to the collection bins where they can gain the maximum value. Hence, efficient demanufacturing of a product is one of the main goals of multi-lifecycle engineering.

For instance, in order to maximize the value and minimize the environmental impacts from a used computer, should we refurbish it for sale? If it has no high reuse value, should we disassemble the computer to get valuable components or materials, or separate toxic materials, or just throw the whole computer away? If some of the inside components are associated with uncertain values, how can we deal with the uncertainty to make our decisions?

Besides the End-of-Life (EOL) management of products by means of product/material recovery to decrease environmental impacts, the concept of Design for Environment (DFE) is proposed to integrate environmental requirements into the design process. "Sustainable development" through eco-efficiency [Fiksel, 1996] has become an essential part of the companies who see a competitive advantage in conservation of resources and environmental stewardship. To evaluate environmental performance and sustainability, it is very important to develop and implement an unambiguous and quantitative measure for environmental impacts of products, processes and activities. Compared to other environmental performance metrics, the Sustainability Target Method (STM) approach [Dickinson, 1999] shifts the focus from environmental impact to sustainability, and explicitly considers economic or market value and is equally applicable to analyzing products/processes, services or facilities. By directly considering economic value, resource productivity measures can be generated, and comparisons can be drawn between products or services that are not necessarily “functionally equivalent” but are instead economically equivalent. STM utilizes the earth’s carrying capacity estimates and economic information to provide a practical sustainability target for individual businesses and products. This approach provides a basis for decision making between different suppliers and manufacturers.

Electronics manufacturers use commodities provided from their supply chain. Most electronic components are standard plug-in parts. Supply chain management shortens the cycle between components, manufacturers and end customers. How to choose the suppliers to balance profitability and environmental performance is among the crucial decisions a company has to make.

For instance, a computer company is going to assemble a computer for sale. The company needs to buy parts from different suppliers. Each part is associated with price and environmental impacts. If a company’s objective is to maximize its profitability, the parts with the lowest prices will be chosen. However, a company has to make a decision to balance profitability and environmental performance to make it sustainable too. The real success of a business enterprise is measured by the "triple bottom line" (TBL): its impact on people, profits, and the planet.

1.2 Objectives

This research intends to investigate the optimal decision making algorithms and methods on sustainable development of electronic products. The specific objectives are:

1) To investigate the optimal decision approaches to maximize the profit from a used product through balancing the resource invested in disassembly processes, the return and the environmental impact caused by them.
2) To develop a Multi-Life Cycle Assessment (MLCA) tool to support designers and decision makers for practicing a full life cycle analysis on products based on Life Cycle Assessment (LCA) and Mutli-LifeCycle Engineering (MLCE) concept.
3) To use fuzzy logic theory to quantify the data uncertainty and develop Fuzzy Reasoning Petri nets (FRPN) model and their reasoning algorithm based on fuzzy production rules.
4) To apply FRPN into a demanufacturing process for dynamic decision making with uncertain information.
5) To investigate decision making methods based on Sustainability Target Method (STM) to provide a company with the best improvement suggestions to raise its product's sustainability.

1.3 Organization

The dissertation is organized as follows: Chapter 1 gives the motivation and objectives of the research work. Chapter 2 surveys the literature associated with current research issues and existing methods in the areas of the related research subjects. Chapter 3 proposes the methods to determine disassembly leveling and bin assignment for product multi-lifecycle analysis and assessment. Chapter 4 develops a fuzzy reasoning Petri net (FRPN) model and reasoning algorithm to deal with uncertainty and make the best decisions. Chapter 5 applies FRPN in demanufacturing process decision making. Chapter 6 proposes the improvement approaches of product sustainability based on sustainability target method. Finally, Chapter 7 presents the conclusions and some future research directions.

CHAPTER 2

LITERATURE REVIEW

2.1 Life Cycle Assessment (LCA) and Multi-Life Cycle Engineering (MLCE)

The need to do something about the environment is undisputed nowadays. There are a number of different system-based approaches to integrating environmental issues into industry, such as life cycle assessment (LCA), design for the environment (DFE), total quality environmental management, green supply chain management, and a number of national and international environmental standards. DFE means the systematic consideration of design performance with respect to environmental health, and safety objectives over the full product and process life cycle [Fiksel, 1993]. LCA is one of methodologies to help implement DFE. It is an objective process to evaluate the environmental burden associated with a product, process, or activity by identifying and quantifying energy and materials used and wastes released to the environment, to assess the environmental impact of those energy and materials, and to analyze possible approaches to improve environmental quality [SETAC, 1993]. The assessment includes the entire life cycle of a product, its processes or activities, encompassing a) product design; b) material processing; b) manufacturing; c) use; and d) recycling or final disposal.

Many LCA tools have been available for implementing a full LCA analysis. Some examples of the available LCA database and software are ECO-it [1999], Gabi software [1997], TEAM [1998], and SimaPro 4.0 [1999]. Since it is unlikely that one single LCA methodology can be optimal for all LCA analyses, differences can be found in these tools, depending on the boundary set by the tool and the specific problems it is designed to solve. They may deal with energy at just one life cycle stage rather than the total life cycle. They may also vary in the type of databases of materials they use.

As an extension to the traditional LCA methodology, Multi-Life Cycle Engineering (MLCE) is a new approach in today's environment [Zhou et al., 1996; 1999; Caudill, 1998]. It is based on the principle of sustainable economy where competitiveness is balanced with environmental responsibility. MLCE is a comprehensive, systems approach to growing a strong industrial economy while maintaining a clean, healthy environment - not only today but for future generations [MERC, 2003]. MLCE takes a systems perspective and considers fully the potential of recovering and reengineering materials and components from one product to create another, not just once but many times.

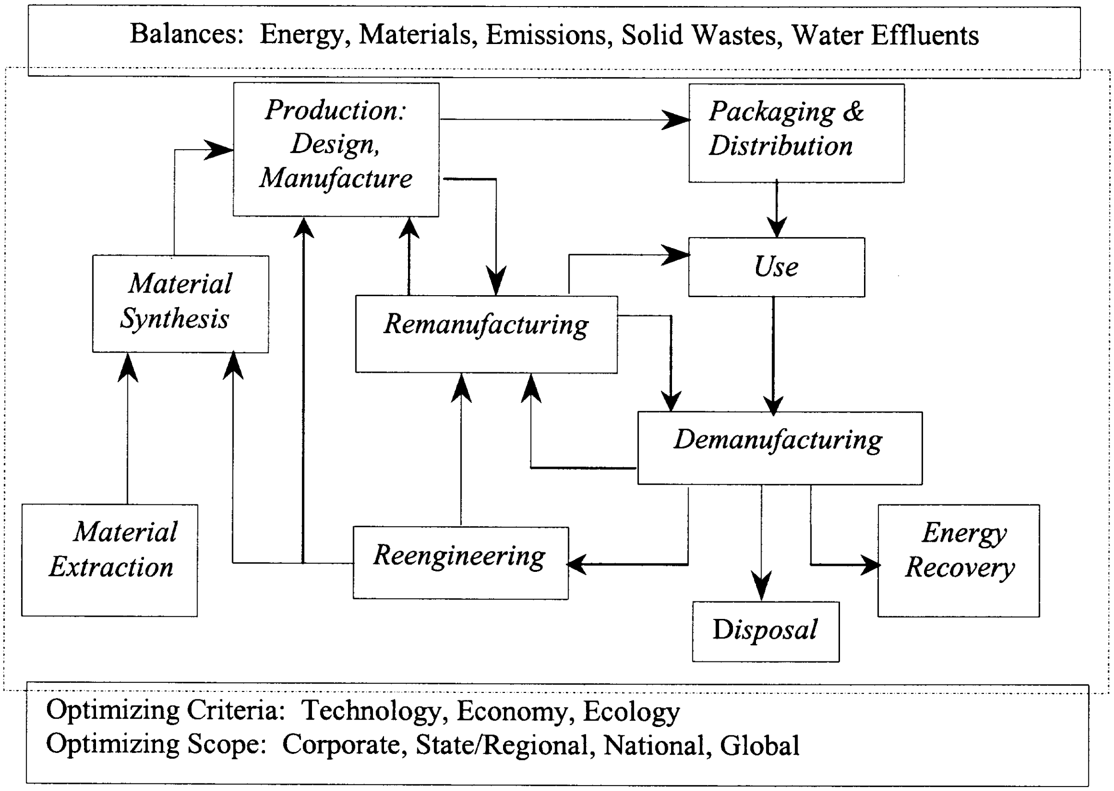

Balances: Energy, Materials, Emissions, Solid Wastes, Water Effluents
Production: Design, Manufacture
Packaging & Distribution
Use
Material Synthesis
Remanufacturing
Demanufacturing
Material Extraction
Reengineering
Energy Recovery
Disposal
Optimizing Criteria: Technology, Economy, Ecology
Optimizing Scope: Corporate, State/Regional, National, Global

Figure 2.1 The total life cycle engineering framework [Caudill, 2001].

Multi-Life Cycle Assessment (MLCA) systematically considers and quantifies the consumption of resources and the environmental impacts associated with a product or process. It extends the structure of traditional LCAs to include explicit consideration of demanufacturing, remanufacturing, reengineering, and reuse - extending LCA to the realm of MLCE [Mahecdra, 2002]. The total life cycle engineering framework in terms of consideration for analysis and modeling can be presented in Figure 2.1 [Jin, 1999].

2.2 Demanufacturing

As mentioned in Section 2.1, MLCA extends the structure of the traditional LCA to the realm of MLCE by including explicit consideration of demanufacturing, remanufacturing, reengineering, and reuse, among which, demanufacturing is the most important concept and procedure in MLCE.

As an integral part of a product life cycle, demanufacturing is a process of disassembling a product and assigning bins for the resultant subassemblies and parts, which are then reused, re-manufactured, reengineered, or disposed of [Shyamsundear et al., 1997]. A demanufacturing procedure is a bridge to connect different lifecycles of products. Hence, efficient demanufacturing of a product is one of the main goals of multi-lifecycle engineering.

In order to maximize the value return from a used product and minimize the environmental impact of a used product, one generally has to make two decisions in demanufacturing stage: 1) how to disassembly a product (disassembly process); and 2) how to assign the disassembled components to the collection bins.

2.2.1 Disassembly Process

The process of disassembly, the first step in demanufacturing, attempts to break up a product into several pieces, with expectation that the pieces together have a net value greater than that of the discarded product. Disassembly has recently gained much attention in the literature due to its role in product recovery [Jovane, 1993]. There are two decisions to be made in disassembly process:

1) *Disassembly Leveling*: identifying the extent to which the disassembly of a product should be performed.
2) *Disassembly Process Planning*: determining the sequence of disassembly tasks.

The disassembly leveling problem can be defined as achieving a disassembly level to which the product of interest is disassembled to keep profitability and environmental features of the process at a desired level [Gungor and Gupta, 1999]. Even though complete disassembly may seem to provide the best way of minimizing the damage to the environment, complete disassembly may not be profitable since the cost of disassembly exceeds the market and environmental benefit. Thus, it is important to find a balance between the resources invested in a disassembly process and the return realized from it.

A Disassembly Process Plan (DPP) is a sequence of disassembly tasks which begins with a product to be disassembled and terminates in a state where all of the parts of interest are disconnected. A plan can, therefore, be either partial or complete disassembly. The objective of disassembly process planning is to find optimal or near-optimal DPPs, in terms of the cost of disassembly or the cost/benefit ratio assuming that a certain level of disassembly is given [Gungor and Gupta, 1999]. In current literatures, however, disassembly leveling is usually integrated into disassembly process planning. Disassembly level and disassembly sequence are decided concurrently.

Researchers in this field have considered various ways to represent the geometric relationships among parts, the most common being graphs and trees, which can be mainly categorized into four types: Connection Graph [Zhang and Kuo, 1996; 1998; Yan and Gu, 1994]; directed graph [Homen and Sanderson, 1990]; AND/OR graph [Homen and Sanderson, 1990; 1991; Lambert 1994; 1997]; and Disassembly Petri Net (DPN) [Moore et al, 1998A; 1998B; 1998C; Zussman et al, 1998; Zussman and Zhou, 2000; 1999].

1) Connection Graph

Connection Graph is an undirected graph, which can be generated according to CAD files of a product. This method can find the feasible disassembly sequences from a CAD system directly and automatically and provides some useful information for designers to evaluate a disassembly problem during the product design. It assumes that the disassembly process is the reversed process of assembly. However, the reversing order between assembly and disassembly process does not always hold.

2) Directed Graph

A directed graph is a graph showing all possible disassembly strategies and associated cost and revenue values. The problem is then converted into a shortest path problem. Its solution techniques are readily available.

3) AND/OR Graph

AND/OR graph is also a directed graph with feasible disassembly sequences. A node representing a product or subassembly can have more than one disassembly method, forming OR-relation; if a disassembly step causes a node (product or subassembly) separated to more than one parts or subassemblies, forming AND-relation. AND/OR graph is also able to show all the possible disassembly sequences as a directed graph we mentioned earlier. Due to the introduction of OR and AND logic relations, AND/OR graph needs less storage space for nodes and edges than the directed graph. This advantage of AND/OR graph becomes greater as the number of parts in a product increases.

It is clear that each kind of disassembly plan corresponds to a tree in the AND/OR graph. Therefore, in order to obtain the optimal disassembly plan, one needs to compute the total cost of each tree from the beginning node (the product) and then chooses the tree with the least cost.

4) Disassembly Petri Nets

Petri nets (PN) [Zhou and Venkatech, 1999] are a graphic modeling method. They are widely used to model and analyze such discrete event systems (DES) as communication, manufacturing, and transportation systems. Petri nets are excellent tools to model and analyze the synchronism, parallel and conflict among the processes in a system. A Disassembly Petri Net (DPN) model extends traditional Petri nets to represent disassembly processes in detail. It is a variant of AND/OR graph. The advantages of using Petri nets are multi-folded. First, we can allow the dynamic behavior to be visualized using a token game from the understanding point of view. Second, both disassembly process and system resources can be represented in a single presentation. Thus concurrent and sequential operations, choice structures, and resource constraints can clearly be described.

Moore et al [1998B] proposed a method to map a AND/OR graph to a DPN and use reachability tree method to generate all feasible disassembly process plans. Cost functions are used to determine the optimal DPP [Moore et al, 1998C].

Zusseman et al. [1998; 1999] propose another disassembly Petri net (DPN) for modeling and adaptive planning of disassembly processes. The objective of their study is to derive the optimal disassembly process plan whose terminal goal is not fixed and the objective function is maximized. Two utility functions are introduced to places and transitions in a DPN. The former represents the end-of-life value for a product, subassembly, or part to be reused, refurbished, material-recycled and utilized, and discarded. The latter represents the cost to perform a particular disassembly operation. To incorporate the uncertainty caused by different product conditions and performance of the external resources, the success rates of disassembly operations are presented by introducing probability values to the corresponding transitions.

5) Other methods

Other methods have also been used in the fields of disassembly process planning. For instance, Huang and Wang [1996] develop a Neural Network-based approach to identify the optimum disassembly sequence that maximizes the profit from the materials recovered. Arai and Iwata [1993] present a simulation-based method for assembly/disassembly planning of a product by utilizing a CAD system and an approach to find the best sequence of disassembly among all possible disassembly sequences with the assumption that a disassembly sequence is just the reverse of an assembly sequence.

6) Problems in existing disassembly process planning methods

Disassembly process planning is the heart of a disassembly system. The key question is how quickly the algorithm can generate a good solution for a system. In the other words, the smaller the complexity, the better the planning [Tang et al., 2000]. However, the methods in current research on disassembly planning process are to find an optimal DPP from all the possible disassembly paths. The number of alternative disassembly paths grows exponentially as the number of the components increase in a product [Gungor and Gupta, 1999]. Therefore, based on the current research, it is not easy to reduce the complexity for a disassembly planning method.

Most of the present research work on disassembly process planning counts on the information of assembly process to generate possible DPPs. First of all, it should not be assumed that the disassembly plan is simply an inverse of the bill-of-materials tree [Das et al., 2002]. Secondly, in practice, the real conditions of some of used products might be unknown. The situation of inside components cannot be known until a product is disassembled to that step, and different individual products (even same type of products) might have different conditions. For example, a hard drive in a computer may be able to be reused while a hard drive in another computer with the same brand, model and age may not. Some components in a product may rust or deform so that they can not be disassembled from other components while they may be in good condition in another product. Those uncertainties make a predetermined plan unrealistic [Tang et al., 2000]. Hence, the current research work on disassembly process planning is hard to be applied in practical disassembly process analysis and planning because of the high complexity and uncertain information.

2.2.2 Bin Assignment

After being disassembled, the resultant subassemblies and components are assigned to collection bins for further processing, such as remanufacturing, reengineering, reuse, and dispose of. This procedure is called bin assignment. The disassembled components could be recovered or discarded. In a typical disassembly facility there are four classes of outputs: 1) retrieved parts or subassemblies; 2) recyclable material mass; 3) waste materials; and 4) hazardous material. Thus, a collection bin could be a "reuse" bin for product recovery (remanufacturing) [EPA, 1997]; a different material bin for material recovery [Porada, 1994]; a "hazard bin" for processing; or a "trash" bin for disposal.

Product recovery includes disassembly, cleaning, sorting, replacing or repairing bad components, reconditioning, testing, reassembly and inspecting. The recovered parts/products are used in repair, remanufacturing of other products and components, and for sale to an outsider.

The recyclable material mass is collected in different material collection bins with certain purity level, which are then transferred to a recycling facility. Disassemblers then contact material recycling vendors for the collection of these bins. Based on the specifications provided by these vendors, dissemblers setup a series of material output bins, like copper, steel and aluminum, which are to be transported to those interested vendors. One can see that the end-of-life value for a part or subassembly depends on what kind of collection bins one sets up. In other words different bin setup affects the outcomes of the disassembly leveling and disassembly process planning. Mani at al. [2001] propose an approach to calculate the material recycling values of parts for a given set of material collection bins using the bill-of-materials information from CAD program.

A bin assignment task has been considered following a disassembly process. Yet, the two processes actually are interweaved. The components in a product are associated with different values if assigned to different bins – reuse value, material values obtained from material recycling, and hazard mitigation value. These values may determine the disassembly operation for a product. Whether a component should be obtained from disassembly is decided by the disassembly leveling. The two processes have been always considered separately in the current literatures. The above analysis shows that such separation may not be the best. It may lead to significant loss in overall demanufacturing gain for certain products if not for every product. Thus, an algorithm integrating the two processes is worth investigating to achieve maximum demanufacturing gain.

2.2.3 Performance Assessment of Demanufacturing at Product Design Stage

To integrate environmental issues into industry, DFE is an approach that product designers evaluate and improve their designs from an environmental perspective. There is a need to develop DFE tools that require minimal use input and can run automatically in the shadow of product design process [Mani et al., 2001].

An MLCA tool is developed by the Multi-Life Cycle Research Center at NJIT for analyzing and comparing the environmental impacts, energy consumption, and cost of different products. With this software, one can practice a full life cycle analysis on products by considering all the effects and options based on product life cycle stages, including product design, material processing, production, use, demanufacturing, remanufacturing and reengineering. The software tool is written for Windows environment using Visual Basic with Microsoft Access as the database package. It aims to support the designers for implementation of the MLCE methodology; develop generic screens that can be used by most consumer electronic producers, with particular focus on telephones, computers, monitors, and televisions; have environmental information on products and processes readily accessible to designers, which simplifies the integration of DFE into a design process; help designers and decision makers track the environmental performance of products by using performance metrics and other indices such as Eco-Compass; and supply a cost model to put the cost issue on line [Mahecdra, 2002].

Demanufacturing stage connects a product's lifecycle to the next. Therefore, the decision making and analysis at demanufacturing stage are crucial to practice a full life cycle analysis on products.

There are following limits and requirements for a designer to practice demanufacturing analysis based on a MLCA tool:

1) Limited information: the only available information at design stage is the data in a product's CAD file;
2) Few user input: users should not input too much data in order to make an analysis;
3) Integration of disassembly leveling and bin assignment: disassembly leveling and bin assignment tasks have to be integrated and execute automatically.

This kind of preliminarily work is explored in [Mani et al, 2001]. Mani et al. proposed an approach to calculate the material recycling values for parts through calculating the contribution of the material in a part to each bin based on Bill of Materials

(BOM). The only use input is reuse values of components. Based on material recycling values and reuse values, they apply complete disassembly first and initialize bin assignment. Then, the result is adjusted using 2 rules. There are some problems with the method:

1) It assumes that all the components in a product are parts, which only have one kind of material. The assumption is not correct;
2) It only provides the approach to calculate the material recycling values for parts, and does not mention the approach to calculation the material recycling values for subassemblies and final subassemblies, which may be composed of more than one kind of material;
3) Hazard penalty is not considered there;
4) Rule 1 there says the parts with low weight won't be disassembled. However, those kind of parts may have to be disassembled first in order to get other desired components;
5) Rule 2 is trying to use a bottom to up approach based on BOM tree to update disassembly levels and bin assignments. However, it assumes that a subassembly's value only comes from its children (a subassembly has no value itself). Therefore, it is not correct;
6) The correctness is not proved; and
7) An illustration example applying the method is missing.

2.3 Environmental Performance Metrics

Inside an LCA and MLCE methodology, it is very important to develop and implement an unambiguous and quantitative measure for environmental impacts of products, processes and activities. There are four major environmental impact assessment methods that are comprehensive in nature and generate a single numerical value reflecting the composite magnitude of global impact associated with a specific product.

1) Eco-indicator 95

The Eco-indicator 95 approach, developed by the National Reuse of Waste Research Programme and Pre Consultants of the Netherlands, is one of the most widely used environmental performance assessment tools. The work is intended to give fundamental and in-depth considerations of how people should evaluate the consequences of impairment of the environment, and how designer can search for more environmentally friendly design alternatives.

The Eco-indicator 95 method classifies, characterizes and normalizes the environmental impacts based upon their effects. The environmental burdens associated with a product are first aggregated into a number of environmental effects caused. Then the effects are characterized according to the degree to which they contribute to ecosystem damage. The characterized effect scores can then be normalized compared with reference values. The result is an environmental profile with standardized effect scores. Finally the different environmental effects are weighted and summed to form an overall environmental index. A single Eco-indicator value is calculated on the basis of impact data from a lifecycle inventory study combined with appropriate weighting techniques. In addition to scientific influence considerations, the weighting is determined based on subjective and political views. The Eco-indicator method uses the degree by which a target level, which reflects the earth's carrying capacity for that effect, is exceeded to weight the different environmental effects. The greater the gap between the current environmental impact and a target level, the higher the rating given to the seriousness of the impact. Detailed information on Eco-indicator 95 can be found in [Goedkoop, 1998].

2) Eco-indicator 99

The Eco-indicator 99 is an extension and update of the Eco-indicator 95 methodology. Both Eco-indicator 99 and 95 use a damage-oriented methodology based on a "less is better" approach. The difference is that, Eco-indicator 95 assesses the potential impact, and a low damage level at which demonstrable but limited damage occurs is set as the target level, while Eco-indicator 99 assesses the impacts on their possible marginal damage, that is to say, no explicit target is defined. Detailed information on Eco-indicator 99 can be found in [Goedkoop and Sepriensma, 1998].

3) Ecological Footprint

The concept of the Ecological Footprint is based on the premise that sustainability depends on maintaining natural capital, and that human consumption and emission must not exceed the natural carrying capacity. This framework [Wackernagal et al., 1999] is intended to track national economy's energy and resource throughput and translate them into biologically productive area necessary to produce these flows. The approach estimates, not only, industrial and commercial usage, but also overall human consumption of resources including food and shelter. The Ecological Footprint represents the critical natural capital requirements of a defined economy or population in terms of the corresponding biologically productive areas. In order to calculate the ecological footprint, recourses consumed and wastes generated have to be estimated. Then, the recourse and waste flow is converted to a biologically productive area necessary to provide these functions. Ecological Footprint is mostly used to calculate area-based indicators of sustainability [Dieoff]. If the footprint area calculated for a region exceeds that available from within its boundaries, the region is deemed to be drawing excessively from the global carrying capacity.

4) Sustainability Target Method (STM)

The Sustainability Target Method (STM) utilizes the earth's carry capacity estimates and economic information in order to provide a practical sustainability target for individual businesses and products. The principle behind the STM approach is to establish a relationship between carrying capacity and both the environmental impact and economic value of products. Initially formulated at Lucent Technologies Bell Laboratories, the STM is under continuing development through collaboration of the Multi-lifecycle Engineering Research Center (MERC) at NJIT with Lucent Technologies and Agere Systems. The key and unique feature of this method is the link between carrying capacity, economic value, and environmental impact to provide an absolute or "target" criteria for sustainability that is practical for use by business [Dickinson et al, 2002].

In general the STM approach involves interpreting different types of environmental impact based on the single dimensionless indicator EI, the environmental impact per production rate. This is accomplished by normalizing environmental impacts using impact reference levels that relate impact with economic value and sustainability. In turn, EI provides the basis for calculating Resource Productivity (RP), and the related indicators Service Productivity (SP) and Value Productivity (VP). These are relative indicators expressing the rate of production, service functionality, and value provided, respectively, per unit of environmental impact. They provide a basis for comparing environmental impact among products (the higher productivity is the better).

Resource productivity and Value productivity are also used to calculate Eco-Efficiency (EE), which is a practical, absolute indicator for sustainability. Essentially it is a product's economic contribution (the percentage of GDP) divided by its environmental burden (the percentage of carrying capacity). Thus the criterion for sustainability is EE ≥100%. EE >100% simply indicates even less impact than the sustainable level given the value being provided. Additional information on STM can be found in [Dickinson et al, 1999; 2001].

Compared to other environmental performance metrics, STM:

1) Shifts the focus from environmental impact to sustainability;
2) Creates relationship between Carry Capacity, Economic Value and Environmental Impact;
3) Provides a practical sustainability target through determining the indicators RP (Resource Productivity) and EE (Eco-Efficiency); and

4) Provides a basis for decision making between deferent suppliers and manufacturers.

For an electronics manufacturer there are commodities provided from the supply line. The STM provides a very effective basis for integrating supply line impacts with production, use and end-of-life by applying the same indicators to evaluate those lifecycle stages [Dickinson et al, 2002]. The STM approach provides a basis for decision making with different suppliers and manufacturers involved, because it uses an absolute "target" criterion for sustainability analysis. In addition, the structure and terminology of STM is consistent with ISO14000 standards.

2.4 Decision Making Methods

Decision making is a process of choosing alternatives based on the values and preferences of a decision maker. Optimal decision making is the strategy of choosing the best possible solution to a problem, discovering as many alternatives as possible and choosing the very best one. The "best" solution depends on the defined problem and the goal of a decision maker. For instance, when a computer company is choosing its suppliers, the company's objectives can be to maximize its profitability, or to balance profitability and environmental performance such that the overall economy or the company most sustainable. Several decision making methods exist, depending on the problems of interest and the criteria of choice.

1) Analytical Techniques

Analytical techniques use mathematical formulas to provide a method for a problem and then derive an optimal solution or predict a certain result. Analytical techniques are used mainly for solving structured problems, usually of a tactical or operational nature, in areas like resource allocation or inventory management. A structured problem is one in which standard solutions exit. In a structured problem, the procedures for obtaining the best (or at least a good enough) solution are known. The dominant methods belong to the field of mathematical programming including linear programming, non-linear programming, and integer programming. In this dissertation, we use an analytical technique to maximize value return from used products and minimize the environmental impact through deciding disassembly level and bin assignment for disassembled components.

2) Search Algorithms

For many cases, even if a mathematical model is built up, the derivation of a solution may not be trivial at all. If such model offers a discrete search space that contains all the solutions, then search algorithms are often used as a method for the solution. Search algorithm is a step-by-step search process for arriving at an optimal solution. Solutions are generated and tested for possible improvements. An improvement is made whenever possible and the new solution is subjected to an improvement test. The process continues until no further improvement is possible.

3) Heuristic Search

The idea of a "heuristic" is a technique which sometimes will work, but not always. It is sort of a rule of thumb. Most of what we do in our daily lives involves heuristic solutions to problems. They usually work, or they usually work well enough, and when they don't work, we then try to find the other ways.

The word "heuristic" is not just used to describe cases where a solution might not be found, but also to describe cases where we are trying to find the best solution. A heuristic rule might help us to find solutions which are good, but perhaps not the very best they can be. It may help us find a good solution faster or reach an optimal solution along a most likely direction. Obviously the measure of goodness, and the assessment of a heuristic technique, is going to be relative to the domain, and to the specific job that problem solving is going to be applied to in that domain. Heuristic search method is used in the situation that the optimal solution is not reachable or to find the optimal solution will be too time-consuming [Zanakis and Evans, 1981].

4) Sensitivity Analysis

Sensitivity analysis [Marshall, 1999] attempts to help decision makers when they are not certain about the accuracy of the relative importance of information, or when they want to know the impact of changes in input information of a model on some results or measures of performance. Through checking the impact of a change in the input data on the proposed solution, the priorities of the input data can be obtained to change proposed solution. For instance, if a computer company wants to make a decision on which components should get more attention and need most urgent improvement such that the company can get the best benefits, the company needs to know the improvement of which component will have the most significant effect on the performance of the whole product.

5) Artificial Intelligence

Artificial intelligence (AI) is basically a theory of how the human mind works. Different from conventional computing, which is based on an algorithm, AI computing is based on symbolic representation and manipulation. In AI, a symbol is a letter, word, or number that is used to represent objects, processes and their relationships. Objects can be people, things, ideas, concepts, events, or statements of fact. By using symbols, it is possible to create a knowledge base that states facts, concepts, and the relationships among them. Then various processes are used to manipulate the symbols to generate advices or recommendation for solving problems.

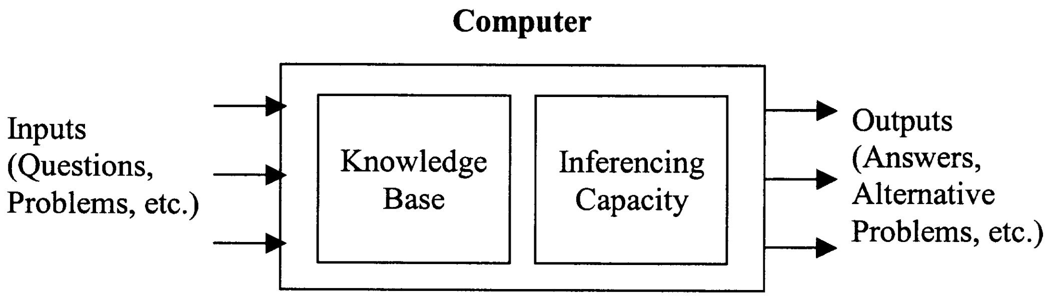


Figure 2.2 Applying AI concepts with a computer.

Figure 2.2 illustrates the concept of a computer using AI in an application. The collection of knowledge related to a problem to be used in an AI system is called a knowledge base. Once a knowledge base is built, AI techniques are used to give the computer the reasoning capacity. With the ability to draw inferences from a knowledge base, a computer can put some practical use as a problem solver and decision maker. Powerful reasoning strategies used to draw inferences are very important in AI.

Several methods exist in the AI research field: evolving solutions - Generic Algorithms (GA) [Grefenstette, 1982; Goldberg, 1989]; simulating human brain to solve problems - Neural Network (NN) [Fu, 1994; Hertz et al., 1991]; expert reasoning - rule based system [Dym and Levitt, 1991]; and dealing with linguistic ambiguity - fuzzy logic [Zadeh, 1975].

Of all the situations one can think of, whether planning, diagnosis, data interpretation, optimization, or social behaviors are involved, many can be expressed in terms of rules. It is not surprising, then, that for several decades rules have served as a fundamental knowledge presentation in artificial intelligence.

Fuzzy logic is basis of a method of reasoning that allows for partial or "fuzzy" descriptions of rules. The power of fuzzy logic comes from the ability to describe a particular phenomenon or process linguistically and then to represent that description in a small number of very flexible rules. The knowledge in a fuzzy system is embedded in its rules and fuzzy sets together, which hold general descriptions of the properties of phenomena.

Neural networks make sense when one has access to a large amount of data and is not in a position to articulate relationships among variables with confidence. Fuzzy rules make sense in just the opposite type of situation: wherever one does not have enough data, but has experts who are able to describe pieces of behaviors in terms of that cannot be disputed.

Which decision making methods should be used in problem solving depends on the problems of interest and the criteria of choice. In this research work, various decision making methods will be used and developed according to the problems and criteria on sustainable development of electronic products.

2.5 Summary

Due to the increasing environmental pressure, sustainable development issues have been caused much interest among manufacturers, governments and consumers. This chapter reviews some of the recent methodology and technology development activities on the sustainable development subject. The issues include the multi-lifecycle engineering for product demanufacturing and recycling, and sustainability analysis at product design and manufacturing stage. This chapter also briefly reviews various decision making approaches in the current literature, some of which will be used and improved in this research work to make decision according to different problems on sustainable development of electronic products.

CHAPTER 3

INTEGRATION OF DISASSEMBLY LEVELING AND BIN ASSIGNMENT FOR DEMANUFACTURING ANALYSIS

As an integral part of a product life cycle, demanufacturing is a process of disassembling a product and assigning bins for the resultant subassemblies and parts, which are then reused, re-manufactured, reengineered, or disposed of [Shyamsundear et al, 1997]. The process of disassembly, the first step in demanufacturing, attempts to break up a product into several pieces, with expectation that the pieces together have a net value greater than the discarded product. Then, the disassembled subassemblies and components are assigned to the target bins where they can gain the maximum value. This process is called bin assignment. Demanufacturing stage connects a product's one lifecycle to the next. Therefore, the decision making and analysis at demanufacturing stage are crucial to practice a full life cycle analysis on products. In order to practice full life cycle analysis at product design stage, designers need to have a DFE tool that require minimal use input and can decide disassembly levels and bin assignment automatically in the shadow of product design process.

Bin assignment task has been considered the one following a disassembly process. Yet, the two processes actually are interweaved. Whether a component is obtained from disassembly is decided by the disassembly leveling. The values of components in a product are associated are decided by bin assignment. On the other hand, these values determine the disassembly operation. However, the two processes are always considered separately in the current literature.

Partially motivated by Mani et al.'s work [Mani et al, 2001]. This chapter is to utilize the information from a CAD system to develop an algorithm, called as a bin assignment algorithm, to complete both the disassembly leveling and bin assignment decisions through balancing the effort involved in disassembly processes, return realized from them, and environmental impact. The algorithm requires minimal user input and can run automatically based on product design information. It has also been implemented into a demanufacturing module of a Multi-Lifecycle Assessment and Analysis (MLCA) [Caudill et al, 2001].

This chapter is organized as follows: Section 3.1 introduces product tree structure, which the algorithm is based on. Section 3.2 presents the bin assignment algorithms design. Section 3.3 focuses on the algorithm implementation and complexity analysis. Section 3.4 briefly introduces MLCA tool. Section 3.5 illustrates the algorithm and its application in MLCA through an example, and Section 3.6 gives summary.

3.1 Product Tree Structure

Bin assignment means to determine disassembly levels and assign disassembled subassemblies and components to target bins where they can gain the maximum value. It is the most important and complicated task in demanufacturing. The algorithm is based

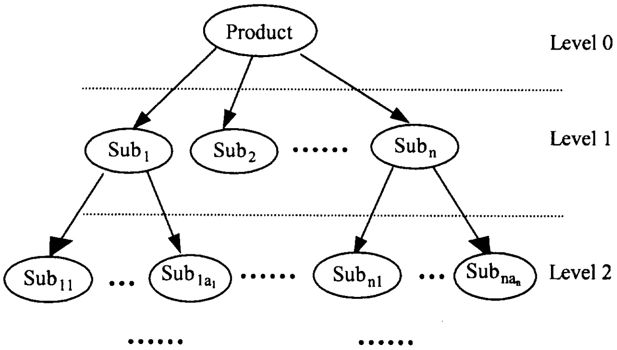


Figure 3.1 Tree structure of a product.

on the product structure, which already exists in the design stage of a product. No other structures need to be built.

The structure of a product can be well represented as a tree as shown in Figure 3.1 [Zhou et al, 1999]. A sample product tree (Zenith Laptop Z Star 433VL) is shown in Figure 3.5. The root of the tree is product itself. The assembly at level k can be disassembled into several subassemblies at level k+1. A subassembly that cannot be separated is a leaf node. If it is made of a single material, it is called a part; otherwise, it is a final subassembly (FS). For example, bolts and nuts are parts; and batteries and chips are FS.

A subassembly at level k has two choices:

1) dealt with as a whole – no further disassembly is needed; and
2) disassembled into subassemblies and/or parts at level k+1.

The disassembled subassemblies and components are assigned to bins for further processing – material or product recovery [Lund, 1996]. Whether a subassembly needs further disassembly and which bin a disassembled component is assigned to can be decided by a bin assignment algorithm automatically, which will be discussed in detail next.

3.2 Bin Assignment Algorithm

3.2.1 Complete Disassembly

In developing the algorithm for bin assignment, users' input should be minimized. In the demanufactuirng stage, the only input data required is readily extracted from common Computer Aided Design (CAD) programs. Specially, the data include product structure, part weight, and part material composition. All the other data are stored in a standardized library.

In this algorithm, it is assumed that the value obtained from a used product only comes from the recycled materials and the product is completely disassembled to the leaf level (parts/final subassemblies). The objective of this algorithm is to calculate the maximum recyclable value of each part/final subassembly and locate its corresponding target bin.

Step1: Calculate the maximum recycling value of the parts and final subassemblies.

Let i be the ith part/final subassembly in a product, j the jth material in part/final subassembly, and β the material recycling bin β. Then the following notations are introduced:

W_{ij} the weight of material j in node i.

$\Phi_{\beta j}$ the recycling yield of material j in bin β.

C_{β} the unit material value in bin β.

The recycling yield represents what percentage of the material is reclaimable when it is sent to a specific bin. Note that $\Phi_{\beta j}$ may be negative for the case where material j generates adverse effect on claiming the main material in bin β. All the above data are stored in database. W_{ij} is extracted from CAD programs, the default values of $\Phi_{\beta j}$ and C_{β} are stored in the standard library. Users with permission can modify the library data through the user interface.

The material recycling value of a part/final subassembly in a specific material bin is the sum of the contribution of each material in the part/final subassembly to the bin.

The contribution is negative if the material generates adverse effect on the recycling the main material in that bin. Hence, the value of a certain part/final subassembly is

$$\underset{\beta}{Max}\{\sum_{j} C_{\beta}\Phi_{\beta j}W_{ij}\}$$

Step 2: Assign the parts and final subassemblies to the corresponding bins.

A part/final subassembly is assigned to bin β if it has the maximum material recycling value in bin β.

3.2.2 Integration of Disassembly Leveling and Bin Assignment

As mentioned before, Algorithm 1 assumes that a product is disassembled to the leaf level. In fact, in order to achieve more benefits from a used product, a subassembly does not need further disassembly if disassembly effort is equal to or lager than the increased value through the disassembly.

At the demanufacturing stage, the first thing needed to do for a product is disassembly. To achieve the maximum benefit from used products, the disassembly level and a reasonable disassembly plan should be decided. Disassembly planning is a complicated task because there are various connections between the parts / subassemblies [Tang et al, 2000]. In this work, it is assumed that the immediate children in a subassembly can be obtained through one aggregate step of disassembly. The assumption does not hold only when some of subassemblies need partial disassembly. The algorithm based on the assumption will work well in case that the most of subassemblies/parts in a product will be deal with as a whole, for example keyboard, hard drive, monitor, and floppy disk compartment in a computer.

This work proposes an algorithm to determine disassembly levels and target bins in an integrated way to maximize the profit from a used product through balancing the effort involved in disassembly processes, the return and the environmental impact. In developing an algorithm for bin assignment, one tends to utilize the information associated to a product, which is readily extracted from its CAD programs. Specially, this data includes a component relationship tree, component material composition and weight.

Each node in a product tree may have reuse value, material recycling value, hazard mitigation value or the value coming from its children. The objective is to calculate the maximum value of the root node (a used product) in the product tree. Here a bottom-up approach is used, which starts from the leaf level of a product tree to the root. The following are the steps of the algorithm, which will be more formally presented in Section 3.3.

Step 1: Calculate the maximum material recycling value of each node in a product tree.

In this step, the material recycling value of a node in each material bin is calculated and the maximum value is selected among them as its material value. Here, a node may be a whole product, or subassembly having children, or leaf (part/FS).

Let i be the i^{th} node in a product, j the j^{th} material in it, and β the material recycling bin β. Then the following notations are introduced:

W_{ij} the weight of material j in node i.

$\Phi_{\beta j}$ the recycling yield of material j in bin β.

C_{β} the unit material value in bin β.

M_i the material value coming from the material recycling of node i.

The recycling yield $\Phi_{\beta j}$ represents the degree of the reclaimable material j's contribution to a specific bin β. Note that $\Phi_{\beta j}$ may be negative for the case that material j generates adverse effect on recycling of the main material in bin β.

The material recycling value of a subassembly in a specific material bin is the sum of the contribution of each material in the subassembly to the bin. Hence, the material value of node i is

$$M_i = \underset{\beta}{Max}\{\sum_j C_\beta \Phi_{\beta j} W_{ij}\} \tag{3.1}$$

Step 2: Determine initial values of all the nodes in a product tree.

R_i the reuse value of node i

H_i the hazard mitigation value of node i

V_i the value of node i to be decided

B_i the target bin for node i to be decided

Hazard mitigation value is a kind of profit a company achieves through reducing the hazardous material's impact on the environment. The target bin B_i for a node is the bin where it has the maximum value. If the subassembly is not available or the product will not be disassembled to that level, B_i is set to zero. Three special bins are defined: The first bin (β=1) is the "Reuse" bin where all reusable subassemblies are going to be reused; the second bin (β=2) is "Hazard" bin where the parts/FS with a high hazard mitigation value are stored for further treatment, and the third bin (β=3) is the "Trash" bin where the components without any value are stored to be disposed of. Material bins are numbered with β>3.

The algorithm uses a bottom-up approach, which starts from the leaf level of a product tree. Hence, the initial value of a non-leaf node is set to zero. For a part/FS, its initial value is decided by its reuse value R_i, hazard mitigation value H_i, and material value M_i which can be calculated using Equation (3.1).

Formally, the initial value for node i having children:

$$V_i = 0$$

Then the target bin of the initial assignment for a node having children is set to zero ($B_i = 0$ means that node i is not available because it is assumed that a subassembly is going to be disassembled into children by default).

For a part/FS i, its material value is compared to its reuse value first. If the reuse value is greater than or equal to the material recycle value, it is going to be reused. The obtained value from the part/FS is the reuse value plus its hazard mitigation value, which is obtained by avoiding throwaway of the component. Otherwise, the part/FS is going to be recycled if it does have material value and target bin can be obtained from Equation 3.1. If it does not have material value at all (M_i ≤0), then check if it has hazard mitigation value. If so, set its value to that value and assign the part/FS into "Hazard" bin. Otherwise, its value is set to zero and assign the part/FS to "Trash" bin. In summary,

If $R_i \geq M_i$ and $R_i > 0$, then $V_i = R_i + H_i$, B_i=1;

else if $M_i > 0$, then $V_i = M_i + H_i$, $B_i = \beta$;

else if $H_i > 0$, then $V_i = H_i$, B_i=2;

else $V_i = 0$, $B_i = 3$.

Step 3: Determination of disassembly levels and update of values and target bins.

A disassembly leveling problem can be defined as achieving a disassembly level to which the product of interest is disassembled to keep profitability and environmental impact at a desired level. Complete disassembly may seem to provide the best way of minimizing the damage to the environment [Gungor and Gupta, 1999]. It is not profitable when the cost of disassembly is more than the market and environmental benefits. Thus, it is important to find a balance between the resources invested in a disassembly process and return realized from it [Bhamra, 1997].

In order to determine the disassembly level, the disassembly effort is required to check if further disassembly is worthy. The disassembly effort for subassembly i to break into its children is denoted by E_i, which can be calculated according to [Mani et al, 2001] or given by users if known. In [Mani et al, 2001], disassembly effort is modeled as a function of the number of links unfastened, the packing density of each subassembly, and the access level of the subassembly. These values can be obtained or estimated from the CAD data, which include the number of the children the subassembly has, their bounding volumes, and the level at which the subassembly is located.

3.1) Assume that L is the total number of levels in a product. Set k=L-1;

3.2) For each subassembly i at level k:

The net value obtained through a disassembly process

$$A_i = \sum_{\gamma} V_{i_\gamma} - E_i$$

where, i_γ is the γ^{th} immediate child of subassembly i, and $\sum_{\gamma} V_{i_\gamma}$ is the total value of immediate children of subassembly i.

The hazard mitigation value associated with subassembly i is

$$H_i = \sum_{\gamma} H_{i_\gamma}$$

i.e., the hazard mitigation value equals the sum of the hazard mitigation values its immediate children may have.

For each subassembly i (having children) at level k,

if $R_i \geq M_i$ and $R_i + H_i \geq A_i$ and $R_i > 0$, then $V_i = R_i + H_i$, $B_i = 1$ and set values and the target bins of all its children to 0;

else if $M_i + H_i \geq A_i$ and $M_i > 0$, then $V_i = M_i + H_i$, $B_i = \beta$ and set values and the target bins of all its children to 0;

else if $A_i > H_i$, then $V_i = A_i$, $B_i = 0$;

else if $H_i > 0$, then $V_i = H_i$, $B_i = 2$ and set values and the target bins of all its children to 0;

else $V_i = 0$, $B_i = 3$ and set values and the target bins of all its children to 0.

The situation here is slightly similar to the value calculation of a part/FS in Step 2. At first, to check if an assembly is going to be reused. If not, then check if the value gain through a disassembly process is greater than the material recycling value of the subassembly as a whole to decide if it needs further disassembly. It is noted that A_i itself includes hazard mitigation value if some of its children have. If the value gain through a disassembly process is not positive, the subassembly will not be disassembled, and whether it is treated as trash directly depends on if it has hazard mitigation value.

The values and target bins of a subassembly may need to be updated. If it needs no further disassembly, the target bins of its children are set to 0 because the product will not be disassembled to that level. Otherwise, it is to be disassembled. Then, its value comes from its children's and the subassembly itself is not available (set its bin to zero.).

3.3) k=k-1. If k<0, the algorithm stops, otherwise go to Step 3.2.

After these three steps, the target bin for each node has been obtained. Then a bin tree can be created according to the above calculation results.

3.3 Algorithm Implementation and Complexity Analysis

3.3.1 Algorithm Implementation

The goal of the proposed algorithm is to determine the value and target bin for all the nodes in a tree according to the information associated to the nodes and the tree structure so that the maximum value can be obtained for each node.

Inputs:

1) Data extracted from product CAD file: tree structure, W_{ij};

2) Date in database: $\Phi_{\beta j}, C_{\beta}, R_i$, and H_i.

Outputs: V_i and B_i.

The flow diagram of the algorithm is shown in Figure 3.2. The algorithm implementation can be represented as follows:

Step 1: Initialization

For leaf node (Node.type='P' or 'FS'),

$$M_i = \underset{\beta}{Max}\{\sum_j C_{\beta}\Phi_{\beta j}W_{ij}\};$$

If $R_i \geq M_i$ and $R_i > 0$, then $V_i = R_i + H_i$, $B_i = 1$;

else if $M_i > 0$, then $V_i = M_i + H_i$, $B_i = \beta$;

else if $H_i > 0$, then $V_i = H_i$, $B_i = 2$;

else $V_i = 0$, $B_i = 3$;

End If;

For a non-leaf node (Node.type='S'), V_i=0 and B_i=0.

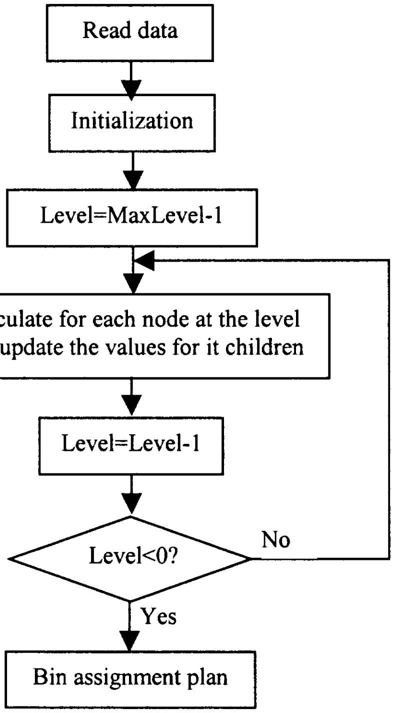


Figure 3.2 The flow diagram of the bin assignment algorithm.

Step 2: Update

```
L=Maximum level of the tree;
k=L-1;
Repeat:
    RecordSet=All nodes at level k where
    node.type='S';
    If RecordSet.RecordCount≠0 then
    RecordSet.MoveFirst;
    End If;
    Do while not end of RecordSet
        For subassembly i,
        Estimate $E_i$ according to an algorithm in [Mani et al, 2001] or use user-supplied data;
        $A_i = \sum_{\gamma} V_{i\gamma} - E_i$;
        $H_i = \sum_{\gamma} H_{i\gamma}$;
        $W_{ij} = \sum_{\gamma} W_{i\gamma,j}$;
        $M_i = \underset{\beta}{Max}\{\sum_{j} C_{\beta}\Phi_{\beta j}W_{ij}\}$;
        If $R_i \geq M_i$ and $R_i + H_i \geq A_i$ and $R_i > 0$, then $V_i = R_i + H_i$, $B_i = 1$ and set values and the target bins of all its children to 0;
        else if $M_i + H_i \geq A_i$ and $M_i > 0$, then $V_i = M_i + H_i$, $B_i = \beta$ and set values and the target bins of all its children to 0;
        else if $A_i > H_i$, then $V_i = A_i$, $B_i = 0$;
        else if $H_i > 0$, then $V_i = H_i$, $B_i = 2$ and set values and the target bins of all its children to 0;
        else $V_i = 0$, $B_i = 3$ and set values and the target bins of all its children to 0;
        End If;
    RecordSet.MoveNext;
Loop;
```

```
k=k-1;
Until k=-1.
```

3.3.2 Complexity Analysis of The Algorithm

In this section, it will be approved that the algorithm maximizes the value of a used product with a polynomial computational complexity under assumption that a subassembly is disassembled into its immediate children in one aggregate step.

Theorem 1: In a product tree, the number of all the nodes (part, FS or subassembly) is less than or equal to 2n-1, where n is the number of leaves (part or FS).

A product tree is a type of tree, each node of which has at least two children except its leaves that have no children. This point is obvious because each subassembly is composed of at least two components. According to this character of a product tree, it is easy to prove that for the fixed number of nodes in a product tree, the number of leaves is minimum when each non-leaf node has exactly two children. In other words, the number of nodes is the maximum for a fixed number of leaves in a product tree when each non-leaf node has exactly two children. In a tree whose non-leaf nodes have exactly two children, if the number of the leaves is n, then the number of the nodes is N=2n-1[Baase 1988]. Therefore, Theorem 1 is proved.

Theorem 2: The algorithm complexity is O(n^2+mn), where m is the number of collection bins and n is the number of leaves in a product tree.

Proof:

In Step 1, in order to initialize the values for leaf nodes, the value of each node in each bin needs to be calculated. Hence the initialization for leaf nodes runs at O(mn); and the initialization for a non-leaf value runs at O(N-n) where N is bounded by 2n-1 according to Theorem 1. Then, the complexity for Step 1 is O(mn)+O(N-n). From Theorem 1, O(mn)+ O(N-n) = O(mn) + O(2n-n) = O(mn).

In Step 2, the calculation is based on each non-leaf node. For each such node, the values of its immediate children are required to calculate A_i, H_i and W_{ij}, and the calculation runs at O(n) in the worst case; The value of each non-leaf node in each bin needs to be calculated to obtain M_i, which runs at O(m); The values of all the children of a non-leaf node may need to be updated, which runs at O(n) in the worst case. Thus, for each non-leaf node, the calculation runs at O(2n+m), i.e., O(n+m) in the worst case. Therefore, the total computational complexity in Step 2 is O((N-n)*(n+m)), i.e. O(n^2+mn) .

Hence, the computational complexity of the algorithm is O(n^2+mn).

Theorem 3: The algorithm maximizes the value of a used product given input data including its tree structure in which a subassembly is disassembled into its immediate children in one aggregate step.

Proof:

The objective of the algorithm is to obtain the maximum value of a used product, which is the value of the root node (k=0) in product tree. Hence, the proof will be done if the value of the root node calculated from the algorithm is proved to be its maximum value.

For each subassembly at a certain level, it is either left intact or disassembled into their immediate children. If it is not disassembled, the subassembly goes to reuse bin

(B_i=1) with the value ofR_i+H_i, material recycle bin (B_i= β) with the value of M_i+H_i, hazard bin (B_i=2) with the value of H_i, or trash bin with no value. Otherwise, the subassembly is disassembled into its immediate children and has the value of A_i. Therefore, the maximum value of the subassembly is Max{R_i+H_i, M_i+H_i, H_i, A_i, 0}.

According to the algorithm:

Consider the first condition of "$R_i \geq M_i$ and $R_i+H_i \geq A_i$ and $R_i > 0$". It is equivalent to "$R_i+H_i \geq M_i+H_i$ and $R_i+H_i \geq A_i$ and $R_i+H_i > H_i$, and $R_i+H_i > 0$", which means that Max{R_i+H_i, M_i+H_i, H_i, A_i, 0}=R_i+H_i. Therefore, according to the algorithm, when Max{R_i+H_i, M_i+H_i, H_i, A_i, 0}=R_i+H_i, the subassembly goes to reuse bin with maximum value of R_i+H_i. In the similar way, It can be proved that when Max{R_i+H_i, M_i+H_i, H_i, A_i, 0}=M_i+H_i, the subassembly goes to material recycle bin with maximum value of M_i+H_i; when Max{R_i+H_i, M_i+H_i, H_i, A_i, 0}=H_i, the subassembly goes to hazard bin with maximum value of H_i; when Max{R_i+H_i, M_i+H_i, H_i, A_i, 0}=A_i, the subassembly is disassembled into its immediate children with the maximum value of A_i which is its immediate children's values minus its disassembly effort; or the subassembly goes to trash bin with zero value.

The algorithm uses a bottom-up approach, which starts from the leaf level of a product tree to the root. At each level, according to the above analysis, the maximum value of each node at that level is obtained. Therefore, for the subassembly at the next upper level, its immediate children have their maximum values so that the subassembly's value can be updated with its maximum value. Continuing this to the root, the value of the root is its maximum value. Therefore, Theorem 3 is proved.

3.4 MLCA Tool

The proposed algorithm is successfully applied to develop a demanufacturing module in a MLCA Tool [Caudill et al, 2001; Gao et al, 2001]. This section gives a brief introduction to the MLCA tool first and then focus on the development of demanufacturing module.

3.4.1 Brief Description of MLCA Tool

The MLCA tool is developed to apply the MLCA methodology to electronic products and help designers obtain environmental performance of their products. An ideal MLCA tool aims [Caudill et al, 2001]:

1) To support the designers to implement the MLCA methodology;
2) To develop generic screens that can be utilized by most consumer electronic designers, producers, and demanufacturer;
3) To have environmental information on products and processes readily accessible to designers, which facilitates the integration of DfE (Design for Environment) into a design process;
4) To help designers and decision-makers to track the environmental performance of products by using performance metrics and other indices such as Eco-Compass [Fussler and James, 1996].

The initial prototype version of a multi-lifecycle assessment software tool has been completed. It starts from product description screens, goes through seven multi-lifecycle stages, finally, ends with final analysis and reports. There are three key levels in the development of the MLCA software:

The first one is the import of product information. The information of a product that consists of certain subassemblies and parts is needed to evaluate a product layout, in a general case, including material type, weight, and quantity of each component in a product, and the tree structure of a product. If the product file is of ProEngineer format, an automatic data retrieval and database population algorithm can be used [Zhou et al., 2001].

The second level is the multi-lifecycle screen development. Information specific to each multi-lifecycle stage of a product is entered and modified in these screens. There are six screens related to each multi-lifecycle stage: Material Processing, Production, Use, Demanufacturing, Reengineering, and Remanufacturing.

The third one is to use algorithms to make decision and transfer the product information into assessment reports mainly on value gain, environmental burdens and energy consumption. The reports of the actual energy consumptions and environmental burdens consumed and generated during the lifecycles of their products can help designers make improvements to their product design. Thus the devised ones have higher profit, less energy consumptions and environmental burdens, while satisfying the same basic functional requirements.

Other than the screen implementation of MLCA software, data storage and retrieval to and from Microsoft Access database is another important aspect of the implementation. SQL queries are the common way to do it. To link SQL queries with

Visual Basic, the process is usually done with statements that create Recordsets, which is an easy way to create an object that gives us access to all the data in the query.

Multi-lifecycle analysis is a systematic approach in analyzing a product, since it accounts for the multiple lifecycles that materials or components pass through. This requires a clear vision and understanding of the product from its raw material extraction through use stage and finally demanufacturing, reengineering and remanufacturing. Hence, the efficient demanufacturing of a product is one of the prime goals of multi-lifecycle engineering. In the following sections, the design and development of the demanufacturing stage module will be introduced.

3.4.2 Demanufacturing Module

3.4.2.1 Bin Assignment. Bin assignment is to assign the disassembled subassemblies and components to the target bins where they can gain the maximum value, which is the most important and complicated task in the demanufacturing stage. In the MLCA tool, the bin assignment is accomplished in two ways: First, the disassembly algorithms (as discussed at Section 3.3) are designed to automatically sort the parts and subassemblies into bins based upon disassembly effort and market value of recovered materials. The second, a manual sort, if needed, is performed where the users can make modification on the tree built up by the algorithm or set up a totally new bin tree manually, which provides additional convenience to users when they need to assess and analyze the environmental performance of a product.

1) Automatic assignment

In the demanufacturing module, the algorithms (as discussed in Section 3.4) are developed to assign collection bins automatically. All the data needed for the algorithm are stored in Access database. The information of a product, in a general case, including material type, weight, and the tree structure of a product can be obtained through an automatic data retrieval and database population algorithm [Zhou et al, 2001]. Original product information is extracted from Pro/Engineer software and written into XML (Extensible Makeup Language) file. Then, the XML file data is retrieved into database through a Java program [Zhou et al, 2001]. The default values of $\Phi_{\beta j}$, C_{β}, R_i and H_i are stored in a standard library. Users with permission can modify the library data through the user interface. A sample screen of the demanufacturing module with assignment results is shown as Figure 3.5. In the MLCA tool, a bin tree is created where a bin name is displayed and then all the parts and subassemblies sorted into that bin are arranged along the tree. An example will be described in Section 6.4.

2) Manual assignment

Users also can make bin assignment manually in the MLCA tool. The functions provided by manual assignment include:

- Add new bin;
- Delete bin/subassembly;
- Clear bin tree;
- Click-and-drag a subassembly from the product tree being displayed on the screen to the appropriate bin;
- Click-and-drag a subassembly between the two different bins in the bin tree to re-assign the bin to the subassembly.

3.4.2.2 Process Specification and Definition. After being assigned to the bins, the subassemblies in each bin are going to be processed for material recycle or disposed of.

There are two kinds of processes may be applied to the bins: Primary process (Reengineering / Remanufacturing / Disposal) and secondary process (Shedding and separation / Decontamination / Demilitarization). A secondary process is a pre-process before a primary process. Users can specify / view / re-assign the processes for the bins. The recyclable materials assigned to reengineering / remanufacturing process are going to the corresponding stages in the tool for further analysis. The efficiency of the process for each material and the cost associated to the processes also need to be specified. The default values are provided to users extracted from the database and users can also update the value through the designed user interface according to their particular application.

3.4.2.3 Result Display. In MLCA tool, reports and comparative analysis can be performed based on the information of a product and processes. An interactive capability exists for analysts to select one or more lifecycle stages, impact categories, and specific environmental burdens of interest. Inventory calculations are made and results can be displayed in both graphical and tabular formats.

In the demanufacturing stage, value return, environmental burden, energy consumption and cost are associated with the disassembly operations and processes for recyclable materials in different collection bins. The developed demanufacturing module allows users to effectively manage the end-of-life options of a product.

3.5 A Computer Product Example

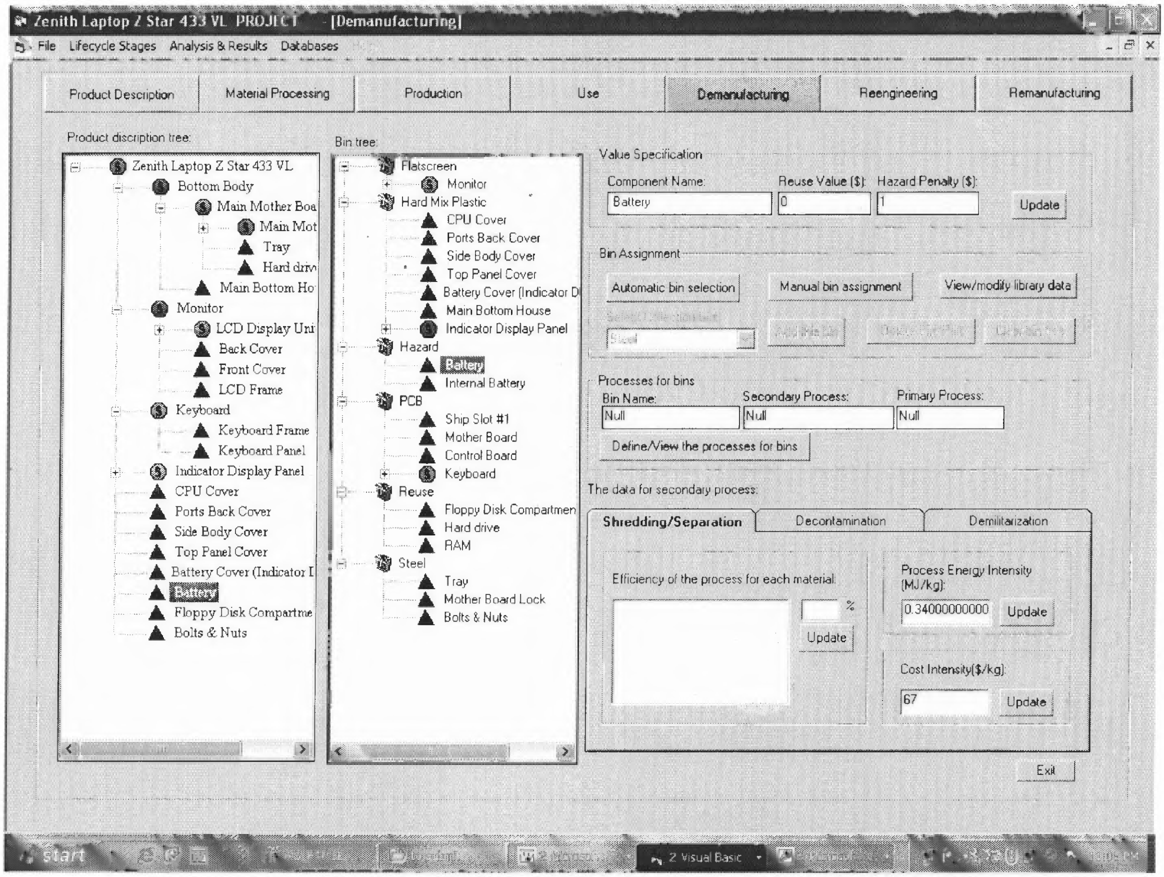
Figure 3.3 A Zenith laptop computer.

Three cases are studied and compared in this section to illustrate the proposed algorithm using Zenith Laptop Z Star 433VL computer (Figure 3.3) as an example with the data and results presented in Appendix. Please note that all the numerical information shown has been adopted only for the illustration purpose. In Case 1, the laptop is completely disassembled no matter what condition it has. In Case 2, the disassembly level is determined using the proposed algorithm in this paper, where the information in CAD file is utilized. Case 3 assumes that the mating relationship in the product is known (shown as in Figure A.1 of Appendix), which is based on a disassembly experiment on a real product [Yossapol and Boontaveekit, 2000]. The comparison of value gains from the product is shown in Table 3.1. The detailed data associated to the product for Case 2 are displayed in Tables A.1-A.3 of Appendix. The information for Case 3 is shown in Tables A.4 and A.5 of Appendix, which is based on [Yossapol and Boontaveekit, 2000]. Please note the disassembly efforts are calculated under the assumption that disassembly cost is $12/hour.

Table 3.1 Value Gains in Three Cases

Value gain ($)		
Case 1	Case 2	Case 3
1.62	5.59	6.93

From this example, it can be observed that the proposed algorithm (Case 2) is 3.5 times better than complete disassembly with the same data input and computational complexity. Case 2 is more than 80% close to Case 3. However, Case 3 needs much more data input than Case 2, such as disassembly sequences. The required information may not be available before a product is disassembled.



Component Name: Battery
Reuse Value ($): 0
Hazard Penalty ($): 1

Bin Name: Null
Secondary Process: Null
Primary Process: Null

The data for secondary process:

Process Energy Intensity (MJ/kg): 0.34000000000

Cost Intensity($/kg): 67

Figure 3.4 Demanufacturing screen of MLCA tool.

A sample bin assignment result for this example is displayed as a bin tree in a MLCA screen as shown in Figure 3.4. In MLCA, the bin tree is created where a bin name is displayed and then all the parts and subassemblies sorted into that bin are arranged along the tree. For example, mother board is assigned to "PCB" bin since it gains the maximum value in that bin. In the same way, Battery enters "Hazard" bin since it has high hazard mitigation value, and hard disk and floppy disk compartment go to "Reuse" bin with high reuse value, and so forth.

3.6 Summary

This chapter is to utilize the information from a CAD system to develop algorithms to complete the disassembly leveling and bin assignment decisions through balancing the effort involved in disassembly processes, return realized from them, and environmental impact caused by them. An algorithm is developed for the first time to integrate the functions of disassembly leveling and bin assignment together to achieve the maximum profit from used products. The developed algorithm is of polynomial computational complexity, i.e., $O(n^2+mn)$. It has the similar complexity as that of complete disassembly-based bin assignment algorithm, i.e., $O(mn)$. Note that m is the number of bins and n is the number of parts and final assemblies. Instead of the exponential complexity others, e.g. [Zussman 1999; Tang et al, 2001], may have, the proposed algorithm has only polynomial complexity owing to the assumption that a subassembly at level k is disassembled into its immediate children in one aggregate step. The proposed algorithm can bring significantly higher gain than the complete disassembly-based algorithm. Its performance can be very close to that of optimal disassembly planning, depending upon the product structure and accuracy of the disassembly effort estimates. It has successfully been used in the demanufacturing module of an MLCA tool. It requires minimal user input and can be run automatically in the shadow of a design process.

This disassembly leveling and bin assignment algorithm is a kind of bottom-to-up search algorithm based on tree structure, which is a step-by-step update process for achieving an optimal solution. The process continues until no further improvement/update is possible. In the following two chapters, an artificial intelligent method based on fuzzy reasoning to implement real time demanufacturing process decision making will be discussed.

CHAPTER 4

FUZZY REASONING PETRI NETS

In Chapter 3, an algorithm is proposed to utilize the information from a CAD system to complete both the disassembly leveling and bin assignment decisions through balancing the effort involved in disassembly processes, return realized from them, and environmental impact caused by them. Because the algorithm requires minimal user input and can execute automatically, it can be used for multi-lifecycle assessment perfectly. However, since products during the use stage may experience very different conditions, their external and internal status can vary significantly. These products, when coming to a demanufacturing facility, are often associated with uncertain information. To be able to deal with uncertainty and complement real time decision making, this chapter presents a Fuzzy Reasoning Petri net model for knowledge representation and reasoning with uncertainty.

This chapter is organized as follows: Section 4.1 introduces basic concepts of Petri nets and reviews the exiting Petri net model for fuzzy reasoning. Section 4.2 introduces fuzzy production rules and their Petri net representation. Section 4.3 proposes the formal definition, execution rules, and examples of Fuzzy Reasoning Petri Net (FRPN). Section 4.4 presents the reasoning algorithm based on an FRPN model. Section 4.5 illustrates the legitimacy and feasibility of the proposed approach through a turbine fault diagnosis expert system. Section 4.6 is the summary.

4.1 Introduction

Petri nets (PN) [Zhou and Venkatech, 1999] are a graphic modeling method. They are widely used to model and analyze such discrete event systems (DES) as communication, manufacturing, and transportation systems. Since Zisman showed that PN could be translated into production rule systems [Zisman, 1978], they have caused great attention in the field of artificial intelligence (AI). PN and AI were combined to achieve some goals difficult to fulfill by using only one of them. In order to deal with uncertainty, several authors from PN and artificial intelligence communities have proposed different kinds of Fuzzy Petri Nets (FPN) based on different notions. For example, in [Valette et al, 1989] FPN are used to model DES under incomplete information about state and time, where fuzzy variables are used to describe the degree of a physical object in a certain place [Cardoso et al, 1999]. Cao and Sanderson (1995) proposed FPN as a presentation of a robotic workcell by defining fuzzy internal states of objects and using transition reasoning rules in PN to impose precedent constraints on key operations for task sequence planning. Another important combination of PN with artificial intelligence is that PN model is applied to fuzzy rule-based reasoning using proposition logic, where tokens represent the state of propositions and rules are marked by a fuzzy value between 0 and 1 [Loony, 1988]. In his model, the transitions serve as rules, and the places serve as propositions. Based on Looney's method, Chen et al. improved its ability in both expressions and reasoning method. They developed an algorithm to determine whether there exists an antecedent-consequence relationship from a proposition to another [Chen et al, 1990; Chen, 2002]. The fuzzy reasoning algorithm is implemented using a reachability based method. Though it can be implemented using a computer, the algorithm is loaded down with much trivial details. The parallel reasoning ability of PN has not been exploited. Since then, many approaches have been proposed to extend PN to FPN [Tazaki and Yoshida, 1992; Bugarin and Barro, 1994; Konar and Mandal, 1996; Scarpelli et al, 1996]. The same name, FPN has been used based on different notions, which has caused the concept confusion in this research field. In the dissertation work, Petri nets are applied into proposition logic for (fuzzy) rule-based reasoning and "Reasoning Petri Nets" concept is proposed to avoid the confusion.

The advantages of applying a PN formalism to rule-based systems are summarized as follows

1) PN's graphical nature allows one to visualize the structure of a rule-based system and make the models relatively simple and legible.
2) PN's mathematical foundation allows one to express the dynamic behavior of a system in algebraic forms.

However, most current researches related to FPN focused on applying a fuzzy reasoning mechanism over the PN structure rather than utilizing the PN formalism to improve the efficiency of fuzzy reasoning [Yang et al, 1997].

The relationship between the ordinary PN and PN for knowledge representation and reasoning remains unexploited in the previous work so that Petri nets theory could not be used into proposition logic reasoning efficiently. Moreover, the existing FPN models for knowledge representation and reasoning are unable to deal with the negation issue in a knowledge base system, which is of critical importance because an antecedent(s) and/or a consequent(s) of a rule in a knowledge base may be a negative literal. This chapter aims to:

1) explore the similarity and difference between DES and rule-based systems;
2) propose the concept of Fuzzy Reasoning Petri Nets (FRPN) for rule-based representation;
3) define formally FRPN and its execution rules;
4) propose an efficient reasoning algorithm with parallel reasoning ability; and
5) illustrate their applications through an expert system.

4.2 Fuzzy Production Rules and Their Representation with FRPN

4.2.1 Formats of Fuzzy Production Rules

In many situations, it is difficult to capture data in a precise form. In order to properly represent real-world knowledge, fuzzy production rules [Negoita, 1985] have been used for knowledge representation.

As addressed in [Chen et al, 1990; Yang at al, 1997], fuzzy production rules can be classified into four types and their fuzzy reasoning results can be summarized as follows:

Type 1: $R_j(c_j): P_1(\theta_1) \wedge P_2(\theta_2) \wedge \ldots \wedge P_{k-1}(\theta_{k-1}) \rightarrow P_k(\theta_k)$

$$\theta_k = \min\{\theta_1,\ \theta_2, \ldots, \theta_{k-1}\} * c_j$$

Type 2: $R_j(c_j): P_1(\theta_1) \rightarrow P_2(\theta_2) \wedge \ldots \wedge P_{k-1}(\theta_{k-1}) \wedge P_k(\theta_k)$

$$\theta_2 = \theta_1 * u_j$$

$$\theta_3 = \theta_1 * u_j$$

$$\ldots$$

$$\theta_k = \theta_1 * u_j$$

Type 3: $R_j(c_j): P_1(\theta_1) \vee P_2(\theta_2) \vee ... \vee P_{k-1}(\theta_{k-1}) \rightarrow P_k(\theta_k)$

$$\theta_k = \max\{\theta_1,\ \theta_2, \ldots, \theta_{k-1}\} * c_j$$

Type 4: $R_j(c_j): P_1(\theta_1) \rightarrow P_2(\theta_2) \vee ... \vee P_{k-1}(\theta_{k-1}) \vee P_k(\theta_k)$

where θ_i denotes the truth degree of antecedent/consequent part P_i in a fuzzy rule R_j, and c_j denotes the confidence degree of applying rule R_j.

Rules of Type 4 are unsuitable for deducing control because they make no specific implication. Therefore, this type of rules is not allowed to appear in a knowledge base [Chen et al, 1990]. In the following, the first three types of rules will be discussed.

4.2.2 Rule Representation

Petri nets used in knowledge representation, called Fuzzy Reasoning Petri Nets (FRPN), stand for the syntactic structure of rule-based knowledge. Propositions can be represented by PN places. Each place may or may not contain a token associated with a truth degree between zero and one. The rule reasoning processes can be expressed by means of firing transitions of FRPN. The confidence of a rule is associated with its corresponding transition. It is also between 0 and 1 where one means the full confidence and zero means none. The mutual causality interconnection among the statements and reasoning rules can be expressed by means of the analogy with the arcs between places and transitions.

According to the above specification, the first three types of rules can be represented with PN structures in Figures 4.1-4.3, respectively. Note that type 3 rules need to be normalized before they are transformed to a PN structure, i.e.,

Type 3: $R_j(c_j): P_1(\theta_1) \vee P_2(\theta_2) \vee ... \vee P_{k-1}(\theta_{k-1}) \rightarrow P_k(\theta_k)$

is normalized as:

$R_{j1}(c_j): \quad P_1(\theta_1) \rightarrow P_k(\theta_k)$

$R_{j2}(c_j): \quad P_2(\theta_2) \rightarrow P_k(\theta_k)$

$\dots$

$R_{j,k-1}(c_j): \quad P_{k-1}(\theta_{k-1}) \rightarrow P_k(\theta_k)$

Each rule in FRPN uses the logic operator "AND" to combine its preconditions and each proposition uses the logic "OR" to combine its antecedent transitions. Hence, if there is an "OR" operation among the antecedent proposition of a rule in the knowledge base, it has to be normalized into more than one rule before being transformed to FRPN, as illustrated in Figure 4.3.

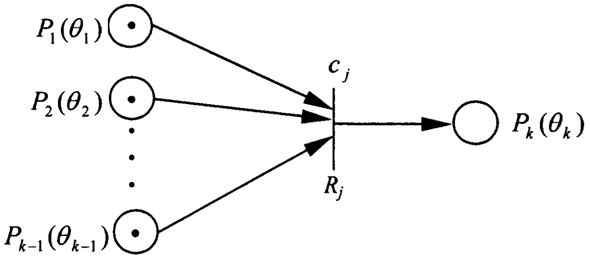


Figure 4.1 The type-1 fuzzy rule presentation with FRPN.

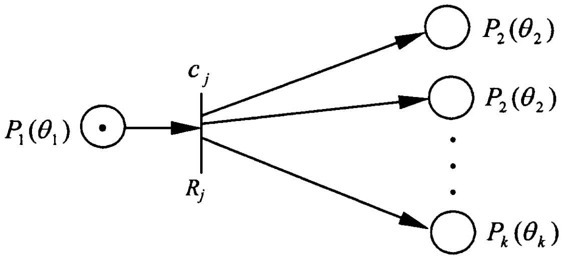


Figure 4.2 The type-2 fuzzy rule presentation with FRPN.

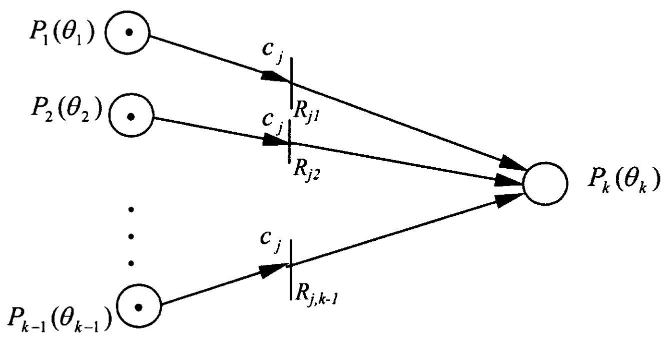


Figure 4.3 The type-3 fuzzy rule presentation with FRPN (after being normalized).

4.2.3 Negation Issue

A proposition (an antecedent or a consequence) in a rule may contain negative literals. Consider the following rules (also shown in Fig. 4 (a)):

R_1 (0.8): IF P_1 AND P_2 AND (NOT P_3) THEN (NOT P_6)

R_2 (0.9): IF P_3 AND P_4 THEN P_6 AND (NOT P_7)

R_3 (0.9): IF P_4 AND P_5 THEN P_8

R_4 (1.0): IF NOT(NOT P_6) THEN P_6

R_5 (0.8): (NOT P_7) and (NOT P_8) THEN P_9

In order to deal with the negation issue, a "complementary arc" is introduced into FRPN. It connects an input place to a transition, and pictorially represented by a directed arc terminated with a small circle. Note that a small circle means "not" in logic circuits.

If a proposition associated with a negation operator is an antecedent in a rule, that is, the proposition is the input of the rule, such as "NOT P_3" in R_1 and "NOT P_8" in R_5, a complementary arc is used to connect the input proposition (place) to the rule (transition). If a proposition associated with a negation operator is a consequent in a rule, that is, the proposition is the output of the rule, such as “NOT P_6” in R_1 and “NOT P_7” in R_2, the whole proposition with the negation operator is considered as a new proposition. For example, $\neg P_6$ and $\neg P_7$ are used to represent “NOT P_6” and “NOT P_7” respectively in Fig. 4. Since NOT(NOT P_i) is equivalent to P_i, if P_i is required based on “NOT P_i” a new rule can be introduced in a rule base, i.e., “IF NOT(NOT P_i) THEN P_i”. One example is P_6 in Figure 4.4.

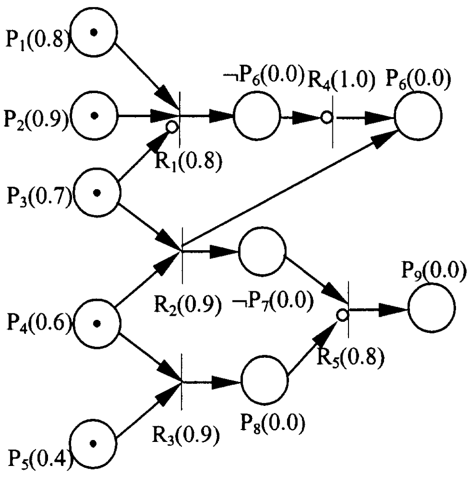
4.2.4 Features of FRPN

FRPN are different from ordinary PN due to the features of fuzzy production rule systems that are different from DES:

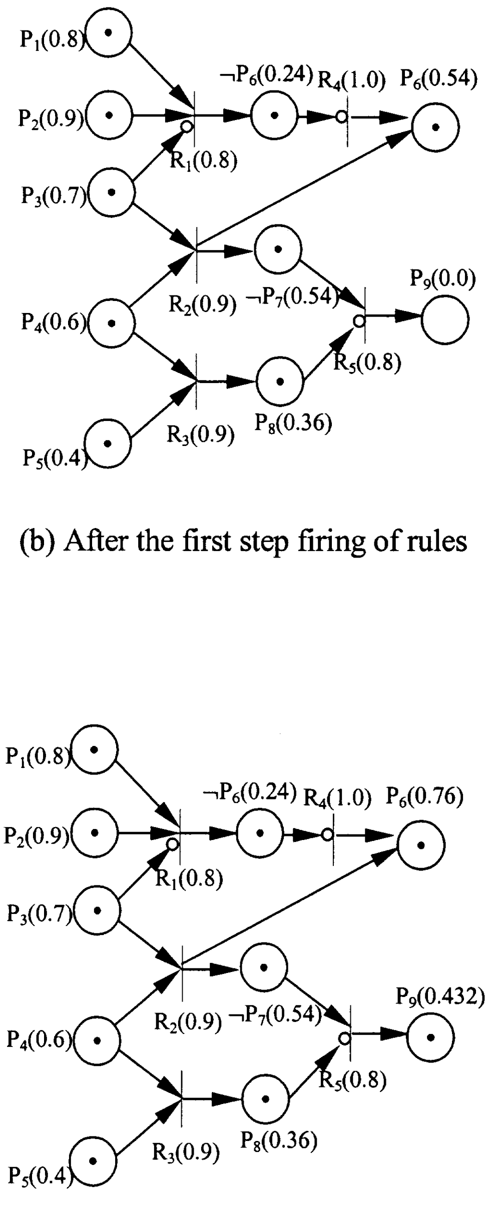
1) In FRPN, the number of tokens in a place cannot be greater than one since a token is associated with a truth degree between zero and one. A token does not represent an “object” while it may likely do so in ordinary PN.
2) FRPN are always conflict-free nets. This point is based on the two facts: a) There is no “resource” concept in FRPN; and b) a proposition may be shared by different rules at the same time. For example, in Fig. 4.4, the proposition P_4 is shared by two rules - R_2 and R_3. R_2 and R_3 can utilize proposition P_4 at the same time and reason in parallel.
3) The tokens are not removed from the input places of a transition after it fires. This is because the evaluation of the rules means only the truth propagation of the propositions. In other words, the truth of a proposition will not disappear because of the rule reasoning. Moreover, a conflict problem would arise if the tokens were removed.

Yang et al. (1997) insist that the tokens are removed from the input places of a transition after the transition fires to keep PN's original firing rules valid. But the conflict problem remains unresolved when an input place is shared by more than one transition. Nazareth (1993) also insists that the tokens are removed from the input places of a transition after it fires. In order to avoid the disappearance of an antecedent propositions' truth after its firing, the structure in Figure 4.5 is proposed. This solves the conflict and token infinity problems. However, the ability to perform the parallel reasoning is weakened when a proposition is shared among more than one rule. Furthermore, a model based on such structures is more complicated.

4) In FRPN, a complementary arc whose input place has a token doesn't inhibit the firing of the rule because a token in it simply means that the truth degree of the corresponding proposition is known and a token is assigned with truth degree between 0 and 1.



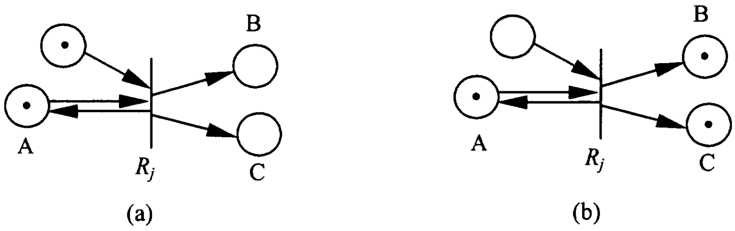
(a) initial state



(b) After the first step firing of rules

(c) After the second step firing of rules

Figure 4.4 A simple FRPN example.



R_j: IF A THEN B and C

Figure 4.5 A kind of rule representation with Petri nets:
(a) Rule representation and
(b) Desired resultant state.

4.3 Formal Definition of FRPN

Fuzzy Reasoning Petri Nets (FRPN) are an extension of PN, which is exclusively used in rule-based knowledge representation and reasoning. The section gives their formal definition and execution rules.

4.3.1 Definition of FRPN

An FRPN can be defined as an 8-tuple:

$$(P, R, I, O, H, \theta, \gamma, C)$$

where

1) $P = \{p_1, p_2, \cdots, p_n\}$ is a finite set of propositions or called places .

2) $R = \{r_1, r_2, \cdots, r_m\}$ is a finite set of rules or called transitions.

3) I: P×R→{0,1}, is an nxm input matrix defining the directed arcs from propositions to rules. I(p_i, r_j) =1, if there is a directed arc from p_i to r_j; and I(p_i, r_j)=0, if there is no directed arcs from p_i to r_j, for i=1,2,...,n, and j=1,2,...,m.

4) O: P×R→{0,1}, is an n×m output matrix defining the directed arcs from rules to propositions. O(p_i,r_j) =1, if there is a directed arc from r_j to p_i; O(p_i,r_j) =0, if there is no directed arcs from r_j to p_i for i=1,2,...,n, and j=1,2,...,m.

5) H: P×R→{0,1}, is an n×m matrix defining the complementary arcs from propositions to rules. H(p_i,r_j) =1, if there is a complementary arc from p_i to r_j; and H(p_i,r_j)=0, if there is no complementary arcs from p_i to r_j, for i=1,2,...,n, and j=1,2,...,m. $I(p,r)\cdot H(p,r)=0, \forall p\in P, and\ r\in R$.

6) θ is a truth degree vector. θ=(θ_1 ,θ_2 $\cdots$,θ_n)^T, where θ_i ∈ [0,1] means the truth degree of p_i, i = 1, 2, ..., n. The initial truth degree vector is denoted by θ^0.

7) γ : P→{0,1}, is a marking vector. γ =(γ_1 ,γ_2 $\cdots$,γ_n)^T, γ_i=1, if there is a token in p_i; and γ_i=0, if p_i is not marked. An initial marking is denoted by γ^0. Tokens are pictured by dots.

8) C = diag{c_1,$\cdots$,c_m}, c_j is the confidence of r_j, j =1, ..., m.

Note that the confidence of rules is represented in a matrix format to facilitate the algorithm development to be discussed later.

The five tuple (P,R,I,O,H) is the basic FRPN structure that defines a directed graph. The updates of the truth degree vector θ through the firing of a set of rules describe the dynamic rule reasoning process of the modeled system. If the truth degree of a proposition is known at a certain reasoning step, a token is assigned to the corresponding proposition, which is associated with the value between 0 and 1. The token is represented by a dot. When a proposition p_i has no token, which means that the truth degree is unknown at that step, $\theta_i = 0$. Hence, $\theta_i = 0$ implies two possible situations - either the absence of token which means truth degree of proposition p_i is unknown or a token with zero value which means the truth degree of proposition p_i is known and equals zero. Marking vector γ can be used to distinguish the two situations.

4.3.2 Execution Rules of FRPN

In order to represent the execution rules of FRPN formally, two operators from Max algebra are cited:

1) ⊕: **A⊕B =D** where **A, B,** and **D** are all m×n-dimensional matrices, such that

$$d_{ij} = \max\{a_{ij}, b_{ij}\}$$

2) ⊗: **A⊗B=D** where **A, B,** and **D** are (m×p), (p×n), (m×n) -dimensional matrices, respectively, such that

$$d_{ij} = \max_{1 \le k \le p}(a_{ik} \cdot b_{kj})$$

Similar to ordinary PN, the execution rules of a FRPN include enabling and firing rules:

1) A rule $r_j \in$R is enabled if and only if

$$p_i \text{ is marked, or } \gamma_i = 1, \ \forall p_i \in \{\text{input propositions of } r_j\}.$$

2) Enabled at marking γ, r_j fires resulting in a new one γ' :

$$\gamma'(p) = \gamma(p) \oplus O(p, r_j), \forall p \in P$$

The truth degree vector changes from θ to θ':

$$\theta'(p) = \theta(p) \oplus c_j \cdot \rho_j \cdot O(p, r_j), \ \forall p \in P.$$

where,

$$\rho_j = \min_{p_i \in {}^{\bullet}r_j}\{x_i \mid x_i = \theta_i \textit{ if } I(p_i,r_j)=1;\ x_i = 1-\theta_i \textit{ if } H(p_i,r_j)=1\}$$

and ${}^{\bullet}r_j = \{p_i \mid I(p_i,r_j)=1 \textit{ or } H(p_i,r_j)=1,\ p_i \in P\}$

3) In FRPN, all the enabled rules can fire at the same time. A firing vector μ is introduced such that $\mu_j=1$ if r_j fires. After firing a set of fuzzy rules, the marking and truth degree vectors of the FRPN become

$$\gamma' = \gamma \oplus [O \otimes \mu] \tag{4.1}$$

$$\theta' = \theta \oplus [(O \cdot C) \otimes \rho] \tag{4.2}$$

where $\rho=[\rho_1, \rho_2, ..., \rho_m]^T$, which is called as control vector. μ : T→{0,1}, is a firing vector. $\mu = (\mu_1, \mu_2 \cdots, \mu_m)^T$.

According to the execution rules, the reasoning process of three types of rules can be analyzed. The truth degrees of the propositions after rules' firing are shown in Figures. 4.6-4.8.

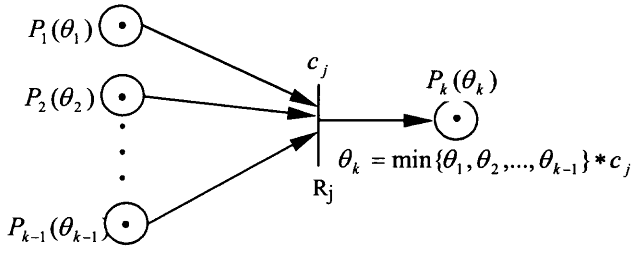


Figure 4.6 The type-1 fuzzy rule presentation with FRPN (after firing rule R$_j$).

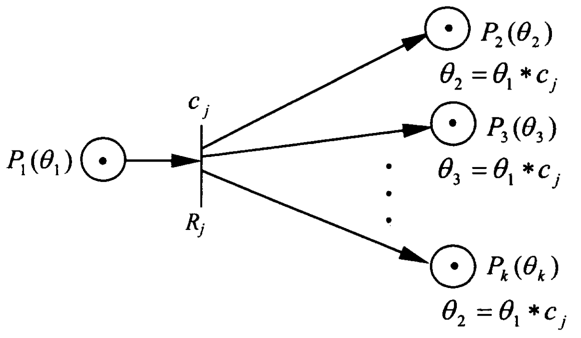
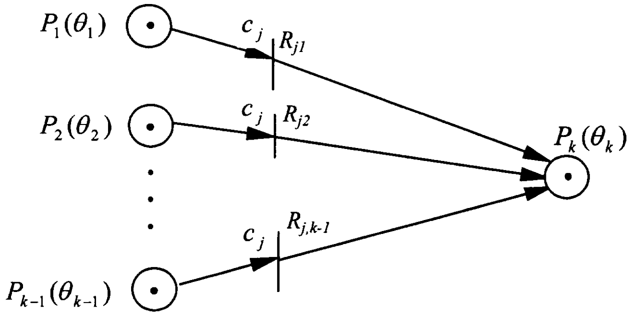

$P_2(\theta_2)$
$\theta_2 = \theta_1 * c_j$
c_j
$P_3(\theta_3)$
$P_1(\theta_1)$
$\theta_3 = \theta_1 * c_j$
R_j
$P_k(\theta_k)$
$\theta_2 = \theta_1 * c_j$

Figure 4.7 The type-2 fuzzy rule presentation with FRPN (after firing rule R$_j$).



$$\theta_k = \max\{\theta_i \mid p_i \in Antecedent(R_j) \wedge p_i \, is \; marked\} * c_j$$

Figure 4.8 The type-3 fuzzy rule presentation with FRPN (after firing rules R$_{j1}$, R$_{j2}$, …, and R$_{j,k-1}$).

Please note that if the arc from P$_i$ to R$_j$ in Figure 4.6-4.8 is a complementary arc, θ_i should be replaced by 1- θ_i.

A practical fuzzy rule-based system may have a combination of these three types of rules. Both P$_i$ and "NOT P$_i$" may exist in a rule-based system. In this case, complementary arcs are needed to model it.

Consider the example shown in Fig. 4.4, where $P=\{P_1, P_2, P_3, P_4, P_5, \neg P_6, P_6, \neg P_7, P_8, P_9\}$, and $R=\{R_1, R_2, R_3, R_4, R_5\}$. According to the definition of FRPN, the output matrix *O* and confidence matrix *C* are:

$$O=\begin{bmatrix}0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\1&0&0&0&0\\0&1&0&1&0\\0&1&0&0&0\\0&0&1&0&0\\0&0&0&0&1\end{bmatrix},\quad C=\begin{bmatrix}0.8&0&0&0&0\\0&0.9&0&0&0\\0&0&0.9&0&0\\0&0&0&1&0\\0&0&0&0&0.8\end{bmatrix}$$

Its initial marking $\gamma^0=[1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^{\mathrm{T}}$ and initial truth degree vector $\theta^0=[0.8\ \ 0.9\ \ 0.7\ \ 0.6\ \ 0.4\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^{\mathrm{T}}$. Initially, R_1, R_2 and R_3 are enabled. The firing and control vectors are

$$\mu^0=[1\ \ 1\ \ 1\ \ 0\ \ 0]^{\mathrm{T}}$$

$$\rho^0=[0.3\ \ 0.6\ \ 0.4\ \ 0\ \ 0]^{\mathrm{T}}$$

According to Equations (4.1) and (4.2), the marking and truth degree vectors of the FRPN after the first step rule firing are:

$$\gamma^1=\gamma^0\oplus[O\otimes\mu^0]=\begin{bmatrix}1\\1\\1\\1\\1\\0\\0\\0\\0\\0\end{bmatrix}\oplus\left(\begin{bmatrix}0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\0&0&0&0&0\\1&0&0&0&0\\0&1&0&1&0\\0&1&0&0&0\\0&0&1&0&0\\0&0&0&0&1\end{bmatrix}\otimes\begin{bmatrix}1\\1\\1\\0\\0\end{bmatrix}\right)=\begin{bmatrix}1\\1\\1\\1\\1\\1\\1\\1\\1\\0\end{bmatrix}$$

$$\theta^1 = \theta^0 \oplus [(O\cdot C) \otimes \rho^0] = \begin{bmatrix} 0.8 \\ 0.9 \\ 0.7 \\ 0.6 \\ 0.4 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \oplus \left(\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0.8 & 0 & 0 & 0 & 0 \\ 0 & 0.9 & 0 & 0 & 0 \\ 0 & 0 & 0.9 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0.8 \end{bmatrix} \otimes \begin{bmatrix} 0.3 \\ 0.6 \\ 0.4 \\ 0 \\ 0 \end{bmatrix} \right)$$

$$=[0.8 \;\; 0.9 \;\; 0.7 \;\; 0.6 \;\; 0.4 \;\; 0.24 \;\; 0.54 \;\; 0.54 \;\; 0.36 \;\; 0]^T$$

At this step, the firing and control vectors are

$$\mu^1 = [1 \;\; 1 \;\; 1 \;\; 1 \;\; 1]^T$$

$$\rho^1 = [0.3 \;\; 0.6 \;\; 0.4 \;\; 0.76 \;\; 0.54]^T$$

After the next step firing of the rules, the marking and truth degree vectors are:

$$\gamma^2 = \gamma^1 \oplus [O \otimes \mu^1] = [1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1]^T$$

$$\theta^2 = \theta^1 \oplus [(O\cdot C) \otimes \rho^1] = [0.8 \;\; 0.9 \;\; 0.7 \;\; 0.6 \;\; 0.4 \;\; 0.24 \;\; 0.76 \;\; 0.54 \;\; 0.36 \;\; 0.432]^T$$

$\theta^3 = \theta^2$ and $\gamma^3 = \gamma^2$ if one more step is calculated. So, the reasoning stops and θ^2 is the final reasoning result. The termination condition will be proved next.

4.4 A Reasoning Algorithm Based on FRPN

4.4.1 Reasoning Algorithm

Based on the model and execution rules of FRPN, a reasoning algorithm can be developed to obtain the final reasoning result about a modeled rule-based system from its initial truth degrees of propositions θ^0. From Equations (4.1) and (4.2), it can be observed that as long as ρ and μ are known, the next step marking and truth degree vectors can be derived from the current values, and so forth. The final reasoning result can be obtained. It will be proved that the reasoning process will terminate. ρ is a vector composed of truth degrees of rules' preconditions. From the view point of control theory, ρ can be called as a "control vector". Hence, it is an important step to obtain the value of ρ. Next, ρ and μ are derived from the model and known parameters so that an integrated reasoning algorithm can be built.

To obtain firing vector and the minimum fuzzy value of the truth degrees among an enabled rule's inputs, a "neg" operator [Tzafestas and Capkovic. 1997] is used. Combining it with the other operators defined in Section 3.2, μ^k and ρ^k can be calculated as the follows:

$$\text{neg}\,\theta^k = \mathbf{1}_\mathbf{m} - \theta^k = \overline{\theta^k}$$

$$\text{neg}\,\gamma^k = \mathbf{1}_\mathbf{m} - \gamma^k = \overline{\gamma^k}$$

$$\mu^k = \overline{(I+H)^T \otimes \overline{\gamma^k}} \tag{4.3}$$

$$\mathbf{v}_\mathbf{k} = (I^T \otimes (\overline{\gamma^k} \oplus \overline{\theta^k})) \oplus (H^T \otimes (\overline{\gamma^k} \oplus \theta^k))$$

$$\rho^k = \text{neg}\ \mathbf{v}_\mathbf{k} = \overline{(I^\mathrm{T} \otimes (\overline{\gamma^k} \oplus \overline{\theta^k})) \oplus (H^T \otimes (\overline{\gamma^k} \oplus \theta^k))} \tag{4.4}$$

where $\mathbf{1}_\mathbf{m} = (1, 1, \ldots, 1)^\mathrm{T}$;

k means the k-th reasoning step;

$\text{neg}\,\theta^k$ is an n-dimensional vector. Its components express the confidence of proposition p_i being false at the kth reasoning step, i=1,...,n;

γ^k is the marking at the kth reasoning step;

μ^k is an m-dimensional firing vector at the kth reasoning step. μ_j^k=1, if r_j is enabled; and μ_j^k=0, if r_j is not enabled, j=1,...,m;

$\mathbf{v_k}$ is an m-dimensional vector. Its components express the confidence of enabled rule r_j's precondition being false at the kth reasoning step, j=1,...,m. The corresponding component equals to one, if the rule is not enabled; and

ρ^k is an m-dimensional control vector at the kth reasoning step. Its components express the truth degrees of enabled rule r_j's preconditions, and ρ_j^k= 0 if rule r_j is not enabled, j=1,...,m.

For example, in Figure 4.4, according to the definition of FRPN, the *I* and *H* are

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad H = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Its initial marking γ^0=[1 1 1 1 1 0 0 0 0 0]^T and initial truth degree vector θ^0= [0.8 0.9 0.7 0.6 0.4 0 0 0 0 0]^T. According to the equations in this section afore, the following results can be obtained:

$$\overline{\theta^0} = [0.2\ 0.1\ 0.3\ 0.4\ 0.6\ 1\ 1\ 1\ 1\ 1]^T.$$

$$\overline{\gamma^0} = [0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1]^T.$$

$$\mu^0 = \overline{(I+H)^T \otimes \overline{\gamma^0}} = [1 \;\; 1 \;\; 1 \;\; 0 \;\; 0]^{\mathrm{T}}.$$

$$\mathbf{v}_0 = (I^T \otimes (\overline{\gamma^0} \oplus \overline{\theta^0})) \oplus (H^T \otimes (\overline{\gamma^0} \oplus \theta^0))$$

$$= [0.7 \;\; 0.4 \;\; 0.6 \;\; 1 \;\; 1]^{\mathrm{T}}.$$

$$\rho^0 = \text{neg}\; \mathbf{v}_0 = [0.3 \;\; 0.6 \;\; 0.4 \;\; 0 \;\; 0]^{\mathrm{T}}.$$

With μ^0 and ρ^0, the next step marking γ^1 and truth degree vector θ^1 can be obtained from Equations (4.1) and (4.2), as discussed in Section 4.3.2. In the same way, the firing vector, control vector, marking and truth degree in the next step can be derived, and so forth.

So far, it has been discussed how to obtain all the necessary parameters for automatic reasoning. Next, the reasoning algorithm based on an FRPN model will be formally described.

From Equation (4.4),

$$\rho^k = \overline{(I^{\mathrm{T}} \otimes (\overline{\gamma^k} \oplus \overline{\theta^k})) \oplus (H^T \otimes (\overline{\gamma^k} \oplus \theta^k))} \tag{4.5}$$

From Equation (4.2),

$$\theta^{k+1} = \theta^k \;\oplus\; [(O \cdot C) \otimes \rho^k] \tag{4.6}$$

From Equations (4.1) and (4.3),

$$\gamma^{k+1} = \gamma^k \oplus [O \otimes \overline{(I+H)^T \otimes \overline{\gamma^k}}] \tag{4.7}$$

where "⊕" and "⊗" are the operators defined in Section 4.3.2.

Then, the FRPN reasoning algorithm can be described as follows:

Step 1: Read FRPN initial inputs: *I, O, H, C,* γ^0 and θ^0;

Step 2: Let k=0;

Step 3:

1) Compute ρ^k according to Equation (4.5);
2) Compute θ^{k+1} from θ^k according to Equation (4.6);
3) Compute γ^{k+1} from γ^k according to Equation (4.7);

Step 4: If $\theta^{k+1} \neq \theta^k$ or $\gamma^{k+1} \neq \gamma^k$, let k=k+1, return to Step 3; otherwise, the reasoning is over.

In the above algorithm, k represents the kth reasoning step.

4.4.2 Complexity of The Algorithm

An acyclic net is a net in which there is no loop or circuit. No circularity exists in most practically used knowledge base [Nazareth, 1993; Koriem, 2000]. Therefore, the FRPN of a fuzzy production rule system is assumed to be an acyclic net.

Theorem 4.1: *The reasoning algorithm for an acyclic FRPN terminates at a finite number of iterations $k \leq h+1$, where h is the number of transitions in the longest place-transition direct path in the net. The computational complexity in the worst case is O(nmh) where n and m are the numbers of places and transitions respectively.*

Proof:

(a) At first, to prove that the reasoning terminates when $\theta^{k+1} = \theta^k$ and $\gamma^{k+1} = \gamma^k$.

If $\theta^{k+1} = \theta^k$ and $\gamma^{k+1} = \gamma^k$, then from Equation (4.5)

$$\rho^{k+1} = \rho^k$$

Then from Equation (4.6),

$$\theta^{k+2} = \theta^{k+1} \oplus [(O \cdot C) \otimes \rho^{k+1}]$$

$$= \theta^k \oplus [(O \cdot C) \otimes \rho^k]$$

$$= \theta^{k+1}$$

And, from Equation (4.7)

$$\gamma^{k+2} = \gamma^{k+1} \oplus [O \otimes \overline{(I+H)^T \otimes \overline{\gamma^{k+1}}}]$$

$$= \gamma^k \oplus [O \otimes \overline{(I+H)^T \otimes \overline{\gamma^k}}]$$

$$= \gamma^{k+1}$$

In the same way,

$\theta^j = \theta^k$ and $\gamma^j = \gamma^k$, if j>k.

Hence, it is proved that the reasoning terminates when $\theta^{k+1} = \theta^k$ and $\gamma^{k+1} = \gamma^k$.

(b) Let h stand for the number of transitions in the longest place-transition direct path. It needs to be prove that $\theta^{h+1} = \theta^h$ and $\gamma^{h+1} = \gamma^h$.

Since h is the number of transitions in the longest place-transition direct path, the components in θ^{h-1} and θ^h are the same except the components in sink places. Also, the components of ρ_F^h express the truth degrees of enabled rules' preconditions, they are not correlative to sink places' truth degrees. From these two view points, it can be derived that $\rho_F^h = \rho_F^{h-1}$.

For the same reason, firing vector $\mu^h = \mu^{h-1}$.

Hence, $\theta^{h+1} = \theta^h \oplus [(O \cdot C) \otimes \rho_F^h]$

$$= \theta^{h-1} \oplus [(O \cdot C) \otimes \rho_F^{h-1}] \oplus [(O \cdot C) \otimes \rho_F^h]$$

$$= \theta^{h-1} \oplus [(O \cdot C) \otimes \rho_F^{h-1}] = \theta^h$$

And, from Equation (4.1)

$$
\begin{aligned}
\gamma^{h+1} &= \gamma^{h} \oplus O \otimes \mu^{h} \\
&= \gamma^{h-1} \oplus O \otimes \mu^{h-1} \oplus O \otimes \mu^{h} \\
&= \gamma^{h-1} \oplus O \otimes \mu^{h-1} \\
&= \gamma^{h}
\end{aligned}
$$

Combining (a) and (b), it can be concluded that the algorithm at the worst case terminates at the (h+1)th step.

Note:

1) The last step reasoning is for checking whether the marking and truth degree vectors still change. If h is known in advance, one can certainly terminate the reasoning at the hth step. Otherwise, the reasoning algorithm will halt at (h+1)th step at the worst case. The algorithm may terminate at a step before h iterations depending on the initial marking, truth degrees of propositions and whether the marking and truth degree vectors change after that step. For example, for the FRPN model in Figure 4, suppose initial marking γ^{0}=[1 1 1 1 1 1 0 1 1 0]^{T} and initial truth degree vector θ^{0}= [0.8 0.9 0.7 0.6 0.4 0.24 0 0.54 0.36 0]^{T}, then γ^{2}= γ^{1}=[1 1 1 1 1 1 1 1 1 1]^{T} and θ^{2}= θ^{1}=[0.8 0.9 0.7 0.6 0.4 0.24 0.76 0.54 0.36 0.432]^{T}, and the reasoning terminates.

2) $\gamma = \mathbf{1}_{\mathbf{n}}$ is a necessary condition for the determination of reasoning but not a sufficient one. This point is indicated in the example in Section 4.5.

4.4.3 Modularization of FRPN

In some cases, a rule-based system can be divided into several sub-systems, e.g., the one in Fig. 4.9. The arrows between sub-systems indicate the propagation of truth degrees. If a rule-based system is composed of several modules, the corresponding FRPN modules can be built up and the algorithm can be executed respectively, which may reduce the complexity. Suppose Sub-system i has the complexity of $O(m_i n_i h_i)$, then the total complexity for the whole rule-based system is $\sum_i O(m_i n_i h_i)$.

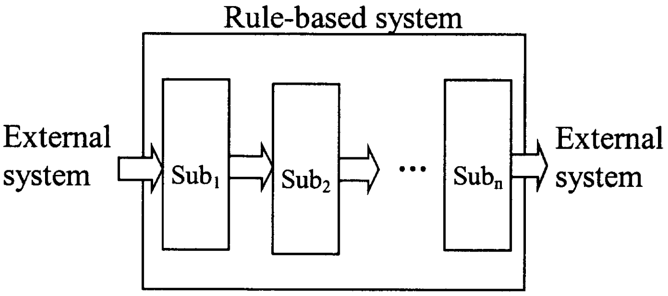


Figure 4.9 A rule-based system composed of several modules.

4.4.4 Procedure to Apply FRPN to Fuzzy Rule-based Reasoning

Given a fuzzy rule-based system and the initial marking and truth degrees of the propositions in the system, the reasoning result can be achieved through the following steps:

Step 1: To normalize the fuzzy rules;

Step 2: To model them using FRPN;

Step 3: To obtain the parameters needed for the reasoning algorithm from the FRPN model;

Step 4: From the initial marking and truth degrees of the propositions to calculate the final reasoning result using the proposed algorithm.

The next section explains the application of the method through a gas turbine fault diagnosis system.

4.5 FRPN for A Gas Turbine Diagnosis System

A rule-based system can be modeled and reasoned using the proposed FRPN model and reasoning algorithm. This section applies the method into a turbine fault diagnosis expert system [Huang, 2000].

4.5.1 Rule Base

Rule 1 (confidence=0.8)

Sympton:

1. The cross section area of turbine's path is too large (p_1)
2. The efficiency of assembling unit is too low (p_2)

Anticipated fault:

Ventilation side of guider's blade of turbine wears and tears (p_3)

Rule 2 (confidence=0.8)

Sympton:

1. The spray head of turbine is broken (p_{25})
2. The flow rate of the fuel in combustor is not too high ($\neg p_7$)

Anticipated fault:

The higher pressure level's spray head of turbine is broken (p_8)

Rule 3 (confidence=0.8)

Sympton:

1. The spray head of turbine is broken (p_{25})
2. The flow rate of the fuel in combustor is too high (p_7)

Anticipated fault:

The higher pressure level's spray head of turbine is broken (p_8)

Rule 4 (confidence=0.8)

Sympton:

1. The outlet gas temperature of turbine is too high (p_9)
2. The efficiency of turbine is too low (p_{10})
3. The flow coefficient of turbine is too low (p_{11})

Anticipated fault:

The blade of turbine scales (p_{12})

Rule 5 (confidence=0.8)

Sympton:

1. The efficiency of turbine is too low (p_{10})
2. The power of assembling unit is too low (p_{13})
3. The efficiency of assembling unit is too low (p_2)

Anticipated fault:

The blade of turbine wears and tears (p_{14})

Rule 6 (confidence=0.8)

Sympton:

1. The efficiency of assembling unit is too low (p_2)
2. The inlet gas temperature of turbine is too high (p_{15})

Anticipated fault:

The blade of turbine burns down (p_{16})

Rule 7 (confidence=0.9)

Sympton:

1. The flow path of compressor wears and tears (p_{17})

Anticipated fault:

The compressor is in turbulence (p_{18})

Rule 8 (confidence=1)

Sympton:

1. The blade of compressor breaks down (p_{19})

Anticipated fault:

The compressor is in turbulence (p_{18})

Rule 9 (confidence=0.8)

Sympton:

1. The conversion flow of compressor is too low (p_{20})
2. The pressurization ratio of compressor is too low (p_5)
3. The compressor has problem (p_{24})

Anticipated fault:

The inlet of compressor freezes (p_{22})

Rule 10 (confidence=1)

Sympton:

1. The efficiency of assembling unit is too low (p_2)
2. The fuel consumption of assembling unit is too high (p_{21})

3. The inlet gas temperature of turbine is too high (p_{15})

Anticipated fault:

The compressor has problem (p_{24})

Rule 11 (confidence=0.9)

Symptom:

1. The pressurization ratio of compressor is too low (p_5)
2. The uniform entropy compression efficiency of compressor is too low (p_{23})

Anticipated fault:

The blade of turbine scales (p_{12})

Rule 12 (confidence=0.8)

Symptom:

1. The conversion flow of compressor is not too low ($\neg p_{20}$)
2. The pressurization ratio of compressor is not too low ($\neg p_5$)
3. The compressor has problem (p_{24})

Anticipated fault:

The flow path of compressor wears and tears (p_{17})

Rule 13 (confidence=0.8)

Symptom:

1. The inlet gas temperature of turbine is too low (p_4)
2. The efficiency of assembling unit is too low (p_2)
3. The pressurization ratio of compressor is too low (p_5)

Anticipated fault:

The spray head of turbine is broken (p_{25})

4.5.2 FRPN Modeling of The Application

Using the procedure in Section 4.4, the first three steps lead to an FRPN model as shown in Figure 4.10. Matrices I, O and H can be obtained from the model.

From the rule base, confidence matrix C = diag (0.8 0.8 0.8 0.8 0.8 0.8 0.9 1.0 0.8 1.0 0.9 0.8 0.8). Suppose that the initial marking is $\gamma^0 = (1\ \ 1\ \ 0\ \ 1\ \ 1\ \ 0\ \ 1\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0\ \ 0\ \ 1\ \ 1\ \ 1\ \ 0\ \ 1\ \ 0\ \ 0)^T$ and the initial truth degree vector is $\theta^0 = (0.6\ \ 1.0\ \ 0.0\ \ 0.0\ \ 0.2\ \ 0.0\ \ 0.0\ \ 0.0\ \ 0.2\ \ 0.9\ \ 0.0\ \ 0.0\ \ 0.8\ \ 0.0\ \ 0.2\ \ 0.0\ \ 0.0\ \ 0.0\ \ 0.0\ \ 0.0\ \ 0.3\ \ 0.0\ \ 0.0\ \ 0.0\ \ 0.0)^T$. Note that initial degrees are obtained by fuzzifying the data gathered from a gas turbine system through data sampling equipment. Given γ^0 and θ^0, the proposed reasoning algorithm can be used to derive the reasoning result.

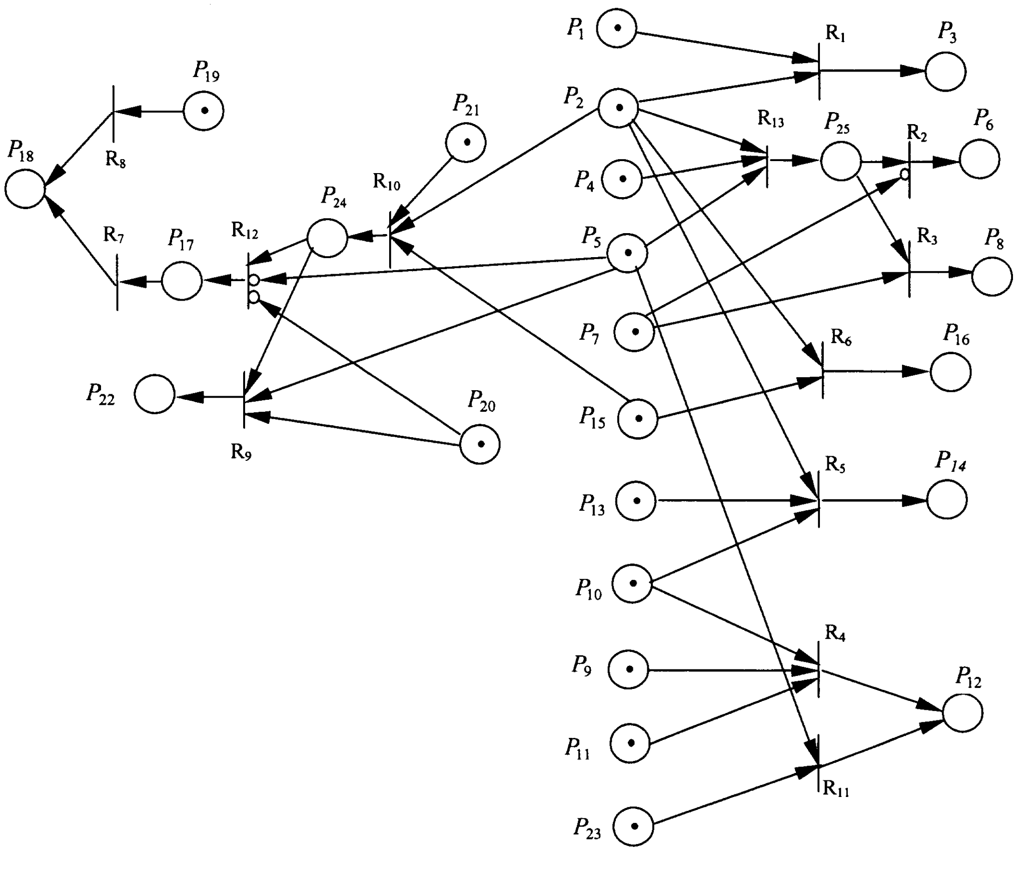
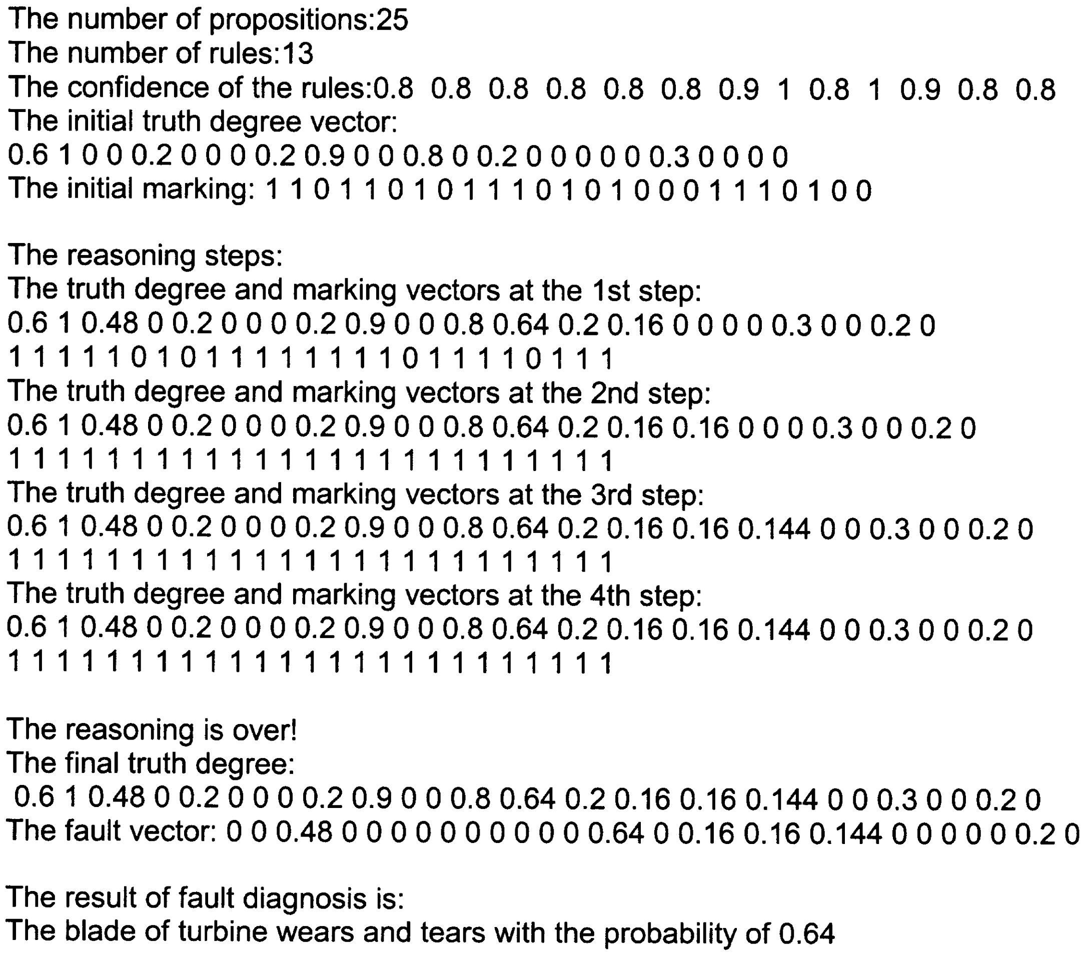


Figure 4.10 The FRPN model of a turbine fault diagram expert system.

4.5.3 Algorithm Execution Results

A C++ program is developed to execute the proposed reasoning algorithm. Because the proposed reasoning algorithm is a formal algorithm based on the operations of matrices, the program is very simple and generic. For different rule based systems, users just need to input appropriate FRPN model parameters, i.e., *I*, *O*, *H*, *C*, γ^0 and θ^0, the algorithm will be executed automatically to obtain the final truth degree of all the propositions. For this example, the execution result is shown in Figure 4.11. Note that the algorithm still



```
The number of propositions:25
The number of rules:13
The confidence of the rules:0.8  0.8  0.8  0.8  0.8  0.8  0.9  1  0.8  1  0.9  0.8  0.8
The initial truth degree vector:
0.6 1 0 0 0.2 0 0 0 0.2 0.9 0 0 0.8 0 0.2 0 0 0 0 0 0.3 0 0 0 0
The initial marking: 1 1 0 1 1 0 1 0 1 1 1 0 1 0 1 0 0 0 1 1 1 0 1 0 0

The reasoning steps:
The truth degree and marking vectors at the 1st step:
0.6 1 0.48 0 0.2 0 0 0 0.2 0.9 0 0 0.8 0.64 0.2 0.16 0 0 0 0 0.3 0 0 0.2 0
1 1 1 1 1 0 1 0 1 1 1 1 1 1 1 0 1 1 1 1 0 1 1 1
The truth degree and marking vectors at the 2nd step:
0.6 1 0.48 0 0.2 0 0 0 0.2 0.9 0 0 0.8 0.64 0.2 0.16 0.16 0 0 0 0.3 0 0 0.2 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
The truth degree and marking vectors at the 3rd step:
0.6 1 0.48 0 0.2 0 0 0 0.2 0.9 0 0 0.8 0.64 0.2 0.16 0.16 0.144 0 0 0.3 0 0 0.2 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
The truth degree and marking vectors at the 4th step:
0.6 1 0.48 0 0.2 0 0 0 0.2 0.9 0 0 0.8 0.64 0.2 0.16 0.16 0.144 0 0 0.3 0 0 0.2 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

The reasoning is over!
The final truth degree:
 0.6 1 0.48 0 0.2 0 0 0 0.2 0.9 0 0 0.8 0.64 0.2 0.16 0.16 0.144 0 0 0.3 0 0 0.2 0
The fault vector: 0 0 0.48 0 0 0 0 0 0 0 0 0 0 0.64 0 0.16 0.16 0.144 0 0 0 0 0 0.2 0

The result of fault diagnosis is:
The blade of turbine wears and tears with the probability of 0.64
```

Figure 4.11 The execution result of the algorithm for the example.

continue at the 2nd step although $\gamma^2=\mathbf{1}_n$. The marking and truth degree in the fourth reasoning step is the same as those in the third step. Thus, the reasoning terminates after 4 reasoning steps. The final truth degree is (0.6 1 0.48 0 0.2 0 0 0 0.2 0.9 0 0 0.8 0.64 0.2 0.16 0.16 0.144 0 0 0.3 0 0 0.2 0)^T, which includes truth degrees of all the propositions, including symptons. The truth degrees of the concerned faults can be picked up by multiplying the other non-concerned elements with zero, shown as the fault vector in Figure 4.10. The fault of "The blade of turbine wears and tears" has the highest value compared to other faults. Therefore the system most likely has the fault of "The blade of turbine wears and tears" with the confidence of 0.64.

4.6 Summary

A particular kind of extended Petri net format – Fuzzy Reasoning Petri Net (FRPN) model is proposed to represent a fuzzy rule based system. A reasoning algorithm based on the proposed model is derived in this research. Even though this new kind of Petri nets is extended from the ordinary Petri nets, its speciality is emphasized to match the rule-based system's characteristics in this research. Both graphical and mathematical advantages of ordinary Petri nets can be well utilized when one employs FRPN for reasoning. Negation issues in PN for knowledge reasoning are for the first time addressed. The proposed algorithm adopts a matrix equation format that is similar to that in the ordinary Petri nets, and exhibits fully parallel reasoning ability. Using this algorithm, the truth degrees of all the propositions can be quickly obtained from the initial ones without using a reachability analysis method. It has great potential to be used to solve complicated problems efficiently. A practical rule-based fault diagnosis system is used to illustrate the proposed method.

The next chapter will apply the method into demanufacturing process decision to handle the uncertain information associated to used products and make dynamic demanufactuing process decisions based on real time information about a product and its components' statuses.

CHAPTER 5

FUZZY REASONING PETRI NETS FOR DEMANUFACTURING PROCESS DECISION MAKING

Practical demanufacturing process planning is extremely important for efficient material recycling and components reuse. Most of the research work on demanufacturing process planning, as discussed in literature review, is intended to generate predictive optimal result based on the information of an assembly [Tang et al, 2000]. The work in Chapter 3 also utilizes the information from a CAD system to complete the disassembly leveling and bin assignment decisions. This kind of work provides predetermined plans and can be used in prediction and assessment. In practice, however, the indeterminate characteristics associated to used products makes the predetermined plan unrealistic and demanufacturing processes have to be decided dynamically adaptive to the products' specific status. Based on theory of FRPN in Chapter 4, this chapter presents a Fuzzy Reasoning Petri net model to represent a used product with uncertainty and its related demanufacturing decision rules, and applies the algorithm based on FRPN model into dynamic decision making in demanufacturing process with uncertain information.

This chapter is organized as follows: Section 5.1 introduces demanufacturing system model. Section 5.2 presents demanufacturing process decision rules and their FRPN Model. Section 5.3 utilizes an example to describe the demanufacturing process decision procedure. Section 5.4 gives summary.

5.1 Demanufacturing System Model

Demanufacturing is an integral part of a product life cycle, which performs a set of functions (such as inspection and disassembly) to recover materials and components for reuse, recycling, or re-manufacturing. A genetic model for a demanufacturing system is shown in Figure 5.1.

The used products coming to a demanufacturing system are separated in collection unit according to their types. Then, for each product, the inspection unit will test its potential reusability, material composition, disassembly success rate and so on. According to the data from the inspection unit and the rules for demanufacturing, the next process for the product will be decided. Finally, the subassemblies or parts getting from the disassembly go to the inspection unit for the next process decision.

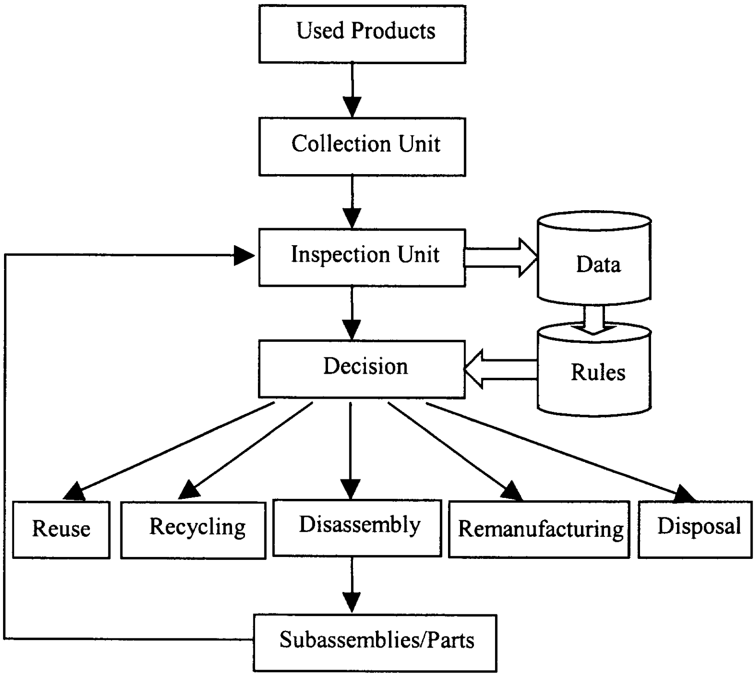


Figure 5.1 A demanufacturing system.

5.2 Demanufacturing Process Modeling with FRPN

Efficient reasoning for a complex electronic product's status based on the present known characteristics is important to ensure highly reliable and efficient disassembly processes. These used products, unfortunately, exhibit high uncertainty in system status since products may experience very different conditions during their use stage. Hence, the demanufacturing processes have to be decided dynamically based on the products' specific status. Moreover, the demanufacturing process usually can't be determined by a single criterion, but multi-criterion rules. The rules themselves are not necessarily of full confidence, which come from the previous experience or expert knowledge. Hence, the complexity and uncertainty have to be handled in a disassembly decision process. An example for demanufacturing process decision making is shown in Figure 5.2.

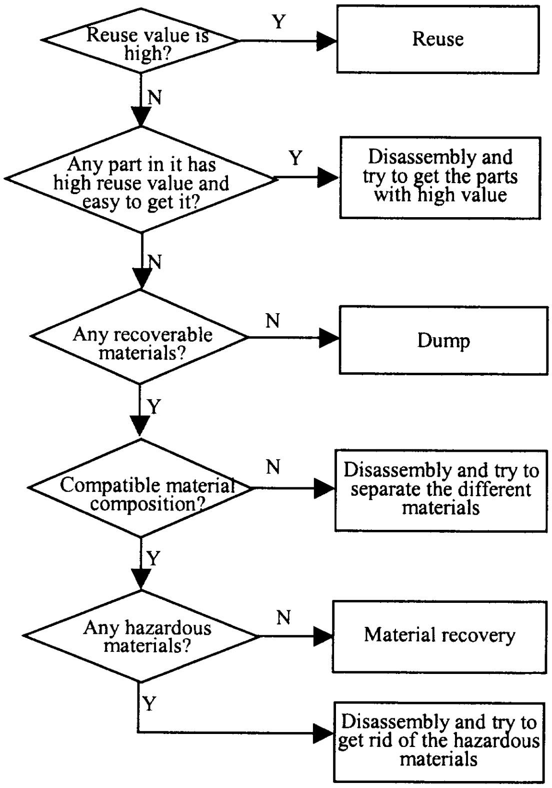

Reuse value is high?
Y
Reuse
N
Any part in it has high reuse value and easy to get it?
Y
Disassembly and try to get the parts with high value
N
Any recoverable materials?
N
Dump
Y
Compatible material composition?
N
Disassembly and try to separate the different materials
Y
Any hazardous materials?
N
Material recovery
Y
Disassembly and try to get rid of the hazardous materials

Figure 5.2 An example for making demanufacturing decisions.

For each product/subassembly, its reuse value is evaluated first. If the reuse value is high enough, the product/subassembly is reused directly. Otherwise, the reuse value of its components (subassemblies/parts) is further investigated. If any subassembly/part has a high reuse value, then try to disassemble the product/subassembly to obtain the subassemblies/parts with high reuse value. Or, one wants to recover the materials from the product/subassembly. If there are no recoverable materials in the product/subassembly at all, it is dumped. If it is composed of recoverable materials, the product/subassembly with compatible material composition doesn't need to be further broken down unless some of its parts pose a hazard.

According to the definition of FRPN, the decision process in Figure 5.2 can be presented as Figure 5.3 using the FRPN model. The definitions of the propositions in Figure 5.3 are the followings:

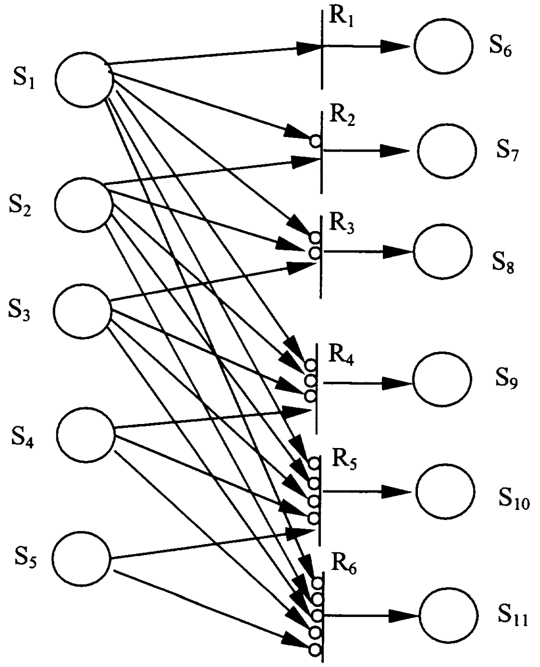


Figure 5.3 The FRPN for demanufacturing process decision.

S1: Reuse value of a product/subassembly is high.

S2: Some parts in it may have high reuse value and it is not difficult to obtain them.

S3: The product/subassembly has no recoverable materials.

S4: The product/subassembly has not the compatible material composition.

S5: No hazardous materials are in the product/subassembly.

S6: The product/subassembly is reused.

S7: The parts with high reuse value are obtained through product/subassembly disassembly.

S8: The product/subassembly is dumped.

S9: The subassemblies/parts with different materials are separated via disassembly.

S10: Material is recovered without further disassembly.

S11: The hazardous materials are removed.

5.3 An Electronic Appliance Example

In this section, the disassembly of a handlight is used as an example to demonstrate the application of the proposed method. The components contained in the handlight include Cover (C), Glass (G), Head-housing (H), Bulb (B), Spring (S) and Main-housing (M). The mental head-housing is screwed onto the same material main-housing, and both the cover and bulb are screwed onto the head-housing. The spring is welded onto the main-housing. All the possible disassembly paths are shown as Figure 5.4, modeled by Disassembly Petri net (DPN) [Zhou and Venkatech, 1999].

Starting with the whole product – handlight, suppose that the following evaluation is obtained according to the data from the inspection unit.

- The possibility for the product to be reused is 0.1;
- "Cover" has high reuse value (certainty =0.9) and the possibility of any other parts' reuse is no more than 0.2;
- Material can be recovered from the product with the certainty of 0.8;
- At this stage, the product does not have the compatible material;
- There are no hazard materials.

Therefore, the initial state for the handlight is $\theta^{0}_{handflight}$= (0.1, 0.9, 0.2, 1, 1, 0, 0, 0, 0, 0, 0)^{T}, and $\gamma^{0}_{handflight} = (1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0)^{T}$. Suppose that all the rules have full confidence i.e. confidence matrix $U = 1_{6\times6}$, According to the reasoning algorithm of FRPN,

$$\theta^{1}_{handlight} = (0.1, 0.9, 0.2, 1, 1, 0.1, 0.9, 0.1, 0.1, 0, 0)^{T}$$

$$\gamma^{1}_{handflight} = (1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1)^{T}$$

$$\theta^{2}_{handlight} = (0.1, 0.9, 0.2, 1, 1, 0.1, 0.9, 0.1, 0.1, 0, 0)^{T}$$

$$\gamma^{2}_{handflight} = (1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1)^{T}$$

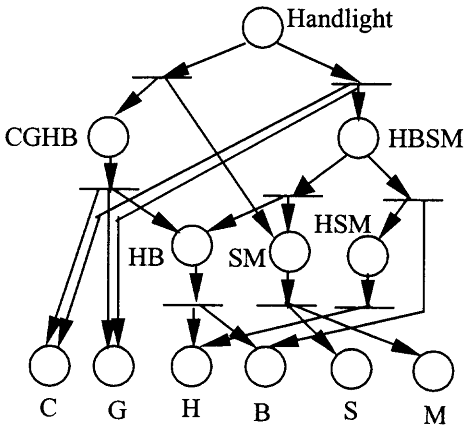


Figure 5.4 Disassembly Petri nets for a handlight.

As $\theta^1_{handlight} = \theta^2_{handlight}$ and $\gamma^1_{handflight} = \gamma^2_{handflight}$, the reasoning is over. The final state of the decision system for the handlight is (0.1, 0.9, 0.2, 1, 1, 0.1, 0.9, 0.1, 0.1, 0, 0)^T. The result of the decision is to disassemble the product/subassembly and try to obtain the part with high reuse value because it has highest truth value of 0.9. In the DPN model, there are two choices to obtain "Cover". Suppose that the path with the minimum disassembly steps is chosen. Hence, transition t_1 is selected.

After disassembling the handlight, a new subassembly - HBSM is obtained. For HBSM, the inspection value is again updated and the similar reasoning process could be used for making further decision and so forth.

Suppose that the initial state for HBSM is θ^0_{HBSM} = (0, 0.2, 0.1, 1, 1, 0, 0, 0, 0, 0, 0)^T. Using the algorithm the decision result for HBSM can be obtained as $\theta^2_{HBSM} = \theta^1_{HBSM}$ = (0, 0.2, 0.1, 1, 1, 0, 0.2, 0.1, 0.8, 0, 0)^T, that is, to separate the different materials from HBSM. Thus, the bulb is disassembled from HBSM, then obtain the new subassembly HSM which has compatible materials. For HSM, θ^0_{HSM} = (0, 0.2, 0, 0.1, 1, 0, 0, 0, 0, 0, 0)^T and $\theta^2_{HSM} = \theta^1_{HSM}$ = (0, 0.2, 0, 0.1, 1, 0, 0.2, 0, 0, 0.8, 0)^T. Thus, HSM is used to recover materials directly without further disassembly.

From the above example, it can be observed that the multi-criterion step-by-step decision process in Figure 5.2 can be modeled by FRPN in a parallel way to obtain the result of reasoning automatically just in one reasoning step. Moreover, for each product/subassembly, the inspection value for decision is updated so that a decision can be made in real time and is closer to the practical situation.

5.4 Summary

The research work for disassembly process planning in literature focuses on the generation of the optimal sequence based on the predictive information of a product. In practice, most of used products' characteristics are unknown, making a predetermined plan unrealistic [Tang et al., 2000]. Hence, the practical approach should consider uncertainty and address a dynamic disassembly system reactive to unpredictive events. This chapter introduces the general demanufacturing system model and applies FRPN modeling and reasoning algorithm into demanufacturing decision making in a demanufacturing system to handle the uncertain information associated to used products. A dynamic demanufactuing decision making process is implemented. Using this method, multi-criterion fuzzy demanufacturing rules, which come from the previous experience and expert knowledge, can be presented in a uniform model to obtain decision making results.

CHAPTER 6

SUSTAINABILITY IMPROVEMENT

In Chapters 3 and 5, the optimal and intelligent decision making methods on demanufacturing processes are proposed to increase the profit and decrease environmental impact from used products. Besides the End-of-Life (EOL) management of products by means of product/material recovery to decrease environmental impacts, the concept of Design for Environment (DFE) is proposed to aim to integrate environmental requirements in a design process [Mahendra and Marcel, 20002]. "Sustainable development" through eco-efficiency has become the rallying cry of a new breed of companies who see competitive advantage in conserving resources and promoting environmental stewardship. How to make its products most sustainable has become one of a company's most important decisions to make. Before making such decision, one has to define the concept of sustainability and has its measurement. The Sustainability Target Method (STM) provides a practical sustainability target for individual businesses and products through determining the relative indicator Resource Productivity (RP) for environmental performance and the absolute indicator Eco-Efficiency (EE) for sustainability [Dickinson et al., 2002].

Based on STM indicators, decision-making methods are investigated in this chapter to provide a company with improvement suggestions and enable a company to improve its product's sustainability at the stages of design and manufacturing.

A product can be viewed generally as an assembly that consists of multiple components. Thus, one of the ways a company can improve its product environmental performance is by improving the performance of these components. Given STM information on the components, how the company utilizes this to improve its product sustainability is a decision-making issue. This chapter investigates the optimal decision-making approach for a company to improve its product's sustainability using the STM indicators RP and EE of the product components. Sensitivity analysis is used to provide a methodology to indicate which components should receive attention to make the overall product more sustainable. The feasibility of the approach is validated by using a generalized product consisting of multiple components, which relates easily to circuit board assemblies and other types of electronic products.

This chapter is organized as follows: Section 6.1 introduces the sustainability target method. Section 6.2 introduces a sensitivity analysis method. Section 6.3 proposes the sensitivity analysis method on STM indicator analysis. Sections 6.4 and 6.5 illustrate the legitimacy and feasibility of the proposed approach through examples. Section 6.6 gives the summary of the chapter.

6.1 Sustainability Target Method

The Sustainability Target Method (STM) approach utilizes the earth's carrying capacity estimates and economic information to provide a practical sustainability target for individual businesses and products. Initially formulated at Lucent Technologies Bell Laboratories, the STM is under continuing development through the collaboration of the Multi-lifecycle Engineering Research Center (MERC) at New Jersey Institute of Technology (NJIT) with Lucent Technologies and Agere Systems. The key and unique feature of this method is the link between carrying capacity, economic value, and environmental impact to provide an absolute or "target" criterion for sustainability that is practical for use by business [Dickinson et al, 2002].

Compared to other environmental impact assessment methods, STM shifts the focus from environmental impact to sustainability and explicitly considers economic or market value of a product. It is equally applicable to analyzing processes, services or facilities. By directly considering economic value, resource productivity measures can be generated, and comparison can be drawn between products or services that are not necessarily "functionally equivalent" but are instead economically equivalent. Thus, the STM approach provides a basis for decision making among different suppliers and manufacturers. This section gives a brief description about STM. The approach is described in detail in [Dickinson, 1999; Dickinson et al., 2001].

STM involves interpreting different types of environmental impact based on a single dimensionless indicator, i.e., the environmental impact per unit production rate, as denoted by EI_{PR}. Its computation is accomplished via normalization by using impact reference levels that relate impact with economic value and sustainability. EI_{PR} provides the basis for calculating the relative indicator Resource Productivity (RP) for environmental performance and the absolute indicator Eco-Efficiency (EE) for sustainability.

A. Environmental Impact (EI)

The Environmental Impact (EI) resulting from an activity, such as manufacturing a product or providing a service, is quantified by normalizing each associated impact using an impact reference level that relates in a specific way to sustainability. EI is aggregated by adding the ratios to obtain a single dimensionless indicator of total impact.

$$EI = I_1 / I_{R1} + I_2 / I_{R2} + \cdots + I_N / I_{RN}$$

Each impact reference level I_R is the level at which the impact is environmentally sustainable at a rate of value generation V_R, i.e., the value reference level. Both the impact level and impact reference level are expressed per unit time (e.g., kg/year for emissions), and the same for V_R (e.g., \$/year).

For production rate P (product items per unit time), the environmental impact per unit production rate EI_{PR} is then:

$$EI_{PR} = EI / P = I_1^* / I_{R1} + I_2^* / I_{R2} + \cdots + I_N^* / I_{RN}$$

where each $I_i^* = I_i / P$ is the impact quantity per product item (e.g., kg CO_2 emissions per kWh of energy or kg of material).

B. Resource Productivity (RP)

Resource Productivity (RP) is the production rate achieved per unit environmental impact:

$$RP = P / EI = 1 / EI_{PR}$$

RP serves as a relative indicator of environmental performance that allows for direct comparison, such as between products or alternative product designs. A larger RP value means a higher production rate achieved per unit environmental impact. An increasing trend in the resource productivity indicator RP would be an indication that the firm is moving in the direction of using resources more efficiently and generating less environmental impact.

C. Eco-Efficiency (EE)

Eco-Efficiency (EE) is defined as:

$$EE = \beta / EI$$

where $\beta = V / V_R$ (value ratio); and V is value creation, e.g. revenue ($/year). The value of the product is established by the market. V_R is the value reference level corresponding to the impact reference levels. Then,

$$\begin{aligned} EE &= (V / V_R) / EI \\ &= (V_{PR} \times P) /(V_R \times EI) \\ &= V_{PR} \times (P / EI) / V_R \\ &= V_{PR} \times RP / V_R \end{aligned}$$

$$\text{i.e. } EE = V_{PR} \times RP / V_R$$

where V_{PR} is the revenue per product unit or value added per unit. EE is an absolute indicator of sustainability. It is the ratio of the actual rate of value generation ($V = V_{PR} \times P$) to the rate that is sustainable given the level of environmental impact ($V_R \times EI$). It can be interpreted as the benefit achieved compared to the minimum allowable benefit given the level of environmental impact incurred. Thus, the criterion for sustainability is $EE \geq 100\%$ (EE>100% simply indicates even less impact than the sustainable level given the value being provided). EE serves as an absolute indicator of sustainability.

Definition: A product is sustainable if its $EE \geq 100\%$.

To make products most sustainable, the values of RP and EE are expected as high as possible. The higher these values, the higher production rate achieved per unit of environmental impact and more sustainable. The methods to compute these values can be found in [Dickinson et al., 2001; 2002].

An example of STM indicators' calculation is provided as follows for illustration. All numerical information shown has been adopted only for this example. The company to be evaluated in this example manufactures a single product at a rate of 1000 units per day (P) and generates revenue of \$400M/year (V). The manufacturing involves three environmental impacts: 1) Global Warming (40K kg/day); 2) Acidification (8K kg/day); and 3) Summer Smog (1 kg/day).

The environmental impact of the manufacturing is

$$EI = I_1 / I_{R1} + I_2 / I_{R2} + I_3 / I_{R3}$$

where:

I_1, I_2, and I_3 are the levels of impacts global warming, acidification, and summer smog respectively;

I_{R1}, I_{R2}, and I_{R3} are the reference levels of impacts global warming, acidification, and summer smog, respectively.

Suppose the reference values are selected as follows: V_R= \$80 Million/year, I_{R1}= 10K kg/day, I_{R2}= 2K kg/day, and I_{R3}= 0.5 kg/day. According to the above description, I_1= 40K kg/day, I_2= 8K kg/day, and I_3= 1 kg/day. Then, the environmental impact of the company is

$$EI = \frac{40K}{10K} + \frac{8K}{2K} + \frac{1}{0.5} = 10$$

The resource productivity indicator, RP, is

$$RP = P / EI = 1000/10 = 100 \text{ (Items per day, per unit environmental impact)}$$

The Eco-Efficiency indicator, EE, is

$$EE = (V / V_R) / EI = (\$400M / \$80M) / 10 = 0.5$$

The environmental impact is twice the sustainable level. In other words, either environmental impact or/and revenue should be improved to make the company sustainable.

6.2 Sensitivity Analysis

In general, a product can be viewed as the one that consists of its components. Thus, a company can improve its product's performances through improving the components' performances. One of the most important performances a company is facing how to improve sustainability, which integrates the strategies for economic success, environmental quality and social equity [Mosovsky et al., 2000]. If a company attempts to make a product more sustainable, it may be able to do so by improving the sustainability of the components of the products.

An issue is raised in this attempt. How does the company utilize the SMT indicators of the components to improve the product's sustainability efficiently? In the next section, a sensitivity analysis approach that enable a company to make the best decision will introduced.

To make the product more sustainable, the company needs to improve the product's RP and EE. If a company expects to improve its product's sustainability through improving the components' performance, the company needs to make a decision on which component should be given the first priority consideration and requires most urgent improvement such that the product is most sustainable.

Sensitivity analysis measures the impact on the outcomes of changing one or more input values. For example, there are three input values. Then an analysis could be performed to see how the outcome changes as each of the three chosen values is considered in turn, with other things being held the same. To improve the sustainability and environment quality, a company expects the values of RP and EE of its products as high as possible. Sensitivity analysis approach can be used to determine which component's performance change has the most notable effect on the product's EE and RP indicators.

Without the loss of generality, it is assumed that a product is produced from three types of components – A, B and C. The quantities of A, B, and C in the product are n_A, n_B, and n_C, respectively. The unit values of A, B, and C are V_{PR_A}, V_{PR_B}, or V_{PR_C}. The RP and EE indicators associated to these three types of components are RP_A, RP_B, and RP_C, and EE_A, EE_B, and EE_C, respectively.

The product's RP:

$$RP = \frac{1}{\frac{n_A}{RP_A} + \frac{n_B}{RP_B} + \frac{n_C}{RP_C} + EI_{PR-Assembly}} \tag{6.1}$$

where $EI_{PR-Assembly}$ is the environmental impact produced in assembling a product from A with quantity n_A, B with n_B, and C with n_C. RP is the production rate achieved per unit environmental impact for the product. From the above equation, it can be seen that the product's RP can be increased through increasing RP_A, RP_B, and RP_C.

The product's EE:

$$EE = V_{PR} \cdot RP / V_R \tag{6.2}$$

Assuming that the change of V_{PR_A}, V_{PR_B}, or V_{PR_C} does not effect RP_A, RP_B, RP_C and V_{PR}, the above two equations indicate that the change of V_{PR_A} (or V_{PR_B}, or V_{PR_C}) does not change the overall RP and EE values of the product. In other words, the improvement on EE_A (or EE_B, or EE_C) by increasing only V_{PR_A} (or V_{PR_B}, or V_{PR_C}) cannot improve the product's RP and EE. Therefore, eventually the product's RP and EE are improved by increasing RP values of the components.

Next section will discuss the sensitivity of the change of a product's RP and EE in terms of the change of its components' RP values.

6.3 Sensitivity Analysis of STM Indicators

6.3.1 Resource Productivity Analysis

$$RP = \frac{1}{\frac{n_A}{RP_A} + \frac{n_B}{RP_B} + \frac{n_C}{RP_C} + EI_{PR-Assembly}}$$

For brevity and clarity, let x $= \frac{n_A}{RP_A} + \frac{n_B}{RP_B} + \frac{n_C}{RP_C} + EI_{PR-Assembly}$, which is the environmental impact per unit production rate of the product.

The sensitivity of RP_A's effect on the product's RP is obtained by taking the first deviation of RP with respect RP_A:

$$\frac{d(RP)}{d(RP_A)} = (-\frac{1}{x^2}) \cdot \frac{dx}{d(RP_A)}$$

$$= (-\frac{1}{x^2}) \cdot n_A \cdot (-\frac{1}{RP_A^2}) = \frac{1}{x^2} \cdot \frac{n_A}{RP_A^2}$$

which is the change of RP of the product per unit change of RP of Component A.

In the same way,

$$\frac{d(RP)}{d(RP_B)} = \frac{1}{x^2} \cdot \frac{n_B}{RP_B^2}$$

$$\frac{d(RP)}{d(RP_C)} = \frac{1}{x^2} \cdot \frac{n_C}{RP_C^2}$$

6.3.2 Eco-Efficiency Analysis

$$EE = V_{PR} \cdot RP / V_R$$

where V_{PR} is the accumulated value per unit product. Assuming that V_{PR} is fixed, V_{PR} / V_R is a constant. Thus, The sensitivity of RP$_A$'s, RP$_B$'s and RP$_C$'s effect on the product's EE is

$$\frac{d(EE)}{d(RP_A)} = \frac{V_{PR}}{V_R} \cdot \frac{d(RP)}{d(RP_A)} = \frac{V_{PR}}{V_R} \cdot \frac{1}{x^2} \cdot \frac{n_A}{RP_A^2}$$

$$\frac{d(EE)}{d(RP_B)} = \frac{V_{PR}}{V_R} \cdot \frac{d(RP)}{d(RP_B)} = \frac{V_{PR}}{V_R} \cdot \frac{1}{x^2} \cdot \frac{n_B}{RP_B^2}$$

$$\frac{d(EE)}{d(RP_C)} = \frac{V_{PR}}{V_R} \cdot \frac{d(RP)}{d(RP_C)} = \frac{V_{PR}}{V_R} \cdot \frac{1}{x^2} \cdot \frac{n_C}{RP_C^2}$$

which are changes of product's EE per unit RP change of Component A, B and C, respectively.

6.3.3 Decision Making Based on RPs of Components

From RP and EE analysis, the following consistent conclusion can be reached. The company should try to improve the RP of the component that can generate the largest improvement to its product's RP and EE. In other words, the component with the maximum value in $\{\frac{n_A}{RP_A^2}, \frac{n_B}{RP_B^2}, \frac{n_C}{RP_C^2}\}$ should be selected for the company to focus on.

However, a company may improve RP of different components with different efforts. Considering this factor, the company should choose the component with the maximum value in $\{\frac{d(RP)}{d(RP_A)\cdot C_A}, \frac{d(RP)}{d(RP_B)\cdot C_B}, \frac{d(RP)}{d(RP_C)\cdot C_C}\}$ and $\{\frac{d(EE)}{d(RP_A)\cdot C_A}, \frac{d(EE)}{d(RP_B)\cdot C_B}, \frac{d(EE)}{d(RP_C)\cdot C_C}\}$, that is, the maximum value in $\{\frac{n_A}{RP_A^2 \cdot C_A}, \frac{n_B}{RP_B^2 \cdot C_B}, \frac{n_C}{RP_C^2 \cdot C_C}\}$, where C_A, C_B, and C_C are the cost per unit change in RP$_A$, RP$_B$, and RP$_C$, respectively. For example, if $\frac{n_A}{RP_A^2 \cdot C_A} > \frac{n_B}{RP_B^2 \cdot C_B}$ and $\frac{n_A}{RP_A^2 \cdot C_A} > \frac{n_C}{RP_C^2 \cdot C_C}$, the company should work on A first, because the improvement of RP$_A$ will cause the most significant improvement on the product's RP per unit cost.

6.3.4 Decision Making Based on EEs of Components

In some cases, RP information of components is not available or the company wants to use components' EE information to make decisions.

From Eq. (5.2), for components A, B, and C:

$$EE_A = V_{PR_A} \cdot RP_A / V_R$$

$$EE_B = V_{PR_B} \cdot RP_B / V_R$$

$$EE_C = V_{PR_C} \cdot RP_C / V_R$$

which are equivalent to

$$RP_A = EE_A \cdot V_R / V_{PR_A}$$

$$RP_B = EE_B \cdot V_R / V_{PR_B}$$

$$RP_C = EE_C \cdot V_C / V_{PR_C}$$

From previous analysis,

$$\frac{d(RP)}{d(RP_A)} = \frac{1}{x^2} \cdot \frac{n_A}{RP_A^2} = \frac{1}{x^2} \cdot \frac{V_{PR_A}^2 \cdot n_A}{EE_A^2}$$

$$\frac{d(RP)}{d(RP_B)} = \frac{1}{x^2} \cdot \frac{V_{PR_B}^2 \cdot n_B}{EE_B^2}$$

$$\frac{d(RP)}{d(RP_C)} = \frac{1}{x^2} \cdot \frac{V_{PR_C}^2 \cdot n_C}{EE_C^2}$$

$$\frac{d(EE)}{d(RP_A)} = \frac{V_{PR}}{V_R} \cdot \frac{1}{x^2} \cdot \frac{n_A}{RP_A^2} = \frac{V_{PR}}{V_R} \cdot \frac{1}{x^2} \cdot \frac{V_{PR_A}^2 \cdot n_A}{EE_A^2}$$

$$\frac{d(EE)}{d(RP_B)} = \frac{V_{PR}}{V_R} \cdot \frac{1}{x^2} \cdot \frac{V_{PR_B}^2 \cdot n_B}{EE_B^2}$$

$$\frac{d(EE)}{d(RP_C)} = \frac{V_{PR}}{V_R} \cdot \frac{1}{x^2} \cdot \frac{V_{PR_C}^2 \cdot n_C}{EE_C^2}$$

Based on EE information of components, the company should choose the component with the maximum value in $\{\frac{V_{PR_A}^2 \cdot n_A}{EE_A^2}, \frac{V_{PR_B}^2 \cdot n_B}{EE_B^2}, \frac{V_{PR_C}^2 \cdot n_C}{EE_C^2}\}$ to focus on because improving the RP of the component can generate the largest improvement to its product's RP and EE.

6.3.5 Results for General Cases

In the above analysis, that a product is composed of three types of components is considered. In practice, no matter how many types of components, the following conclusions can be obtained.

Rule 1. A company is producing a product, which is composed of N types of components. The number of type-i component is n_i; the RP associated to it is $RP_{i,}$ and the cost per unit change in RP_i is C_i, i=1,..,N. In order to make the product more sustainable, the company should choose the component with the maximum value of $\frac{n_i}{RP_i^2 \cdot C_i}$ first to improve the component's RP.

Rule 2. In the case that the cost for resource productivity improvement is not an issue or unknown, a company should select the component with the maximum value of $\frac{n_i}{RP_i^2}$ to improve first.

Rule 3. When n_i=1, i=1,2,…,N, and C_i is unknown, to choose the component with the minimum RP.

Rule 4. A company is producing a product, which is composed of N types of components. The number of type-i component is n_i; the value of type-i component is V_{PR_i}; and the EE associated to it is $EE_{i,}$ i=1,..,N. In order to make the product more sustainable, the company should choose the component with the maximum value of $\frac{V_{PR_i}^2 \cdot n_i}{EE_i^2}$ first to improve the component's RP.

Rules 1-3 are based on RP information of components while Rule 4 is based on V_{PR_i} and EE_i.

6.4 An Illustration Example

In this section, a simple example is used to illustrate the above analysis and verify the above conclusions. It assumes that a company is producing a printed circuit board, which is composed of 4 type-A chips (ID: A) and 2 type-B chips (ID: B) and Base Board (ID: C) respectively. The relevant parameters are shown in Table 6.1.

Table 6.1 Components of a Product, and Their Production Rates per Unit Environmental Impact (RP), Costs per Unit Change of RP (C), Eco-Efficiency (EE), and Unit Value (V_{PR})

ID	Number	RP	C	EE	V_{PR}
A	4	200	0.05	1	2
B	2	120	0.08	1.2	4
C	1	150	0.07	1.1	3

6.4.1 Based on RP and Costs Per Unit Change of RP

$$\frac{n_A}{RP_A^2 \cdot C_A} = \frac{4}{200^2 \cdot 0.05} = \frac{1}{500}$$

$$\frac{n_B}{RP_B^2 \cdot C_B} = \frac{2}{120^2 \cdot 0.08} = \frac{1}{576}$$

$$\frac{n_C}{RP_C^2 \cdot C_C} = \frac{1}{150^2 \cdot 0.07} = \frac{1}{15750}$$

$$\frac{n_A}{RP_A^2 \cdot C_A} > \frac{n_B}{RP_B^2 \cdot C_B} > \frac{n_C}{RP_C^2 \cdot C_C}$$

According to Rule 1 in previous section, in this example, the company should focus on Chip A's performance improvement first, then Chip B, finally the base board because Chip A's RP has the most significant effect on the product per unit cost, and base board has the least effect.

Initially, RP and EE of the product:

$$RP = \frac{1}{\frac{n_A}{RP_A} + \frac{n_B}{RP_B} + \frac{n_C}{RP_C} + EI_{PR-Assembly}}$$

$$= \frac{1}{\frac{4}{200} + \frac{2}{120} + \frac{1}{150} + 0.01} = 18.87$$

$$EE = V_{PR} \cdot RP / V_R = 20 \times 18.87/400 = 0.94$$

Assume that RPs of Chip A, Chip B and the base board are increased by 50 per unit environmental impact. Their effect on the product's sustainability are analyzed, respectively.

First consider that RP of Chip A is increased by 50 per unit environmental impact, and RPs of Chip B and the base board keep unchanged, i.e., RP_A=250, RP_B=120, and RP_C=150 per unit environmental impact. It is assumed that $EI_{PR-assembly}$=0.01, V_{PR}=\$20 per unit product, V_R=\$400 per unit environmental impact.

$$RP = \frac{1}{\frac{n_A}{RP_A} + \frac{n_B}{RP_B} + \frac{n_C}{RP_C} + EI_{PR-Assembly}}$$

$$= \frac{1}{\frac{4}{250} + \frac{2}{120} + \frac{1}{150} + 0.01} = 20.26$$

$$EE = V_{PR} \times RP / V_R = 20 \times 20.26/400 = 1.013$$

The cost for improvement is:

$$cost = C_A \cdot \Delta RP_A = 0.05 \times 50 = 2.5$$

$$\frac{\Delta RP}{cost} = \frac{20.26 - 18.87}{2.5} = 0.556$$

$$\frac{\Delta EE}{cost} = \frac{1.013 - 0.94}{2.5} = 0.029$$

In the same way, if RP of Chip B is increased by 50 per unit environmental impact, and RPs of Chip A and the base board keep unchanged,

$$RP = 20.63$$

$$EE = 1.032$$

$$cost = 4$$

$$\frac{\Delta RP}{cost} = 0.44$$

$$\frac{\Delta EE}{cost} = 0.023$$

If RP of the based board is increased by 50 per unit environmental impact, and RPs of Chip A and Chip B keep unchanged.

$$RP = 19.34$$

$$EE = 0.967$$

$$cost = 3.5$$

$$\frac{\Delta RP}{cost} = 0.14$$

$$\frac{\Delta EE}{cost} = 0.0068$$

From the above calculation, it can be concluded that Chip A has the most significant effect on the product's sustainability by the same performance improvement as Chip B and the base board per unit cost. Thus, in this example, the company should focus on Chip A's RP performance improvement first to improve the product's sustainability.

6.4.2 Based on RP Information of Components

When the cost per unit change of a component's RP is not an issue, unknown, or equal.

$$\frac{n_A}{RP_A^2}=\frac{4}{200^2}=\frac{1}{10000}$$

$$\frac{n_B}{RP_B^2}=\frac{2}{120^2}=\frac{1}{7200}$$

$$\frac{n_C}{RP_C^2}=\frac{1}{150^2}=\frac{1}{22500}$$

$$\frac{n_B}{RP_B^2}>\frac{n_A}{RP_A^2}>\frac{n_C}{RP_C^2}$$

According to Conclusion (2), without considering the cost for RP improvement, the company should focus on Chip B's performance improvement first, then Chip A, and finally the base board. In this example, EE of the product is improved to 1.032 if RP of Chip B is increased by 50, while the EE is improve to 1.013 and 0.967 respectively if Chip A and the base board are selected to be improved. It can be observed the Chip B has the most significant effect on the product's sustainability by the same performance improvement as Chip A and the base board. Therefore, the result is verified.

6.4.3 Based on EE Information of Components

$$\frac{V_{PR_A}^2\cdot n_A}{EE_A^2}=\frac{2^2\cdot 4}{1^2}=16$$

$$\frac{V_{PR_B}^2\cdot n_B}{EE_B^2}=\frac{4^2\cdot 2}{1.2^2}=22.2$$

$$\frac{V_{PR_C}^2\cdot n_C}{EE_C^2}=\frac{3^2\cdot 1}{1.1^2}=7.4$$

$$\frac{V_{PR_B}^2 \cdot n_B}{EE_B^2} > \frac{V_{PR_A}^2 \cdot n_A}{EE_A^2} > \frac{V_{PR_C}^2 \cdot n_C}{EE_C^2}$$

According to Conclusion (3), based on EE information of components, the company should focus on Chip B's performance improvement first, then Chip A, and finally the base board. The result is the same as the result based on RP information of components when the cost for improvement is under no consideration.

6.4.4 Discussion

From above analysis, the following interesting results are obtained: to improve a product's sustainability, a company is not necessary to pick up the component with worst RP or EE to focus on first. In this example, $RP_A > RP_C > RP_B$ and $EE_B > EE_C > EE_A$. However, the sequence of priority to be improved is A→B→C (based on RP and costs per unit change of RP) and B→A→C (based on RP or EE information without considering costs per unit change of RP).

6.5 Application to A Handy Board

In this section, "Handy Board" is used as an example (Figure 6.1) to illustrate the above analysis and verify the above conclusion. The Handy Board is a 68HC11-based controller board designed for an experimental mobile robot. The Handy Board is based on the 52-pin Motorola MC68HC11 processor, and includes 32K of battery-backed static RAM, four outputs for DC motors, a connector system that allows active sensors to be individually plugged into the board, an LCD screen, and an integrated, rechargeable battery pack. This design is ideal for experimental robotic project, but the Handy Board

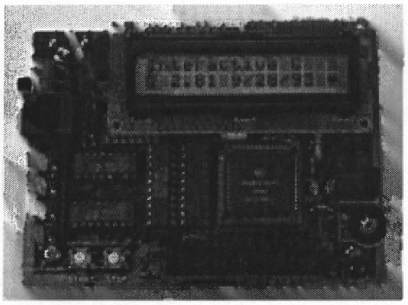


Figure 6.1 The Handy Board.

can serve other embedded control applications as well. The main components in the board are listed in Table 6.2, where RP and C are the environmental impact and costs per unit change of RP respectively.

It is assumed that the environmental impact produced in assembling the product from its components $EI_{PR-assembly}=$ 0.07; unit value of the product V_{PR}=\$88 per unit product; value reference level V_R=\$100 per unit environmental impact.

Initially, the RP and EE of the product:

$$RP = \frac{1}{\sum_{i=1}^{14} \frac{n_i}{RP_i} + EI_{PR-Assembly}}$$

$$= \frac{1}{0.83+0.07} = 1.1$$

$$EE = V_{PR} \times RP / V_R = 88 \times 1.1/100 = 0.968$$

In order to improve the product's sustainability, one can try to improve its components' RPs. From Table 6.2 and Rule 1 in Section 6.3.5, it can be observed that the company should focus on Component No. 9 -16x2 LCD first, because it has the maximum value of n/(RP^2*C), which means that it has the most significant effect on the whole product per unit cost. When the cost per unit change of a component's RP is not an issue or unknown, according to Table 6.2 and Rule 2 in Section 6.3.5, the company should focus on Nicad battery because it has the maximum value of n/RP^2.

Assume that RPs of LCD and battery are increased by 10 per unit environmental impact. Their effect on the product's sustainability respectively will be analyzed next. First consider that LCD's RP is increased by 10 per unit environmental impact, and RPs of other components' RPs keep unchanged, then RP and EE of the product become:

$$RP = \frac{1}{\sum_{i=1}^{14} \frac{n_i}{RP_i} + EI_{PR-Assembly}}$$

$$=1.15$$

$$EE = V_{PR} \times RP / V_R = 88 \times 1.15/100 = 1.012$$

The cost for improvement is:

$$cost = C_A \cdot \Delta RP_A = 0.05 \times 10 = 0.5$$

$$\Delta RP = 1.15\text{-}1.1 = 0.05$$

$$\Delta EE = 1.012\text{-}0.968 = 0.044$$

$$\frac{\Delta RP}{cost} = \frac{0.05}{0.5} = 0.1$$

$$\frac{\Delta EE}{cost} = \frac{0.044}{0.5} = 0.088$$

In the same way, if RP of battery is increased by 10 per unit environmental impact, and RPs of other components keep unchanged, then the RP and EE of the product become:

$$RP=1.26$$

$$EE=1.109$$

$$cost=5$$

$$\Delta RP=1.26\text{-}1.1=0.16$$

$$\Delta EE=1.109\text{-}0.968=0.141$$

$$\frac{\Delta RP}{cost}=0.032$$

$$\frac{\Delta EE}{cost}=0.023$$

From the above calculation, it can be seen that the battery has more significant effect on product's sustainability for the same performance improvement than LCD does. However, if the cost for RP improvement is considered, the LCD has more significant effect on product's sustainability for the same performance improvement than the battery per unit cost does. The results are consistent with the conclusions derived in Section 6.3.5.

Table 6.2 Main Components of a Handy Board

ID	**Component Name**	**n**	**RP**	**n/RP^2**	**C**	**n/(RP^2*C)**
1	Nicad battery	1	6	0.028	0.5	0.056
2	AC/DC adaptor	1	26	0.0015	0.05	0.03
3	LED	15	355	0.00012	0.05	0.0024
4	32K static CMOS RAM	1	26	0.0015	0.2	0.0075
5	Motor driver	2	34	0.0017	0.05	0.034
6	Voltage regulator	3	112	0.00024	0.1	0.0024
7	RS232 converter	1	52	0.0004	0.05	0.008
8	6811 Microprocessor	1	13	0.0059	0.2	0.0295
9	16x2 LCD	1	15	0.0044	0.05	0.088
10	SPDT switch	2	23	0.0038	0.05	0.075
11	DIP socket	9	100	0.0009	0.05	0.018
12	Connector	1	65	0.00024	0.05	0.0048
13	Infrared demodulator	1	34	0.00087	0.05	0.017
14	PCB Base Board	1	20	0.0025	0.1	0.025

6.6 Summary

Based on STM, this chapter investigates the approaches to support a company to improve its product's sustainability using the product's components' STM indicators, and adopts sensitivity analysis method to provide a company with improvement suggestions on which components should receive more attention and perform urgent improvement to make its products more sustainable. Through the analysis, the result is obtained that a company is not necessary to pick up the component with worst RP or EE to focus on first in order to improve a product's sustainability. The future work includes investigating the application of STM in supply chain management and lifecycle assessment, such as supplier selection, analysis of service, use and recycling activities as well as the design and manufacturing processes. The data acquisition approaches for calculation and the decision-makings based on the uncertain information are also future discussion issues.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

Increasing global population and product consumption are causing declining natural and social systems. Sustainable development addresses these issues by integrating strategies for economic successes, environmental quality, and social equity. Sustainable development for the Earth and its future generations depends on turning environmental responsibility into a competitive advantage through End-of-Life (EOL) management of discarded products and new product design to minimize waster and maximize resource utilization.

The changing needs of customers not only require a great variety of products, but also make the products obsolete quickly. Most old electronic products are either in storage or thrown in landfills rather than recycled. EOL electronics contain a significant amount of toxic materials, which are not a problem when consumers use electronic products, but they can create health and environmental hazards if they are not properly disposed of at end-of-life. Multi-Lifecycle Engineering is a comprehensive and systematic approach to growing a strong industrial economy while maintaining a clean, healthy environment. The use of materials and components from discarded products is enhanced so that they could be used over more than one product lifecycle. As an integral part of a product life cycle, demanufacturing is a process of disassembling a product and assigning bins for the resultant subassemblies and parts, which are then reused, re-manufactured, reengineered, or disposed of. Efficient demanufacturing of a product is fundamental to fulfill the goals of multi-lifecycle engineering. Bin assignment task has been considered the one following a disassembly process in the current literatures. Yet, the two processes are actually interweaved. In this doctoral research, the information from a CAD (Computer Aided Design) system is utilized to develop a methodology to integrate the disassembly leveling decision with the bin assignment task. An example is performed by applying the methodology to a computer product analysis.

Since products during the use stage may experience different conditions, their external and internal status can vary significantly. In this research, an Fuzzy Reasoning Petri net model and fuzzy reasoning algorithm are proposed to deal with uncertain information, and are successfully applied to demanufacturing process decision making so that the demanufacturing processes are dynamically decided.

Besides the End-of-Life (EOL) management of products by means of product/material recovery to decrease environmental impacts, the concept of Design for Environment (DFE) is proposed to aim to integrate environmental requirements in a design process. Based on Sustainability Target Method (STM), decision-making methods are investigated in this research to provide a company with improvement suggestions and enable a company to choose their suppliers to improve the company's sustainability.

In addition to the theoretical research, the proposed methods are implemented into a software package. Using today's advanced computing technologies, the Multi-LifeCycle Assessment and Analysis (MLCA) tool is developed to gather and analyze the information of the whole lifecycle of a product, including product design, material processing, production, use stage, and up to demanufacturing, remanufacturing, reengineering stage. It provides product developers with a set of systematic approaches to green product and system development and therefore increase significantly the efficiency and profitability of manufacturing and demanufacturing industries, and improve the quality of environment.

The contributions of this dissertation are summarized into three aspects:

1) Bin assignment algorithm for multi-lifecycle assessment tool development

   An integrated methodology to complete both the disassembly leveling and bin assignment decisions is developed through balancing the effort involved in disassembly processes, value return, and environmental impact. The proposed method is successfully implemented into a demanufacturing module of MLCA software so that a product can be analyzed through its whole life cycle. It utilizes the information from a CAD system and requires minimal user input and can be automatically executed during a design process. It is proved that the maximum value can be retrieved and minimum environmental burden can be produced from used products using the proposed method, under the assumption that a subassembly is disassembled into its immediate children in one aggregate step. The algorithm can derive relatively high value return from a product with low computational complexity. An example is used to illustrate the methodology.

2) Decision making with uncertainty

   Since products during the use stage often experience different conditions, used products are thus associated with uncertain information so that the demanufacturing processes have to be decided dynamically based on the products' specific status. To deal with uncertain information, a particular kind of extended Petri net format – a Fuzzy Reasoning Petri Net (FRPN) model is proposed to represent a fuzzy rule based system. A reasoning algorithm based on the proposed model is derived. FRPN can be used in generic fuzzy rule based systems for modeling and reasoning. Both graphical and mathematical advantages of ordinary Petri nets can be well utilized when one employs FRPN for reasoning. The proposed algorithm exhibits fully parallel reasoning ability. The method is applied into demanufacturing process decision to handle the uncertain information associated to used products and implement dynamic demanufactuing process decision.

3) Sustainability improvement

Besides decision making approaches based on end-of-life management, this research also investigates the methodologies to improve a product's sustainability in the design and manufacturing stage. Based on sustainability target method, a sensitivity analysis decision-making method is proposed to provide a company with improvement suggestions and enable a company to improve its product's sustainability effectively. The method allows a company to focus on a right component to improve its performance and thus achieve the needed product sustainability.

This research has the following limitations:

1) In the bin assignment investigation, it is assumed that a subassembly at a certain level is disassembled into its immediate children at the next level in one aggregate step. This assumption does not always hold in practice;
2) In FRPN for fuzzy reasoning, it is assumed that the rule base is error free. If the rule base itself has errors, the reasoning results may be inconsistent. The FRPN has reasoning ability but does not have learning ability;

3) In sustainability improvement based on STM approach, this research only considers one kind of decision making problem, which is to improve a product's sustainability through improving its components' performance.

7.2 Future Research

There are several ways in which this work could be extended in the future. Some important and promising directions are listed as follows:

1) In bin assignment algorithm, it is assumed that a subassembly at a certain level either does not need disassembly or is disassembled into all of its immediate children at the next level. This assumption holds sometimes but not always. If all the disassembly sequences are considered, the computational complexity will be very large (the complexity exponentially increases in terms of the number of components) and all the link relations have to be known, which makes the development of the algorithm in the multi-lifecycle assessment software not practical. A tradeoff method needs to be developed to obtain better solution without too much input information and too high complexity. In the research, the bin assignment results can be modified by manual assignment. In the future, the improvement work may be to develop some kind of heuristic rules which are added following the proposed algorithm to make the disassembly leveling and bin assignment more practical;
2) Based on the improved bin assignment algorithm, the MLCA tool can also be improved. Besides, more effort is required to implement more methodologies into the software package to make the tool more powerful and complete. For example,

the decision making approaches based on STM can be integrated into the software. The MLCA tool is developed based on local database. Distributed online version is required to widen its application;

3) As mentioned in the limitations, it is assumed that the rule base is correct in FRPN for fuzzy reasoning. The validation and testing of rule base embedded in rule reasoning can be the future research work. To improve knowledge rules by combining the learning mechanism with the FRPN model is a promising research work. Moreover, additional applications should be explored in the future;

4) Further integrating the method of dynamic disassembly process decision making with disassembly line design to develop an integrated disassembly system is a promising research direction. Thus the end of life value of used products, the disassembly system throughput and precious resource utilization can be optimized;

5) More decision making issues based on STM approach can be implemented. How to make decision on supply chain management based on STM method is another direction for the future work.

6) Industrial case studies are required to be developed. Although several examples are performed in the research, they are limited to laboratory tests. Collaboration with industrial firms would be valuable to putting the theoretical research results into practical uses. Especially collection of accurate data represents a tedious yet extreme important task. On the other hand, more appropriate design and implementation methodologies may be required to be developed for particular industrial applications.

APPENDIX

DATA OF ZENITH LAPTOP CASE STUDY

Zenith Laptop Z Star 433VL example is used to illustrate the proposed bin assignment algorithm in Chapter 3. The data and results are presented in the appendix.

Table A.1 Product Tree Structure, Material Composition and the Results

Components		Tree Structure			Material Composition		Results	
ID	Name	Parent	Type	Level	Material	Weight	Target Bin (B_i)	Value (V_i)
1p	Zenith Laptop Z Star 433 VL	N/A	S	0	N/A	6660.1	N/A	5.59
p191	CPU Cover	1p	P	1	Plastic	0.004	Mix Plastic	0.00044
p192	Ports Back Cover	1p	P	1	Plastic	0.011	Mix Plastic	0.00121
p193	Side Body Cover	1p	P	1	Plastic	0.002	Mix Plastic	0.00022
p194	Top Panel Cover	1p	P	1	Plastic	0.061	Mix Plastic	0.00671
p195	Battery Cover	1p	P	1	Plastic	0.012	Mix Plastic	0.00132
p196	Battery	1p	FS	1	Ni/Cu Polyester	1360	Hazard	1
p197	Floppy Disk Compartment	1p	FS	1	Aluminum	242	Reuse	3
p198	Bottom Body	1p	S	1	N/A	477.34	N/A	2.08
p199	Main Bottom House	p198	P	2	Plastic	0.22	Mix Plastic	0.0242
p200	Main Mother Board Tray	p198	S	2	N/A	477.12	N/A	2.232
p201	Tray	p200	P	3	Steel	0.082	Steel	0.00164
p202	Hard drive	p200	FS	3	Aluminum	171	Reuse	2
p203	Main Mother Board	p200	S	3	N/A	306.038	N/A	0.5632
p204	Internal Battery	p203	FS	4	Ni/Cu Polyester	28	Hazard	0.2
p205	Chip Slot #1	p203	FS	4	Circuit Board	14	PCB	0.007
p206	Mother Board	p203	FS	4	Circuit Board	203	PCB	0.1015
p207	Mother Board Lock	p203	P	4	Steel	0.038	Steel	0.00076
p208	Control Board	p203	FS	4	Circuit Board	53	PCB	0.0265
p209	RAM	p203	FS	4	Circuit Board	8	Reuse	1
p210	Monitor	1p	S	1	N/A	1397.841	Flatscreen	0.072
p211	Back Cover	p210	P	2	Plastic	0.164	N/A	0
p212	Front Cover	p210	P	2	Plastic	0.114	N/A	0
p213	LCD Frame	p210	P	2	Aluminum	0.087	N/A	0
p214	LCD Display Unit	p210	S	2	N/A	920.136	N/A	0
p215	Circuit Board for Front cover	p214	FS	3	Circuit Board	16	N/A	0
p216	EPSON LCD Screen	p214	FS	3	Leaded Glass	427	N/A	0
p217	Keyboard	1p	S	1	N/A	1585.863	PCB	0.0885
p218	Keyboard Frame	p217	P	2	Plastic	0.022	N/A	0
p219	Keyboard Panel	p217	FS	2	Circuit Board	188	N/A	0
p220	Indicator Display Panel	1p	S	1	N/A	1596.883	Mix Plastic	0.0016
p221	Circuit Board #1	p220	FS	2	Circuit Board	2	N/A	0
p222	Circuit Board #2 and Speaker	p220	FS	2	Circuit Board	9	N/A	0
p223	Display Panel	p220	P	2	Plastic	0.02	N/A	0
p226	Monitor Wires	p214	P	3	Wire	0.016	N/A	0
p227	Bolts & Nuts	1p	P	1	Steel	0.041	Steel	0.006

Table A.2 Reuse Value (R_i) and Hazard Mitigation Value (H_i)

ID	Name	Reuse Value	Hazard Mitigation Value
p196	Battery	0	1
p204	Internal Battery	0	0.2
p209	RAM	1	0
p202	Hard Drive	2	0
p197	Floppy Disk Compartment	3	0

Table A.3 Unit Material Value in Bin β (C_β), and Recycling Yield of Material j in Bin β ($\Phi_{\beta j}$)

Bin Name	Unit Value	Steel	Circuit Board	Ni/Cu Polyester	Aluminum	Leaded Glass	Wire	Plastic
Aluminum High	$0.49	-0.5	-0.5	-0.5	1	-0.5	-0.5	-0.5
Aluminum Mix	$0.19	-0.5	-0.5	-0.5	0.9	-0.5	-0.5	-0.5
Cables	$0.18	-0.5	-0.5	-0.5	-0.5	-0.5	0.2	-0.5
Cardboard	$0.05	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Cards	$0.95	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Clarkboard	$4.00	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Connectors	$0.94	-0.5	-0.5	-0.5	-0.5	-0.5	0.5	-0.5
Copper Foil	$0.08	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Copper High	$0.67	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Copper Mix	$0.08	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Copper Trim	$0.08	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
CRTs	$0.20	-0.5	-0.5	-0.5	-0.5	0.9	-0.5	-0.5
Flatscreen	$0.50	-0.5	-0.5	-0.5	-0.5	0.8	-0.5	-0.5
Gold Ends	$0.93	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Hard Mix Plastic	$0.11	-0.5	-0.5	-0.5	-0.5	-0.5	1	1
Keyboards	$0.23	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Magnets	$0.02	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Mods	$1.25	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Other Mix	$0.12	0	0.5	0	0	0	0	0
PCB	$0.50	-0.5	1	-0.5	-0.5	-0.5	-0.5	-0.5
Photodrum	$0.06	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Recycle Plastic	$0.02	-0.5	-0.5	-0.5	-0.5	-0.5	0.5	1
Softfoam	$0.06	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Stain High	$0.23	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Stain Mix	$0.10	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Steel	$0.02	1	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Styrofoam	$0.05	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
Toner	$0.05	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
UPS	$0.02	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5

Note: The negative value in this table means that material generates adverse effect on recycling of the main material in the corresponding bin.

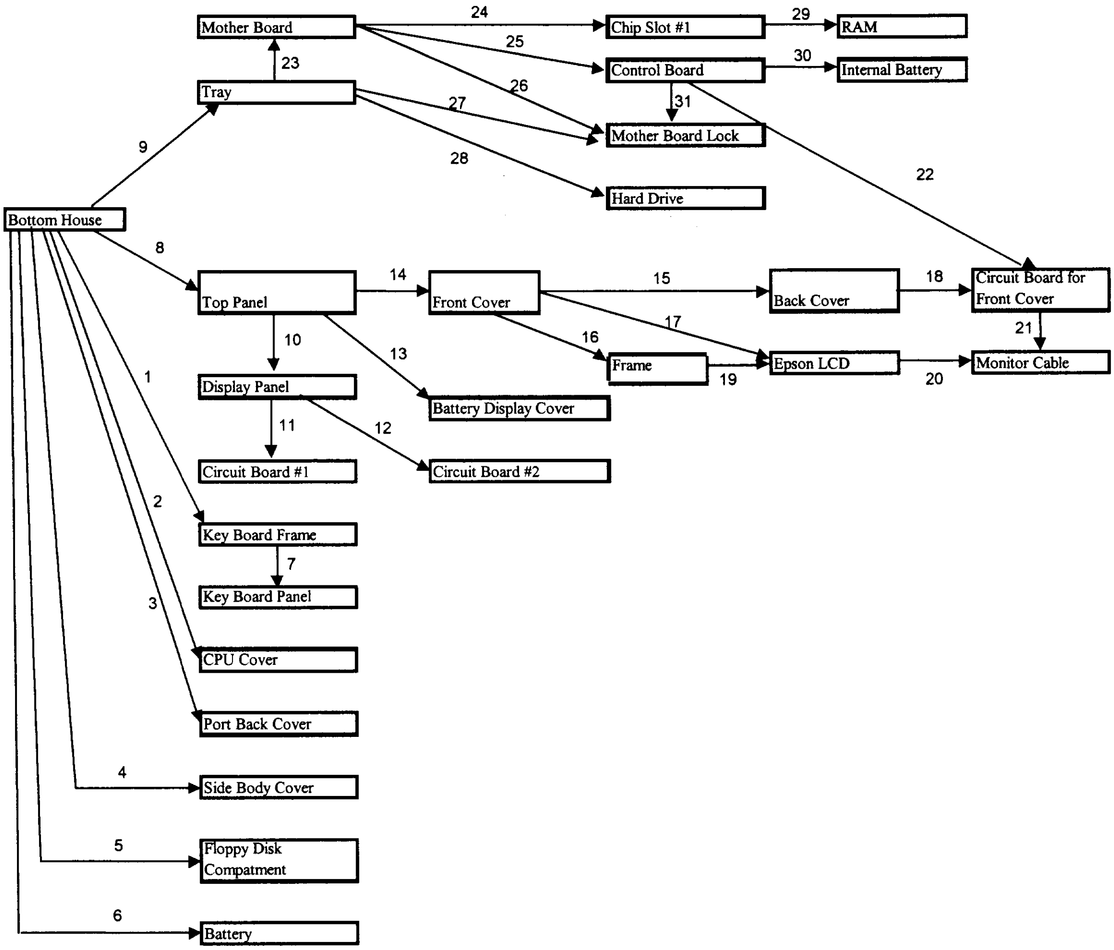


Figure A.1 Mating diagram of Zenith laptop computer [Yossapol and Boontaveekit, 2000].

Table A.4 Disassembly Procedure and Data

Part/Subassembly Name	Tool/Procedure	Time		Effort
		(min)	(sec)	($)
MAIN BODY				
Battery	Hand		10	0.03
CPU Cover	Hand		5	0.02
Floppy Disk Compartment	Hand		30	0.10
Port Back Cover	Hand		5	0.02
Slide Body Cover	Screwdriver		20	0.07
Top Panel Cover	Screwdriver	1		0.20
Battery Cover (Indicator Display Panel)	Hand		30	0.10
Bolts & Nuts	Screwdriver		30	0.10
BOTTOM BODY				
Main Bottom House	Hand		15	0.05
Main Mother Board Tray	Hand		30	0.10
Tray	Star and Screwdriver		30	0.10
Hard Drive	Screwdriver	1		0.20
Main Mother Board	Star and Screwdriver	1		0.20
Internal Battery	Hand		5	0.02
Chip Slot #1	Screwdriver		20	0.07
Mother Board Lock	Screwdriver	1	20	0.27
Control Board	Screwdriver		20	0.07
RAM	Hand		5	0.02
Mother Board	Hand		5	0.02
MONITOR	Screwdriver	2	30	0.50
Back Cover	Screwdriver	1		0.20
Front Cover	Screwdriver		30	0.10
LCD Frame	Screwdriver		30	0.10
LCD Display Unit	Screwdriver	2		0.40
Circuit Board for Front Cover	Hand		30	0.10
Monitor Wire	Hand		10	0.03
Epson LCD Screen	Hand		20	0.07
KEYBOARD	Screwdriver		20	0.07
Keyboard Frame	Hand		20	0.07
Keyboard Panel	Hand		20	0.07
INDICATOR DISPLAY PANEL	Screwdriver		10	0.03
Circuit Board#1	Screwdriver		30	0.10
Circuit Board#2 and Speaker	Hand		10	0.03
Display Panel	Hand		10	0.03

Table A.5 Disassembly Results for Case 3

Disassembled Components	Value ($)	Disassembly Effort ($)
Floppy Disk Compartment	3	0.1
Hard Drive	2	0.2
Keyboard	0.09	0.07
Battery	1	0.03
Internal Battery	0.5	0.02
RAM	1	0.02
Chip Slot #1	0.01	0.07
Tray	0.002	0.1
Main Mother Board	0.232	0.2
Bolts & Nuts	0.006	0.1
Remainder	--	--
	7.84 (Total value)	0.91 (Total effort)
	6.93 (Net value gain)	

REFERENCES

1. Arai E., and Iwata K., "CAD system with product assembly/disassembly planning function," *Robotics & Computer - Integrated Manufacturing,* vol. 10, pp. 41-48, 1993.

2. Bugarin A. J., and Barro S., "Fuzzy reasoning supported by Petri nets," *IEEE Trans. On Fuzzy System*, vol. 2, No. 2, pp. 135-149, 1994.

3. Cao T., and Sanderson A. C., "Task sequence planning using fuzzy Petri nets," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 25, No. 5, pp. 755 - 768, 1995.

4. Cardoso J., Valette R., and Dubois D., "Possibilistic Petri nets," *IEEE Transaction on System, Man, Cybernetics, Part B: Cybernetics*, vol. 29, No. 5, pp. 573-582, 1999.

5. Caudill R. J., "A Final Proposal to the National Science Foundation Engineering Research Center Program," Multi-Life Cycle Research Center, New Jersey Institute of Technology, Newark, NJ, September 1997.

6. Caudill R. J., "Multi-lifecycle Engineering and Demanufacturing of Discarded Electronic Products," Technical Report, Multi-lifecycle Engineering Research Center, New Jersey Institute of Technology, Newark, NJ, October 1998.

7. Caudill R. J., Zhou M. C., Yan P., and Jin J., "Multi-life Cycle Assessment: An Extension of Life Cycle Assessment," in *Mechanical Lifecycle Engineering: Good Environmental Design and Manufacturing*, Marcel Dekker, NY, 2001.

8. Chen S. M., Ke J. S, and Chang J. F., "Knowledge representation using fuzzy Petri nets," *IEEE Transation on Knowledge and Data Engineering*, vol. 2, No. 3, pp. 311-319, 1990.

9. Chen S. M., "Weighted fuzzy reasoning using weighted fuzzy Petri nets," *IEEE Transactions on Knowledge and Data Engineering,* vol. 14, No. 2, pp. 386-397, 2002.

10. Dickinson, D. A., Mosovsky, J.A., Caudill, R.J., Watts, D.J., and Morabito, J.M., "Application of the Sustainability Target Method: Supply Line Case Studies," *2002 IEEE International Symposium on Electronics and the Environment*, 2002, pp. 139-143.

11. Dickinson D. A., "A Proposed Universal Environmental Metric," Bell Laboratories Internal Memorandum, August 1999.

12. Dickinson D. A, Mosovsky J. A., and Morabito J. M., “Sustainability: An evaluation & Target Method for Business – Summary & Reference Levels,” Lucent Technologies Bell Laboratories Technical Memorandum, May 2001.

13. Dieoff: http://dieoff.org, February 2003.

14. Dym C., and Levitt R, *Knowledge-Based Systems in Engineering*, New York: McGraw-Hill, 1991.

15. EPA: United States Environmental Protection Agency, Remanufacturing Products: good as new. EPA530-N-002, 1997.

16. Fiksel J., “Design for Environment: The New Quality Imperative,” *Corporate Environmental Strategy*, vol. 1, No. 3, December 1993.

17. Fiksel J., *Design for Environment: Creating Eco-Efficient Products and Process,* McGraw-Hill Inc., 1996.

18. Fu L. M., *Neural Networks in Computer Intelligence*, New York: McGraw-Hill, 1994.

19. Fussler C. and James P., *Driving Eco Innovation: A Breakthrough Discipline for Innovation and Sustainability*, London: Pitman Publishing, 1996.

20. Gao M., Wu Z. M., and Zhou M. C., “A Petri Net-based Formal Reasoning Algorithm for Fuzzy Production Rule-based Systems,” *Proceeding of 2000 IEEE International Conference on System, Man & Cybernetics,* Nashville, TN, Oct. 2000, pp. 3093-3098.

21. Gao M., Zhou M. C., and Caudill R, “Development of Demanufacturing Module for Multi-Lifecycle Assessment and Analysis of Products,” *Proc. of the 2001 IEEE Int. Conf. Systems, Man, and Cybernetics,* October 2001, Tucson, AZ, pp. 2016-2021.

22. Gao M., Zhou M. C., and R. J. Caudill, “A Bin Assignment Algorithm for Demanufacturing Automation,” *Proc. of the 2000 IEEE International Conference on Robotics and Automation,* Washington D. C., May 11-15, 2002, pp. 2351-2356.

23. Gao M., Zhou M. C., and Caudill R. J., “Integration of Disassembly Leveling and Bin Assignment for Demanufacturing Automation,” *submitted to IEEE Transactions on Robotics and Automation,* vol. 18, No. 6, pp. 867-874, December 2002.

24. Gao M., Zhou M. C., Huang X., and Wu Z., "Fuzzy Reasoning Petri Nets," To appear at *IEEE Transactions on Systems, Man, and Cybernetic,* June 2003.

25. Goedkoop M., "The Eco-indicator 95 Final Report," Pre Consultants, the Nertherlands, 1998.

26. Goedkoop M., and Sepriensma R., "The Eco-indicator 99, A Damage Oriented Method for Life Cycle Impact Assessment, Methodology Report," Second Edition, Pre Consultants, the Nertherlands, April 2000.

27. Goldberg D. E., *Genetic Algorithms in Search, Optimization and Machine Learning,* Reading, MA: Addison-Welsey.

28. Grefenstette J., "Optimization of Control Parameters for Genetic Algorithms", *IEEE Transactions on Systems, Management and Cybernetics*, vol. 16, No.1, 1982.

29. Gungor A., and Gupta S. M., "Issues in Environmentally Conscious Manufacturing and Product Recovery: A Survey," *Computers and Industrial Engineering,* vol. 36, No. 4, pp. 811-853, 1999.

30. Hertz J., Krogh A., and Palmer R., *Introduction to the Theory of Neural Computation,* Reading, MA: Addison-Wesley, 1991.

31. Homen de Mello L. S., and Sanderson A. C., "AND/OR Graph Representation of Assembly Plans," *IEEE Trans. on Rob. & Aut.*, vol. 6, No. 2, pp. 188-199, 1990.

32. Homen de Mello L. S., and Sanderson A. C., "Representation of Mechanical Assembly Sequences," *IEEE Trans. on Rob. & Aut.*, vol. 7, No. 2, pp. 211-217, 1991.

33. Huang H. H., and Wang M. H., "Optimal disassembly sequence generation using Neural Network," *Proceedings of the Fourth International Conference on Environmentally Conscious Design and Manufacturing*, Cleveland, OH, 23-25, July 1996, pp. 231-238.

34. Huang X. G., "A Study on Fault Diagnosis of Gas Turbine Based on Thermal Parameters," *Ph.D. Dissertation, Shanghai Jiao Tong University*, P.R.C., June 2000.

35. Jin J., "Multi-Life Cycle Assessment Design Tolls and Software Development," *M.S. Thesis*, Electrical and Computer Engineering Department, New Jersey Institute of Technology, Newark, NJ, May 1999.

36. Jovane F., Alting L., Armillotta A., and et al., "A key issue in product life cycle: disassembly," *Annals of the CIRP*, vol. 42, No. 2, 1993, pp. 640-672.

37. Konar A., and Mandal A. K., "Uncertainty management in expert systems using fuzzy Petri nets," *IEEE Tran. On Knowledge and Data Engineering*, vol. 8, No. 2, pp. 96-105, 1996.

38. Koriem S. M., "A fuzzy Petri net tool for modeling and verification of Knowledge-based systems", *The Computer Journal*, vol. 43, No. 3, pp. 206-223, 2000.

39. Lambert A. J. D., "Optimal disassembly of complex products," *International Journal of Production Research*, vol. 35, No. 9, pp. 2509-2523, 1997.

40. Lambert A. J. D., "Optimal disassembly of complex products," *Proceedings of the Northeast Decision Sciences Institute Conference*, Portsmouth, NewHampshire, 1994, pp. 74-80.

41. Lambert A. J. D., "Optiaml disaeembly of complex products", *International Journal of Production Research,* vol. 35, No. 9, 1997, pp. 2509-2523.

42. Lambert A. J. D., "Optiaml disaeembly of complex products", *Proceedings of the First International Working Seminar on Reuse*, Eindhoven, The Netherlands, Nov. 1996, pp. 213-222.

43. Looney C. G., "Fuzzy Petri nets for rule-based decisionmaking," *IEEE Trans. Syst., Man, Cybern.*, vol. 18, No. 1, pp. 178-183, 1988.

44. Lund R., *The Remanufacturing Industry: Hidden Giant.* Boston, MA: Boston University, 1996.

45. *Mechanical Life Cycle Handbook-Good Environmental Design and Manufacturing,* Edited by Mahendra S. Hundal, Marcel Dekker, Inc., 2002.

46. Mani V., Das S., and Caudill R., "Disassembly complexity and recyclability analysis of new designs from CAD file data," *Proc. of the 2001 IEEE Int. Symp. on Electronics and the Environment,* 2001, pp. 10-15.

47. Marshall H. E., "Sensitivity Analysis", *Technology Management Handbook,* Chapter 8.12, CRC Press LLC, Boca Raton, FL, Dorf, R. C., Editor, pp. 8.59-63, 1999.

48. Multi-lifecycle Engineering Research Center, http://merc.njit.edu, March 2003.

49. Moore, K. E., Gungor, A., and Gupta, S. M., "Petri Net Approach to Disassembly Process Planning," *Computers and Industrial Engineering*, vol. 35, No. 1-2, Oct., 1998, pp. 165-168.

50. Moore, K. E., Gungor, A., and Gupta, S. M., "Disassembly Petri Net Generation in the Presence of XOR precedence Relationships," *Proc. of the IEEE Int. Conf. on Systems, Man, & Cybernetics,* Vol. 1, San Diego, California, Oct. 11-14, 1998, pp. 13-18.

51. Moore, K. E., Gungor, A., and Gupta, S. M., "Disassembly Process planning using Petri nets," *Proc. of the IEEE Int. Symposium on Electronics, and the environment,* Oak Brook, IL, 1998, pp. 88-93.

52. National Safety Council, http://www.nsc.org/ehc/epr2/baseline.htm, February 2003.

53. Navin-Chandra D. "The recovery problem in product design," *Journal of Engineering Design,* vol. 5, No. 1, pp. 65-86, 1994.

54. Navin-Chandra D. "ReStar: a design tool for environmental recovery analysis," *Proceeding of ICED'93: 9th International Conference on Engineering Design,* The Hague, 17-19 August 1993, pp. 780-787.

55. Negoita C. V., *Expert Systems and Fuzzy Systems*, Massachusetts: Benjanmin/Cummings, 1985.

56. Penev K., and Ron A, "Determination of a disassembly strategy," *International Journal of Production Research*, vol. 34, No. 2, 1996, pp. 495-506.

57. Porada T., "Materials recovery: Asset alchemy," *Proceeding of the IEEE International Symposium on Electronics and the Environment,* San Francisco, CA, 2-4, May 1994, pp. 171-174.

58. Ron A, and Penev K., "Disassembly and recycling of electronic consumer products: an overview," *Technovation*, vol. 16, No. 6, pp. 363-367, 1995.

59. Scarpelli H., Comide F., and Yanger R., "A reasoning algorithm for high-level fuzzy Petri nets," *IEEE Trans. On Fuzzy Systems,* vol. 4, No. 3, pp. 282-294, 1996.

60. Society for Environmental chemistry and toxicology (SETAC), Guideline for Life Cycle Assessment: A Code of Practice, Workshop held at Sesimbra, Portugal, March 31-April 3, 1993.

61. Shyamsundear H., Srinivassan H., and Gadh R., "Virtual De-manufacturing Via Virtual Disassembly to Design Environmentally Conscious Products," *Int. Journal of Environmentally Conscious Design & Manufacturing,* vol. 6, No. 1, pp. 37-50, 1997.

62. Tang Y., Zhou M. C., Caudill R., and Zussman E., "Disassembly Modeling, Planing and Application: A Review," *Proc. of the 2000 IEEE Int. Conf. on Robotics & Automation,* April 2000, pp. 2197-2202.

63. Tang Y., Zhou M. C., and Caudill R. J., "An Integrated Approach to Disassembly Planning and Demanufacturing Operation," *IEEE Trans. on Robotics and Automation*, vol. 17, No. 6, pp. 773-784, Dec. 2001.

64. Tazaki E., and Yoshida K., "A fuzzy Petri net model for approximate reasoning and its application to medical diagnosis," *Proc. of IEEE International Conference on System, Man and Cybernetics*, Chicago, IL, 1992, pp. 627-631.

65. Tzafestas S. G., and Capkovic F., "Petri net-based approach to synthesis of intelligent control systems for DEDS," *Computer-assisted Management and Control of Manufacturing Systems*, Springer, 1997, pp. 523-351.

66. Valette R., Cardoso J., and D. Dubois, "Monitoring manufacturing systems by means of Petri nets with imprecise markings," *Proc. 1989 IEEE Int. Symp. Intell. Contr.*, Albany, NY, Sept. 1989, pp. 233-238.

67. Wackernagal et al, "National Natural Capital Accounting with the Ecological Footprint Concept," Ecological Economics, vol. 29, pp. 375-390, 1999.

68. Yan X., and Gu P., "A graph based heuristic approach to automated assembly palnning," *Flexible Assembly Systems*, vol. 73, pp. 97-106, 1994.

69. Yang S. J. H., Chu W. C., Lee J., and Huang W. T., "A fuzzy Petri nets based mechanism for fuzzy rules reasoning," *Proc. of The Twenty-First Annual International Computer Software and Applications Conference,* 1997, pp. 438-443.

70. Yossapol C. and Boontaveekit V., term project report for MNE 638: Multi-Lifecycle Engineering, New Jersey Institute of Technology, December 2000.

71. Zadeh L., "Fuzzy Logic and Approximate Reasoning," *Synthese* 30, pp. 407-428, 1975.

72. Zanakis S. H., and Evans J. R., "Heuristic Optimization: Why, When, and How to Use it," *Interface,* October 1981.

73. Zhang H. C., Kuo T. C., and Huang, S. H., "Recycling model for end-of-life products," *Janpan/USA Symposium on Flexible Automation*, 1996, pp. 1127-1130.

74. Zhang H. C., and Kuo T. C., "A Graph-Based Disassembly Sequence Planning for EOL Product Recycling," *21st IEEE/CPMT Int. Electronics Manufacturing Technology Symp*, Austin, TX, Oct. 13-15, 1997, pp. 140-151.

75. Zhang H. C., and Kuo T. C., "A Graph-Based Approach to Disassembly Model for End-of-life Product Recycling," *19th Proc. of Int. Electronics Manufacturing Technology Symp.*, Austin, TX, Oct. 14-16, 1996, pp. 247-254.

76. Zhou M. C., Zhang B., Caudill R. J., and Sebasrian D., "A Cost Model for Multi-lifecycle Engineering Design," *Proceedings of the 1996 IEEE International Symposium on Emerging Technologies and Factory Automation*, Hawaii, Nov. 1996, pp. 385-391.

77. Zhou M. C, Caudill R. J., Sebastian D., and Zhang B., "Multi-lifecycle Product Recovery for Electronic Products," *Journal of Electronics Manufacturing*, vol. 9, No. 1, pp. 1-15, March 1999.

78. Zhou M. C., and Venkatech K., *Modeling, simulation and control of flexible manufacturing systems-A Petri net approach*, World Scientific Publisher. 1999.

79. Zhou M. C., He J., and Caudill R. J., "A CAD-MLCA interface for next-generation DfE tools," Working Paper, Department of Electrical and Computer Engineering, New Jersey Institute of Technology, May, 2001.

80. Zussman E, "Planning of disassembly systems," *Assembly Automation*, vol. 15, No. 4, 1995, pp. 20-23.

81. Zussman E., Kriwet A, and Seliger G., *"Disassembly-oriented assessment methodology to support design for recycling,"* Annals of the CIRP, vol. 43, No. 1, 1994, pp. 9-14.

82. Zussman E., Reiter, B. S., and Scharke, H., "Modeling and Planning of Disassembly Processes," *Proc. of the IFIP WG5.3 Int. Conf. on Life-Cycle Modeling for Innovative Products and Processes,* Berlin, Germany, Nov./Dec., 1995, pp. 221-232.

83. Zussman E., Zhou, M.C., and Caudill, R., "Disassembly Petri Net Approach to Modeling and Planning Disassembly Processes of Electronic Products," *Proc. of IEEE Int. Symp. on Electronic and Environment*, Oak Brook, IL, May 4-6, 1998, pp. 331-337.

84. Zussman E. and Zhou, M.C., "Methodology for Modeling and Adaptive Planning of Disassembly Processes," *IEEE Trans. on Rob. & Aut.*, vol. 15, No. 1, Feb. 1999, pp. 190-194.

85. Zussman E. and Zhou, M.C., "Design and Implementation of an Adaptive Process Planner for Disassembly Processes," *IEEE Trans. on Rob. & Aut.*, vol. 16, No. 2, April, 2000, pp. 171-179.